Contents lists available at ScienceDirect

Food and Bioproducts Processing

journal homepage: www.elsevier.com/locate/fbp

Optimal production planning and scheduling in breweries

Georgios P. Georgiadis, Apostolos P. Elekidis, Michael C. Georgiadis*

Department of Chemical Engineering, Aristotle University of Thessaloniki, 54124 Thessaloniki, Greece

ARTICLE INFO

Article history:
Received 2 October 2020
Received in revised form 13 November 2020
Accepted 16 November 2020
Available online 24 November 2020

Keywords:
Planning and scheduling
MILP
Decomposition
Brewery

ABSTRACT

This work considers the optimal production planning and scheduling problem in beer production facilities. The underlying optimization problem is characterized by significant complexity, including multiple production stages, several processing units, shared resources, tight design and operating constraints and intermediate and final products. Breweries are mainly differentiated to the rest of the beverage industries in terms of long lead times required for the fermentation/maturation process of beer. Therefore, synchronizing the production stages is an extremely challenging task, while the long time horizon leads to larger and more difficult optimization problems. In this work we present a new MILP model, using a mixed discrete-continuous time representation and the immediate precedence framework in order to minimize total production costs. A number of test cases are used to illustrate the superiority of the proposed model in terms of computational efficiency and solution quality compared with approaches developed in other research contributions. The proposed model provides consistently better solutions and improvements of up to 50% are reported. In order to address large-scale problem instances and satisfy the computation limitations imposed by the industry, a novel MILP-based solution strategy is developed, that consists of a constructive and an improvement step. As a result, near-optimal solutions for extremely large cases consisting of up to 30 fermentation tanks, 5 filling lines and 40 products are generated in less than two hours. Finally, the proposed method is successfully applied to a real-life case study provided by a Greek brewery and near-optimal schedules are generated in relatively short CPU times.

1. Introduction

Current markets are characterized by increased competitiveness that forces industries to operate with miniscule profit margins. Moreover, intense cost pressures lead to the need of removing buffers and reduce stock of inventories. Especially in the food and beverage industries market trends call for an increased and highly volatile demand of a large number of high-quality products. Therefore, a more flexible and agile industrial production is critical for the viability and future growth of all industries. A holistic improvement and management of the production process can be achieved through efficient production planning and scheduling. This is a crucial decision-making process that is concerned with the allocation of scarce resources (e.g. equipment, utilities, manpower etc.) among competing activities over a given time horizon, in order to optimize one or more objectives (Pinedo, 2016). The quality of the production plans and schedules significantly affect the supply chain in its operational level. Efficient production planning and scheduling is extremely beneficial to all industries, since some of the induced benefits are increased productivity, lower production costs and reduced energy needs and waste. Therefore, a systematic way to take planning and scheduling decisions should be a cornerstone to any company that seeks to achieve economic and environmental sustainability.

Digitalization of manufacturing is attracting a lot of attention within all process industries, such as food and beverage, chemicals, pharmaceuticals, etc., and is expected to have a

* Corresponding author.
E-mail address: mgeorg@auth.gr (M.C. Georgiadis).
https://doi.org/10.1016/j.fbp.2020.11.008

significant impact on how the industry operates (Isaksson et al., 2017). However, the current industrial reality is different, since in most cases the production schedules are generated manually by the production engineers and the operators. The decision-making process and consequently the quality of the schedules is dependent on rules and heuristics that stem from the engineers' experience and understanding of the production process. Real-life scheduling problems are extremely complex since they consider complicated production facilities that comprise of multiple processing stages, parallel machines and production routes, an ever-expanding product portfolio, utilities (e.g. cold water, steam, electricity, etc.) and numerous constraints. Therefore, the manual generation of production schedules becomes an extremely difficult and tedious task, that requires, on a daily basis, a significant effort. In some cases, industries utilize commercially available scheduling tools (Intelligen Inc, 2020), in order to automate the procedure and to generate fast and feasible production schedules. However, the schedules are created based on simple heuristics that solely ensure their feasibility. Consequently, either when schedules are manually generated by the engineers or when simulation-based tools are employed, the extracted solutions are far from being optimal. Furthermore, generated and later executed schedules cannot be evaluated in terms of their efficiency, so production managers cannot assess the true potential benefits realized on the plant. As a result, productivity is reduced, resources are often underutilized, customers are dissatisfied and there are significant profit losses, which result to a decrease in the industries' competitiveness. The deployment of optimization-based tools in industrial problems can address these issues by assisting the production engineers into systematically improving their decisions, thus leading to important economic, environmental and social benefits (Harjunkoski, 2016).

The importance of production scheduling has been long recognised by academia and a plethora of contributions across different scientific communities can be found in the literature (Harjunkoski et al., 2014). An abundance of optimization-based algorithms has been proposed to address the production scheduling problem. Most of them express the production scheduling problem as a mixed-integer linear programming (MILP) problem, since it has been proved to be extremely flexible and rigorous, while ensuring optimality. The initial efforts of academic research have been focused on the introduction of general algorithms that would cover all types of scheduling problems. These efforts resulted into two major breakthroughs, the State-Task-Network (STN) (Kondili et al., 1993) and the Resource-Task-Network (RTN) (Pantelides, 1994). Both representations were generic enough and allowed researchers to efficiently address arbitrary networks of processes while their utilization resulted into simple discrete time MILP models. These seminal works inspired many researchers to further investigate the production scheduling problem, leading to a large number of contributions utilizing the basics of the proposed mathematical programming framework. However, the pursuit of academia to propose a unified general mathematical framework has been quickly abandoned, due to the intrinsic diversification of scheduling problems. Therefore, research turned to the development of MILP models that take advantage of problem and industry-specific characteristics (e.g. production and market environment, process type, interaction with other planning functions, storage policies etc.). Based on the way sequencing and timing of tasks are considered, the emerged MILP models are categorized into precedence-based and time-grid based. In precedence-based models, the representation of time is continuous, while the timing and sequencing constraints are expressed as precedence relationships between batches/lots, that can be either enforced between all pairs of batches (general precedence models) (Mendez and Cerda, 2004) or only between consecutive pairs (immediate precedence models) (Méndez et al., 2000). On the contrary, time-grid based models enforce timing and sequencing constraints by utilizing a single (Giannelos and Georgiadis, 2002) or multiple (Velez and Maravelias, 2013) external time references on which events (e.g. starting of a task) are mapped. Depending on the representation of time, these models are further classified into discrete (Yee and Shah, 1998) and continuous (Pinto and Grossmann, 1995; Castro et al., 2001). Both time representations display unique advantages and drawbacks. Discrete-time formulations can be easily applied to model features common to any industry, such as inventory levels, material balances and availability of utilities without introducing any non-linearities. However, high quality solutions require a fine discretization of time, which results into large models that often become intractable. This led researches into investigating continuous-time models (either time-grid based or precedence-based), especially when dealing with large production scheduling problems, since they tend to generate models that require fewer variables, while providing more precise solutions. A major disadvantage of continuous models compared to their discrete counterparts is that monitoring of material balances is not straightforward and may lead to nonlinear models. However, recently Cerdá et al. (2020) proposed a novel precedence-based model for optimal scheduling of multiproduct batch facilities that allows batch mixing and splitting as well as monitoring of inventories, without the need of introducing a discrete time grid. In recent years hybrid algorithms that take advantage of the strengths of both approaches have been proposed (Lee and Maravelias, 2018, 2019). An excellent review on the available MILP modelling approaches for production scheduling can be found in Méndez et al. (2006).

Production planning and production scheduling are two distinct but highly interconnected problems, since the output of the planning level is the input of the scheduling level. In particular, the planning level sets the production goals of the scheduling level. Their main differences lie on the studied time horizon and the type of decisions made. On the one hand, production planning problems have a time horizon of weeks to months, while batching/lot-sizing decisions are made to define production and inventory levels that fulfil the given orders at a minimum cost. On the other hand, the time horizon of production scheduling problems is much shorter (hours to a week), while the decisions made are more detailed and concern the unit allocation, timing and sequencing of processing tasks. The integrated production planning and scheduling problem displays a high combinatorial complexity, therefore early studies have mostly focused on separately addressing the two sub-problems. However, it has been clearly illustrated that the integrated and coordinated decision-making process across the various levels of manufacturing supply chain can yield a significantly improved overall economic performance (Castro et al., 2018). This is further enforced by the digitalization of manufacturing that allows for a seamless information flow between the various levels of decision-making. In recent years many studies have proposed optimization-based solutions for the integrated production planning and scheduling problem (Li and Ierapetritou, 2010; Kopanos et al., 2011; Aguirre et al., 2017), with a few of them illustrating implementations in industrial case studies (Kopanos et al., 2010; Sel et al., 2015).

All scheduling problems are NP-hard meaning that no known solution algorithm exists that is of polynomial complexity in the problem size (Kallrath, 2002). Despite the persistent efforts of the scientific community that led to efficient and improved modelling frameworks (Mostafaei and Harjunkoski, 2020; Lee and Maravelias, 2017; Janak and Floudas, 2008; Castro et al., 2004), the direct application of commercial solvers (e.g. CPLEX) has been illustrated to the solution of small- to medium-sized problems which in most cases do not represent the industrial reality. Real-life industrial problems are usually characterized by significant complexity, thus resulting in large MILP models that are computationally intractable. Another important consideration in industrial applications is the tight computational limitations. Managers and operators require methods that can promptly generate production schedules. This way the decision-making process can be agile in case of production disruptions e.g. machine breakdown, cancellation of an order etc. Furthermore, the scheduling solutions can be used to perform what-if analyses, which often offer important managerial insights. These industrial considerations guided many researchers into suggesting alternative methods, such as rule-based scheduling (Pierreval and Mebarki, 1997), variable neighbourhood search (Almada-Lobo et al., 2008), evolutionary algorithms (Pezzella et al., 2008), timed automata (Panek et al., 2008) and tabu-search (Zhang et al., 2007), which generate fast and feasible schedules, but cannot ensure optimality. A very successful approach is the combination of mathematical programming with decomposition techniques that reduce the size of the initial problem, thus large industrial problems can be solved in acceptable computational times, while providing near optimal solutions. Bassett et al. (1996) proposed multiple time-based decomposition algorithms that can address large-scale scheduling problems. A temporal decomposition scheme has been also suggested by Dimitriadis et al. (1997), through the employment of an RTN-based rolling horizon algorithm. A two-step decomposition algorithm that is based on the idea of quickly constructing an initial feasible solution that is later improved through reallocation or reinsertion actions have been initially proposed by Roslöf et al. (2000). Inspired by this work, Kopanos et al. (2010) successfully addressed the production planning and scheduling problem of a yoghurt production line. The scheduling problem of a pharmaceutical industry has been tackled by Stefansson et al. (2011) through a spatial decomposition strategy which sequentially optimized the schedule for each processing stage. Aguirre et al. (2017) proposed an iterative-improvement algorithm in combination with a rolling horizon algorithm to address the integrated planning and scheduling problem of multi-stage plants. Borisovsky et al. (2019) were able to achieve significant reductions in changeover times for real-life applications by proposing a novel MILP-based solution method that combined an evolutionary algorithm and a decomposition technique. In recent years, order-based decomposition approaches have successfully addressed large scale, real-life industrial applications. Elekidis et al. (2019) proposed two MILP-based decomposition strategies for the scheduling problem of a large consumer goods industry. The authors achieved great changeover reductions, improving the overall plant productivity. Georgiadis et al. (2020) proposed a three-step scheduling solution based on an aggregated MILP model to consider a large multistage food industrial facility. Recently, Basán et al. (2020) addressed the scheduling problem of flexible manufacturing plants that involve multiple multipurpose units. They developed an MILP-decomposition algorithm to integrate optimal scheduling with redesign options, in order to avoid bottlenecks and ensure a balanced production.

