Modeling Production Sequences in the Fast Moving Consumer Goods Industry

–

The Impact of Changeover Structure in Integrated Lot-Sizing and Scheduling

vorgelegt von:

Dipl.-Kfm. Mario Lueb

aus Köln

von der Fakultät VII - Wirtschaft und Management
der Technischen Universität Berlin
zur Erlangung des akademischen Grades
Doktor der Wirtschaftswissenschaften

- Dr. rer. oec. -

genehmigte Dissertation

Promotionsausschuss

Vorsitzender: Prof. Dr. Hans Hirth
Gutachter: Prof. Dr. Hans-Otto Günther
Gutachter: Prof. Dr. Christian Almeder

Tag der wissenschaftlichen Aussprache: 6. Februar 2014

Berlin 2014

D 83

Acknowledgement

The following work was created during my time at the Technical University of Berlin as a research assistant at the department of Production Management. Looking back, it could not have been done without the support of several people, to which I want to express my gratitude.

First and foremost, my deepest thanks go to Prof. Günther for his support, guidance and providing valuable insights and advice. I wish to thank the whole staff of my former department for a very friendly, yet productive, working environment, being it our student assistants, our secreatary, my colleagues from the department and from other universities or Prof. Günther himself.

Then, I wish to thank the whole PhD committee for the assessment of this dissertation.

Furthermore, my gratidude goes to my family and friends for their ongoing support and encouragement. Special thanks go to my good friend F. Semmo, for his assistance with graphics design and layout in LaTeX.

Mario Lueb
Berlin, Germany, February 14, 2014

Contents

List of Figures

List of Tables

List of Abbreviations

ACO	Ant Colony Optimization.
ACT	Average Changeover Time.
APS	Advanced Planning System.
CLSP	Capacitated Lot-Sizing Problem.
CSLP	Continuous Lot-Sizing and Scheduling Problem.
CU	Capacity Units.
DLSP	Discrete Lot Sizing and Scheduling Problem.
EOQ	Economic Order Quantity.
ERP	Enterprise Resource Planning.
FF	Full Flexibility.
FMCG	Fast Moving Consumer Goods.
GA	Genetic Algorithms.
GLSP	General Lot-Sizing and Scheduling Problem.
L	Large.
LC	Limited Changeover.
LHV	Large with High Demand Variation.
LP	Linear Programming.
LSPSD	Lot-Sizing and Scheduling-Problem with Sequence-Dependent Setup Costs and Times.
M	Medium.
MILP	Mixed Integer Linear Programming.
MIP Gap	Mixed Integer Programming Gap.
MRP	Material Requirements Planning.
MRP II	Manufacturing Resource Planning.
NS	Natural Sequence.
PLSP	Proportional Lot-Sizing and Scheduling Problem.
S	Small.

SA	Simulated Annealing.
SIULSP	Single-Item Uncapacitated Lot-Sizing Problem.
TS	Tabu-Search.
VRP	Vehicle Routing Problem.
XL	Extra Large.

List of Symbols

$\overline{\alpha_i}$	Latest end time of block i.
$\underline{\alpha_i}$	End time respectively start time of the macro period of block i.
$\overline{\alpha_m}$	End time respectively start time of the macro period m.
$\underline{\alpha_m}$	End time respectively start time of the macro period m.
$\alpha_i \geq 0$	Start time for block i.
α_m	Start time of macro period m.
$\alpha_o \geq 0$	Start time of production order o (α_1 given).
α_t	Start time of micro period t.
$\delta_i \geq 0$	Duration of block i.
$\delta_o \geq 0$	Duration of production order o.
$\varphi \in P(f)$	Set of families which can be predecessor of family f.
B	Sufficiently large number.
C	Capacity of each macro period.
c	Capacity of each micro period.
C_m	Capacity of macro period m.
c_t	Capacity of micro period t.
d_{jm}	Demand of product j at the end of macro period m.
d_{jt}	Demand of product j at the end of micro period t.
d_k	Quantity of demand element k.
$e_{ij} \geq 0$	Inventory of product j at the end of the planning horizon built up by production in block i.
$f \in F$	Set of families.
$h, j \in J$	Set of products ($j = 1, \ldots, \lvert J \rvert$).
h_j	Holding cost per period and per unit of product j.
$i \in I$	Set of blocks.
$i \in I^f$	Subset of blocks that are the first of each macro period.
$I_{jm} \geq 0$	Inventory of product j at the end of macro period m.
$I_{jt} \geq 0$	Inventory of product j at the end of micro period t (I_{j0} given).
$i \in I(k)$	Subset of blocks including those that can be used to satisfy demand element k.

