RAIRO-Oper. Res. 60 (2026) 1701–1732
https://doi.org/10.1051/ro/2026039

RAIRO Operations Research
www.rairo-ro.org

ENHANCING INDUSTRIAL EFFICIENCY IN AN INTEGRATED LOT SIZING AND SCHEDULING PROBLEM

Asmaa Djama Maafa and Lamia Triqui Sari*

Abstract. This study investigates an integrated production planning and scheduling problem within a complex industrial environment featuring non-identical parallel production lines. It focuses on reducing the impact of downtime, which incurs substantial setup costs. To this end, we introduce advanced transition mechanisms between production periods, motivated by real-world industrial practices, such as setup carryover and setup crossover, in order to enhance inter-period coordination, reduce operational disruptions, and improve overall system efficiency. A mixed-integer mathematical formulation is developed and solved using CPLEX for medium-scale instances, while a Simulated Annealing metaheuristic is designed to handle large-scale cases. The proposed solution framework yields realistic and cost-efficient plans that jointly minimize downtime-related costs, setup costs, production costs, and inventory holding costs.

Mathematics Subject Classification. 35L05, 35L70.

Received January 1, 2025. Accepted April 6, 2026.

1. Introduction

Production planning and scheduling constitute the core operational functions that determine industrial performance. While planning establishes optimal production quantities and timing (lot sizing) to meet demand at minimal cost, scheduling sequences these jobs on available resources. Traditionally addressed sequentially, this decoupled approach often yields plans that are infeasible to schedule or schedules that are suboptimal due to rigid pre-defined plans, leading to significant operational inefficiencies. To overcome this, the Integrated Lot Sizing and Scheduling Problem (ILSSP) has emerged, simultaneously optimizing both decision levels. Research has extended this problem to real-world features, including multi-item production, parallel machines, sequence-dependent setups, and continuity configurations like setup carryover (SCA) and crossover (SCR).

However, a critical industrial reality remains insufficiently addressed: the impact of production stoppages (downtime/idle time) due to maintenance, breakdowns, or material shortages. In capital-intensive continuous industries (*e.g.*, heavy manufacturing, process industries), restarting a production line after any stop incurs substantial time and cost penalties. Current models, focusing on production continuity (SCA/SCR), fail to effectively model costly restart transitions following a line stoppage.

Keywords. Production planning, lot sizing, scheduling, transitions between periods, metaheuristics.

Manufacturing Engineering Laboratory of Tlemcen (MELT), University of Tlemcen, BP 230 Tlemcen, Algeria.
*Corresponding author: triquilamia@yahoo.fr & lamia.triqui@univ-tlemcen.dz

This work fills this gap by formally studying integrated lot sizing and scheduling with explicit stoppage and costly restart transitions. We focus on heavy industries where each line interruption imposes significant restart costs. Our objective is to determine an optimal production plan that meets demand and capacity constraints while minimizing total costs, which now explicitly include line restart costs, in addition to setup, production, and inventory holding costs.

We ground this study in a real industrial case from CHIALI TUBES, a manufacturer of tubes, where changeovers are costly and generate waste, making meticulous planning essential. To solve this complex problem, we develop a Mixed-Integer Linear Programming (MILP) model and propose a Simulated Annealing (SA) metaheuristic.

The remainder of this paper is structured as follows. Section 2 provides a detailed literature review. Section 3 formally describes the problem, presents the MILP model, and details the SA algorithm. Section 4 discusses the numerical validation and results analysis on the industrial case. Section 5 presents managerial insights. Finally, Section 6 presents conclusions and future research directions

2. Literature review

The lot-sizing problem have been widely studied in the literature since many years ago. It has been studied and applied by several researches in various industries: Marinelli *et al.* [65] in the packaging lines in process industries, Das and Patnaik [12] and Geršak [35] in clothing industry, Garre *et al.* [34] in food industry, Walton [85] in the escape and rescue ladders industry, de Matta and Guignard [14] in tile manufacturing, Jans and Degrave [48] in tire industry, Ghosh *et al.* [13] in plastic injection molding, Dumoulin and Vercellis [25] and Silva and Magalhaes [77] in textile industry, Gupta and Magnusson [42] in paper production, Santos-Meza *et al.* [21] in metallic alloy molding, Maafa Djama *et al.* [20] and Maafa Djama and Triqui [19] in plastic tube industry.

The lot sizing problem with capacity constraints (CLSP) is one of the most widely used problems in industry, since it effectively formulates production systems problems with large product sizes planned over discrete periods and with deterministic demands. Karimi *et al.* [52] presented the factors that significantly influence the formulation and complexity of production plans, which can be expressed as: capacity constraint, nature of demand, number of products, inventory shortage, product deterioration, planning horizon and number of levels. Gicquel *et al.* [36] brought together in their review the different CLSP works and their evolution in the industrial environment, from single-level models with a single resource, through intermediate models, to more complex models with several resources and several levels. Buschkühl *et al.* [74] gave a literature review about the lot sizing problems with capacity constraints in which they had taken into consideration the aspects of modeling approaches and presented the algorithmic solution approaches. Jans and Degraeve [48] were interested to the different extensions of CLSP, considering different supplementary constraints and presenting the different solutions for each proposed extension. Quadt *et al.* [70] reviewed about the problems with extensions related to: sequencing of products, parallel resources, setup carryover and backorders.

Besides the CLSP, there is the scheduling problem which can be of great importance when the setups are sequence-dependent in terms of times and/or costs. It provides the sequencing of the products in each period while respecting capacity constraints. The integration of the lot sizing and the scheduling problem or as it can be expressed in the literature considering the two problems "simultaneously", is more advantageous than when considering each problem separately by providing better production planning and represents a challenge for researchers and entrepreneurs [1, 7, 15, 33]. Maravelias *et al.* [64] discussed in their review about the major approaches of integrating the planning and scheduling which are: (1) detailed scheduling models that consist on injecting the scheduling model as a sub-model into each period of the planning model or on considering one global detailed scheduling model on the planning horizon. This approach leads to a computationally difficult problem. For this first approach, we can cite the work of Sung *et al.* [82]. (2) Relaxation or aggregation of the detailed scheduling model, consisting in creating an approximation of the original model by relaxing certain constraints or grouping certain decisions, as in the works of Erdirik-Dogan *et al.* [26, 27]. And (3) off-line surrogate models that consist of calculating off-line to generate constraints that define the feasible region of the scheduling model, such as the work of Wan et. al. [86]. The literature includes several reviews on the integrated lot sizing and scheduling problem such as the works of Drexl *et al.* [24], Zhu *et al.* [92], Guimarães *et al.* [41], Copil *et al.* [11], Clark *et al.* [10] and Guzman *et al.* [43]. It is shown that simultaneous lot sizing and scheduling problem is motivated by many industrial applications and that there are many formulations that have been developed for when considering one global model.

To get closer to the reality of certain industrial fields and to model production systems more effectively, it is important to consider transition configurations between consecutive periods. In the last decades, several researchers have tend to incorporate setup carryover and/or setup crossover as transition configurations in their works because it makes the models more applicable in a way that gives a true image of the situations considered and try to find better solutions. As far as is known, only Fiorotto *et al.* [29] reviewed and classified researches with setup carryover and setup crossover and analyzed the impact of considering it on lot sizing problems. In their review, they classified works related to considering different forms of transitions between periods, starting with setup crossover, setup carryover, or transitions that consider both simultaneously, in big bucket or small bucket, short or long setup times, and whether considering scheduling or not. In this work, it is shown that considering setup carryover and setup crossover in the capacitated lot sizing problem is significant and more beneficial compared to the classical and standard CLSP assumption, it allows to find feasible and better solutions. Dillenberger *et al.* [16] who were firsts to publish a work that takes into consideration the setup carryover in the production planning problem, had applied a fix-and-relax approach to solve the problem for semi-conductor manufacturing industries. Gopalakrishnan *et al.* [38] were motivated in their research by a production planning problem in paper mills, they presented a framework considering the modelling of the single machine CLSP with setup carryover and extended it into two formulations to include multiple identical machines and tool requirements planning. Gupta and Magnusson [42] had considered scheduling in the capacitated lot sizing problem with single machine, sequence-dependent setup times and costs, they were motivated in their work by a problem confronted by a company that produces sandpaper rolls, had developed a constructive heuristic named ISI to solve the problem and it has three main steps: Initialize, Sequence, and Improve. The first step enables to find an initial feasible solution, the second step premise to find a sequencing with the least cost, and in the last step the solution is refined. Kopanos *et al.* [59] proposed a model for production planning and scheduling of parallel continuous processes for a beer production facility with sequence-independent setups for lots of products in the same family and sequence-dependent setup times and costs for product families. Motivated by a semiconductor assembly facility, Quadt and Kuhn [71] study the production planning and scheduling problem considering setup carryover, setup times, back-orders and parallel machines. They addressed the proposed problem using a heuristic incorporating an aggregate model with integer variables. Zhang *et al.* [91] proposed two mixed integer programming models for the multi-item problem with setup carryover and sequence-dependent setup, and a data-driving branching and selection method to solve the problem.

There are few researches that integrate setup crossover and setup carryover simultaneously in the production planning problem, references [3, 51, 62, 66, 67, 73, 81] in single-level single-resource problems. Carvalho and Nascimento [6], Kopanos *et al.* [59], and Mahdieh *et al.* [63] in single-level parallel-resource problems, [5] in multi-level parallel-resource problem, and [2] in multi-level single-resource problem with the big bucket. Sung *et al.* [81] developed a model that consider sequence independent setup times and costs, long setup times that can take longer than one period of time and non-uniform time periods. They also extended the formulation to model product's families, parallel units, idle time, backlogged demand and lost sales. Menezes *et al.* [66] had considered non-triangular and sequence-dependent setups in their formulation which enforce minimum lot sizes and had developed a method that enable to identify and eliminate disconnected sub tours. Mohan *et al.* [67] extended in their paper the formulation proposed by Suerie and Stadtler [80] modelling the capacitated lot sizing problem with setup carryover, in order to include setup crossover. They had concluded that in 12 out of 18 tested instances, their model with setup crossover and setup carryover presents better solutions or removes infeasibility then the model without setup crossover. Fiorotto *et al.* [28] made a comparison between the formulations of Mohan *et al.* [67] and Menezes *et al.* [66] throw a computational study, and had concluded that the formulation proposed by Menezes *et al.* [66] performs better than the formulation of Mohan *et al.* [67]. Fiorotto *et al.* [28] had also proposed symmetry-breaking constraints that ensure the elimination of alternative solutions that can lead to decrease the resolution time. Belo-Filho *et al.* [3] proposed two formulations that include backlogging besides the setup carryover and crossover, a first one based on model of Suerie and Stadtler [80] for the capacitated lot sizing with setup carryover, hybridized with the capacitated lot sizing with setup carryover, setup crossover and backlogging model of Sung and Maravelias [81]. In the second formulation, a disaggregated preparation variable was used, derived from the classic reformulation of Krarup and Bilde [60], that trace back the starting and termination times of the setup operation. The two formulations had been compared to the formulation proposed by Sung and Maravelias [81] and the impact of considering setup crossover was shown. Ramya *et al.* [73] proposed two mathematical models and had considered two situations: in the first situation the setup and holding costs are time independent and product dependent, and in the second one, those costs are time dependent and product dependent. The mathematical models were compared to the model proposed by Mohan *et al.* [67] where it was shown throw an example that this last one gives infeasible solutions. Kang [51] proposed a mathematical formulation with triangular and sequence-dependent setup times and setup carryover and crossover. In his formulation, he used an exponential number of generalized subtour elimination constraints (GSECS) for which he employed a separation routine to generate the violated GSECs. Y. Lee and K. Lee [62] used an approximate dynamic programming algorithm to solve the integrated lot sizing and scheduling problem considering sequence-dependent setups, setup carryover and setup crossover. Kopanos *et al.* [59] presented a model that handles the backlogging and product families' aspects where the products that share the same characteristics can be classified into families. The concept assumes that the setup times and costs are sequence-independent for products that belong to the same family and are sequence-dependent for product families. They had shown that the model leads to better solutions when tested on a real-world problem compared to commercial tools. Mahdieh *et al.* [63] presented a formulation that considers non-triangular and sequence-dependent setup times and costs and enforces minimum lot sizes for products. The formulation was first developed for single machine then extended to flexible flow lines and parallel machines. It has been shown through computational tests that the model with setup carryover and setup crossover offers greater flexibility and effectiveness to the lot sizing model. Carvalho and Nascimento [6] considered the integrated lot sizing and scheduling problem with parallel machines, setup carryover and non-triangular sequence-dependent setup times and costs. They proposed a combination of heuristics to solve the problem. Camargo *et al.* [5] addressed the lot sizing and scheduling problem for two-stage manufacturing processes where they proposed three formulations, one of these takes into account setup carryover and setup crossover. In this study, it was concluded throw computational tests that this formulation provides more flexibility to incorporate setup-related features despite the worst performance delivered in terms of CPU times compared to the two other formulations proposed. Behnamian *et al.* [2] considered the multi-level lot sizing and scheduling problem with sequence-dependent family setup times and costs and setup carryover. A lower bound and a Genetic Algorithm were applied to solve the problem with large-size instances.

In the literature, to our knowledge, other transition configurations related to production line and product launches after stop-times have not been explicitly considered and detailed. Table 1 lists classification of papers considering transition configurations, mainly setup carryover (SCA) and setup crossover (SCR). It is inspired by the work of Fiorotto *et al.* [29] and is completed to include works up to 2022.