Many practical cases of optimization-based scheduling can be found in the open literature for a wide variety of industrial sectors, such as food industries (Georgiadis et al., 2019a), chemical (Nie et al., 2014), pharmaceutical (Liu et al., 2014), steel (Gajic et al., 2017) and consumers goods industries (Van Elzakker et al., 2012). Despite the wide range of real-life applications, the industrial need for new optimization schemes is still growing. Many open issues remain for the efficient industrial application of optimization-based scheduling solutions. Most importantly, the majority of the applications are limited in handling small or medium sized study cases, rather than realistic large-scale industrial problems. Other remaining challenges are related to issues such as the reduction of the required computational time, dynamic rescheduling, the connection with the plant's IT infrastructure e.g. ERP system, and the interactive visualization of the generated schedules through the incorporation of the suggested methods into proper computer-aided tools. A comprehensive review on optimization-based scheduling of real-life industrial case studies, alongside an analysis on the major issues that impede their application can be found in Georgiadis et al. (2019b).

Beverage industrial facilities display specific production characteristics e.g. multiple mixed batch and continuous processes, an ever-expanding product portfolio, intermediate due dates, etc., that make the optimal production planning and scheduling of real-life industrial problems extremely challenging. Only few contributions have addressed the integrated production scheduling of the soft drink industry, using either optimization-based (Ferreira et al., 2009, 2012; Sel and Bilgen, 2014; Toscano et al., 2020) or non-exact methods (Toledo et al., 2009). The optimal production scheduling of breweries is an even more difficult problem. This originates mainly from the very long lead times that typically characterize these industries. In particular, liquid preparation (fermentation and maturation) lasts from 3 to 41 days, therefore, a long planning and scheduling horizon is necessary making the synchronization of the various processing stages a very difficult task. Moreover, larger models are unavoidably generated which become intractable when studying real-life industrial cases. As a result, few works have addressed the production planning and scheduling problem in breweries. Kopanos et al. (2011) proposed a novel mixed discrete-continuous MILP model for the optimal production planning and scheduling of parallel continuous processes. The proposed model effectively addressed industrial-scale problems of a real brewery, requiring very low computational times. However, their analysis focused solely on the bottling lines and was based on the assumption that the packing stage constitutes the production bottleneck, which is not always the case. Baldo et al. (2014) were the first to study the optimal integrated production planning and scheduling problem of a beer production facility. They assumed that the production can be divided into two processing steps, liquid preparation and bottling. Based on this valid simplification they developed a novel MILP model and proposed MILP-based heuristics to solve large-scale problems. Recently, Maravelias et al. (Lee and Maravelias, 2020) employed the general discrete and continuous algorithm (DCA) (Lee and Maravelias, 2018) for the optimal production lot-sizing and

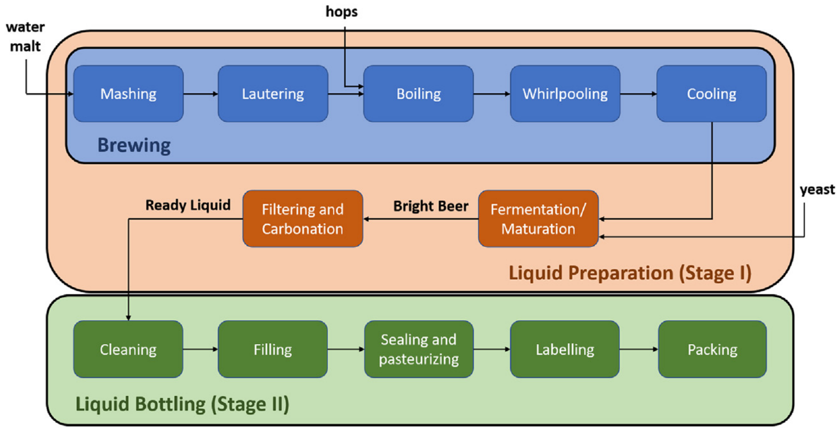

Fig. 1 – Description of the beer production process.

scheduling of a large brewery. The authors modelled the beer production as a four-stage problem including brewing, fermentation, maturation and bottling and optimized schedules were generated. Due dates were not modelled, rather monthly production targets were imposed, and the main objective was profit maximization.

The main contribution of this work is the development of a novel optimization-based solution approach for the integrated planning and scheduling problem of breweries. A new MILP model based on a mixed discrete-continuous time representation is developed. In order to reduce the size of the generated model, the production bottlenecks of the process are explicitly modelled, while the considered horizon is divided into two sub-horizons. In the first one a detailed optimal production schedule is extracted, and in the second only planning decisions are taken. To the best of our knowledge the only model in the literature that addresses this problem has been proposed by Baldo et al. (2014). An extensive analysis is included that proves the superiority of the developed model both in terms of solution quality and computational time. However, the large number of involved tanks, production lines and products and most importantly the large lead times, lead to computationally intractable models especially when dealing with industrial cases. Therefore, we propose a novel solution strategy that consists of a constructive and an improvement step. In the first step an initial solution is generated which is then improved in the second step of the proposed algorithm. Finally, the proposed model and solution strategy is successfully applied to an industrial case study concerning a large Greek beer production facility.

The rest of this work is organized as follows. In Section 2 the beer production process is described in detail and the optimization problem is formally stated. A thorough presentation of the developed MILP model and the proposed solution strategy is provided in Section 3. Section 4 presents the validation and computational assessment of the proposed methods in a number of test cases followed by their application in a real-life industrial case. Finally, concluding remarks are drawn in Section 5.

2. Problem statement

Beer production is a complex process that comprises of multiple production steps utilizing numerous shared resources. Any beer type consists of four main ingredients, in particular, water, malt (from barley grains), hop (responsible for the bitter taste of beer) and yeast (saccharomyces cerevisiae for ale beer or saccharomyces pastorianus for lager beer). The various beer products are diversified in terms of raw materials and the required processing time in each production step. Despite the distinct process required for each beer type, all products go through the same processing steps, which can be categorized into two main production stages, liquid preparation and bottling (Fig. 1).

Some breweries produce their own malt; therefore, a malting process is taking place prior to the brewing process. The malting process is divided into three subprocesses; steeping, where the humidity of the grain is increased, germination, which transforms the grains into malt and finally drying in kiln, to remove most of the humidity from the malt. In this study we assume that the malt is a raw material that is ready to be brewed, therefore the malting process is not considered. In the liquid preparation stage two main processes take place, specifically, brewing and fermentation/maturation. The brewing process consists of several batch tasks, namely mashing, lautering, boiling, whirlpooling and cooling, that transform the raw materials into different worts. Mashing involves the addition of water into the prepared malt and the heating of the mixture, while in lautering the mixture is filtered from any solids. Then the hops are added, and the mixture is heated in the boiling process. Finally, the wort is filtered (whirlpooling) and quickly cooled (cooling). In the next processing step, the yeast is added into the cooled wort and the fermentation/maturation process begins. This subprocess constitutes one of the main production bottlenecks, since it lasts 3–41 days, depending on the type of beer produced. At this moment beer of a given wort type is obtained, referred to as bright beer. Finally, bright beer is transferred from the fermentation/maturation tanks to bright beer tanks (BBTs), where it is filtered, diluted and carbonated. At the end of the liquid preparation process the beer is referred to as ready beer or ready liquid.

Bottling is the last stage of the production process, where the ready liquid is bottled in cans, bottles or kegs and then the final products are packed and palletized. Multiple subprocesses take place during the bottling process. First the returnable bottles are cleaned and sterilized, while cans and kegs are simply washed. Next the filling subprocess takes place, which is the main production bottleneck of the bottling stage. The products are then sealed and pasteurized in a bath of hot water to ensure that they are not infected by any harmful microorganisms. Finally, labelling, packing and palettizing takes place and the final products are loaded on a transport vehicle or stored in a warehouse.

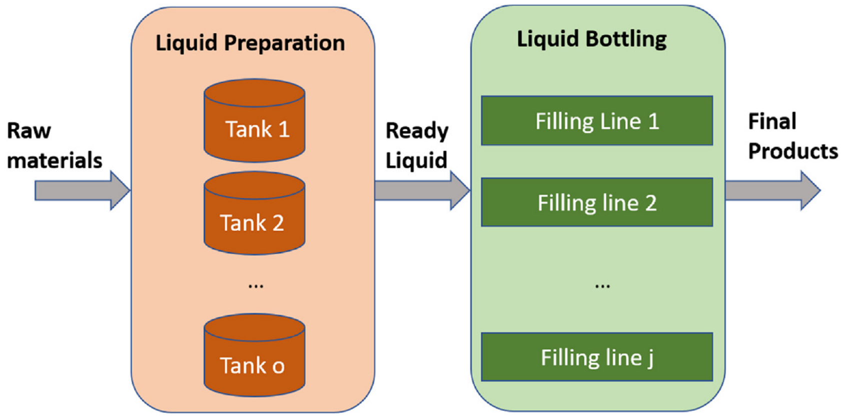

Fig. 2 – Simplified process description focusing on production bottlenecks.

The brewery industry, like most food and beverage industries, can be described as a make-and-pack facility, where in the initial stages the raw materials are processed based on a given production recipe and then are packaged in the desired final form. In order to efficiently address the optimal production planning and scheduling problem of breweries, only processes that constitute the main bottlenecks of production are explicitly modelled. The most challenging task in the first stage is the proper utilization of the fermentation and maturation tanks. The rest of the subprocesses of this stage only take a few hours, when the fermentation/maturation task lasts multiple days, thus making it the bottleneck of the liquid preparation stage. Similarly, the limited capacity of the filling subprocess makes it the most difficult task of the second stage and it can therefore be considered as the production bottleneck. Moreover, the more ready liquid is bottled in the filling process, the faster the tanks empty and therefore become available to process a new batch of liquid. As a result, the beer production process is simplified, leading to relatively small sized models, without sacrificing any kind of information required for the generation of feasible and optimal production schedules (Fig. 2).

Based on the aforementioned simplifications, the brewery plant can be described as a multiproduct, multistage facility that combines both a batch (fermentation/maturation) and a continuous (filling) process with multiple parallel units. The first stage involves a number of tanks, which are non-identical in terms of capacity, but can process all liquids. In contrast, each filling line of the second stage can only process a specific subset of the final products, depending on the packing and bottling type of the line. Tanks can only prepare a single liquid at a time and likewise filling lines can only bottle a single product at a time. In terms of availability of connections, a tank can simultaneously supply multiple lines with ready liquid, however each line can receive ready beer from a single tank at a time. Furthermore, tanks must be cleaned in-between the fermentation/maturation process of two different batches, thus a sequence-independent setup time is necessary. On the contrary sequence dependent setup times for cleaning and/or machine adjustments are required in the filling lines, whenever a changeover of liquids and/or packages occurs. There are no intermediate storage vessels, however the ready liquid can be temporarily stored in the fermentation and maturation tanks.

The current industrial reality in most plants imposes the production plans and schedules to be generated manually by the decision makers. The large number of decisions to be taken alongside the tight operational, logistical and technical constraints result to an extremely complex optimization problem. In addition, the long lead times require extended long planning horizon compared to other industries, while the generated schedules should ensure the proper synchronization of the liquid preparation and bottling stages. Thus, it is very difficult for the production engineers to consider the integrated planning and scheduling problem even using simple heuristic rules. In order to propose feasible schedules, the decision-making process is divided into two steps. First, the production plan for the sterilization and maturation tanks is generated. In this step the timing of all filling and emptying operations in each tank and the allocation of liquids into tanks is defined. The plans are determined for a monthly horizon based on the given demand and the capacity limitations of the units. At this point the goal of the production engineers is to utilize the tanks as much as possible while trying to reduce backlogs and maintain a relatively small inventory. Then the plan is thrown over the wall to the department responsible for production scheduling, which generates a feasible schedule for the filling lines. Here, the tank to filling line connections are determined (which tank will provide liquid to which line), and it is decided when will each filling process take place (timing) and at what order will every final product be processed (sequencing).

The decision-making procedure described above lacks efficiency since the two main production stages of the plant are considered separately without the employment of optimization-based methods. Therefore, the realized production plans and schedules are far from being optimal, thus productivity is decreased, and total profits are reduced. It is clear that the efficient integration of both planning and scheduling decision is an area with great potential for improvement, with significant benefits for the brewery industry. The main goal of this work is to develop an MILP-based solution method for the integrated production planning and scheduling problem that provides near-optimal decisions in short computational times.