$i \in I(m)$	Set of blocks belonging to macro period m.
$j' \in J'$	Set of products including dummy product ($j' = 0, \dots, \lvert J' \rvert$).
$j \in J(f)$	Set of products belonging to family f.
$j \in J(i)$	Subset of products, including those belonging to the family of block i.
$j(k)$	Product, to which demand element k refers.
$k \in K$	Set of demand elements.
$k \in K(i)$	Set of demand elements which may be satisfied from production in block i.
L	Length of a macro period.
$m \in M$	Set of macro periods.
mls	Minimum family lot size.
mls_j	Minimum lot size of product j.
MS	Makespan until all production is completed.
$o \in O$	Sequence of production orders over all blocks.
$o \in O(j,m)$	Set of production orders in period (block) m which can produce product j.
pt^o	Variable processing time per unit of production order o.
pt_j	Processing time per unit of product j.
$q_{ik} \geq 0$	Quantity of demand element k that is satisfied from production in block i.
q_j	Quantity of product j that can be produced per micro period.
$q_{jm} \geq 0$	Quantity of product j produced in macro period m.
$q_{jt} \geq 0$	Quantity of product j produced in micro period t.
$q_{st} \geq 0$	Units produced using the setup state s in micro period t.
$q_t \geq 0$	Quantity produced in micro period t.
$sc_{j'j}$	Cost for a changeover from product j' to product j.
sc^o	Capacity required for a changeover to any production order o.
sc_j	Cost for a changeover to product j.
$\overline{\overline{st}}$	Maximum of all setup times.
st^f	Capacity required for a changeover to any family.
$st_{j'j}$	Capacity required for a changeover from product j' to product j.
st^o	Capacity required for a changeover to any production order o.
st^p	Capacity required for a changeover to any product.
$ST_{jt} \geq 0$	Capacity required for changeover to product j in period t.

$t \in T$	Set of micro periods.
$t \in T^f$	Subset of micro periods that are the first of any macro period.
$t \in T(f)$	Set of micro periods belonging to family f.
$t \in T(f, m)$	Set of micro periods belonging to family f and macro period m.
$t \in T(j, m)$	Set of micro periods that can produce product j and belong to macro period m.
$t \in T(m)$	Set of micro periods that belong to macro period m.
$u_m \in \{0; 1\}$	1, if macro period m is used, else 0.
$u_t \in \{0; 1\}$	1, if micro period t is used, else 0.
$X_{ft} \in \{0; 1\}$	1, if a changeover to family f takes place in micro period t, else 0.
$x_{j'jt} \in \{0; 1\}$	1, if a changeover from product i to product j takes place in micro period t, else 0.
$x_{jt} \in \{0; 1\}$	1, if a changeover to product j takes place in micro period t, else 0.
$x_o \geq 0$	Size of production order o.
$Y_{ft} \in \{0; 1\}$	1, if family f is set up at the end of micro period t, else 0.
$Y'_i \in \{0; 1\}$	1, if all blocks until block i are used, else 0.
$Y_i \in \{0; 1\}$	1, if block i is used, else 0.
$y_{ij} \in \{0; 1\}$	1, if product j is set up in block i, else 0.
$y_{ij'j} \in \{0; 1\}$	1, if a changeover from product j' to product j is performed at the beginning of block i, else 0.
$y_{j'jt} \in \{0; 1\}$	1, if product j is set up in micro period t by a changeover from product j', else 0.
$y_{jm} \in \{0; 1\}$	1, if product j is set up in macro period m, else 0.
$y_{jt} \in \{0; 1\}$	1, if product j is set up at the end of micro period t, else 0.
$y_o \in \{0; 1\}$	1, if production order o produces a positive quantity, else 0.
$y_t \in \{0; 1\}$	1, if micro period t is set up to the corresponding product, else 0.
$z_{fi} \in \{0; 1\}$	1, if family f is assigned to block i, else 0.