Furthermore, recent works have broadened the perspective by explicitly integrating demand uncertainty and developing hybrid metaheuristic approaches to solve more realistic lot-sizing and scheduling problems. For instance, stochastic and robust models have been proposed to manage uncertain demand in multi-level, multi-period environments [61,76]. A machine-learning rolling horizon heuristic [89] and hybridized decomposition models incorporating emissions control [9] exemplify advanced frameworks tackling these complexities. In parallel, hybrid metaheuristics and matheuristics have demonstrated superior efficacy. For example, the PSNS-GAII algorithm [53], cooperative coevolutionary hyper-heuristics [90], matheuristics for integrated production [31], and comparisons of algorithms like GA, WOA, and PSO for problems with remanufacturing [68] have proven effective for bi-objective optimization and problems with complex constraints like sequence-dependent

TABLE 1. Classification of papers considering transition configurations.

Author(s)	Transition configurations	Scheduling	Bucket Big	Bucket Small	Level multi (m)/single (s)	Machine parallel (p)/single (s)	Resolution approach heuristic (h)/exact (e)	Domain
Dillenberger *et al.* [16]	SCA	x	x		s	p	h/fix-and-relax approach	manufacturing
Haase [44]	SCA		x		s	s	h/randomized-regret-based	general
Drexl and Haase [22]	SCA	x		x	s	p	h/randomized-regret-based	general
Gopalakrishnan *et al.* [38]	SCA		x		s	p	e/solver	paper
Drexl and Haase [23]	SCA	x		x	s	s	h/randomized-regret-based	general
Kimms [54]	SCA	x		x	m	s	h/randomized-regret-based	general
Kimms [55]	SCA	x		x	m	s	h/randomized-regret-based	general
Haase [45]	SCA		x		s	s	h/randomized-regret-based	general
Sox and Gao [78]	SCA		x		s	s	h/Lagrangian decomposition	general
Kimms [56]	SCA	x		x	m	p	h/genetic algorithm	general
Gao [32]	SCA		x		s	s	h/period-by-period	general
Gopalakrishnan [37]	SCA		x		s	s	e/solver	general
Gopalakrishnan *et al.* [39]	SCA		x		s	s	h/tabu-search	general
Porkka *et al.* [69]	SCA		x		s	s	h/post-optimal modification	general
Suerie and Stadtler [80]	SCA		x		m	s	h/time-oriented decomposition	general
Gupta and Magnusson [42]	SCA	x	x		s	s	h/constructive	sandpaper
Suerie [79]	SCR			x	s	s	e/solver	general
Sung and Maravelias [81]	SCA + SCR		x		m	s	e/solver	chemical
Tempelmeier and Buschkuhl [84]	SCA		x		m	p	h/Lagrangian relaxation	general
Kaczmarczyk [49]	SCR			x	s	s	e/solver	general
Quadt and Kuhn [71]	SCA	x	x		s	p	h/period-by-period	semiconductor
Sahling *et al.* [74]	SCA		x		m	s	h/fix-and-optimize	general
Menezes *et al.* [66]	SCA + SCR	x	x		s	s	e/solver	general
Kopanos *et al.* [59]	SCA + SCR	x	x		s	p	e/solver	beverage
Wu and Shi [87]	SCA		x		m	p	h/relax-and-fix	general
Camargo *et al.* [5]	SCA + SCR	x	x		m	p	e/solver	general
Goren *et al.* [40]	SCA		x		s	s	h/hybrid approach	general
Mohan *et al.* [67]	SCA + SCR		x		s	s	e/solver	general
Wu [88]	SCA		x		m	p	h/time-oriented decomposition	general
Kaczmarczyk [50]	SCR	x		x	s	p	e/solver	general
Belo-Filho *et al.* [3]	SCA + SCR		x		s	s	e/solver	general
Ramya *et al.* [73]	SCA + SCR		x		s	s	e/solver	general
Fiorotto *et al.* [28]	SCR		x		s	s	e/solver	general
Mahdieh *et al.* [63]	SCA + SCR	x	x		s	p	e/solver	general
Kang [51]	SCA + SCR		x		s	s	h/Branch-and-Cut Algorithm	general
Carvalho and Nascimento [6]	SCA + SCR	x	x		s	p	h/hybrid approach	general
Zhang *et al.* [91]	SCA	x	x		s	p	h/Data-driven branching and selection algorithm	general
Behnamian *et al.* [2]	SCA + SCR	x	x		m	s	h/genetic algorithm	general
Y. Lee and K. Lee [62]	SCA + SCR	x	x		s	s	h/approximate dynamic programming algorithm	general

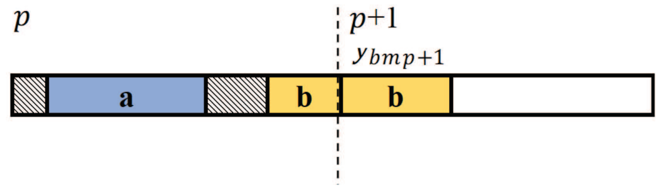


FIGURE 1. Transition case 1.

setups. Earlier applications, such as in plastic injection molding [58], underscore the practical relevance of these hybrid approaches. These advances collectively highlight the critical role and growing sophistication of hybrid, stochastic, and data-driven methods in contemporary production planning and scheduling research.

Contribution

The existing literature highlights several significant gaps in the modeling and optimization of complex production systems. In particular, few studies comprehensively address the costs and time associated with relaunching production lines after downtime, despite their substantial impact on industrial performance. Moreover, researches generally focus on isolated concepts of production continuity, such as setup carryover or setup crossover, with very few studies addressing these two notions simultaneously, as shown in Table 1. Furthermore, these studies fail to account for other possible transition configurations between consecutive periods, especially those related to end-of-period downtimes (idles times) and production line relaunches. These limitations reduce the practical applicability of existing approaches in real industrial contexts.

This research stands out for its ambition to bridge existing gaps by proposing an innovative decision-support method. This method is built on two complementary pillars:

(1) A new mathematical model: Designed to accurately represent the complexity of the studied production systems, explicitly integrating line launches and sequence-dependent setups during transitions between periods in various forms.
(2) A performant metaheuristic inspired from the Simulated Annealing algorithm: Adapted to efficiently solve the joint optimization problem of production planning and scheduling.

One of the key contributions of this work lies in the simultaneous integration of five distinct transition configurations between consecutive periods within the developed model as illustrated[1] in Figures 1–5 below.

Case 1: Setup carryover
The setup is carried over from period p to period $p+1$ without requiring a new setup in $p+1$. Production continues seamlessly across periods.

Case 2: Setup crossover
Setup from product a to product b starts in period p and finishes in $p+1$. The associated setup cost is incurred in period $p+1$, covering both tools change and product launch.

Case 3: Manufacturing continuity with setup at period start
Production of product a ends at the close of period p. At the start of $p+1$, a setup for product b begins without idle time or line launch cost.

Case 4: Product change with line launch
After idle time at the end of period p, period $p+1$ begins with a line launch and setup for a new product b. Costs include both line launch and product setup which involves tools change and product launch.

[1]The notation used for the transition configurations corresponds to the notation employed for the developed mathematical model: LC_m is the production line launch cost, SC_{ab} is the setup cost for the change from a to b, $SC0_a$ is the setup cost for launching the item a. y_{bmp+1}, z_{abmp+1}, w_{abmp+1} and w_{aamp+1} are the transition variables.

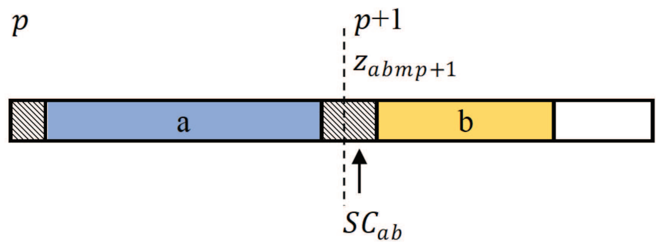


Figure 2. Transition case 2.

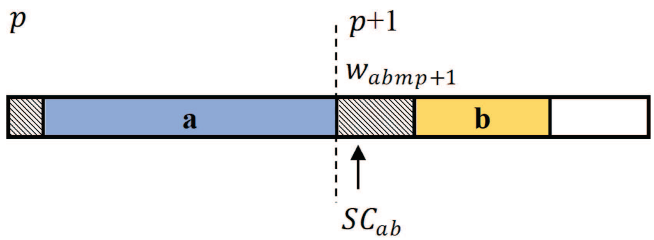


Figure 3. Transition case 3.

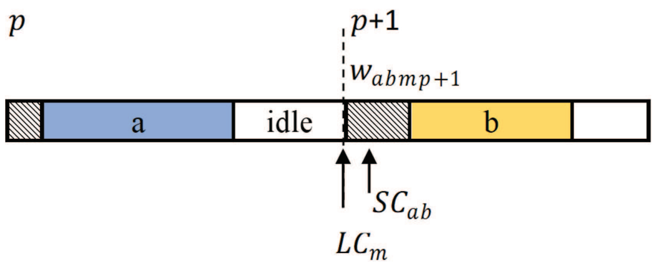


Figure 4. Transition case 4.

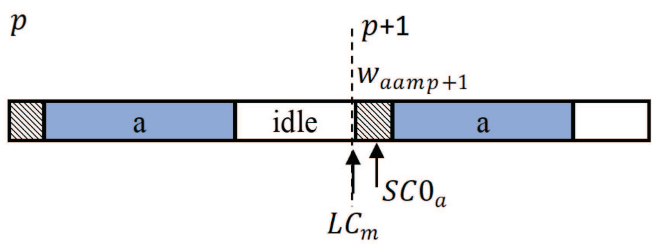


Figure 5. Transition case 5.

Case 5: Relaunch of same product after idle time

Following idle time, period $p+1$ starts with a line launch and setup for the same product a as in period p. Costs include line launch and product relaunch.

These configurations, rarely or never simultaneously addressed in the literature, reflect the complexity of real-world industrial scenarios. Their modeling and optimization within a unified framework represent a significant scientific advancement.

Furthermore, this work highlights the importance of interactions between production planning and scheduling in production systems. Through its application to a real case study, it demonstrates the effectiveness of the proposed method in improving the overall performance of the studied system by reducing operational costs while maximizing resource utilization.

Highlights

- Development of a mathematical model simultaneously incorporating five configurations of sequence-dependent setups in transitions between consecutive periods, including setup carryover, setup crossover, production continuity with changes at the beginning of the period, product relaunch after idle time, and product change with line launch.
- Proposal of an optimization method combining production planning and scheduling in environments where production line launch costs after idle times are critical.
- Design of a metaheuristic based on Simulated Annealing to efficiently solve complex problems and reduce operational costs.
- Validation of the approach through real-world case study, demonstrating significant results.
- A theoretical and practical advancement addressing the requirements of complex production systems with realistic and costly transitions.

3. Problem statement

In this work, we study a production planning and scheduling problem in a production unit that manufactures several types of products by adopting the principle of lot sizing (each type is produced by lot). Each product type can be manufactured on several production lines (m lines) where each line works with a specific and appropriate rate for the product type, to meet varying and periodic demands on a given time horizon, divided into several periods. We are in the framework of a multi-line, multi-product and multi-period production unit that takes into account all the possible transitions between consecutive periods in order to establish the best management of the production unit that satisfies the customer's demands in the expected time, reducing waiting, non-productive times, and production costs.

In order to analyze the posed problem, we have proposed a model formulated by a Mixed Integer Linear Programming (MILP) and solved by two different approaches; an exact method and a metaheuristic. In terms of exact method, the proposed model was implemented under the CPLEX solver, while for metaheuristics, simulated annealing was applied on large instances. Several configurations have been considered in each method. Configurations are obtained by varying the different parameters belonging to production lines, products and periods. Different formulations and configurations are detailed in part 3.1 (CPLEX modeling), part 3.2 (implementation of the metaheuristic) and part 4 (experiments). The objective is to propose an optimal planning and scheduling plan that takes into account all possible transitions between consecutive periods while minimizing all costs. As a framework for applying our proposed approach, we have chosen the company CHIALI TUBES, which manufactures a wide variety of polyethylene tubes on several production lines. This company serves as a digital support for data that can be used in experimentation.

Tube production is characterized by its continuous flow of materials through production lines where they can operate 24 h a day, 7 days a week. Each production line has a launch cost and a specific capacity; it can produce a range of product type with a specific speed for each type. Each final product is characterized by its diameter and thickness, a production time and cost on each production line. Thus, the setup of an item depends on the previous item in the production line Hence, the manufacturing of multi-items in this industry domain constraints the production workshop to be multi-lines and the setups to be sequence-dependent. So, we consider in this paper a Multi-Items Multi-Lines Capacitated Lot Sizing Problem with Sequence dependent setups, with different configurations for the transition between consecutive periods, including Setup Carryover and Setup Crossover. In order to modeling system, we consider the following assumptions:

- The planning horizon is made up of several equitable periods of the same length, noted T.
- Demand is deterministic and known in advance, with the condition of being satisfied at each period.
- The storage capacity is not limited.
- The workshop is organized in non-identical parallel lines with different capacities.
- The production flow is continuous and can extend over several periods without interruption. This allows the possibility of a setup carryover or a setup crossover between consecutive periods.
- There are several different items that can be produced in different production lines.
- Each production line can manufacture certain types of items.
- Each item has a production rate in each production line, which leads to different production costs and times on the different production lines.
- We can have idle time in a period that can be equal or less than the length of the period.
- After an idle time, the production line is relaunched for manufacturing.
- When a production line is launched in a period, it starts with a setup for a product.
- The setup time and cost of an item depend on the item produced before.
- An item can be the first launched at period after being the last one manufactured at a previous period with an idle time left at the end of the previous period. In this case, the setup cost is less than the setup cost when realizing a change from a different item.
- A period can start with a setup between different items without needing a launch of the production line and this is in the case where there was no idle time at the previous period. It is considered as a kind of production continuity.