The problem under consideration can be formally stated as follows.

Given:

- A planning horizon H that is divided into a set of time periods $t \in T$. The horizon is further divided into two subsets of time periods, $t_1 \in T_1$and $t_2 \in T_2$ $(T = T_1 \cup T_2)$. In the first precise production schedules are determined, while in the latter only production plans are generated.

- A set of fermentation/maturation tanks $o \in O$ and a set of filling lines $j \in J$.
- A set of liquids $l \in L$ to be prepared and a set of final products $i \in I$ that have to be produced within the given horizon.
- The multidimensional set I_l that denotes whether product i contains liquid l.
- The mapping set I_j that defines the set of products i that can be processed on filling line j.
- All production related parameters, in particular, demand $\zeta_{i,t}$, liquid preparation time λ_l, filling rate for each final product $\rho_{i,J}$, capacity of each tank χ_o, and quantity of liquid required for a single unit of product i $\pi_{i,l}$.
- A sequence-dependent setup for cleaning and/or machine changes whenever there is a changeover of production between two final products i and i'. Every changeover task requires a specific time $\gamma_{i,i'j}$.
- The cost coefficients associated with inventory σ_i, backlog β_i and changeover operations $k_{i,i'j}$.

Determine:

- The planning decisions for the liquid preparation stage. More specifically, determine the filling and emptying operations in each tank as well as the material balance (amount of ready liquid) in each tank.
- The amount of liquid that is being transferred from each tank to each filling line.
- The allocation of products into filling lines, as well as the sequencing between products in each line and the completion time of each filling operation.
- The production amounts of final products as well as the product inventories and backlogs.

In order to minimize an economic objective that includes inventory, backlog and changeover costs.

All data used are deterministic, and as such any type of uncertainty is not considered in this work. It is also assumed that raw materials are always available. Resource limitations, such as manpower or utilities, e.g. cold water, electricity, are not considered. No intermediate storage vessels exist; however, the ready liquid can be stored in the fermentation and maturation tanks. We assume an instantaneous transfer of liquid between the two stages and that the fermentation/maturation process in a tank only starts at the beginning of a time period and is completed at the end of a time period.

3. MILP-based solution method

An MILP model is presented to efficiently address the integrated production planning and scheduling problem for a multistage multiproduct facility typically found in the brewing production process. The model is based on a precedence-based framework that utilizes a mixed discrete-continuous time representation, inspired by the works of Kopanos et al. (2011) and Baldo et al. (2014). Operational and technical constraints, such as demand requirements and tank capacities, as well as, specific characteristics of the production are incorporated to produce feasible plans that minimize total production costs, which, in this study, comprises of the inventory, backlog and changeover cost terms.

3.1. MILP model

The detailed modeling of all processing steps of the brewery facility would result to large and complicated models. Therefore, only the main production bottlenecks are considered, in particular, the fermentation/maturation process in the liquid preparation stage and the filling process in the liquid bottling stage. Thus, the facility at hand is reduced to a two-stage multiproduct plant. This make-and-pack type of process is very common in food and beverage facilities. Therefore, an abundance of various techniques that can optimally solve this type of problems can be found in the literature. However, beer production displays characteristics that significantly differentiate them to other production processes. Compared to other food and beverage industries, the preparation step (fermentation/maturation in the case of breweries) requires a large processing time that spans from some days to multiple weeks, leading to large production lead times. Hence, planning must be considered in synchronization with short term scheduling decisions, since product preparation lasts more than the usual scheduling horizon (one week). In case of optimizing just the scheduling decisions of the filling process (Stage 2), there is a high risk of generating schedules that overestimate the capacity of the fermentation/maturation tanks (Stage 1), thus leading to an infeasible solution.

In order to address this optimization problem, an MILP model has been developed, relying on a mixed discrete-continuous time representation. The discrete time grid has a period length of one day and is used to seamlessly monitor the production, inventory and backlog levels of both stages. A lot-sizing model is introduced for the planning decisions of both stages, that considers the given processing times of the tasks and the capacity of the production units. Within each time period a continuous-time representation is utilized, and constraints inspired by the immediate precedence framework are incorporated to determine the sequencing decisions in the liquid bottling stage (Stage 2). Note that sequencing decisions are not required in the first stage, since it does not involve any sequence-dependent setup times. The planning horizon of interest is divided into two sub-horizons. In the first one (T_1), both planning and scheduling decisions are considered, while in the second one (T_2) a coarser optimization is done, than only determines the lot-sizing and unit utilization decisions. Fig. 3 portrays the employed time grid, as well as the decisions taken for each stage in each sub-horizon. The orange oval shape contains the considered decisions for Stage 1, while the green oval shape displays the made decisions for Stage 2. Blue-colored text denotes planning decisions, while red-colored text signifies the scheduling decisions. The intersection of the two shapes encloses key decisions that connect the two stages, specifically, the amount of liquid that is transferred from the tanks to the filling lines.

Let us describe the main decision variables of the developed model for the integrated planning and scheduling problem. The liquid is transferred into the tanks and the fermentation/maturation process starts. When the required processing time, λ_l, passes, then an amount $L^P_{o,l,t}$ of liquid gets ready. Binary variables, $Y^1_{o,l,t}$, denote that liquid l in tank o gets ready in time period t. The ready liquid is either used on filling lines j to produce items i in period t ($L^T_{o,j,i,t}$) or is stored in tank o for future production ($L^S_{o,l,t}$). In case the ready liquid is used to produce items on filling lines, then an amount of item i ($Q_{o,j,i,t}$) is processed on filling line j in period t made of liquid fed by tank o. This amount is used to satisfy the demand on the current or previous time periods ($t' \leq t$) or is stored to meet future demand of item i ($t' > t$). Note that the outputs of Stage 1 (liquid preparation) are the inputs of Stage 2 (liquid bottling), so the production in the filling lines takes place only when

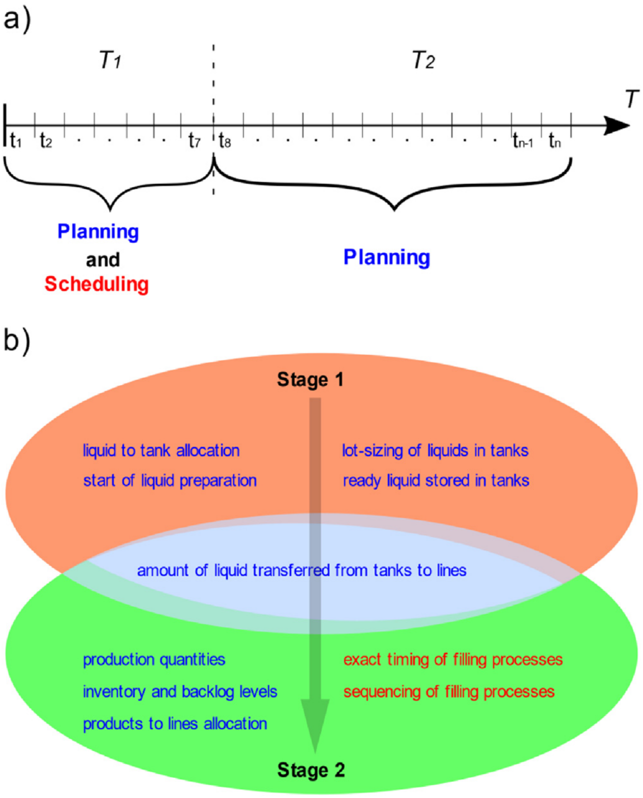

Fig. 3 – a) Time representation and description of sub-horizons, b) considered planning and scheduling decisions in each stage.

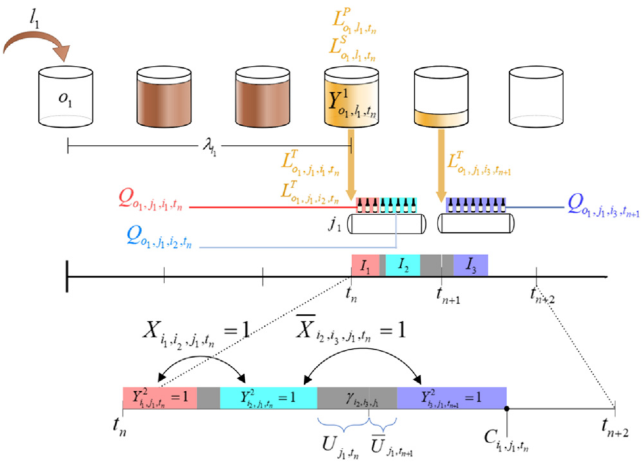

Fig. 4 – Illustration of main decision variables.

there is available ready liquid to be fed from the fermentation/maturation tanks. In terms of scheduling decisions, unit allocation variables ($Y^2_{i,j,t}$) are used to denote that a product *i* is processed in line *j* in time period *t* and two sets of immediate precedence variables ($X_{i,i',j,t}$ and $\overline{X}_{i,i',j,t}$) are employed to indicate direct precedence of tasks. The first is enabled whenever there is direct precedence or production between two final products, *i* and *i'*, in line *j* in the same period *t*, while the latter indicates precedence of filling tasks between consecutive periods. Continuous variables $U_{j,t}$ and $\overline{U}_{j,t}$ are used to properly model changeovers between tasks in consecutive time periods. Lastly timing variables $C_{i,j,t}$ are employed to signify the completion of a filling task of product *i* in time period *t*. An overview of the main decision variables is illustrated in Fig. 4.

Next, we present the developed model, categorizing the constraints based on the production stage and the types of decisions taken. To facilitate the presentation of the model, we use lowercase Latin letters for indices, uppercase Latin letters for variables and lowercase Greek letters for parameters. In the future we will refer to this model as GEG.

Stage 1 (Liquid preparation)

In the first stage, constraints are imposed for the accurate modelling of the lot-sizing of the fermentation/maturation tanks. More specifically, they must guarantee that the processed liquid batches do not exceed the capacity of the fermentation tanks and that the liquids remain in the tanks at least for the required fermentation/maturation processing time.

Constraints (1) ensure that if a liquid gets ready in time period t ($Y^1_{o,l,t} = 1$), then no ready liquid is stored in the tank during the previous λ_l time periods. During this period the fermentation/maturation process of the liquid takes place. In order to have an amount $L^P_{o,l,t}$ of ready liquid in time period t, the tank must be empty in time period $t - (\lambda_l + 1)$, so that it can receive the liquid to initiate its preparation (fermentation/maturation) process. Constraints (2) are introduced to guarantee that at most one batch of liquid gets ready in a tank within a time segment equal to the fermentation/maturation time. Finally, constraints (3) impose the upper bound on the amount of liquid getting ready based on the available capacity of the fermentation/maturation tanks.

$$\sum_{l'} \sum_{t'=t-\lambda_{l'}-1}^{t-1} L^S_{o,l',t'} \le M \cdot (1 - Y^1_{o,l,t}) \quad \forall o,\ l,\ t \qquad (1)$$

$$\sum_{l} \sum_{t'=t-\lambda_{l'}}^{t} Y^1_{o,l,t'} \le 1 \quad \forall o,\ t \qquad (2)$$

$$L^P_{o,l,t} \le \chi_o \cdot Y^1_{o,l,t'} \quad \forall o,\ l,\ t \qquad (3)$$

Stage 1 and stage 2

Constraints (4) are responsible for connecting the decision variables of the two stages, while they monitor the liquid balance between them. More specifically, they state that the stored amount of liquid *l* in tank *o* in time period *t* ($L^S_{o,l,t}$) is equal to the stored amount in the previous period plus the amount of liquid getting ready in period t ($L^P_{o,l,t}$), minus the liquid that is transferred to the filling lines.

$$L^S_{o,l,t} = L^S_{o,l,t-1} - \sum_{i\in I_l} \sum_{j\in J_i} L^T_{o,j,i,t} + L^P_{o,l,t} \quad \forall o,\ l,\ t \qquad (4)$$

Stage 2 (Liquid bottling)

The second stage requires a more detailed model since additional to the lot-sizing and unit allocation constrains it also considers the timing and sequencing decisions for the filling lines. In order to generate the required modelling constraints, the immediate precedence framework is employed within a mixed discrete-continuous time representation (Kopanos et al., 2011).