1. Introduction

The *Fast Moving Consumer Goods industry* (FMCG industry) is responsible for producing goods of regular, often daily, need. Common FMCG products are e. g. food and dairy products, glassware, paper products, pharmaceuticals, electronics, plastic goods, printing goods, household products, photography, beverages etc. In 2010, consumer goods sales had a gross revenue of 555 billion € in Germany alone. In 2009, Europe's biggest consumer goods producing company Nestlé's worldwide gross revenue was 100 Billion US-$[1]. The consumer goods industry is therefore one of the most important sectors in today's economy. Due to this big leverage increases in efficiency can, even if they appear small in relative numbers, lead to great savings in absolute numbers.

Due to high competition in this sector, efficient management on all levels is paramount for a company's success. Customers often have frame contracts, which allow to adjust order quantities to be flexible within certain ranges on short notice. Therefore it can be challenging to meet the demands in time, however a high service level is generally required to prevent loosing of customers in this competition-intense market. Production systems need to retain flexibility to be able to produce the required quantities on time. However, production systems are in most cases highly automated with capital-intense machines, therefore it is generally difficult to keep sufficient capacities at bay to meet with unexpected demands on short notice — it is important that capital intense resources are highly utilized.

Most consumer goods industry face an increasing number of products and product variants. E. g., where in yoghurt production a couple of decades ago there was only natural flavor and perhaps a hand full of different flavors, today it is not only an increased number of flavors, but diet variants with decreased sugar and/or fat, special variants for Lactose-intolerant customers, variants based on soy milk for vegan customers, variants using only biological produced ingredients and so on. Using modern production technology, the production capacity of a single machine is usually much too high to be utilized by a single variant, therefore several variants are produced on the same machines. This requires *changeover activities*, also called setup activities, for cleaning and preparing the machine to switch from one variant to another, comsuming time and therefore reducing available capacity for production. Creating production plans which minimize the capacity loss of expensive machines due to such changeovers is an important and often difficult problem, especially if the required time for this changeover time depends on both the predecessor and the successor product (this is commonly referred to as a *sequence-dependent changeover*).

[1]Cf. Statista GmbH (2013)

This problem is treated in two planning steps: lot-sizing and scheduling. The term lot-sizing describes the grouping of production orders, which are necessary to fulfill real or forecasted product demands, into continuous production lots of the same product which are to be produced without being interrupted by production of other products on the same resource. In scheduling, the production lots are assigned to time windows on specific resources. This two steps can be carried out successively, though this can result in inferior solutions or even infeasibilities, e. g. if in the lot-sizing procedure the defined production lots are large and for some products there is a limited time until first demands occur, then it may not possible to produce the production lots for all such products in time, where with smaller lots this would be possible. Due to this effect, in many production environments there is an increasing tendency to solve the lot-sizing and scheduling in a single integrated step. The drawback is the high complexity of this problem, making it more difficult to find good or even optimal solutions.

While due to the high complexity early approaches were based on successively planning the lot-sizes and then scheduling them, often by means of simple heuristics, computing technology has developed rapidly in the last decades. On the hardware side, CPU power, memory, network speed etc. have increased, and new and adapted software in regard to databases, multi-tasking and mathematical solving were developed. Also, production planners are more accustomed to using computing technology. These developments opens new possibilities in production planning which were partly exploited by the development of Enterprise Resource Planning software in the 1980s, and more recently the development of *Advanced Planning Systems* (APS). However, APS still lack sophisticated procedures for integrated lot-sizing and scheduling problems, especially for complex changeover structures[2].