Given the structural complexity of the problem, it is important to note that even simplified versions of capacitated lot sizing are computationally challenging. Bitran and Yanasse [4] and Florian *et al.* [30] classified the single-item capacitated lot sizing problem as NP-hard, while Chen *et al.* [8] demonstrated that the multi-item version with independent setups is strongly NP-hard. The problem addressed here; which integrates multi-item, multi-production line, sequence-dependent setups, and inter-period transitions, is therefore strongly NP-hard. This classification justifies the adoption of a dual solution strategy: an exact MILP formulation for small to medium-scale instances, and a metaheuristic approach for larger, practically sized problems.

3.1. Mathematical formulation

The objective in this study is to determine the production quantities of items on the different production lines at each period of the planning horizon while minimizing production costs, setup costs and holding costs. To begin with, we will list the different used parameters, and then the objective function is given, followed by the different constraints relating to the studied system.

3.1.1. Parameters

I	Set of items.
M	Set of production lines.
P	Set of periods.
i, j	Indexes of items, $i, j \in I$
m	Index of production lines, $m \in M$
p	Index of periods, $p \in P$
R	A large number.
K_{im}	A Boolean matrix that states, for each item i, the possibility to be produced on the production line m.
D_{ip}	Demand of item i at period p.
LC_m	Launch cost of production line m.
PC_{imp}	Production cost by item i on the production line m at period p.
SC_{ij}	Setup cost from item i to item j.
$SC0_i$	Setup cost of item i at the beginning of a period if the item was the last produced at the previous period.
HC_i	Cost of holding item i in inventory.
C_{mp}	Production capacity of the production line m at period p in time unit.
PT_{im}	Production time of item i on production line m.
ST_{ij}	Setup time from item i to item j.
$ST0_i$	Setup time of item i at the beginning of a period if the item was the last produced at the previous period.

3.1.2. Decision variables

q_{imp}	The amount of production of item i on production line m at period p.
s_{ip}	The inventory level of item i at the end of period p.
r_{imp}	$= 1$, item i is produced on production line m in period p, $= 0$, otherwise
x_{ijmp}	$= 1$, there is a setup from item i to item j on production line m at period p $= 0$, otherwise
y_{imp}	$= 1$, the setup state of item i is carried over from period $p-1$ to the beginning of period p on production line m, $= 0$, otherwise
w_{ijmp}	$= 1$, the production line m starts at period p with a setup from item i to item j, $= 0$, otherwise
z_{ijmp}	$= 1$, the setup from item i to item j is split between period $p-1$ and period p on production line m, $= 0$, otherwise
f_{ijmp}	The time part at the start of period p of the setup from item i to item j that is split between period $p-1$ and period p on production line m.
l_{ijmp}	The time part at the end of period p of the setup from item i to item j that is split between period p and period $p+1$ on production line m.
idle_{mp}	The idle time at the end of period p on production line m.
idle_b_{mp}	$=1$, there is an idle time on production line m at the end of period p, $= 0$, otherwise
α_{imp}	$= 1$, item i is produced first at the beginning of period p on production line m, $= 0$, otherwise

β_{imp}	$= 1$, item i is produced last on production line m at the end of period p, $= 0$, otherwise
u_{imp}	An integer variable to track the sequence of item i on production line m at period p.
lc_{mp}	$= 1$, the production line m is launched at period p, $= 0$, otherwise
sauv_{imp}	$= 1$, item i is the last one produced on production line m until period p, $= 0$, otherwise

3.1.3. Objective function

$$\text{Min } F_1 + F_2 + F_3 + F_4 + F_5$$

$$F_1 = \sum_{i \in I} \sum_{m \in M} \sum_{p \in P} PC_{imp}\, q_{imp} \tag{1.1}$$

$$F_2 = \sum_{i \in I} \sum_{m \in M} \sum_{p \in P} SC0_i\, w_{iimp} \tag{1.2}$$

$$F_3 = \sum_{i \in I} \sum_{\substack{j \in I \\ j \neq i}} \sum_{m \in M} \sum_{p \in P} SC_{ij}\,(x_{ijmp} + w_{ijmp} + z_{ijmp}) \tag{1.3}$$

$$F_4 = \sum_{m \in M} \sum_{p \in P} LC_m\, \mathrm{lc}_{mp} \tag{1.4}$$

$$F_5 = \sum_{i \in I} \sum_{p \in P} HC_i\, s_{ip} \tag{1.5}$$

The objective function minimizes the total cost composed of five functions (1.1)–(1.5), function (1.1) corresponds to the production cost of all the products in the production lines over the planning horizon. function (1.2) is related to the total products launch cost, function (1.3) represents the sequence dependent setup cost. functions (1.4) and (1.5) are the production line launch cost and products holding cost respectively.

3.1.4. Constraints

Inventory and production capacity constraints

$$s_{ip} = s_{i(p-1)} + \sum_{m \in M} q_{imp} - D_{ip} \qquad \forall i, \forall p > 1 \tag{2}$$

$$s_{i1} = \sum_{m \in M} q_{im1} - D_{i1} \qquad \forall i \tag{3}$$

$$\begin{aligned} &\sum_{i \in I} q_{imp} PT_{im} + \sum_{i \in I} \sum_{\substack{j \in I \\ j \neq i}} (x_{ijmp} + z_{ijmp} + w_{ijmp}) \cdot ST_{ij} \\ &+ \sum_{i \in I} w_{iimp} ST_{0i} + \sum_{i \in I} \sum_{\substack{j \in I \\ j \neq i}} (f_{ijmp} + l_{ijmp}) + \mathrm{idle}_{mp} = C_{mp} \qquad \forall m, \forall p \end{aligned} \tag{4}$$

Equation (2) and (3) are material balance constraints ensuring the satisfaction of demand. Constraint (4) denotes the capacity constraint in time unit by production line and by period.

Variable linking constraints

$$q_{imp} \leq r_{imp} \cdot \sum_{p \in P} D_{ip} \qquad \forall p, \forall i, \forall m \tag{5}$$

$$r_{imp} \leq K_{im} \cdot q_{imp} \qquad \forall p, \forall i, \forall m \tag{6}$$

$$r_{imp} = y_{imp} + w_{iimp} + \sum_{j \in I, j \neq i} (x_{jimp} + z_{jimp} + w_{jimp}) \qquad \forall p, \forall i, \forall m \tag{7}$$

Constraints (5), (6) and (7) explicit the relationship between the different variables. For a given product, the binary variable r_{imp} takes 1 if there is a produced quantity on a production line that can manufacture it, and this is either following a setup within the period, a setup at the start of the period, a setup starting at the end of the previous period, or by carrying over the production from the previous period.

Transition logic constraints

$$\sum_{i \in I} y_{imp} + \sum_{i \in I} w_{iimp} + \sum_{i \in I} \sum_{\substack{j \in I \\ j \neq i}} z_{jimp} + \sum_{i \in I} \sum_{\substack{j \in I \\ j \neq i}} w_{jimp} \leq 1 \qquad \forall m, \forall p \tag{8}$$

$$\sum_{i \in I} y_{imp} + \sum_{i \in I} w_{iimp} + \sum_{i \in I} \sum_{\substack{j \in I \\ j \neq i}} z_{jimp} + \sum_{i \in I} \sum_{\substack{j \in I \\ j \neq i}} w_{jimp} \geq \frac{\sum_{i \in I} r_{imp}}{R} \qquad \forall m, \forall p \tag{9}$$

Constraints (8) is about the transition between two consecutive periods. Among the different possible cases in the transition, only one of them can occur at a time separately. By constraint (9), we ensure that one of the transition cases is possible if and only if there is at least one product that is manufactured at the period.

Setup crossover constraints

$$f_{ijmp} + l_{ijmp} = z_{ijmp}\ ST_{ij} \qquad \forall p > 1, \forall i, \forall j \neq i, \forall m \tag{10}$$

$$f_{ijm1} = 0 \qquad \forall i, \forall j \neq i, \forall m \tag{11}$$

$$f_{ijmP} = 0 \qquad \forall i, \forall j \neq i, \forall m \tag{12}$$

Constraints (10), (11) and (12) concern the setup crossover, ensuring that the first period can't start with a split setup, in addition the last period can't finish with a split setup too.

Setup carryover constraints

$$\begin{aligned} y_{imp} \leq\ & y_{imp-1} + w_{iimp-1} + \sum_{\substack{j \in I \\ j \neq i}} z_{jimp-1} + \sum_{\substack{j \in I \\ j \neq i}} w_{jimp-1} \\ & + \sum_{\substack{j \in I \\ j \neq i}} x_{jimp-1} \qquad \forall p > 2, \forall i, \forall m \end{aligned} \tag{13}$$

Constraint (13) concerns the setup carryover between two consecutive periods.

Production line activity status and launch constraints

$$\mathrm{idle}_{mp} \leq R \cdot \mathrm{idle\_b}_{mp} \qquad \forall m, \forall p \tag{14}$$

$$\mathrm{idle\_b}_{mp} \leq \mathrm{idle}_{mp} \qquad \forall m, \forall p \tag{15}$$

$$\mathrm{lc}_{mp} \leq \mathrm{idle\_b}_{mp-1} \qquad \forall m, \forall p > 1 \tag{16}$$

$$\mathrm{lc}_{mp} \geq \mathrm{idle\_b}_{mp-1} + \frac{\sum_{i \in I} r_{imp}}{R} - 1 \qquad \forall m, \forall p > 1 \tag{17}$$

$$\mathrm{lc}_{m1} \geq \frac{\sum_{i \in I} r_{im1}}{R} \qquad \forall m \tag{18}$$

Constraints (14) and (15) define the Boolean variable $\mathrm{idle\_b}_{mp}$ which takes one if there is an idle time on the production line m at the end of the period p, and zero otherwise. Constraints (16), (17) and (18) govern the production lines launches. For the first period, a production line is launched if there is at least one product that is manufactured on it. For subsequent periods, it is launched if, in addition to having a product undergoing setup, the line experienced idle time in the previous period. Constraint (16) makes sure that if no idle time is observed at the end of a period, there is no need to relaunch the production line for the next period.

First product of period constraints

$$\sum_{i \in I} \alpha_{imp} \leq 1 \qquad \forall p, \forall m \tag{19}$$

$$\alpha_{imp} \leq r_{imp} \qquad \forall p, \forall i, \forall m \tag{20}$$

$$\alpha_{imp} \geq y_{imp} + w_{iimp} + \sum_{\substack{j \in I \\ j \neq i}} (z_{jimp} + w_{jimp}) \qquad \forall p, \forall i, \forall m \tag{21}$$

$$\sum_{\substack{j \in I \\ j \neq i}} x_{jimp} \leq r_{imp} - \alpha_{imp} \qquad \forall p, \forall i, \forall m \tag{22}$$

Constraints (19)–(22) define the conditions for determining the first product manufactured on each production line during a given period. They state the relationship with the transition variables and the sequence dependent setup variable.

Last product of period constraints

$$\sum_{i \in I} \beta_{imp} \leq 1 \qquad \forall p, \forall m \tag{23}$$

$$\beta_{imp} \leq r_{imp} \qquad \forall p, \forall i, \forall m \tag{24}$$

$$\sum_{\substack{j \in I \\ j \neq i}} x_{ijmp} \geq r_{imp} - \beta_{imp} \qquad \forall p, \forall i, \forall m \tag{25}$$

$$\beta_{imp-1} \geq y_{imp} + \sum_{\substack{j \in I \\ j \neq i}} z_{ijmp} \qquad \forall i, \forall m, \forall p > 1 \tag{26}$$

$$\sum_{\substack{j \in I \\ j \neq i}} r_{jmp} \leq (2 - \alpha_{imp} - \beta_{imp}) \cdot (|I| - 1) \qquad \forall p, \forall i, \forall m \tag{27}$$

Constraints (23)–(27) define the conditions for determining the last product manufactured on each production line during a given period. They state the relationship with the continuity variables and the sequence dependent setup variable. Constraint (27) ensures that, if a product is the first one and at the same time the last one in a period, so there is no other product that can be manufactured at the period.