Material balance constraints. Material balances for every final product are guaranteed by constraints (5). At the end of each time period t, the inventory, $S_{i,t}$, and backlog, $B_{i,t}$, are monitored based on the daily production, demand and the inventory and backlog levels in the previous time period *t-1*. The number of products *i* that use liquid fed by tank *o* and processed in line *j* at time period *t* is expressed by constraints (6).

$$S_{i,t} - B_{i,t} = S_{i,t-1} - B_{i,t-1} + \sum_{j \in J_i} \sum_{o} Q_{o,j,i,t} - \zeta_{i,t} \quad \forall i,\ t \tag{5}$$

$$Q_{o,j,i,t} = \pi_{i,l} \cdot L^{T}_{o,j,i,t} \quad \forall o, j, i \in I_j,\ t \tag{6}$$

Line utilization constraints. The constraints below introduce the line utilization variable, which is enabled, i.e. $V_{j,t} = 1$, when a filling line *j* is used in time period *t*. In particular, constraints (7) ensure that a filling line *j* is utilized in time period *t*, if at least one product *i* is processed in this line and time period. Furthermore, constraints (8) force the unit utilization variable to take a value of 0, in case no product is processed in that particular line and time period t.

$$V_{j,t} \geq Y^{2}_{i,j,t} \quad \forall i,\ j \in J_i,\ t \in T \tag{7}$$

$$V_{j,t} \leq \sum_{i} Y^{2}_{i,j,t} \quad \forall j \in J_i,\ t \in T \tag{8}$$

Sequencing and timing constraints. Binary variables $X_{i,i',j,t}$ are introduced to define the immediate precedence relation between two products *i* and *i'* in line *j* and time period *t*. Moreover, we employ the binary variables $W^{F}_{i,j,t}$ and $W^{L}_{i,j,t}$, which define the first and last product being processed in line *j* and time period *t* accordingly. Constraints (9) and (10) guarantee that if a product is processed in filling line *j* and time period t ($Y^{2}_{i,j,t} = 1$), it will have at most one predecessor and one successor. In case product *i* is processed first in line *j* and time period t, then it has no predecessor and similarly if it is processed last it has no successor. Finally, tightening TSP-based constraints (11) are introduced, which specify the exact number of active sequencing variables. More specifically, they ensure that if line *j* is used in time period *n*, then the total number of enabled sequencing variables is equal to the number of products being processed minus 1. Otherwise, all sequencing variables for that specific line and time period are forced to zero.

Timing considerations are imposed by the next two constraints. Constraints (12) guarantee that the filling process for a product *i'* that is processed right after product *i*, must be completed after the completion of product *i* plus the required processing and changeover time. This type of constraint is formulated as a big-M constraint, meaning that when the succession relation is absent ($X_{i,i',j,t} = 0$), then the constraint becomes inactive. The big-M parameter used is ω, which corresponds to the daily time availability of each filling line. In this particular study this is assumed to be 24 h. Furthermore, constraints (13) are employed to ensure that the filling process for each product is completed after the required processing time.

$$\sum_{i' \neq i, i' \in I_j} X_{i',i,j,t} + W^{F}_{i,j,t} = Y^{2}_{i,j,t} \quad \forall i,\ j \in J_i,\ t \in T_1 \tag{9}$$

$$\sum_{i' \neq i, i' \in I_j} X_{i,i',j,t} + W^{L}_{i,j,t} = Y^{2}_{i,j,t} \quad \forall i,\ j \in J_i,\ t \in T_1 \tag{10}$$

$$\sum_{i \in I_j} \sum_{i' \neq i, i' \in I_j} X_{i,i',j,t} + V_{j,t} = \sum_{i \in I_j} Y^{2}_{i,j,t} \quad \forall j,\ t \in T_1 \tag{11}$$

$$C_{i',j,t} \geq C_{i,j,t} + \sum_{o} \rho_{i,j} \cdot Q_{o,j,i,t} + \gamma_{i,i',j} \cdot X_{i,i',j,t} - \omega \cdot (1 - X_{i,i',j,t}) \quad \forall i,\ i' \neq i,\ j \in (J_i \cap J_{i'}),\ t \in T_1 \tag{12}$$

$$C_{i,j,t} \geq \sum_{o} \rho_{i,j} \cdot Q_{o,j,i,t} \quad \forall i,\ j \in J_i,\ t \in T_1 \tag{13}$$

Sequencing constraints between adjacent periods. In order to model changeover operations between processes that take place in adjacent periods, we introduce binary variables $\overline{X}_{i',i,j,t}$. Constraints (14) and (15) state that this variable is active only for products *i'* that are processed first in line *j* and in time period *t* and products *i* that are processed last in line *j* and time period *t-1*. Furthermore, continuous and positive variables, $U_{j,t}$ and $\overline{U}_{j,t}$, are introduced, to represent time fractions of changeover operations between adjacent periods. Constraints (16) are imposed to facilitate the proper incorporation of these newly introduced variables in the model. Assume there is a changeover between a product processed in period *t-1* and another one processed in period *t*. Then the fraction of the changeover operation that is performed in time period *t-1* is represented by $U_{j,t-1}$, while the time fraction of the changeover that takes place in time period *t* is modeled by $\overline{U}_{j,t}$. Of course, the addition of these times must equal the total changeover time $\gamma_{i,i',j}$.

$$W^{F}_{i,j,t} = \sum_{i' \neq i, i' \in I_j} \overline{X}_{i',i,j,t} \quad \forall i,\ j \in J_i,\ t \in T_1 \tag{14}$$

$$W^{L}_{i,j,t-1} = \sum_{i' \neq i, i' \in I_j} \overline{X}_{i,i',j,t} \quad \forall i,\ j \in J_i,\ t \in T_1 : t > 1 \tag{15}$$

$$\overline{U}_{j,t} + U_{j,t-1} = \sum_{i \in I_j} \sum_{i' \neq i, i' \in I_j} \gamma_{i,i',j} \overline{X}_{i,i',j,t} \quad \forall j,\ t \in T_1 : t > 1 \tag{16}$$

Time availability constraints. Constraints (17) bound the operations in a filling line, based on the available production time. In particular, the summation of the changeover times, either within the same time period or between adjacent time periods, and the total processing time of all products being processed, must be less than the total available production time of the line. Note that all sequencing constraints are specified only for those subperiods that belong to the planning and scheduling sub-horizon ($t \in T_1$).

$$\overline{U}_{j,t} + U_{j,t-1} + \sum_{i \in I_j} \sum_{o} \rho_{i,j} \cdot Q_{o,j,i,t} + \sum_{i \in I_j} \sum_{i' \neq i, i' \in I_j} \gamma_{i,i',j} X_{i,i',j,t} \leq \omega \quad \forall j,\ t \in T_1 : t > 1 \tag{17}$$

Lot-sizing constraints. Finally, constraints (18) and (19) bound the production in the liquid bottling stage based on the given processing rates for each product and the available daily production time for each line. Note that in contrast to the timing and sequencing constraints, lot-sizing constraints are constructed for all time periods of the given horizon.

$$\sum_{o} Q_{o,j,i,t} \leq \frac{\omega}{\rho_{i,j}} \cdot Y^{2}_{i,j,t} \quad \forall i,\ j \in J_i,\ t \tag{18}$$

$$\sum_{o} \sum_{i \in I_j} \rho_{i,j} \cdot Q_{o,j,i,t} \leq \omega \quad \forall j \in j_i,\ t \tag{19}$$

Objective function
The overarching goal of the optimization problem is to minimize the total production costs, which is modelled by three cost terms, inventory, backlog costs and changeover cost. The changeover cost term is only defined for the subperiods of the planning and scheduling horizon ($t \in T_1$), since sequencing decisions are considered only for these time periods.

$$\text{minimize} \quad \sum_i \sum_t (\sigma^i \cdot S_{i,t} + \beta^i \cdot B_{i,t}) + \sum_i \sum_{i', i' \neq i} \sum_{j \in (J_i \cap J_{i'})} \sum_{t \in T_1} \kappa_{i,i',j} \cdot (X_{i,i',j,t} + \overline{X}_{i,i',j,t}) \tag{20}$$

3.2. MILP-based solution strategy

For the solution of the above MILP model, the direct application of commercially available solvers, e.g. CPLEX, GUROBI etc., requires large computational effort, with significant solution times and suboptimal production plans. This is especially noticeable when dealing with real-life industrial applications that may not be solved due to their inherent complexity. This is unacceptable, since the developed method must always generate a feasible solution. Moreover, the industry works on a very tight schedule, therefore strict time limitations are imposed to the generation of a solution. To ensure the viability of the proposed method as a computer-aided tool that can be a part of the infrastructure in the plant, it must provide solutions in computational times accepted by the industry. Thus, to satisfy these prerequisites a decomposition strategy is employed that guarantees the generation of near optimal production plans while reducing the combinatorial complexity of the optimization problem. A two-step decomposition technique, consisting of a constructive and an improvement step, is proposed. In the first part, an initial good solution is promptly generated, while in the second part an iterative method is used to improve the initial solution. The following subchapters describe the developed solution algorithm in detail.

3.2.1. Constructive step

In order to generate a feasible and good initial solution, a spatial decomposition approach is introduced, where the two production stages are considered independently. Main goal of this method is to disaggregate the binary decisions of each stage, thus decreasing the complexity of the initial model. We end up with two MILP-subproblems, one for Stage 1 (GEG_S1) and one for Stage 2 (GEG_S2), which are solved in that order. More specifically, GEG_S1 is solved to determine decisions related to the fermentation/maturation tanks (in which tank will the liquids be prepared, when they are going to be ready and the corresponding amount that will be ready during the given horizon). Then this information is used in GEG_S2 to optimize the planning and scheduling decisions of the filling lines and finally generate the production plan for the whole process. The order in which the models are solved (first Stage 1 and then Stage 2) has been decided since the alternative (first GEG_S2 and then GEG_S1) could potentially lead to infeasibilities. This may occur due to an overestimation in the capacity of resources of the first stage. The productions plans for the filling lines generated by GEG_S2 are inapplicable in case the required amount of ready liquid exceeds the available capacity of the tanks in the first stage. On the other hand, this is not an issue in the suggested solution strategy, since Stage 2 is more flexible than Stage 1. Due to the natural flow of material in the problem at hand and the capability of storing or backlogging final products, the filling lines can always adapt to the production plans of the fermentation/maturation tanks. This is crucial since the proposed solution method must ensure the generation of production plans and schedules for any possible case that could occur in the industrial facility. So, the constructive step is further split into two steps. The first one focusing on Stage 1 and the second on Stage 2.

Sub-step 1 (Stage 1). In order to develop the model for the liquid preparation stage, we utilize a subset of the constraints from model GEG. Despite our emphasis on the first stage, we must also take into account additional constraints from the liquid bottling stage. It is essential to include this information in order to avoid the generation of bad production plans that would lead to increased inventory and backlogging costs. If we ignore the incorporation of this information in the model, we could even end up with infeasible production plans. For example, if we do not consider the processing capability of the filling lines, then the model could impose a tank filling plan that prepares an amount of liquid that overwhelms the filling lines. So, the tanks could not be emptied in time and could not be ready for the initiation of the fermentation/maturation process of the next batch, thus making the generated plan inapplicable.

The goal of this model is to determine the tank filling operations by minimizing potential inventory and backlogging costs (21). Constraints (1) – (3) are included to ensure that the operational constraints for the first stage are considered. Constraints (4) must be incorporated in the model to properly model the interaction of liquid between Stage 1 and Stage 2. Furthermore, constraints (5) are necessary to monitor the inventory and backlog levels based on the given demand and optimized production. Finally, constraints (18) and (19) are responsible for providing the capacity information of the filling lines so that infeasible solutions are avoided. The optimized planning decisions for the tanks, in particular the time period t in which each liquid l gets ready in tank o ($Y^1_{o,l,t}$) and the corresponding amount ($L^P_{o,l,t}$), are saved in parameters $\hat{Y}^1_{o,l,t}$ and $\hat{L}^P_{o,l,t}$ respectively, to be later used in the second sub-step of the constructive step.