In this thesis, the problem of lot-sizing and scheduling in the FMCG industry shall be observed with focus on the impact of different changeover structures to the solution process. This shall help practitioners and researchers to select appropriate basic methods when developing solution methods for specific applications of (integrated) lot-sizing and scheduling. In the following, in chapter 2 the specific properties of the FMCG industry are discussed. Chapter 3 outlines approaches to treat the lot sizing and scheduling problem. In chapter 4, three scenarios which differ by the changeover structure and some related minor aspects as they are typical for the FMCG industry are defined and models are developed based on different standard model formulations to solve the integrated lot-sizing and scheduling problem for those scenarios. Then, in chapter 5 the different modeling approaches are evaluated and compared to each other in regard to quality and efficiency, and how they are impacted by problem size and the ratio of required and available production time. Finally, chapter 6 draws conclusions from the previous results and suggestions for further research and applications in real industry are given.

[2]Cf. Tempelmeier (2008, p. 423)

2. Fast Moving Consumer Goods Industry

Production in today's industry can have many appearances which need to be regarded in all stages of management. The building of a bridge is different than designing and manufacturing special machines, producing mass-customized cars or producing bottles of fruit juice. Therefore, problems that arise in planning and production and methods to treat them are not easily transferable from one type of industry to another. This is especially true for the lot-sizing and scheduling problem. While this is neglected by conventional *Production Planning Systems*, that are based on the *Material Requirements Planning* (MRP) or *Manufacturing Resource Planning* (MRP II) concepts, which form the base for industrial applications for production planning in *Enterprise Resource Planning-Systems* (ERP-Systems)[1], more modern *Advanced Planning Systems* (APS) often offer, among other aspects, more industry-specific methods or interfaces to attach external solution methods[2]. Therefore, when treating management problems it is necessary to define the characteristics of the underlying industrial environment and adjust solution methods to this environments.

The *Fast Moving Consumer Goods industry* (FMCG industry) covers a wide range of products of daily use. According to Beck (2002) "the term Fast Moving Consumer Goods Sector is used [. . .] to mean those retailers and their suppliers who provide a range of goods sold primarily through supermarkets and hypermarkets. The core of their business is providing 'essentials' such as various fresh and processed foodstuffs, but they also stock a wide selection of other goods as well, including health and beauty products, tobacco, alcohol, clothing, some electrical items, baby products and more general household items." Cooper et al. (1994) define Fast Moving Consumer Goods as "products sold for everyday use in large quantities; applied to items such as processed foods, snacks, detergents, toothpaste, and so on."

As can be seen from above definitions, the FMCG industry covers a large variety of products. The perhaps most important examples are:

- Food (e. g. meat, fish, vegetables, fruits, dairy products, convenience food)
- Beverages (e. g. bottled water, soft drinks, fruit juices, alcoholic beverages)
- Tobacco products (e. g. cigarettes, cigars, cigarillos)
- Paper products (e. g. Newspapers and Magazines, print-out paper, cardboard boxes)
- Chemical products (e. g. soaps, cleaning preparations, perfumes)

[1]Cf. e. g. Günther and Tempelmeier (2012, pp. 333–335)
[2]Cf. e. g. Günther and Tempelmeier (2012, pp. 359–373)

This industry forms a major part of today's economy. For example, food industry in Germany consists of 5970 enterprises of different sizes, employing 555,000 workers, having a turnover of 169.3 billion € in 2012[3].

In literature, several case studies belonging to the FMCG industry have been regarded under different aspects. For food processing industry, Van Donk (2001) gave an overview of characteristics relevant for production planning, which he derived from five other publications. The tobacco industry was treated e. g. by Van Dam et al. (1999), describing the packaging process in a production plant. In paper industry, e. g. Bouchriha et al. (2007) described a cyclic approach for lot-sizing. Ferreira et al. (2010), Christou et al. (2007) and Bilgen and Günther (2010) gave some major aspects of the beverage industry.

Derived from these descriptions, the following characteristics are typical for most of FMCG industry:

- Mass production of standardized products or product variants is common,
- usually several product variants require a similar task schedule/routing,
- products usually have a long life-cycle, product variants may have a long or short life-cycle,
- customers are price-sensitive and profit margins are low due to a high competition market.