Sequencing constraints

$$u_{imp} \leq r_{imp} \cdot N \qquad \forall p, \forall i, \forall m \tag{28}$$

$$u_{imp} \geq r_{imp} \qquad \forall p, \forall i, \forall m \tag{29}$$

$$u_{imp} \leq \sum_{i \in I} \sum_{\substack{j \in I \\ j \neq i}} x_{ijmp} + 1 \qquad \forall p, \forall i, \forall m \tag{30}$$

$$u_{imp} + 1 \leq u_{jmp} + |I| \cdot (1 - x_{ijmp}) \qquad \forall p, \forall i, \forall j \neq i, \forall m \tag{31}$$

Constraints (28)–(31) enable the tracking of each product's position on production lines across all periods, ensuring proper sequencing and the elimination of subtours throughout the planning horizon. In the proposed model, each product is processed only once per period to optimize operations. This approach groups all units of the same product type together, launching them only once to minimize setup times and prevent sequencing loops within a single period. These constraints are critical to eliminating such scenarios.

$$\sum_{i \in I} \mathrm{sauv}_{imp} \leq 1 \qquad \forall m, \forall p \tag{32}$$

$$\mathrm{sauv}_{imp} \geq \beta_{imp} + r_{imp} - 1 \qquad \forall p, \forall i, \forall m \tag{33}$$

$$\mathrm{sauv}_{imp} \leq \beta_{imp} - r_{imp} + 1 \qquad \forall p, \forall i, \forall m \tag{34}$$

$$\mathrm{sauv}_{imp} \geq \mathrm{sauv}_{imp-1} - \frac{\sum_{j \in I} r_{jmp}}{R} \qquad \forall i, \forall m, \forall p > 1 \tag{35}$$

$$\sum_{t=1}^{p-1} \mathrm{lc}_{mt} \geq t_{mp} \qquad \forall m, \forall p > 1 \tag{36}$$

$$\sum_{t=1}^{p-1} \mathrm{lc}_{mt} \leq (p-1) \cdot t_{mp} \qquad \forall m, \forall p > 1 \tag{37}$$

$$\mathrm{sauv}_{imp-1} \geq w_{ijmp} + \alpha_{jmp} - 1 \qquad \forall i, \forall j, \forall m, \forall p > 1 \tag{38}$$

$$\sum_{i \in I} \mathrm{sauv}_{im1} \geq \sum_{i \in I} r_{im1} \qquad \forall m \tag{39}$$

Constraints (32)–(39) manage the tracking of the last product manufactured across periods. To formulate the logical condition in constraints (36)–(38), a binary auxiliary variable t_{mp} and v_{mp} is introduced, where $t_{mp} \in \{0, 1\}$. Constraint (32) ensures that at most one product is recorded as the last one produced up to a given period. Constraints (33) and (34) link the last product produced in p to the overall production sequence, distinguishing it from other items manufactured in p. Constraints (35)–(39) handle continuity: they save the last product from p-1 when p has no production, relate setups in p to the last product from p-1, and ensure that no product is recorded if the first period has no production.

Period-start setup initialization constraints

$$\sum_{i \in I} \sum_{j \in I} w_{ijmp} \geq \mathrm{lc}_{mp} \qquad \forall m, \forall p > 1 \tag{40}$$

$$\sum_{i \in I} w_{iim1} \geq \mathrm{lc}_{m1} \qquad \forall m \tag{41}$$

$$\sum_{i \in I} w_{iimp} \geq \mathrm{lc}_{mp} + \sum_{t=1}^{p-1} v_{mt} - p \cdot \sum_{t=1}^{p-1} \mathrm{lc}_{mt} \qquad \forall i, \forall m, \forall p > 1 \tag{42}$$

$$v_{mt} \geq r_{imt} \qquad \forall i, \forall m, \forall t = 1, \ldots, p-1 \tag{43}$$

$$v_{mt} \leq \sum_{i \in I} r_{imt} \qquad \forall m, \forall t = 1, \ldots, p-1 \tag{44}$$

$$w_{iimp} \leq \mathrm{lc}_{mp} \qquad \forall i, \forall m, \forall p > 1 \tag{45}$$

Constraints (40)–(45) address setups at the beginning of a period. To formulate the logical condition in constraints (42)–(44), a binary auxiliary variable v_{mp} is introduced, where $v_{mp} \in \{0, 1\}$. Constraints (42)–(44) ensure that a production line launch starts with a product setup. Additionally, Constraint (45) prevents product setups at the start of a period p¿1 unless the line is launched. Together with other constraints (*e.g.*, Eqs. 9 and 17), this guarantees that setups align with transitions, such as setup carryover or crossover, without idle time at the previous period.

Variable domain constraints

$$q_{imp} \in \mathbb{N}^+ \qquad \forall p, \forall i, \forall m \tag{46}$$

$$s_{ip} \in \mathbb{N}^+ \qquad \forall i, \forall p \tag{47}$$

$$\mathrm{idle}_{mp} \in \mathbb{R}^+ \qquad \forall m, \forall p \tag{48}$$

$$r_{imp}, y_{imp}, \alpha_{imp}, \beta_{imp}, \mathrm{sauv}_{imp} \in \{0, 1\} \qquad \forall p, \forall i, \forall m \tag{49}$$

$$x_{ijmp}, z_{ijmp}, w_{ijmp} \in \{0, 1\} \qquad \forall p, \forall i, \forall j, \forall m \tag{50}$$

$$f_{ijmp}, l_{ijmp} \in \mathbb{R}^+ \qquad \forall p, \forall i, \forall j, \forall m \tag{51}$$

$$\mathrm{lc}_{mp}, \mathrm{idle\_b}_{mp} \in \{0, 1\} \qquad \forall m, \forall p \tag{52}$$

$$u_{imp} \in \mathbb{N}^+ \qquad \forall p, \forall i, \forall m \tag{53}$$

Constraints (46)–(53) are about the definition of the variables. q_{imp}, s_{ip}, and u_{imp} are positive integer variables. idle_{mp}, f_{ijmp}, and l_{ijmp} are non-integer positive variables. r_{imp}, x_{ijmp}, y_{imp}, z_{ijmp}, w_{ijmp}, lc_{mp}, α_{imp}, β_{imp}, u_{imp}, sauv_{imp}, and $\mathrm{idle\_b}_{mp}$ are Boolean variables.

Given the strongly NP-hard nature of the problem, a metaheuristic has been developed to efficiently solve large-size instances. This metaheuristic is presented in detail in the following section. For validation purposes, solutions for small- and medium-size instances obtained via the metaheuristic are compared with those derived from the exact resolution of the mathematical model using the CPLEX solver.

3.2. Metaheuristic: simulated annealing

Through works of James *et al.* [47] and Carvalho and Nascimento [6], it has been shown that local-search based methods can offer good performances in solving problems and are efficient for improving the quality of the solutions. Based on these results, we have opted for the choice of the local search algorithm "Simulated Annealing" for our work.

Simulated Annealing (SA) was introduced by Scott Kirkpatrick [57] in 1983. It is one of the most widely used meta-heuristics for solving combinatorial optimization problems [46, 72, 75, 83]. The Simulated Annealing (SA) algorithm is a local search method inspired by thermodynamics. It starts with a first solution, an initial temperature step accompanied by an acceptance probability. For each cooling temperature, a set of iterations is generated, and for each iteration a neighborhood solution is given and will be tested on the objective function.

This solution becomes the new current solution if it improves the objective function or if its probability is greater than the acceptance probability. Otherwise, the new solution is rejected, while the solution from the previous iteration is maintained. For our work, the original algorithm has been modified in such a way that at each temperature level, an initial solution is generated and the best solution is saved at each iteration. This solution is then compared with the global solution. If the best solution at each temperature level is superior to the global solution, it becomes the new global solution. The aim of this process is to increase the probability of exploring the entire search space and obtaining the best results. The steps of the modified simulated annealing algorithm used in our work are detailed in Algorithm 1

Algorithm 1: Simulated annealing.

Inputs: t_0: initial temperature, max: maximum number of iterations, t_f: final temperature
Outputs: solution

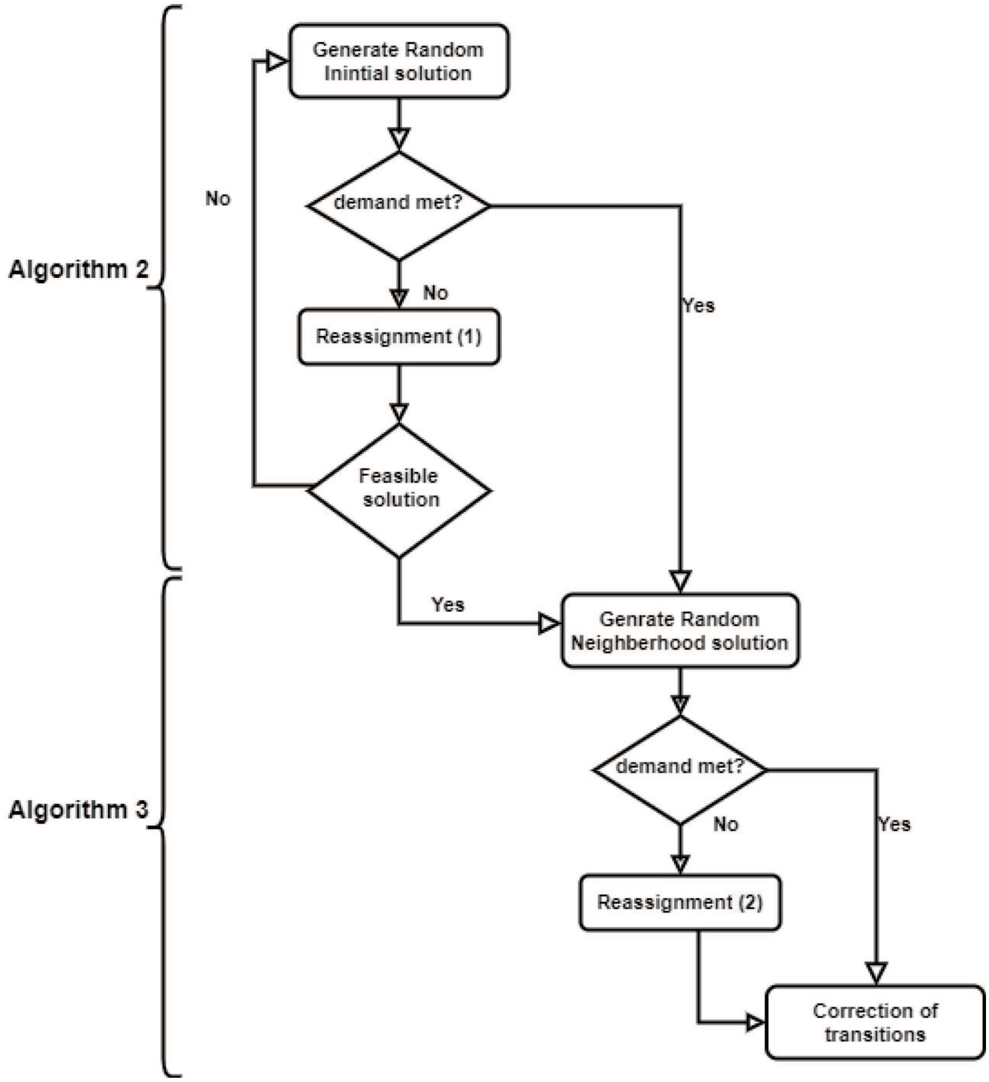
1. $t \leftarrow t_0$
2. iter $\leftarrow 0$
3. Solution $\leftarrow$ Genrate_Random_Solution (Dip, Cmp)
4. while ($t < t_f$) do
    (a) $S_0 \leftarrow$ Genrate_Random_Solution (Dip, Cmp)
    (b) $S \leftarrow S_0$
    (c) If (fitness(S) < fitness(Solution)) then Solution $\leftarrow S$
    (d) while (iter < max) do
        iter++
        $s\_neighborhood \leftarrow$ Genrate_Random_Neighberhood(S)
        $\Delta \leftarrow$ fitness($s\_neighborhood$) $-$ fitness(S)
        If ($\Delta < 0$) then $S \leftarrow s\_neighborhood$
            If (fitness($s\_neighborhood$) < fitness(Solution)) then Solution $\leftarrow s\_neighborhood$
        Else
            $x \leftarrow$ random
            if ($x < e^{-\Delta/t}$) then $S \leftarrow s\_neighborhood$
5. $t \leftarrow t \times \alpha$
6. iter $\leftarrow 0$
7. Return Solution

The adaptation of the algorithm is governed by its used parameters which differ from one application to another and from one case to another. In our work, the modified SA method is divided into several distinct steps. These steps are outlined below, with the objectives of each step described sequentially.

To provide a clearer understanding of our resolution approach, Figure 6 illustrates the connections between the proposed steps.

The generation of a random initial solution defines a plan and schedule for different products across production lines and all periods of the planning horizon. Based on this solution, a test is conducted: if the demand is not satisfied within a period, it is reallocated accordingly. At this stage, two outcomes are possible. Either reallocation is not feasible, prompting the generation of a new random solution, or reallocation succeeds, yielding a feasible initial solution, as detailed in Algorithm 2.

From this initial solution, a neighborhood solution is determined and its feasibility is verified. If the demand is not satisfied, reallocation is carried out, and corrections to transitions between consecutive periods may be necessary, as detailed in Algorithm 3. The process is repeated for each temperature level, and the optimal solution is identified by comparing the results of all iterations.



H

Figure 6. The different steps of the proposed method.

3.2.1. Initial solution generation

Algorithm 2: Generate_Random_Solution.

Inputs: D_{ip}, C_{mp}
Outputs: Random solution S ($w_{ijmp}, x_{ijmp}, y_{imp}, z_{ijmp}, q_{imp}, r_{imp}, s_{ip}, f_{ijmp}, l_{ijmp}, \mathrm{lc}_{mp}$, $\alpha_{imp}, \beta_{imp}, u_{imp}, \mathrm{idle}_{mp}, \mathrm{idle\_b}_{mp}, \mathrm{sauv}_{imp}$)
Repeat until finding a feasible solution S
1– for all item i and period p, $dem_{ip} \leftarrow D_{ip}$
2– Generate the first items for the first period on each production line m
$w_{iim1} \leftarrow$ *random Boolean*
$q_{im1} \leftarrow$ *random integer*$(1, \min(dem_{i1}, C_{m1}))$
For all the periods of the planning horizon, repeat steps from 3 to 6
3– Define the setups for the change from item i to item j inside the period on the launched lines
$x_{ijmp} \leftarrow$ *random Boolean*
$q_{imp} \leftarrow$ *random integer*$(1, \min(dem_{ip}, C_{mp}))$
4– Reassignment (1) of the unmet demand of the period of each item
2– Calculate the inventory of each item s_{ip}
6– Define the transition variables on each production line for the next period
Return solution S

Algorithm 2 generates an initial solution by randomly assigning products to production lines with random production quantities, while respecting demand and capacity constraints. Each product is produced on only one production line at a time. Sequence dependent setups are made to produce the remaining products, always within the limits of demand and capacities. Unsatisfied demands are reassigned to ensure their coverage, and inventories are calculated. Finally, transition variables are defined to plan for subsequent periods, as illustrated in the flowchart of Figure 7. This process is repeated for all periods.