GEG_S1.

$$\text{minimize} \quad \sum_i \sum_t (\sigma^i \cdot S_{i,t} + \beta^i \cdot B_{i,t}) \tag{21}$$
$$\text{s.t.} \quad \text{Constraints} \quad (1) - (5), (18), (19)$$

Sub-step 2 (Stage 2). In the next step, the proposed method solves model GEG_S2 for the second stage considering the solution of GEG_S1. In particular, it receives as inputs the optimized decisions that determine when a liquid gets ready and the corresponding amount. This information is considered in the model by incorporating constraints (22) and (23). More specifically, constraints (22) ensure that a liquid gets ready only at the time imposed by the solution of the first substep ($\hat{Y}^1_{o,l,t} = 1$) and guarantees that the capacity limitations of the tanks are not violated. Note that the binary decisions for the timing of the filling plan are fixed to the solution provided by the previous step ($Y^1_{o,l,t} = \hat{Y}^1_{o,l,t}$). On the contrary, the amount that gets ready is reoptimized in this step to increase the flexibility of the proposed method. Of course, the respective non-negative variable is lower bounded by the solution of the previous step, so that the tank filling plans generated by GEG_S1 are respected. Additionally, constraints (23) guarantee that the tank will be empty and ready to receive the liquid and that the liquid will solely occupy the tank during the fermentation/maturation process. Furthermore, constraints (4) from model GEG are included to ensure that a production in the filling lines occurs only if there is a ready liquid available. Moreover, we include all constraints related to the second stage (5)-(19). Finally, the objective of this model is to minimize the total production cost (inventory, backlog and changeover costs).

$$\text{minimize} \quad \sum_{i}\sum_{t}(\sigma^{i}\cdot S_{i,t}+\beta^{i}\cdot B_{i,t})+\sum_{i}\sum_{i',i'\neq i}\sum_{j\in(J_i\cap J_{i'})}\sum_{t\in T_1}\kappa_{i,i',j}(X_{i,i',j,t}+\overline{X}_{i,i',j,t}) \tag{20}$$

$$\text{s.t.} \quad \hat{L}^{P}_{o,l,t}\cdot\hat{Y}^{1}_{o,l,t}\leq L^{P}_{o,l,t}\leq\chi_{o}\cdot\hat{Y}^{1}_{o,l,t} \quad \forall o\in O, l\in L, t\in T \tag{22}$$

$$\sum_{l'}\sum_{t'=t-\lambda_{l'}-1}^{t-1} L^{S}_{o,l',t'}\leq M\cdot(1-\hat{Y}^{1}_{o,l,t}) \quad \forall o\in O, l\in L, t\in T \tag{23}$$

$(4)-(19)$

3.2.2. *Improvement step*

An iterative method is used to further improve the initial feasible solution generated in the constructive step. A number of improvement operators based on the fix-and-optimize heuristic are introduced (Sahling et al., 2009). The applied methodology is similar to the approach proposed by (Baldo et al. (2014). The main idea of the fix-and-optimize heuristic is to define subsets of the model's binary variables, relax and re-optimize them, in the search for a better solution. Thus, two disjunctive subsets of the model's binary variables B^{υ} are generated. The first one defines, which binary variables are relaxed B^{R}_{υ}, and the second denotes the subset of binary variables whose values remain fixed B^{F}_{υ}. As a result, an MILP subproblem is created that considers only a small portion of the initial problem. Therefore, each subproblem can be solved to optimality in relatively small CPU times. In case the objective of the new solution is better, than the best solution found, the binary variables are updated, otherwise, the best solution found so far is kept. Note that all continuous variables are relaxed since they do not significantly increase the complexity of the model. This procedure is repeated through an exhaustive iterative approach that ensures that all subsets of binary variables are visited. A runtime limit is set to avoid prohibitive computational times that would constitute the application of the method impractical. We use the model presented in subchapter 3.1, in order to address the integrated planning and scheduling planning of the whole production process.

Algorithm. Pseudocode of fix-and-optimize heuristic

Given the initial solution of the constructive step S^{C} with objective value $F(S^{C})$

Define the number of iterations required to visit all subsets (k)

Define the computational limit ($limit$)

$iter = 0$

$S^{best} = S^{c}$

While ($CPU \leq limit$ *and* $iter \leq k$) **do**

Define subsets B^{R}_{υ} and B^{F}_{υ} according to defined rules

Solve generated MILP-subproblem (S^{new})

If ($F(S^{new}) < F(S^{best})$) **then**

Update binary variables

$F(S^{best}) = F(S^{new})$

end-if

end-while

Four improvement operators based on the aforementioned heuristic framework are employed. These operators are differentiated by the way they partition the problem's binary variables to form the various MILP-subproblems that will be solved iteratively. The rules used to define the subsets of the fix-and-optimize heuristic are based on temporal and/or spatial decomposition of the initial problem.

The fix-and-optimize forward (FO_F) operator employs a time decomposition scheme that starts at the beginning and finishes at the end of the planning horizon (Fig. 5a). In each iteration the binary variables of both stages are released for a specific number of time periods. In other words, the production plan is reoptimized for a partition of time. The length of this partition is equal to the maximum duration of fermentation and maturation of the involved liquids $max\{\lambda_l\}$. The algorithm then moves to the next time partition. The step of this movement is equal to the minimum duration of the fermentation and maturation process $min\{\lambda_l\}$. So, in case $max\{\lambda_l\} \neq min\{\lambda_l\}$ overlapping occurs, meaning that in each MILP-subproblem we include some of the decision variables of the previous iteration. This procedure continues until all variables have been revisited and reoptimized. The fix-and-optimize backward (FO_B; Fig. 5b) operator is similar to FO_F, with the only difference being that the iterative procedure starts at the end of the horizon and finishes at the beginning.

The next two improvement operators FO_F21 (Fig. 5c) and FO_B21 (Fig. 5d) employ a bi-level temporal and spatial decomposition strategy. Their main difference to the first two operators is that in each iteration the binary variables of only one stage are relaxed, in particular first the ones of Stage 2 and then the ones of Stage 1.

Fig. 6 presents a general overview of the proposed solution strategy for the optimal production planning and scheduling problem for beer production facilities. First an initial good and feasible solution is constructed, by disaggregating decisions of the two processing stages. GEG_S1 is employed to solve Stage 1, which then sends the relevant information to GEG_S2, which in turn is solved to consider the second stage and generate the solution of the constructive step. This solution is then fed to the improvement step, where a set of improvement operators based on the fix-and-optimize heuristic are applied. The order in which these operators are applied in the proposed method will be discussed in the next section. Conclusively, we are able to consider large-scale industrial cases and generate near-optimal production plans in reasonable computational times.

4. Computational analysis

In this section numerous test studies are examined in order to evaluate the efficiency of the proposed model and solution strategy. Furthermore, the applicability of the developed solution framework in a real-life industrial case study of a brewing facility in Greece is considered. In all cases, the planning horizon is 42 days, while scheduling decisions are taken over a week. All models and solution algorithms were developed using the GAMS 31.1 interface (Brooke et al., 1998) and all problem instances were solved using CPLEX 12.0 in a PC

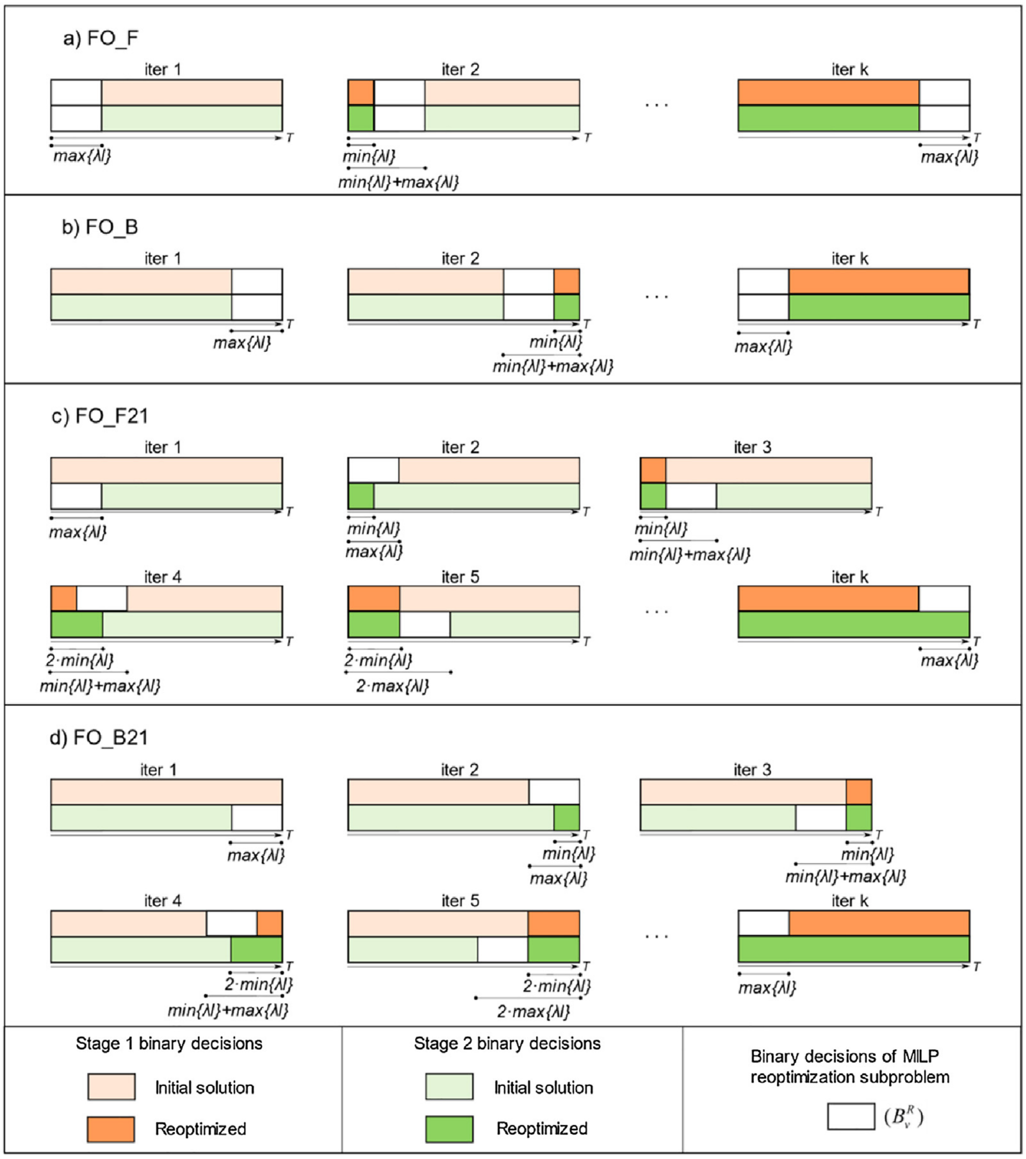

Fig. 5 – Fix and optimize improvement operators.

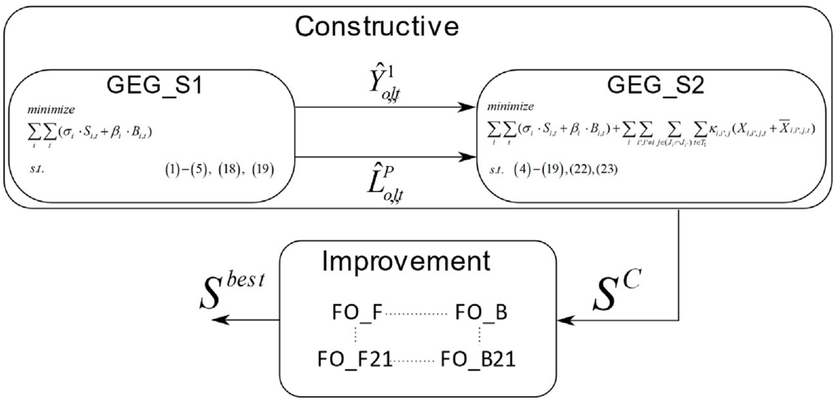

Fig. 6 – Overview of proposed solution strategy.

equipped with an Intel Core i7 @3.4 GHz CPU and 16 GB of DDR4 RAM.