The reassignment process is structured into two phases:

(1) Reassignment within the same period: Priority is given to increasing quantities on available lines, favoring production continuity (setup carryover, setup crossover, changes at the beginning of the period) or performing setups within the periods, or launching new lines if necessary, until the demand is satisfied.
This prioritization approach helps build a good and feasible initial solution. This can be achieved by optimizing both setup costs and production line launching costs.
(2) Reassignment to earlier periods: If demand remains unmet after the first phase, production quantities are adjusted in earlier periods with available capacity, beginning with the most recent periods.

The transition states generation mechanism ensures synchronization between production line states and transition variables. For period $p + 1$, decisions are made based on the remaining capacity and the last product manufactured in period p:

– No remaining capacity: The mechanism evaluates the possibility of a setup carryover (y_{imp+1}) or production continuity at the beginning of the period $p + 1$ by a sequence dependent setup (w_{ijmp+1}), to assign random production quantities.
– Remaining capacity: It considers the feasibility of a setup crossover (z_{ijmp+1}) or a changeover at the beginning of period $p + 1$ (w_{ijmp+1}), with random quantities to be assigned.

The allocated quantities are constrained by the production line's capacity and the unmet demand, while any unused production lines remain idle.

3.2.2. Neighborhood solution generation

The choice of the neighborhood search method employed in this study (Algorithm 3) was made after testing two alternative approaches. The first method involved advancing production by reallocating production lines

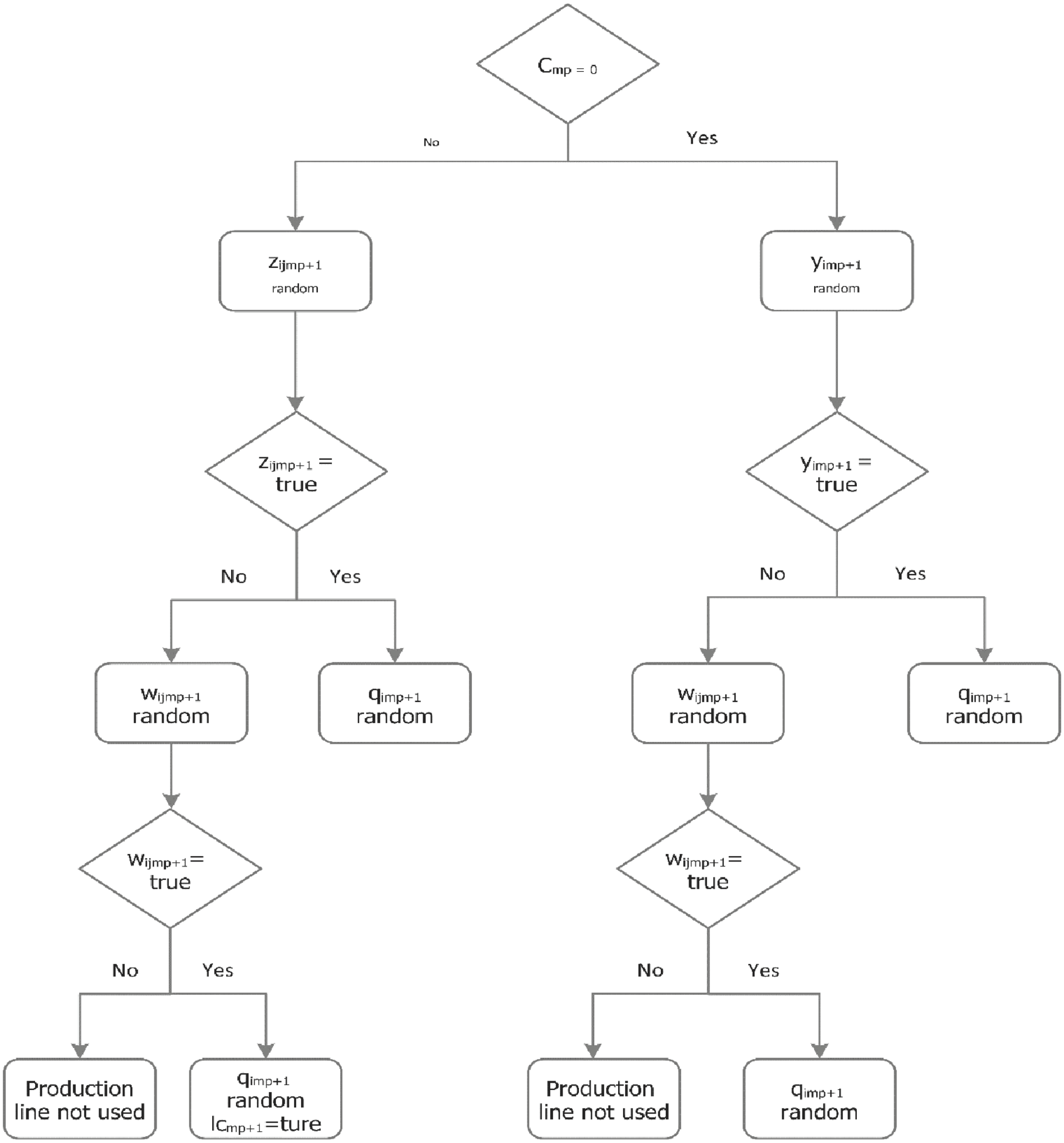


Figure 7. Flowchart of the transitions between two consecutive periods.

to earlier periods, while the second postponed production to later periods. However, testing these two methods yielded inconclusive results and did not lead to significant improvements in the solution. Advancing or postponing entire production schedules while maintaining the same sequencing on each production line only impacts storage costs and production line launching costs. It does not affect setup costs, which are sequence-dependent, or production costs, are production line-dependent and product-dependent.

Algorithm 3: Generate_Random_Neighberhood.

Inputs: initial solution S_0

Outputs: neighbor solution S

1– $S \leftarrow S_0$

2– Select randomly an item manufactured with the indices $m_1 \in M$, $p_1 \in P$ and $n_1 \in N$

3– Delete the production of item n_1 on the production line m_1 in period p_1

4– If items exist before and/or after n_1 on production line m_1, adjust production sequences accordingly. Otherwise, proceed to step 5.

From period $h = 1$ to period $h = P$ repeat steps from 5 to 7:

5– Reassignment (2) of the unmet demand of period h of each item of the set of items

6– Calculate the inventory level of each item at the end of period h

7– Correction of the transitions on each period t, from $t = 0$ to $t = h$, for each production line

8– Return S

The method adopted in this work focuses on the detailed planning and sequencing of products on production lines. In addition to improving production line launching costs and storage costs, it optimizes setup costs and production costs. At each iteration, this method deals with a single product assigned to a specific production line for a given period.

The method initializes a neighborhood solution by inheriting the variables of the initial solution (assignments, quantities, scheduling, and inventories). A position is then randomly selected to cancel the production of a specific product, disrupting capacities, transitions, and associated demands.

The following steps aim to address these disruptions:

Step 4: Adjust the scheduling by linking the products before and after the canceled product.

Step 5: Reallocate unmet demands to production lines, prioritizing earlier periods and promoting continuity to reduce setup costs and improve the solution.

Step 6: Recalculate inventories.

Step 7: Adjust transition states between periods.

These steps are performed across all periods since each iteration impacts the entire planning horizon. The corrected solution is then returned.

For the studied system, several parameters must be considered, characterized by high variability: a fluctuating number of products, multiple production lines, and a planning horizon composed of several periods. This results in a large number of variables within the system. The technical complexity of this approach can be visualized through various scenarios and decision-making possibilities when adjusting the resulting scheduling at step 4, it is illustrated by Figure 8.

The reassignment of unmet demand at step 5 in the neighborhood search method is similar to that of the initial solution generation method, with priority given to earlier periods where the product has already been manufactured. This approach utilizes the remaining capacities to promote continuity, reduce setup costs, and improve the solution.

The transition correction step 7 aims to adjust the solution to ensure its feasibility after modifications and changes within and between periods, whether during the production cancellation phase or the reallocation of unmet demand. This phase is one of the most complex in the metaheuristic due to the large number of parameters to consider, the variety of possible cases to address, and the numerous configurations that can be envisioned for each case. These factors significantly increase the complexity of the studied system and the metaheuristic.

Here is an example of transition correction: Production continuity (y_{imt+1}).

Case1: if there is no more remaining capacity at period t but the last manufactured item is different from the one at the beginning of the next period, then there is no more setup carryover. However, continuity at the beginning of period $t+1$ can be ensured by a changeover ($w_{saveimt+1} = 1$) if the remaining capacity at $t+1$ is sufficient to carry out the setup (change from *save* to i). If this is not possible, the produced amount of i is reduced in such a way to allow setup for the change from *save* to i. Even if, by reducing the production

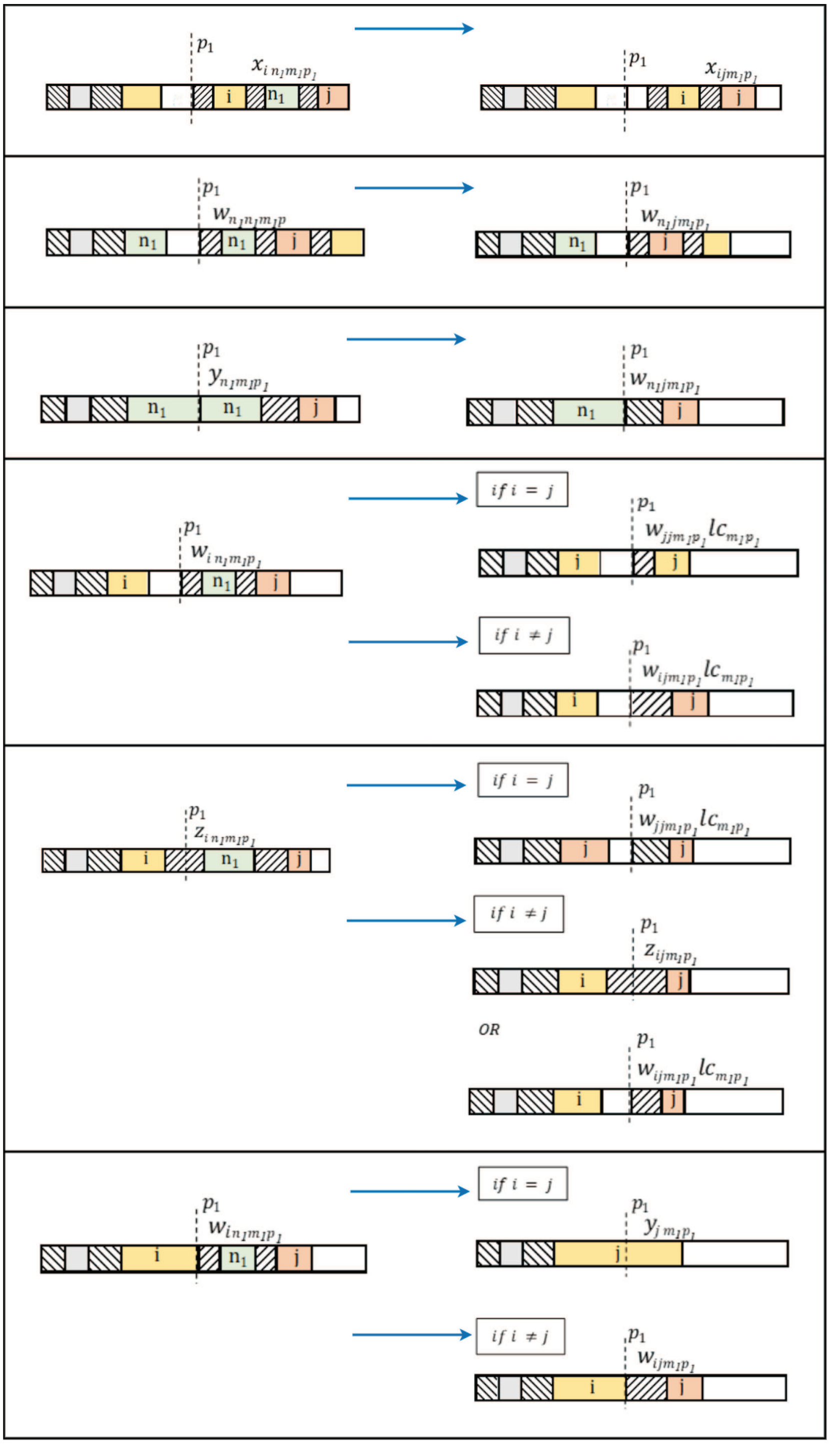


FIGURE 8. Possible scheduling corrections to make after the removal of item n_1.

of i to the minimum possible, the setup is still not feasible due to lack of capacity. Then, in this case, the production of item i is cancelled ($q_{imt+1} = 0$) and the one that follows (j) is linked, if it exists ($x_{ijmt+1} = 1$), to the last item (*save*) manufactured in period t, in such case two configurations are possible:
- If $save = j$ then $y_{jmt+1} = 1$
- Else, $w_{savejmt+1} = 1$

Case 2: if there is a remaining production capacity at the end of period t, then we need to check the last manufactured item against the first manufactured item in the next period.
- If $save = i$, then we just launch the product i at the beginning of period $t+1$ ($w_{iimt+1} = 1$) accompanied by the launch of the production line ($\mathrm{lc}_{mt+1} = 1$), of course, if the capacity is sufficient for this. In the opposite case, we reduce the amount of the production of i or we cancel it completely in the same way as case 1 and we link with the next item if it exists ($x_{ijmt+1} = 1$) to have $w_{savejmt+1} = 1$.
- If $save \neq i$, then we first test the possibility of making a setup crossover and this if the remaining capacity at t as well as the remaining capacity at $t+1$ are sufficient to perform the setup from *save* to i to have $z_{saveimt+1} = 1$. If this is not feasible, then we consider a setup changeover at the beginning of period $t+1$ ($w_{saveimt+1} = 1$) combined with a launch of the production line ($\mathrm{lc}_{mt+1} = 1$) if the remaining capacity at $t+1$ is sufficient. If the capacity is not sufficient, then in the same way as case 1, we reduce the production amount of i or we cancel it completely by the need to perform the changeover from *save* to the product that succeeds i ($x_{ijmt+1} = 1$) to have $w_{savejmt+1} = 1$.

4. Experiments

As stated in Section 3, the study concerns the planning and scheduling of a production system composed of several production lines to manufacture several types of products on a multi-period planning horizon, by considering several transition states between consecutive periods. To solve this problem, we proposed and developed a mathematical algorithm considering all parameters and constraints that constitute the proposed model complexity. The efficiency of the developed algorithm was measured by implementing a mathematical model under the CPLEX solver. In this context, several tests were carried out on an Intel (R) Core (TM) i5-1035G1 CPU @ 1.00 GHz 1.19 GHz and 20 GB of RAM computer under Windows 10 (64 bit).

To be able to implement the proposed model, a careful collection of field data was carried out for the various costs, processing times, demand forecasts and production rates for the different products on the different production lines.

In the framework of small instances, obtained results under CPLEX were compared with those obtained by the coded metaheuristic using IntelliJ IDEA 2020.2.1. For medium and large instances, where CPLEX presented limitations, we solve the problem using the metaheuristic.

From the obtained results, we noticed several possible scenarios which depend on the demand nature and which influence the system behavior. In this context, we have identified three possible situations. The first one concerns the case where the demands are greater than the capacities of the production lines of the planning horizon, while being aware that the production in this system is carried out in a continuous way by 3 shifts working alternately, in 7 days a week. So, there is no possibility to add overtime to satisfy a very high demand.

The second situation is when the demand is low: the total sum of demands is well lower than the capacities of the production lines over all the available periods. In this case, and according to the experiments carried out, the obtained solutions reduce the number of production lines launched by using a minimum of periods of the planning horizon while privileging manufacturing continuity. The algorithm aims to minimize the utilization rate of the production lines by planning all the demands of the later periods in grouped periods since the storage costs are relatively low compared to production lines launch costs and setup costs.

The third situation concerns medium demands where the sum of the demands is equal to or slightly less than the capacities of the different production lines over all the periods. In this situation, the challenge is to find the best planning and scheduling that improves the overall performance of the system while satisfying the demands for each product type and over each period. Also, since the setups are sequence-dependent and the

TABLE 2. Results of the tests on CPLEX.

Configuration	Products	Production lines	Periods	Objective function	Time	Unit
1	3	3	3	645 264	11	sec
	5	3	3	640 749	2	min
	6	3	3	NO RESULTS	–	–
2	3	6	3	1 136 108	18	sec
	3	8	3	1 459 212	2	min
	3	10	3	1 788 702	8	min
	3	12	3	NO RESULTS	–	–
3	3	3	6	1 135 549	9	min
	3	3	8	1 597 957	55	min
	3	3	10	NO RESULTS	–	–

setup times are significant, it is difficult to find the best scheduling that minimizes the overall costs on one hand and respects the capacities on each period of each production line on the other hand, thus reflecting the way the studied benchmark performs. This is why the proposed algorithm becomes necessary to give an optimal solution.

The different experimentations presented in this article are part of this last situation, and were grouped into three configurations according to the parameter to be varied:

– In the first configuration, we varied the number of products.
– In the second configuration, we varied the number of production lines.
– In the third configuration, we varied the number of periods.

The data for the three configurations are available from [18].

For all the configurations, obtained results under CPLEX are represented in the following Table 2.

From the obtained results, we noticed that the variation of a parameter influences the resulting solution and shows the limits of CPLEX in solving problems of large instances, such is the case for each configuration. In the first configuration, beyond 6 products, CPLEX does not provide results even if the number of periods and production lines is restricted (3 production lines and 3 periods). As for the second configuration, by increasing the number of production lines in the system, CPLEX no longer works beyond 10 production lines with 3 products and 3 periods. This situation was noticed even for the 3rd configuration; by increasing the number of periods of more than 8 periods, the solver no longer gives results with 3 products and 3 production lines. To this fact, we made call to a metaheuristic in solving large-size instances of the studied system. The total costs of each configuration tests are presented in the graphics of Figure 9.

For each experiment where CPLEX was able to reach an optimal solution, a sequencing of the products on each production line in each period, corresponding to the minimum objective function, was given. For example, the sequencings obtained for the experiments 3 products/ 3 production lines/ 3 periods and 3 products/ 8 lines/ 3 periods are respectively given in Figures 10 and 11.

We can see that the sequencing obtained for each configuration has minimized or eliminated the waiting time between periods on each launched production line and, therefore, allows the manufacturing continuity (setup carryover, setup crossover and the start of the period with a setup for the change to a different product without relaunching the production line) between consecutive periods. The proposed algorithm also allowed to minimize the number of setups and to limit the use of production lines, according to the needed capacity in each period, required for the satisfaction of the demand at a lower cost: if there is the possibility to group or allocate the production to other periods and other production lines, then the use of some production lines is avoided, since the launch costs of the production lines and the setup costs are significant compared to the storage costs. This possibility of grouping the different demands and planning them over a time horizon, while respecting

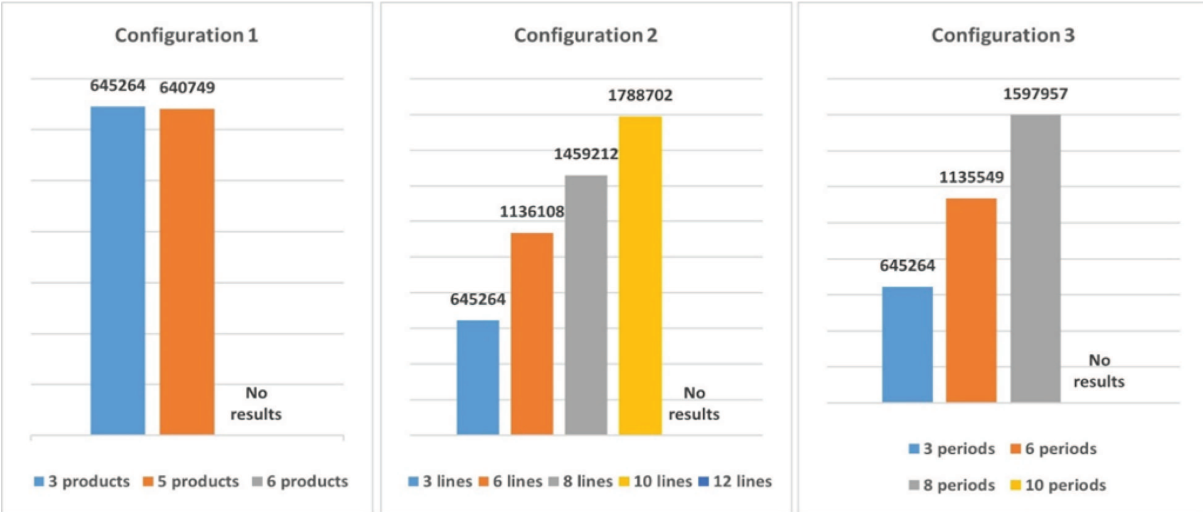


Figure 9. Total costs obtained by CPLEX for the three configurations.

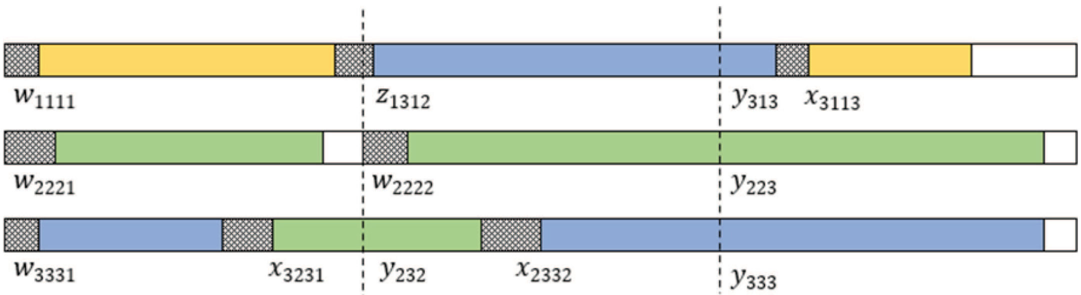


Figure 10. Result of the experimentation with 3 items, 3 production lines and 3 periods.

the delivery delays, allowed reducing the total costs, thus increasing the company's profits and ensuring the synchronous exploitation of production units.

For each configuration among the three already mentioned configurations, and from a certain rank of the parameter to be varied, CPLEX was unable to give any result. In this situation, the use of metaheuristics becomes a necessity.

The parameters for the Simulated Annealing (SA) metaheuristic were determined through an initial experimental tuning phase. A final temperature (T_f) of 0.1 was set. Subsequently, preliminary tests were conducted to identify a suitable initial temperature ($T0$); a value of 20 was selected as it provided a balanced exploration of the solution space. With $T0$ fixed, further tests were performed to calibrate the geometric cooling rate (α). A cooling rate of 0.99 was found to yield the most effective convergence towards improved final solutions. Once established, this parameter set ($T0 = 20$, $T_f = 0.1$, $\alpha = 0.99$) was used consistently across all SA-based experiments. For each specific problem instance, multiple runs with varying iteration counts were executed, and the best-obtained results are reported in this study.

In a first step, we tested the same parameters of the different proposed configurations on CPLEX. Then, for each type of configuration, we increased the variation of the considered parameter. The obtained results are presented in Table 3.

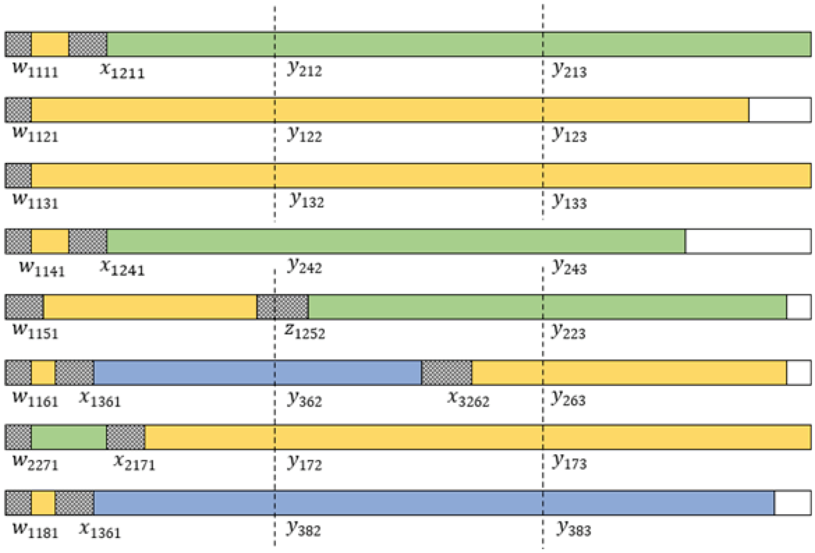


FIGURE 11. Result of the experimentation with 3 items, 8 production lines and 3 periods.

TABLE 3. Comparison between CPLEX results and SA algorithm results.

				CPLEX			SA			
	Products	Production lines	Periods	Objective function	Time	Unit	Objective function	GAP	Time	Unit
Configuration 1	3	3	3	645 264	11	sec	656 683	1,8%	39	sec
	5	3	3	640 749	2	min	653 243	1,9%	6	min
	8	3	3	NO RESULTS	–	–	784 037		5	min
	10	3	3	NO RESULTS	–	–	792 901		15	min
	30	3	3	NO RESULTS	–	–	280 576		15	h
Configuration 2	3	6	3	1 136 108	18	sec	1 172 920	3,2%	6	min
	3	8	3	1 459 212	2	min	1 511 915	3,6%	89	min
	3	10	3	1 788 702	8	min	1 873 976	4,8%	4	min
	3	15	3	NO RESULTS	–	–	2 949 713		27	min
	3	20	3	NO RESULTS	–	–	3 852 004		20	min
Configuration 3	3	3	6	1 135 549	9	min	1 161 859	2,3%	7	min
	3	3	8	1 597 957	2	min	1 676 669	4,9%	13	min
	3	3	10	NO RESULTS	–	–	1 918 044		20	min
	3	3	15	NO RESULTS	–	–	3 247 620		26	min

The metaheuristic demonstrated robust feasibility, consistently generating valid production schedules for medium and large-scale instances, even where CPLEX failed to return a feasible solution within resource limits. This robustness is crucial for practical deployment, as the modeled constraints (manufacturing continuity across periods, sequence-dependent setups on parallel lines) significantly increase combinatorial complexity.