4.1. *Evaluation of the proposed MILP model*

The developed MILP model (GEG) is used for the solution of various cases that represent small to medium-sized integrated planning and scheduling problems of brewing facilities. In order to evaluate the quality of the generated production plans, we compare solutions generated by our model to the ones using the MILP model of Baldo et al. (2014), which to the best of our knowledge is the only available model that addresses the specific optimization problem. This model will be referred to as BSAM. A total of 28 test cases have been studied, which can be categorized in 7 groups based on the different number of lines, tanks, liquids and products (Table 1).

Table 1 – Main design specifications of the examined case studies.

	Tanks	Lines	Liquids	Products
Cases 1.*	3	1	1	5
Cases 2.*	3	1	2	10
Cases 3.*	2	2	2	10
Cases 4.*	4	2	3	15
Cases 5.*	8	2	3	15
Cases 6.*	8	3	4	20
Cases 7.*	10	4	5	25

For each group, 4 alternative test cases are studied, that are differentiated in terms of the rest of the production characteristics, such as, demand mixture (size of orders and due dates), processing times, changeover times and cost term coefficients. In order to simulate the main characteristics of realistic study cases, we employ the methodology of Baldo et al. (2014). The specific data for each study case are randomly generated by a set of possible values, that simulate production parameters typically met in real-life breweries. We now present the interval of values used in the considered cases. The demand for final products in number of items is in the interval [60, 256710] and the due date of each product is randomly set within the given horizon. Each final product requires an amount of liquid, $r_{l,i}$, that is chosen from the set {1.98, 4.00, 4.80, 5.00, 6.00, 6.00, 6.60, 7.92, 12.00, 17.82, 20.00, 30.00, 50.00}. Processing rates of filling lines range between 0.028 units/second and 9.6 units/second, while the fermentation/maturation process may last from 5 to 21 days. We assume that all filling lines can process any final product. Regarding the changeover times, we randomly equate them to the following values {30, 40, 45, 60, 75, 90, 100, 120, 150, 160, 165, 180, 195, 210, 240, 260, 300, 380, 480, 900}. Furthermore, the unitary inventory cost coefficient (σ_i) of each product over a single time period are defined from [0.012, 0.45], while the backlog cost coefficient (β_i) is set to be one hundred times the inventory cost coefficient, since the priority is to meet the customer demands prior to the given due date. In order to define the changeover cost coefficients, we multiply the respective changeover time with a factor in the range of [10, 100]. Finally, the capacity of the fermentation/maturation tanks is specified based on the specific production characteristics of each problem instance. For more details regarding the generation or realistic study cases refer to Baldo et al. (2014).

Our GEG model is used to optimally solve the 28 test instances. A comparison with the solution generated by BSAM is made. For both models a computational limit of half an hour is set. Table 2 summarizes the results of this analysis. More specifically, the objective value for each case study, the solution time, and the optimality gap of the solution is provided for both models. Finally, the improvement achieved using the suggested MILP model (GEG) is also reported using the following equation:

$$Improvement = \frac{S_{BSAM} - S_{GEG}}{S_{BSAM}} \cdot 100$$

The improvements reported in most cases when using the GEG model illustrates the superiority of the proposed model, which is especially notable in larger problem instances (5.*, 6.* and 7.*). There are very few cases in which our model was not able to provide a better solution, e.g. 4.C and 7.B, however, the solution generated by BSAM in these cases is only marginally better (<5%). In contrast, the utilization of the proposed GEG model, can immensely reduce production costs. Specifically, an improvement between 10% and 55% is achieved in most of the instances. As expected, the larger the problem size the more difficult its solution is and thus a larger potential for improvement exists. It is interesting to note that even in small-sized cases, where the BSAM solution reaches its global optimal solution (0% gap), the proposed model can further reduce the objective value. Another important conclusion of this analysis is that the proposed model is much faster than BSAM. In the smaller study cases (1.* - 3.*), GEG achieves similar or better quality solutions using only a fraction of the computational time required by BSAM. For larger problem instances, both models reach the computational limit, except for case 5.C, and in almost all cases GEG generates a better solution. However, the results also show the limitations of GEG model. With the exception of small test instances (1.* - 3.*), the solution is characterized by a large integrality gap. Thus, a monolithic MILP approach does not suffice, and the development of a sophisticated solution strategy is necessary.

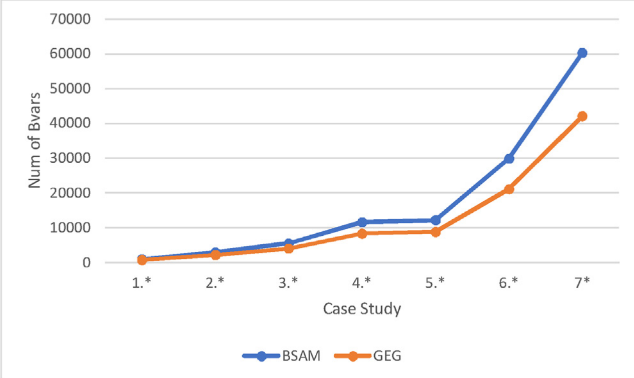

Fig. 7 – Comparison of binary variables between the GEG and BSAM model.

The combinatorial complexity of an MILP model is mostly affected by the number of binary variables. Fig. 7 illustrates this metric for both GEG and BSAM. Obviously, the proposed model requires fewer binary variables. In the largest cases, the difference in the number of binary variables between the two models is significantly increased. In particular, up to 30% fewer binary variables are used in the proposed GEG model. Consequently, this model is generally faster and can generate better solutions in the same computational time.

4.2. *Evaluation of the construction heuristic*

The analysis of the previous subsection has uncovered both the advantages and limitations of the proposed GEG model. Therefore, a solution strategy is proposed based on that model, in order to solve large-scale problems. As described in subsection 3.2, this method consists of a constructive and an improvement step. It is crucial to promptly generate a good initial solution in the constructive step, in order to improve the performance of the solution algorithm. This is not possible if the original MILP model is used, since it lacks computational efficiency, especially for large-scale problems. Instead, we employ a spatial decomposition approach, that consists of models GEG_S1 and GEG_S2 presented in subsection 3.2.1. In this subsection we test how this approach compares with the monolithic MILP GEG model. In total 7 cases of divergent complexity are considered, which are a subset of the test instances studied in the previous subsection. Each test case has been solved using the GEG model for two different

Table 2 – Comparison between the BSAM and GEG models.

	BSAM			GEG			
Case	Objective	CPU (s)	GAP (%)	Objective	CPU (s)	GAP (%)	Improvement (%)
1.A	11177	13.3	0	10067	11.8	0	9.9
1.B	5083	<1	0	5080	<1	0	0.1
1.C	10232	0.15	0	10234	0.15	0	0
1.D	16885	0.5	0	16555	0.2	0	1.9
2.A	247461	1800	20.8	203674	225	0	17.7
2.B	59074	1800	52.5	52274	600	0	11.5
2.C	14874	1800	0.4	14662	224	0	1.4
2.D	1060844	1800	18.3	929332	950	0	12.4
3.A	2825319	1443	0	2336326	3.7	0	17.3
3.B	467805	310	0	467325	104	0	0.1
3.C	80731	444	0	80731	144	0	0
3.D	2657138	484	0	2656117	131	0	0
4.A	2592330	1800	50.5	2122710	1800	31.1	18.1
4.B	1112320	1800	62.2	1066094	1800	53.9	4.2
4.C	24518	1800	70.5	25643	1800	61.4	−4.6
4.D	4628488	1800	83.1	3401497	1800	78.8	26.5
5.A	324848	1800	91.8	200925	1800	80.6	38.1
5.B	32325	1800	37	32491	1800	29.5	−0.5
5.C	45363	24.8	0	46034	6	0	−1.4
5.D	506431	1800	3.5	309656	1800	1.8	38.9
6.A	2546127	1800	65.7	1693735	1800	47.4	33.5
6.B	31386	1800	21.7	28298	1800	8.7	9.8
6.C	18467	1800	57.3	16744	1800	61.4	9.3
6.D	2193641	1800	96.6	1351917	1800	93.88	38.4
7.A	6916373	1800	32.5	5541703	1800	16	19.9
7.B	440177	1800	94.3	451862	1800	93.9	−2.6
7.C	339429	1800	58.3	154366	1800	6.8	54.5
7.D	10446975	1800	100	6858949	1800	91.1	34.3

computational limits (600 s, 1800s) and furthermore using the proposed decomposition approach with a limit of 600 s. To compare the three approaches, we use the following expression:

$$R = \frac{\text{Found} - \text{Best}}{\text{Found}}.100$$

The best solution found (Best) is compared to the solution generated by each approach (Found). The better the quality of the solution is, the closer the value of R is to zero. Table 3 shows a summary of the results. We found that in small cases there is no difference in the quality of the solution, however the decomposition approach is able to generate a solution much faster. This is not the case for medium-sized problem instances. Using the same time limit, the solution of the decomposition approach is always better. This effect is stronger in larger cases, were an improvement of up to 50% is reported. The monolithic approach cannot outperform the decomposition method even when we allow three times the computational time. The only exception is case 4.A, where the solution of the decomposition strategy is insignificantly worse but requires only a third of the CPU time. Conclusively, it is shown that the decomposition strategy significantly improves the solution generated by the constructive step.

4.3. *Evaluation of the developed MILP-based solution strategy*

In subsection 3.2.2 four improvement operators (FO_F, FO_B, FO_F21 and FO_B21) have been introduced, based on the relax-and-optimize heuristic for the further enhancement of the quality of the initial solution. Preliminary tests have been made to assess the impact of the different operators. For each test, an initial solution was generated based on the proposed constructive heuristic and then each operator was applied separately, and potential improvements were reported. The tests showed that the best performer is FO_B, followed by FO_B21, FO_F and finally FO_F21. Based on this information we create two improvement schemes, that differentiate in the order in which the improvement operators are applied. In the first, denoted as IMP.A, a greedy approach is employed were the different operators are applied from best to worst (FO_B -> FO_B21-> FO_F -> FO_F21). In the second denoted as IMP.B, a reverse order is followed. To evaluate the two improvement schemes, 10 large-scale problem instances are generated. The characteristics of these cases are as follows. The number of fermentation tanks is in the range of [20, 30], while 5 filling lines comprise the liquid bottling stage. Depending on the considered case, 35–40 products, requiring 5–7 different liquids, are to be processed. The procedure of generating each problem's parameters is the same as the one described in subsection 4.1. For each case study we have used the two alternative improvement schemes and the monolithic approach. Moreover, two solution time limits (1 h and 2 h) were considered for each method. Consequently, 6 different runs were done for each case study. Notice that the improvement schemes are applied to the initial solution provided by the constructive heuristic. Therefore, the available computational time must be shared between the two steps of the proposed solution strategy. Preliminary tests showed that better results were achieved, when a small CPU time is allocated to the generation of the initial solution. Therefore, a time limit of 450 s is set for the constructive step. The rest of the available CPU time (3150 or 6750 s depending on the test instance) is imposed on the iterative improvement step. Table 4 summarizes the results. The relative quality of each solution is reported using the R value described in the previous subsection. Solutions generated by

Table 3 – Comparison between the monolithic GEG model and the decomposition approach.

	Monolithic (GEG)				Decomposition (GEG_S1 + GEG_S2)	
	limit 600 s		limit 1800s			
Case	R (%)	CPU (s)	R (%)	CPU (s)	R (%)	CPU (s)
1.A	0	12	0	12	0	9
2.A	0	225	0	225	0	57
3.A	0	4	0	4	0	1
4.A	5.28	600	0	1800	1.79	600
5.A	10.32	600	2.06	1800	0	600
6.A	16.00	600	8.23	1800	0	600
7.A	52.70	600	41.62	1800	0	600

Table 4 – Comparison between the monolithic MILP model and the solution strategies for large-scale problems.

	Limit (3600 s)				Limit (7200 s)		
Case	GEG (%)	Constructive (%)	IMP.A (%)	IMP.B (%)	GEG (%)	IMP.A (%)	IMP.B (%)
L1	85.4	68.3	22	29.3	49.7	19.3	0
L2	39.3	18.6	7.4	7.4	39.3	0	1.8
L3	55.7	27.7	12.2	10.5	38.2	11.3	0
L4	52.5	33.3	32	4.9	49.6	17.8	0
L5	18.4	3.5	0.4	0.4	15.8	0	0.3
L6	68	62.7	47.4	19.7	43.7	27.9	0
L7	73.7	40.9	1.6	23.1	50.9	0	2
L8	55.7	29.2	4.3	3.2	9.5	2.3	0
L9	67.1	19.2	2.1	3	54.8	0.7	0
L10	72.7	85.7	41	2.3	40.5	25.9	0
Average	58.85	38.91	17.04	10.38	39.2	10.52	0.41

any of the two proposed methods is much better than the solutions obtained by the MILP model, even when the computational time is doubled. It should be underlined that on average the initial solutions provided by the constructive step are better than the ones obtained by the model using any time limit. Notice that the constructive heuristic runs only for a very small fraction of time compared to the GEG model. On average IMP.B leads to best solutions, except for a few cases where IMP.A outperforms it. More clear conclusions can be drawn when a time limit of 2 h is imposed. Here IMP.B is clearly the better approach, since it provides the best solution in nearly all cases. Conclusively, both solution strategies seem promising, since they outperform the direct solution of the MILP model (GEG), for each large-scale problem. Thus, the results indicate that the proposed methods can successfully address real-life industrial problems. It should be also noted that as the runtime limit increases, the performance of both methods is improved. Finally, the order of applying the improvement operators affects the performance of the improvement step. In particular, better solutions are obtained in most cases, when operators from worst to best are applied.

4.4. *Industrial application*

The applicability of the proposed solution strategy in real-life industrial problems is tested in this subsection. In particular, a real-life case study provided by a brewery located in Northern Greece is considered. The facility under consideration consists of 31 fermentation/maturation tanks and 2 filling lines. The tanks are divided in tree types, small, medium and large, depending on their capacity. Regarding the filling lines, the first one can process all products that use aluminum cans or glass bottles, while the second only produces final items that use kegs. A total of 9 products that require 2 types of liquids are produced in the facility. However, multiple orders for each final product that usually have different amounts and due dates must be satisfied in the considered time horizon, thus increasing the complexity of the problem. The planning horizon is set to 6 weeks, while the scheduling decisions are required over a weekly horizon. The plant operates throughout the clock, so there is a 24/7 availability for all processing units. Due to confidentiality reasons all data of the plant cannot be disclosed. In this case a total of 36 orders must be met. The proposed solution method is employed in order to generate optimal production plans that minimize total production costs (inventory, backlog and changeover) of the facility. In the improvement step, the operators are applied from worst to best (IMP.B approach), due to its superior performance. The chosen computational limit is set to 2 h.

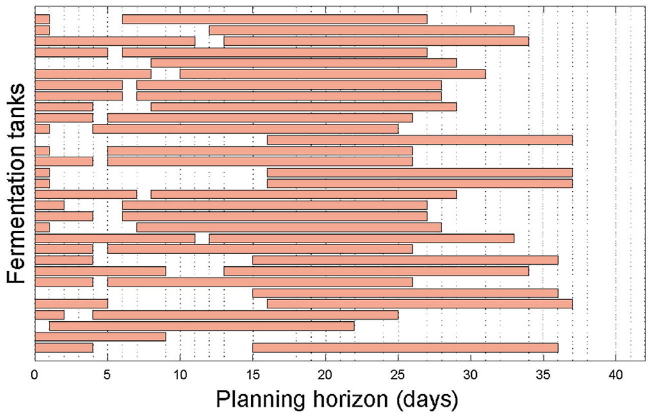

Fig. 8 – Gantt chart of fermentation/maturation tanks.

Fig. 8 illustrates the Gantt chart of the optimized solution for each fermentation/maturation tank. Each block signifies the fermentation/maturation process of a liquid that takes places in a tank. Note that by the end of the planning hori-

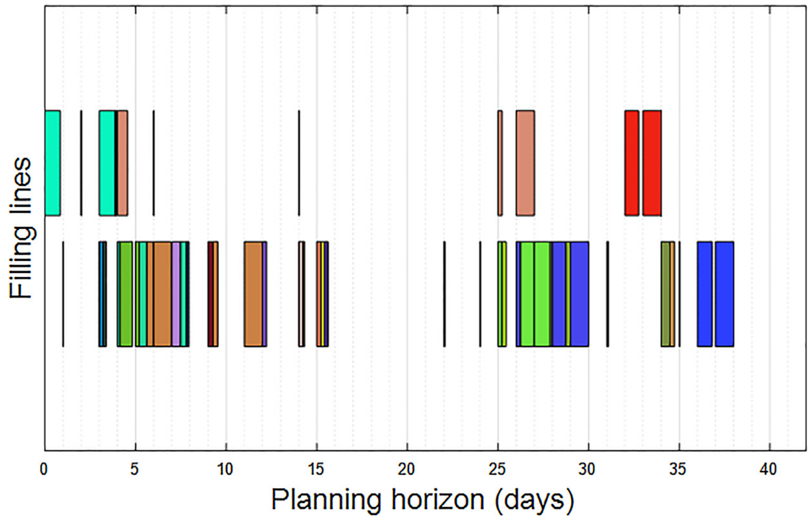

Fig. 9 – Gantt chart of filling lines.

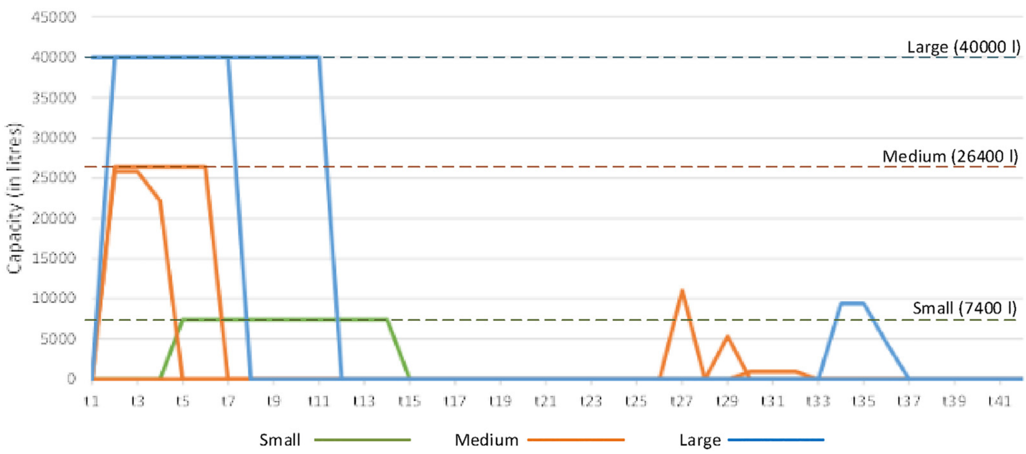

Fig. 10 – Profile of liquid stored in selected fermentation tanks.

zon no fermentation process occurs. This is justified by the limited time horizon. The fermentation/maturation process requires a total of 21 days, consequently, no liquid can be prepared in the available time, therefore no additional process can start.

Fig. 9 illustrates the Gantt chart of the filling lines. Each colored block denotes that a filling process for a specific order is taking place. At a first look one would say that there is no need for incorporating this stage and that the production bottleneck is the fermentation/maturation stage. It is true that the results, underline an overdesign issue of the filling lines in the facility under consideration. However, they must be included in the optimization problem for two reasons. Firstly, the changeovers that take place in the liquid filling stage must be incorporated since they induce significant costs either due to the loss of production time or due to involvement of various resources required for the cleaning operations (e.g. water and manpower). Moreover, filling lines must always be available for the tanks to empty the ready liquid and be refilled to initiate the next fermentation process. It must be ensured that the capacity of the filling lines is never violated, otherwise no feasible production plan can be achieved for the fermentation tanks.

Finally, Fig. 10 depicts the amount of stored ready liquid in selected fermentation tanks. It can be observed that tank capacity limitations are satisfied throughout the planning horizon.

5. Conclusions

This work presents a new MILP-based solution framework for the optimal production planning and scheduling problem of breweries. The overall production process consists of a batch (liquid preparation) and a continuous (liquid bottling) processing stage. Numerous parallel non-identical units i.e. fermentation/tanks and filling lines are available in each stage, while many orders must be satisfied as close as possible to their corresponding due dates. A salient characteristic of this process is the long lead times originating from the large processing time required for the fermentation/maturation process. Therefore, a long planning horizon must be considered, resulting in a complex optimization problem. In order to efficiently address the problem, first a new MILP model is developed that is based on the immediate precedence framework employing a mixed discrete-continuous time representation. A comprehensive analysis demonstrated that the developed model performs better to a relevant literature model. However, the direct application of the MILP model is limited to small problem instances. Therefore, an optimization-based solution strategy is introduced, in order to tackle large-scale problems which represent the industrial reality. The proposed algorithm consists of a constructive step, that utilizes a spatial decomposition heuristic to propose an initial good solution and an improvement step, where four operators based on the relax-and-optimize heuristic are iteratively applied to achieve high quality solutions. The overall solution framework strategy is applied to a real-life industrial problem of a Greek brewery. Optimized production plans that minimize total production costs are generated in relatively low CPU times. The developed optimization framework is suitable for the development of a computer-aided tool, that will facilitate the decision-making process in any brewing facility. As a result, near-optimal production plans can be promptly generated, leading to significant economic benefits thus improving the competitive power of the brewing industry. Future work will focus on the introduction of buffers between the two processing stages and the consideration of rescheduling actions using a rolling horizon technique.

Nomenclature

Indices

$i, i' \in I$	products to be processed within the planning horizon
$l, l' \in L$	liquids required for the final products
$o \in O$	fermentation/maturation tanks
$j \in J$	filling lines
$t, t' \in T$	set of time periods for the whole planning horizon

Sets

T_1	subset of time periods that comprise the first part of the planning horizon
T_2	subset of time periods that comprise the second part of the planning horizon
I_j	mapping set defining filling lines *j* that can process product *i*
J_i	mapping set defining products *i* that can be processed by filling line *j*
I_l	mapping set defining products *i* that are made of liquid *l*

Variables

Binary

Stage 1

$Y^1_{o,l,t}$	=1 when liquid *l* gets ready in tank *o* in time period *t*

Stage 2

$Y^2_{i,j,t}$	=1 when product *i* is processed in filling line *j* in time period *t*
$V_{j,t}$	=1 when filling line *j* is utilized
$W^F_{i,j,t}$	=1 when product *i* is processed first in filling line *j* in time period *t*
$W^L_{i,j,t}$	=1 when product *i* is processed last in filling line *j* in time period *t*‘
$X_{i,i',j,t}$	=1 when product *i* is processed right before product *i*’ in line *j* and time period *t*
$\overline{X}_{i,i',j,t}$	=1 when product *i* is processed first in line *j* and in time period *t* and products *i*’ is processed last in line *j* and time period $t-1$

Continuous

Stage 1

$L^P_{o,l,t}$	amount of liquid *l* that gets ready in tank *o* in time period *t*
$L^S_{o,l,t}$	amount of stored liquid *l* that gets ready in tank *o* in time period *t*

Stage 1+2

$L^T_{o,j,i,t}$	amount of liquid *l* being transferred from tank *o* to line *j* in time period *t*

Stage 2

$Q_{o,j,i,t}$	number of items *i* that use liquid from tank *o* and are processed in line *j* in time period *t*
$C_{i,j,t}$	completion time of the filling process for product *i* in filling line *j* and time period *t*
$U_{j,t}$	time within period *t* used for a changeover operation that is completed in the next period in filling line *j*
$\overline{U}_{j,t}$	time within period *t* used for a changeover operation that started in the previous period in filling line *j*
$S_{i,t}$	inventory level of product *i* in time period *t*
$B_{i,t}$	backlog level of product *i* in time period *t*

Parameters

λ_l	fermentation/maturation time required for liquid *l*
χ_o	maximum capacity of fermentation/maturation tank *o*
$\pi_{i,l}$	amount of liquid *l* required for each unit of product *i*
$\rho_{i,j}$	processing rate of product *i* in filling line *j*
$\gamma_{i,i',j}$	necessary changeover time between products *i* and *i*’ in filling line *j*
$\zeta_{i,t}$	demand of product *i* in time period *t*
σ_i	inventory cost coefficient
β_i	backlog cost coefficient
$\kappa_{i,i',j}$	changeover cost coefficient
ω	available processing time in each time period
M	big-M parameter used for the lot-sizing constraints of the liquid preparation stage

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research is co-financed by Greece and the European Union (European Social Fund-ESF) through the Operational Programme “Human Resources Development, Education and Lifelong Learning 2014-2020” in the context of the project “Optimization techniques for production scheduling in the process industries” (MIS 5047892).