The empirical results (Tab. 3) reveal a clear and expected relationship between instance size and computational effort for the SA algorithm. The solution space of the problem grows combinatorially with the product of the parameters $|I|$ (products), $|M|$ (lines), and $|P|$ (periods). The SA's runtime is primarily governed by the feasible solution construction, the neighborhood exploration and the number of iterations.

Consequently, as shown in Table 3, runtime generally increases with the size of the considered parameter in each configuration. For example, in Configuration 2, scaling production lines from 6 to 15 increased SA runtime from 6 to 27 min. This predictable scaling is a key strength, contrasting with the exponential worst-case time

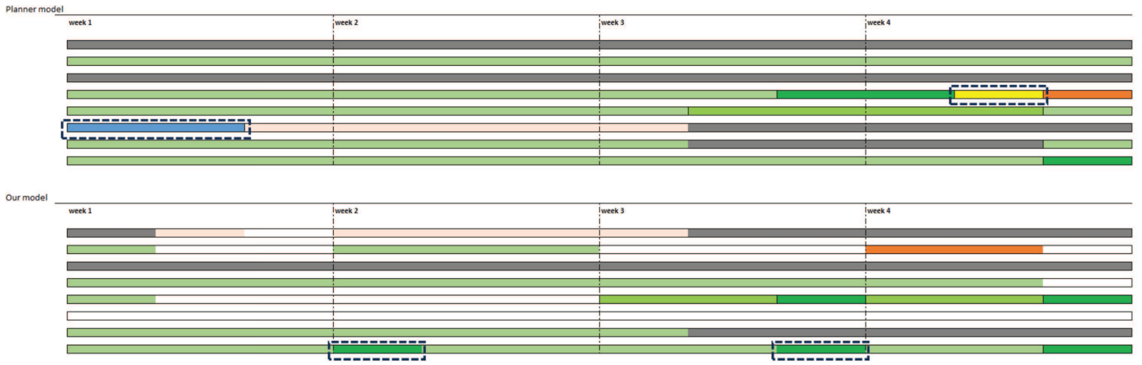


Figure 12. Comparison between the production plan provided by the planner and our solution.

complexity of the exact MILP solved by CPLEX, which becomes intractable for larger instances (*e.g.*, ≥ 8 products).

Notably, runtime does not increase monotonically in all cases (*e.g.*, 10 periods required 20 min while 15 periods required 26 min). This can be attributed to the problem's data structure: when period demands are closer to total line capacities, the search space of feasible schedules is effectively constrained, allowing the algorithm to converge more rapidly to a high-quality solution.

For small instances where CPLEX obtained provably optimal solutions (OF1), the SA's objective value (OF2) was compared using the relative gap: $\text{GAP} = \frac{\text{OF2}-\text{OF1}}{\text{OF1}} \times 100\%$ The observed gaps ranged from 1.8% to 4.9% confirming the metaheuristic's effectiveness in near-optimal search. The increase in gap for larger instances is attributed to the compounded variability from sequence-dependent setups and heterogeneous production lines, which enlarges the local optimum landscape.

As illustrated in Table 3, the objective function value increases with system size across all configurations. This is a direct consequence of the experimental design, where added demand stress-tests the system near its capacity, thereby increasing total production, setup, line launching, and storage costs, all accurately captured by the model.

The problem addressed integrates multiple real-world complexities: parallel heterogeneous lines, sequence-dependent setups, and multi-period continuity constraints. This integration results in a high-dimensional, constrained combinatorial space. The proposed SA algorithm effectively navigates this space, providing scalable, near-optimal solutions where exact methods fail, while maintaining computational times acceptable for tactical planning. The results confirm that the algorithm is a viable and effective decision-support tool for the integrated production planning and scheduling problem under study.

5. Industrial implementation and decision-making support

The proposed model has been implemented in an industrial company, and its results have been compared with production plans manually developed by production planners. These traditional plans relied primarily on the planners' experience and intuition, using paper records and manual calculations, without the support of a formal optimization or decision-support tool.

To illustrate the operational impact of the model, a real industrial case from the June 2022 planning period is analyzed. The case involves 8 products, 8 production lines, and a planning horizon of 4 periods (weeks). Figure 12 presents a direct comparison between the manual production plan proposed by the scheduler and the plan generated by the optimization model.

For an identical demand scenario, the two approaches lead to significantly different decisions regarding the allocation of products to production lines. The manual plan mainly focuses on minimizing immediate production costs, without explicitly considering inventory holding costs or setup costs. In contrast, the proposed model adopts a system-wide perspective by strategically exploiting initial inventories, reducing production quantities, and deliberately introducing idle times to release future capacity. For instance, two products (highlighted on lines 4 and 6 in the manual plan) are not scheduled in the optimized solution, as the model identifies inventory as a more cost-effective means to satisfy demand. Moreover, the model better accommodates period-specific demand requirements, such as scheduling a product on the last production line during the second week, an allocation that is absent from the manual plan.

Overall, the proposed model improves the effective utilization of production resources, enhances the robustness of the production plan in the presence of demand variability, and results in an average reduction of approximately 10% in total costs over the planning horizon. These results highlight the value of explicitly integrating cost components that are often neglected in practice (*e.g.*, inventory holding, setups, and unused capacity) and demonstrate the model's relevance as a practical decision-support tool.

5.1. Managerial implications and practical usage guidelines

To strengthen the connection with industrial decision-making and respond to practitioners' needs, this section clarifies how production planners can use the model in practice and which managerial policies it supports.

The use of the model can be structured into three main steps:

(1) Data Input and Parameterization: The planner inputs standard operational data (forecast demand, initial inventories, setup times, machine capacities and production rates) as well as strategic parameters translated into cost terms (inventory holding cost, setup cost, production cost, line launch cost). A key contribution of the model lies in forcing the explicit formalization and quantification of trade-offs that were previously handled intuitively.
(2) Plan Generation: The model generates a detailed production plan specifying what to produce, on which line, and at what time, along with key performance indicators such as machine utilization rates, projected inventory levels, and total costs.
(3) Analysis and Validation: The planner's role evolves from plan constructor to analyst and decision validator. Potentially counterintuitive recommendations from the model (*e.g.*, leaving a line idle or postponing production) are evaluated in light of qualitative constraints that are not explicitly modeled. In this sense, the model provides an objective quantitative basis for informed managerial adjustments rather than acting as a black box that enforces a rigid solution.

5.2. Managerial policies and strategies supported by the model

The outputs of the model provide tangible decision support for several key managerial policies:

- Inventory Policy: The model identifies appropriate buffer inventory levels that balance holding costs and responsiveness, supporting the definition of inventory turnover targets at the product level.
- Batching and Sequencing Strategy: By explicitly quantifying setup costs, the model helps determine economically efficient batch sizes and production sequences that minimize downtime.
- Order Acceptance Rules: Improved visibility into available capacity and marginal costs enables the definition of objective guidelines for accepting or rejecting urgent or additional orders based on their true profitability impact.
- Capacity Investment Decisions: The systematic identification of bottlenecks and the strategic use of idle time provide data-driven arguments to justify or postpone investments in new machines or additional labor.

Following this industrial experimentation, the company has adopted the model as a central decision-support tool for production planning. The solution has also been successfully deployed in several subsidiaries, with adaptations to local constraints, thereby demonstrating the robustness, scalability, and direct practical relevance of the proposed methodology.

6. Conclusion

This research presents a significant step forward in addressing the complexities of modern production systems by proposing a comprehensive decision-support methodology. The developed mathematical model and metaheuristic provide an innovative solution to the joint optimization of production planning and scheduling, particularly in environments where production line launch costs and transition configurations play a critical role.

A key contribution of this work lies in the simultaneous integration of five distinct sequence-dependent setup configurations, including setup carryover, setup crossover, production continuity with the start with a setup, product relaunch after idle times, and product change with line launch. These configurations, rarely addressed together in the literature, reflect the intricate realities of industrial operations. The proposed model successfully captures these dynamics, while the Simulated Annealing-based metaheuristic demonstrates its effectiveness in solving large-size instances and reducing operational costs. The approach was validated using real-world case study, highlighting its practical applicability and ability to enhance overall system performance by balancing costs and resource utilization. This dual contribution – both theoretical and practical – bridges a critical gap in the literature and offers valuable insights for practitioners and researchers alike.

Future work will focus on several promising avenues to extend the methodology's scope and robustness. Specifically, a sensitivity analysis on key cost drivers (*e.g.*, setup and launch costs) will be conducted to assess the economic robustness of the approach under varying operating conditions and to offer more practical managerial insights. Beyond this, research could explore the inclusion of additional factors such as energy consumption, sustainability considerations, or multi-objective optimization frameworks. Additionally, testing the approach across diverse industrial settings could further demonstrate its adaptability. By addressing these avenues, the proposed research lays a strong foundation for advancing decision-support systems in production management.

Acknowledgments

The authors thank the company CHIALI TUBES for making their data available for this research.

Data availability statement

The data supporting the findings of this study are available online in the Mendeley Data repository: https://data.mendeley.com/preview/xd9ttyt8sb?a=73d00b22-e0e7-4c54-8804-3714e4a5f4c0 [17].

References

[1] M.H. Bassett, P. Dave, F.J. Doyle, G.K. Kudva, J.F. Pekny, G.V. Reklaitis, S. Subrahmanyam, D.L. Miller and M.G. Zentner, Perspectives on model-based integration of process operations. *Comput. Chem. Eng.* **20** (1996) 821–844.

[2] J. Behnamian, S.M.T. Fatemi Ghomi, B. Karimi and M. Fadaei Moludi, Capacitated lot-sizing and production sequence problem with setups complexity. *Int. J. Sci. Technol.* (2022).

[3] M.A.F. Belo-Filho, B. Almada-Lobo and F.M.B. Toledo, Models for capacitated lot-sizing problem with backlogging, setup carryover and crossover. *J. Oper. Res. Soc.* **65** (2013) 1735–1747.

[4] G.R. Bitran and H.H. Yanasse, Computational complexity of the capacitated lot size problem. *Manag. Sci.* **28** (1982) 1174–1186.

[5] V.C.B. Camargo, F.M. Toledo and B. Almanda-Lobo, Three time-based scale formulations for the two-stage lot sizing and scheduling in process industries. *J. Oper. Res. Soc.* **63** (2012) 1613–1630.

[6] D.M. Carvalho and M.C.V. Nascimento, Hybrid matheuristics to solve the integrated lot sizing and scheduling problem on parallel machines with sequence-dependent and non-triangular setup. *Eur. J. Oper. Res.* **296** (2022) 158–173.

[7] P.M. Castro, I.E. Grossmann and Q. Zhang, Expanding scope and computational challenges in process scheduling. *Comput. Chem. Eng.* **114** (2018) 14–42.

[8] W.H. Chen and J.M. Thizy, Analysis of relaxations for the multi-item capacitated lot-sizing problem. *Ann. Oper. Res.* **26** (1990) 29–72.

[9] N.T. Chowdhury, M.F. Baki and A. Azab, A modeling and hybridized decomposition approach for the multi-level capacitated lot-sizing problem with setup carryover, backlogging and emission control. *Oper. Res. Forum* **5** (2024) 68.

[10] A. Clark, B. Almada-Lobo and C. Almeder, Lot sizing and scheduling: industrial extensions and research opportunities. *Int. J. Prod. Res.* **49** (2011) 2457–2461.

[11] K. Copil, M. Wörbelauer, H. Meyr and H. Tempelmeier, Simultaneous lotsizing and scheduling problems: a classification and review of models. *OR Spectr.* **39** (2016) 1–64.

[12] S. Das and A. Patnaik, Production planning in the apparel industry. Garment Manufacturing Technology (2015).

[13] G.S. Dastidar and R. Nagi, Scheduling injection molding operations with multiple resource constraints and sequence dependent setup times and costs. *Comput. Oper. Res.* **32** (2005) 2987–3005.

[14] R. De Matta and M. Guignard, Dynamic production scheduling for a process industry. *Oper. Res.* **42** (1994) 492–503.

[15] L.S. Dias and M.G. Ierapetritou, From process control to supply chain management: an overview of integrated decision-making strategies. *Comput. Chem. Eng.* **106** (2017) 826–835.

[16] C. Dillenberger, L.F. Escudero, A. Wollensak and W. Zhan, On practical resource allocation for production planning and scheduling with period overlapping setups. *Eur. J. Oper. Res.* **75** (1994) 275–286.

[17] A. Djama, Data set for integrated lot sizing and scheduling problem (2025).

[18] A. Djama, Data set for integrated lot sizing and scheduling problem (2026).

[19] A. Djama Maafa and L. Triqui Sari, Multi-item capacitated lot-sizing problem with parallel resources, setup times and costs: a case study, in 2022 2nd International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET). IEEE (2022) 1–5.

[20] A. Djama Maafa, L. Triqui Sari and F. Belkaid, Multi-periods production planning for an industrial company, in IEEE 13th International Colloquium of Logistics and Supply Chain Management (LOGISTIQUA) (2020) 1–5.

[21] E. Dos Santos-Meza, M. Oliveira dos Santos and M. Nereu Arenales, A lot-sizing problem in an automated foundry. *Eur. J. Oper. Res.* **139** (2002) 490–500.