References

Aguirre, A.M., Liu, S., Papageorgiou, L.G., 2017. Mixed integer linear programming based approaches for medium-term planning and scheduling in multiproduct multistage continuous plants. Ind. Eng. Chem. Res. 56 (19), 5636–5651.

Almada-Lobo, B., Oliveira, J.F., Carravilla, M.A., 2008. Production planning and scheduling in the glass container industry: a VNS approach. Int. J. Prod. Econ. 114 (1), 363–375.

Baldo, T.A., Santos, M.O., Almada-Lobo, B., Morabito, R., 2014. An optimization approach for the lot sizing and scheduling problem in the brewery industry. Comput. Ind. Eng. 72 (1), 58–71.

Basán, N.P., Cóccola, M.E., García del Valle, A., Méndez, C.A., 2020. Scheduling of flexible manufacturing plants with redesign options: a MILP-based decomposition algorithm and case studies. Comput. Chem. Eng. 136.

Bassett, M.H., Pekny, J.F., Reklaitis, G.V., 1996. Decomposition techniques for the solution of large-scale scheduling problems. AIChE J. 42 (12), 3373–3387.

Borisovsky, P., Eremeev, A., Kallrath, J., 2019. Multi-product continuous plant scheduling: combination of decomposition, genetic algorithm, and constructive heuristic. Int. J. Prod. Res., 1–19.

Brooke, A., Kendrik, D., Meeraus, A., Raman, R., Rosenthal, R.E., 1998. GAMS-A User's Guide. GAMS Development Corporation, Washington, DC.

Castro, P., Barbosa-Póvoa, P.F.D., Matos, H., 2001. An improved RTN continuous-time formulation for the short-term scheduling of multipurpose batch plants. Ind. Eng. Chem. Res. 40 (9), 2059–2068.

Castro, P.M., Barbosa-Póvoa, A.P., Matos, H.A., Novais, A.Q., 2004. Simple continuous-time formulation for short-term scheduling of batch and continuous processes. Ind. Eng. Chem. Res. 43 (1), 105–118.

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

Cerdá, J., Cafaro, V.G., Cafaro, D.C., 2020. Synchronizing operations in multiproduct batch facilities producing semifinished and final products. Ind. Eng. Chem. Res. 59 (29), 13113–13131.

Dimitriadis, A.D., Shah, N., Pantelides, C.C., 1997. RTN-based rolling horizon algorithms for medium term scheduling of multipurpose plants. Comput. Chem. Eng. 21 (97), S1061–S1066.

Elekidis, A., Corominas, F., Georgiadis, M., 2019. Production Scheduling of Consumer Goods Industries. Ind. Eng. Chem. Res. 58 (51), 23261–23275.

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

Ferreira, D., Clark, A.R., Almada-Lobo, B., Morabito, R., 2012. Single-stage formulations for synchronised two-stage lot sizing and scheduling in soft drink production. Int. J. Prod. Econ. 136 (2), 255–265.

Gajic, D., Hadera, H., Onofri, L., Harjunkoski, I., Di Gennaro, S., 2017. Implementation of an integrated production and electricity optimization system in melt shop. J. Clean. Prod. 155, 39–46.

Georgiadis, G.P., Kopanos, G.M., Karkaris, A., Ksafopoulos, H., Georgiadis, M.C., 2019a. Optimal production scheduling in the dairy industries. Ind. Eng. Chem. Res. 58 (16), 6537–6550.

Georgiadis, G.P., Elekidis, A.P., Georgiadis, M.C., 2019b. Optimization-based scheduling for the process industries: from theory to real-life industrial applications. Processes 7 (7).

Georgiadis, G.P., Mariño Pampín, B., Adrián Cabo, D., Georgiadis, M.C., 2020. Optimal production scheduling of food process industries. Comput. Chem. Eng. 134.

Giannelos, N.F., Georgiadis, M.C., 2002. A simple new continuous-time formulation for short-term scheduling of multipurpose batch processes. Ind. Eng. Chem. Res. 41, 2178–2184.

Harjunkoski, I., 2016. Deploying scheduling solutions in an industrial environment. Comput. Chem. Eng. 91, 127–135.

Harjunkoski, I., et al., 2014. Scope for industrial applications of production scheduling models and solution methods. Comput. Chem. Eng. 62, 161–193.

Intelligen Inc., "SchedulePro." [Online]. Available: https://www.intelligen.com/products/schedulepro-designer/schedulepro-overview/.

Isaksson, A.J., Harjunkoski, I., Sand, G., 2017. The impact of digitalization on the future of control and operations,". Comput. Chem. Eng. 114, 122–129.

Janak, S.L., Floudas, C.A., 2008. Improving unit-specific event based continuous-time approaches for batch processes: integrality gap and task splitting. Comput. Chem. Eng. 32 (4–5), 913–955.

Kallrath, J., 2002. Planning and scheduling in the process industry. OR Spectr. 24 (3), 219–250.

Kondili, E., Pantelides, C.C., Sargent, R.W.H., 1993. A general algorithm for short-term scheduling of batch operations-I. MILP formulation. Comput. Chem. Eng. 17 (2), 211–227.

Kopanos, G.M., Puigjaner, L., Georgiadis, M.C., 2010. Optimal production scheduling and lot-sizing in dairy plants: the yogurt production line. Ind. Eng. Chem. Res. 49, 701–718.

Kopanos, G.M., Puigjaner, L., Maravelias, C.T., 2011. Production planning and scheduling of parallel continuous processes with product families. Ind. Eng. Chem. Res. 50 (3), 1369–1378.

Lee, H., Maravelias, C.T., 2017. Mixed-integer programming models for simultaneous batching and scheduling in multipurpose batch plants. Comput. Chem. Eng. 106, 621–644.

Lee, H., Maravelias, C.T., 2018. Combining the advantages of discrete- and continuous-time scheduling models: Part 1. Framework and mathematical formulations. Comput. Chem. Eng. 116, 176–190.

Lee, H., Maravelias, C.T., 2019. Combining the advantages of discrete- and continuous-time scheduling models: Part 2. Systematic methods for determining model parameters. Comput. Chem. Eng. 128, 557–573.

Lee, H., Maravelias, C.T., 2020. Combining the advantages of discrete- and continuous-time scheduling models: Part 3. General algorithm. Comput. Chem. Eng. 139, 106848.

Li, Z., Ierapetritou, M.G., 2010. Rolling horizon based planning and scheduling integration with production capacity consideration. Chem. Eng. Sci. 65 (22), 5887–5900.

Liu, S., Yahia, A., Papageorgiou, L.G., 2014. Optimal production and maintenance planning of biopharmaceutical manufacturing under performance decay. Ind. Eng. Chem. Res. 53 (44), 17075–17091.

Méndez, C.A., Henning, G.P., Cerdá, J., 2000. Optimal scheduling of batch plants satisfying multiple product orders with different due-dates. Comput. Chem. Eng. 24 (9), 2223–2245.

Mendez, C., Cerda, J., 2004. An MILP framework for batch reactive scheduling with limited discrete resources. Comput. Chem. Eng. 28 (6), 1059–1068.

Méndez, C.A., Cerdá, J., Grossmann, I.E., Harjunkoski, I., Fahl, M., 2006. State-of-the-art review of optimization methods for short-term scheduling of batch processeszz. Comput. Chem. Eng. 30 (6–7), 913–946.

Mostafaei, H., Harjunkoski, I., 2020. Continuous-time scheduling formulation for multipurpose batch plants. AIChE J. 66 (2), 1–16.

Nie, Y., Biegler, L.T., Wassick, J.M., Villa, C.M., 2014. Extended discrete-time resource task network formulation for the reactive scheduling of a mixed batch/continuous process. Ind. Eng. Chem. Res. 53 (44), 17112–17123.

Panek, S., Engell, S., Subbiah, S., Stursberg, O., 2008. Scheduling of multi-product batch plants based upon timed automata models. Comput. Chem. Eng. 32 (1–2), 275–291.

Pantelides, C.C., 1994. Unified frameworks for optimal process planning and scheduling. Proc. Second Conf. Found. Comput. Aided Process Oper., 253–274.

Pezzella, F., Morganti, G., Ciaschetti, G., 2008. A genetic algorithm for the flexible job-shop scheduling problem. Comput. Oper. Res. 35 (10), 3202–3212.

Pierreval, H., Mebarki, N., 1997. Dynamic scheduling selection of dispatching rules for manufacturing system. Int. J. Prod. Res. 35 (6), 1575–1591.
Pinedo, M.L., 2016. Scheduling: Theory, Algorithms and Systems, 5th ed. Springer.
Pinto, J.M., Grossmann, I.E., 1995. A continuous time mixed integer linear programming model for short term scheduling of multistage batch plants. Ind. Eng. Chem. Res. 34 (9), 3037–3051.
Roslöf, J., Harjunkoski, I., Björkqvist, J., Karlsson, S., Westerlund, T., 2000. An MILP-Based reordering algorithm for complex industrial scheduling and rescheduling. Comput. Aided Chem. Eng. 8, 13–18.
Sahling, F., Buschkühl, L., Tempelmeier, H., Helber, S., 2009. Solving a multi-level capacitated lot sizing problem with multi-period setup carry-over via a fix-and-optimize heuristic. Comput. Oper. Res. 36 (9), 2546–2553.
Sel, Ç., Bilgen, B., 2014. Hybrid simulation and MIP based heuristic algorithm for the production and distribution planning in the soft drink industry. Int. J. Ind. Manuf. Syst. Eng. 33 (3), 385–399.
Sel, C., Bilgen, B., Bloemhof-Ruwaard, J.M., van der Vorst, J.G.A.J., 2015. Multi-bucket optimization for integrated planning and scheduling in the perishable dairy supply chain. Comput. Chem. Eng. 77, 59–73.
Stefansson, H., Sigmarsdottir, S., Jensson, P., Shah, N., 2011. Discrete and continuous time representations and mathematical models for large production scheduling problems: A case study from the pharmaceutical industry. Eur. J. Oper. Res. 215 (2), 383–392.
Toledo, C.F.M., França, P.M., Morabito, R., Kimms, A., 2009. Multi-population genetic algorithm to solve the synchronized and integrated two-level lot sizing and scheduling problem. Int. J. Prod. Res. 47 (11), 3097–3119.
Toscano, A., Ferreira, D., Morabito, R., 2020. Formulation and MIP-heuristics for the lot sizing and scheduling problem with temporal cleanings. Comput. Chem. Eng. 142, 107038.
Van Elzakker, M.A.H., Zondervan, E., Raikar, N.B., Grossmann, I.E., Bongers, P.M.M., 2012. Scheduling in the FMCG Industry: An Industrial Case Study.
Velez, S., Maravelias, C.T., 2013. Multiple and nonuniform time grids in discrete-time MIP models for chemical production scheduling. Comput. Chem. Eng. 53, 70–85.
Yee, K.L., Shah, N., 1998. Improving the efficiency of discrete time scheduling formulation. Comput. Chem. Eng. 22, S403–S410.
Zhang, C., Li, P., Guan, Z., Rao, Y., 2007. A tabu search algorithm with a new neighborhood structure for the job shop scheduling problem. Comput. Oper. Res. 34 (11), 3229–3242.