[22] A. Drexl and K. Haase, Proportional lot sizing and scheduling. *Int. J. Prod. Econ.* **40** (1995) 73–87.

[23] A. Drexl and K. Haase, Sequential-analysis based randomized-regret-methods for lot-sizing and scheduling. *J. Oper. Res. Soc.* **47** (1996) 251–265.

[24] A. Drexl and A. Kimms, Lot sizing and scheduling - survey and extensions. *Eur. J. Oper. Res.* **99** (1997) 221–235.

[25] A. Dumoulin and C. Vercellis, Tactical models for hierarchical capacitated lot-sizing problems with setups and changeovers. *Int. J. Prod. Res.* **38** (2000) 51–67.

[26] M. Erdirik-Dogan and I.E. Grossmann, Planning models for parallel batch reactors with sequence-dependent changeovers. *AIChE J.* **53** (2007) 2284–2300.

[27] M. Erdirik-Dogan and I.E. Grossmann, Slot-based formulation for the short-term scheduling of multistage, multi-product batch plants with sequence-dependent changeovers. *Ind. Eng. Chem. Res.* **47** (2008) 1159–1183.

[28] D.J. Fiorotto, R. Jans and S.A. de Araujo, An analysis of formulations for the capacitated lot sizing problem with setup crossover. *Comput. Ind. Eng.* **106** (2016) 338–350.

[29] D.J. Fiorotto, J.D.C. Huaccha Neyra and S.A. de Araujo, Impact analysis of setup carryover and crossover on lot sizing problems. *Int. J. Prod. Res.* **58** (2020) 6350–6369.

[30] M. Florian, J.K. Lenstra and A.H.G. Rinnooy Kan, Deterministic production planning algorithms and complexity. *Manag. Sci.* **26** (1980) 669–679.

[31] M. Furlan, B. Almada-Lobo, M. Santos and R. Morabito, Matheuristic for the lot-sizing and scheduling problem in integrated pulp and paper production. *Comput. Ind. Eng.* **192** (2024) 110183.

[32] Y. Gao, A heuristic procedure for the capacitated lot sizing problem with setup carryover. *Control Theory Appl.* **17** (2000) 937–940.

[33] D.J. Garcia and F. You, Supply chain design and optimization: Challenges and opportunities. *Comput. Chem. Eng.* **81** (2015) 153–170.

[34] A. Garre, M.C. Ruiz and E. Hontoria, Application of machine learning to support production planning of a food industry in the context of waste generation under uncertainty. *Oper. Res. Perspect.* **7** (2020) 100147.

[35] J. Geršak, Planning and organisation of clothing production. Design of Clothing Manufacturing Processes (2013) 87–104.

[36] C. Gicquel, M. Minoux and Y. Dallery, Capacitated lot sizing models: a literature review (2008).

[37] M. Gopalakrishnan, A modified framework for modelling set-up carryover in the capacitated lot-sizing problem. *Int. J. Prod. Res.* **38** (2000) 3421–3424.

[38] M. Gopalakrishnan, D.M. Miller and C.P. Schmidt, A framework for modelling setup carryover in the capacitated lot sizing problem. *Int. J. Prod. Res.* **33** (1995) 1973–1988.
[39] M. Gopalakrishnan, K. Ding, J.M. Bourjolly and S. Mohan, A tabu-search heuristic for the capacitated lot-sizing problem with setup carryover. *Manag. Sci.* **47** (2001) 851–863.
[40] H.G. Goren, S. Tunali and R. Jans, A hybrid approach for the capacitated lot sizing problem with setup carryover. *Int. J. Prod. Res.* **50** (2012) 1582–1597.
[41] L. Guimarães, D. Klabjan and B. Almada-Lobo, Modeling lot-sizing and scheduling problems with sequence dependent setups. *Eur. J. Oper. Res.* **239** (2014) 644–662.
[42] D. Gupta and T. Magnusson, The capacitated lot-sizing and scheduling problem with sequence-dependent setup costs and setup times. *Comput. Oper. Res.* **32** (2005) 727–747.
[43] E. Guzman, B. Andres and R. Poler, Models and algorithms for production planning, scheduling and sequencing problems: a holistic framework and a systematic review. *J. Ind. Inf. Integr.* (2022) 100287.
[44] K. Haase, Lot Sizing and Scheduling for Production Planning. Springer, Berlin (1994).
[45] K. Haase, Capacitated Lot-Sizing with Linked Production Quantities of Adjacent Periods. Springer, Berlin (1998).
[46] I. Ilhan, An improved simulated annealing algorithm with crossover operator for capacitated vehicle routing problem. *Swarm Evol. Comput.* **64** (2021) 100911.
[47] R.J.W. James and B. Almada-Lobo, Single and parallel machine capacitated lotsizing and scheduling: new iterative mip-based neighborhood search heuristics. *Comput. Oper. Res.* **38** (2011) 1816–1825.
[48] R.A.F. Jans and Z. Degraeve, An industrial extension of the discrete lot-sizing and scheduling problem. *IIE Trans.* **36** (2004) 47–58.
[49] W. Kaczmarczyk, Modeling multi-period set-up times in the proportional lot-sizing problem. *DMMS* **3** (2009) 15–35.
[50] W. Kaczmarczyk, Modeling set-up times overlapping two periods in the proportional lot-sizing problem with identical parallel machines. *DMMS* **7** (2013) 43–50.
[51] J. Kang, Capacitated lot-sizing problem with sequence-dependent setup, setup carryover and setup crossover. *Processes* **8** (2020) 785.
[52] B. Karimi, S. Fatemi Ghomi and J. Wilson, The capacitated lot sizing problem: review of models and algorithms. *OMEGA* **31** (2003) 365–378.
[53] V. Kayvanfar, M. Zandieh and M. Arashpour, Hybrid bi-objective economic lot scheduling problem with feasible production plan equipped with an efficient adjunct search technique. *Int. J. Syst. Sci.-Oper. Logist.* **10** (2023) 2059721.
[54] A. Kimms, Competitive methods for multi-level lot sizing and scheduling: Tabu search and randomized regrets. *Int. J. Prod. Res.* **34** (1996) 2279–2298.
[55] A. Kimms, Multi-level, single-machine lot sizing and scheduling (with initial inventory). *Eur. J. Oper. Res.* **89** (1996) 86–99.
[56] A. Kimms, A genetic algorithm for multi-level, multi-machine lot sizing and scheduling. *Comput. Oper. Res.* **26** (1999) 829–848.
[57] S. Kirkpatrick, C. Daniel Gelatt Jr and M.P. Vecchi, Optimization by simulated annealing. *Science* **220** (1983) 671–680.
[58] N. Klement, M. Amine Abdeljaouad, L. Porto and C. Silva, Lot-sizing and scheduling for the plastic injection molding industry – a hybrid optimization approach. *Appl. Sci.* **11** (2021) 1202.
[59] G.M. Kopanos, L. Puigjaner and C.T. Maravelias, Production planning and scheduling of parallel continuous processes with product families. *Ind. Eng. Chem. Res.* **50** (2011) 1369–1378.
[60] J. Krarup and O. Bilde, Plant location, set covering and economic lot size: an 0 (mn)-algorithm for structured problems. Numerische Methoden bei Optimierungsaufgaben Band 3: Optimierung bei graphentheoretischen und ganzzahligen Problemen (1977) 155–180.
[61] J. Kumar, G. Yadav and B.P. Agrawal, A hybrid stochastic optimization model for lot sizing and scheduling problem. *Cuest. Physioter.* **54** (2025) 2007–2018.
[62] Y. Lee and K. Lee, New integer optimization models and an approximate dynamic programming algorithm for the lot-sizing and scheduling problem with sequence-dependent setups. *Eur. J. Oper. Res.* **302** (2022) 230–243.
[63] M. Mahdieh, A. Clark and M. Bijari, A novel flexible model for lot sizing and scheduling with non-triangular, period overlapping and carryover setups in different machine configurations. *Flex. Serv. Manuf. J.* **30** (2017) 884–923.
[64] C.T. Maravelias and C. Sung, Integration of production planning and scheduling: overview, challenges and opportunities. *Comput. Chem. Eng.* **33** (2009) 1919–1930.
[65] F. Marinelli, M.E. Nenni and A. Sforza, Capacitated lot sizing and scheduling with parallel machines and shared buffers: a case study in a packaging company. *Ann. Oper. Res.* **150** (2007) 177–192.

[66] A. Menezes, B. Clark and B. Almada-Lobo, Capacitated lot-sizing and scheduling with sequence-dependent, period-overlapping and non-triangular setups. *J. Sched.* **14** (2010) 209–219.

[67] S. Mohan, M. Gopalakrishnan, R. Marathe and A. Rajan, A note on modelling the capacitated lot-sizing problem with set-up carryover and set-up splitting. *Int. J. Prod. Res.* **50** (2012) 5538–5543.

[68] A. Osati, E. Mehdizadeh and S. Ebrahimnejad, Metaheuristic approaches for flexible lot-sizing and scheduling with remanufacturing and sequence-dependent setup time. *Jordan J. Mech. Ind. Eng.* **19** (2025).

[69] P. Porkka, A.P.J. Vepsäläinen and M. Kuula, Multiperiod production planning carrying over set-up time. *Int. J. Prod. Res.* **41** (2003) 1133–1148.

[70] D. Quadt and H. Kuhn, Capacitated lot-sizing with extensions: a review. *4OR* **6** (2008) 61–83.

[71] D. Quadt and H. Kuhn, Capacitated lot sizing and scheduling with parallel machines, back-orders and setup carry-over. *Nav. Res. Logist. (NRL)* **56** (2009) 366–384.

[72] C. Ramesh, R. Kamalakannan, R. Karthik, C. Pavin and S. Dhivaharan, A lot streaming based flow shop scheduling problem using simulated annealing algorithm. *Mater. Today Proc.* **37** (2021) 241–244.

[73] R. Ramya, C. Rajendran and H. Ziegler, Capacitated lot-sizing problem with production carry-over and set-up splitting: mathematical models. *Int. J. Prod. Res.* **54** (2016) 2332–2344.

[74] F. Sahling, L. Buschkühl, H. Tempelmeier and S. Helber, Solving a multi-level capacitated lot sizing problem with multi-period setup carry-over via a fix-and-optimize heuristic. *Comput. Oper. Res.* **36** (2009) 2546–2553.

[75] M.S. Sarbijan and J. Behnamian, Multi-fleet feeder vehicle routing problem using hybrid metaheuristic. *Comput. Oper. Res.* **141** (2022) 105696.

[76] M. Schlenkrich and S.N. Parragh, Capacitated multi-item multi-echelon lot sizing with setup carry-over under uncertain demand. *Int. J. Prod. Econ.* **277** (2024) 109379.

[77] C. Silva and J.M. Magalhaes, Heuristic lot size scheduling on unrelated parallel machines with applications in the textile industry. *Comput. Ind. Eng.* **50** (2006) 76–89.

[78] C.R. Sox and Y. Gao, The capacitated lot sizing problem with setup carry-over. *IIE Trans.* **31** (1998) 173–181.

[79] C. Suerie, Modelling of period overlapping setup times. *Eur. J. Oper. Res.* **174** (2006) 874–886.

[80] C. Suerie and H. Stadtler, The capacitated lot-sizing problem with linked lot sizes. *Manag. Sci.* **49** (2003) 1039–1054.

[81] C. Sung and C.T. Maravelias, A mixed-integer programming formulation for the general capacitated lot-sizing problem. *Comput. Chem. Eng.* **32** (2008) 244–259.

[82] C. Sung and C.T. Maravelias, A projection-based method for production planning of multiproduct facilities. *AIChE J.* **55** (2009) 2614–2630.

[83] P.P. Tambe, Selective maintenance optimization of a multi-component system based on simulated annealing algorithm. *Procedia Comput. Sci.* **200** (2022) 1412–1421.

[84] H. Tempelmeier and L. Buschkuhl, A heuristic for the dynamic multi-level capacitated lotsizing problem with linked lotsizes for general product structures. *OR Spectr.* **31** (2009) 385–404.

[85] S. Walton and R. Metters, Production planning by spreadsheet for a start-up firm. *PPC* **19** (2008) 556–566.

[86] X. Wan, J.F. Pekny and G.V. Reklaitis, Simulation-based optimization with surrogate models – application to supply chain management. *Comput. Chem. Eng.* **29** (2006) 1317.

[87] T. Wu and L. Shi, Mathematical models for capacitated multi-level production planning problems with linked lot sizes. *Int. J. Prod. Res.* **49** (2011) 6227–6247.

[88] T. Wu, K. Akartunali, J. Song and L. Shi, Mixed integer programming in production planning with backlogging and setup carryover: modeling and algorithms. *DEDS* **23** (2012) 1–29.

[89] Y. Zeiträg and J. Rui Figueira, A novel machine-learning rolling horizon heuristic for dynamic lot-sizing and job shop scheduling problems. *Int. J. Prod. Res.* (2025) 1–27.

[90] Y. Zeiträg, J. Rui Figueira and G. Figueira, A cooperative coevolutionary hyper-heuristic approach to solve lot-sizing and job shop scheduling problems using genetic programming. *Int. J. Prod. Res.* **62** (2024) 5850–5877.

[91] C. Zhang, D. Zhang and T. Wu, Data-driven branching and selection for lot-sizing and scheduling problems with sequence-dependent setups and setup carryover. *Comput. Oper. Res.* **132** (2021) 105289.

[92] X. Zhu and W.E. Wilhelm, Scheduling and lot sizing with sequence-dependent setup: a literature review. *IIE Trans.* **38** (2006) 987–1007.