This product profile encourages usage of mass production technology[4], as is common to be seen in today's consumer goods industry. Usually, flow production systems are established to mass produce one or, as the demand for one product often does not require the full capacity of modern production lines, several product variants. The following major aspects regarding the lot sizing and scheduling are therefore to be considered:

- Due to low profit margins and expensive, yet high productive machines, a high utilization of the available resource capacity is paramount,
- capacities are limited and can usually not be expanded by overtime due to already high utilization,
- while usually several production lines co-exist, often certain product types are exclusively produced on one production line due to technical requirements or decisions of the management,
- demands have to be met in time, as otherwise customers/retailers might switch to a competitor[5],
- since several products are to be produced using the same resources, changeover activities are common which may require changeover costs, e. g. for residue cleaning, and/or capacity consuming changeover time,

[3]Cf. Bundesvereinigung der Deutschen Ernährungsindustrie (2013, p. 11)
[4]Cf. e. g. Günther and Tempelmeier (2012, pp. 60–62))
[5]For applications where stockouts are acceptable, see e. g. Liu and Tu (2008)

- those changeover requirements are usually high for changeovers between different product types and small for changeovers between product variants of the same product type,
- production usually needs several processes to turn the input materials into the desired products,
- in some cases, shelf-life of the products is limited, making high inventory levels undesirable.

In the following, some of these aspects are treated more in detail. First, the production environment is further explored regarding the necessities of production planning. Second, the problem of dimensioning the capacities on a mid- to long-term basis is discussed. Third, different kinds of setup structures as they are found in the FMCG industry are explained. Then, the question of choosing an objective funtion, which decides which of all possible feasibles plans should be chosen, is treated. The chapter is finished by a short summary, outlining the most relevant aspects of the FMCG industry as they are considered in the remainder of this research.

2.1. Production Environment

The production process in the FMCG industry is usually organized in flow production systems with several variants of a product on each production line. While production is in most cases a so called *multi-level production*, meaning several production steps are required to transform the ingoing materials into the final product, in many FMCG industry environments one dominant resource type can be identified. E. g., in bottling companies, raw materials are a concentrate, plain water, and plastic bottle pre-forms. The first production step is to mix the concentrate with plain water and perhaps additional ingredients in an industrial mixer to create a fluid as intermediate product. The fluid is then transported by pipe to a filling machine, which creates the bottles from the pre-forms using a moulding device and directly fills the liquid into the bottle. Often it is also labeling the filled bottles and packaging them, creating the final product to be shipped to customers. Here, the industrial mixer is of minor importance from an economical point of view. Adding capacity to the mixing step is fairly cheap, while adding capacity to the filling machine is rather expensive. Therefore it is of paramount importance to utilize the filling machines capacity as highly as possible. To achieve this, the production line is usually designed in a way that the capacity of the mixer is never a limiting factor (breakdowns excluded). Then, the line can be regarded as a single level *make-and-pack-production*[6] — the production plan for the filling machine is determined and its requirements are propagated to the industrial mixer. This is a common concept

[6]Cf. Lütke-Entrup et al. (2005) give the following definition: 'In literature, a production environment which is characterized by a single production stage and a subsequent packaging stage is named 'make and pack production"

for flow production systems if they do not have a so called *shifting bottleneck*[7]. For other production concepts like e. g. job shop production, usually multi-level production has to be considered since the large product portfolio with changing demands does not allow identifying a single dominant resource, but several capacities can become a restricting bottleneck during a single day[8]. For the FMCG industry, in regard to production planning a single-level make-ans-pack production system is common and will be considered in the remaining of this thesis.

2.2. Capacity Dimensioning

In the mid-term production planning, the dimensioning of capacities is an important problem impacting the options of subsequent planning steps. If no stochastic effects like resource breakdowns should be considered, capacity is consumed by two activities: process times and changeover times. Process times can roughly be distinguished being either constant for a range of lot sizes or being constant per unit. In most FMCG industries, constant process times per unit are given, as will be assumed in the following, therefore the total capacity requirement for processing is given by the constant process time per unit and the demands. Demand figures for specific products or product families face two characteristics, the *demand forecast accuracy* and the *demand levels*:

- In production planning, future demands can stem from real customer orders (also referred to as a *Make-To-Order Production*) and/or statistical demand forecasts (also referred to as *Make-To-Stock Production*) or a combination of both. While in environments where capacities are tight and stockouts or backorders are an option, it can be reasonable to treat both differently and prioritizing real customer demands over forecasted demands. In FMCG industry usually stockouts/backorders are to prevented and all demands have to be fulfilled on time. While demand quantities and delivery dates for real customer orders are usually known (though order modification or cancellation can still incur some degree of uncertainty), demand forecasts always face a degree of uncertainty. Since the modeling of distribution functions for uncertain demands would result in highly complex models which are hardly solvable, it is common to assume the demands forecasts are deterministic, treating the degree of uncertainty in different ways like e. g. defining safety stocks which can be used to satisfy demands which are higher than expected, leading to a third source of demand figures, the adaption of expected inventory levels to the safety stock levels. By this, other steps in production planning usually can assume deterministic demand levels.

[7] A "shifting bottleneck" describes a multi-stage production environment, in which the capacity usage of different process stages is so dependent on the production plan, that in different time windows different resources form the bottleneck resource, limiting the capacity of the production line. In such environments, the generation of a feasible production plan usually requires the modeling of the capacitated multi-stage production process.

[8] Other publications consider multi-stage production environments, see e. g. Akkerman and Donk (2008)

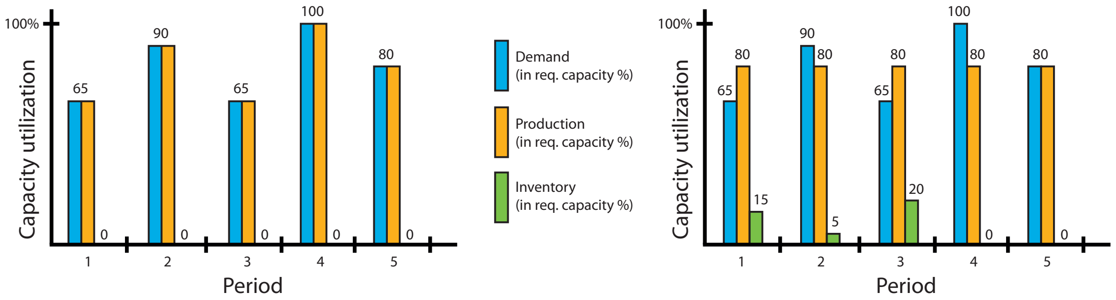

Figure 2.1.: *Demand synchroneous production (left) vs. Demand anticipative production (right).*

- Demand levels can be stable or varying over time. The degree of variability depends on the product and market environments, but also on parameters like periodization[9] or product aggregation[10]. Choosing appropriate period sizes and aggregation methods is therefore of importance, but out of scope for this research. In case of stable demands, dimensioning of capacities is trivial, however the more common case are varying demand levels. Basically, there are two ways to meet varying demands levels (illustrated in figure 2.1): first, production can match demands, which means capacities need to be sufficient to meet high demand periods, which is an attractive option if capacities are cheap or can easily be extended[11]; second, capacities can be utilized at a more stable rate, building up inventory in low demand periods to satisfy demands of later high demand periods.

So called *lot-sizing and scheduling* methods are implemented in the operative planning level to generate production plans which can satisfy the given demands under limited capacities under consideration of company goals like profit maximization/cost minimization or retaining high flexibility. Since in the mid-term tactical planning level in which decistions regading the capacity dimensioning ar made exact lot sizes and changeovers are not known, but neglecting those would result in capacities to be sized to low, an estimate for capacity loss due to changeover activities is necessary. It needs to be sufficient to give the operative planning level not only to find feasible plans, but to give it enough flexibitily to find economically desirable production plans, and of course it should not be too high, as this would lead to expensive capacity not being utilized in an economical sensible way anymore in the subsequent planning steps. In the FMCG industry, many products face seasonal demands, e. g. demand for many beverages is higher in summer than in winter. Even for products like print-out paper, which on the customer side has a relatively stable demand, due to the *bullwhip effect*[12], demands for production can vary significantly over time. E. g., Zotteri (2013) reports in the personal care sector that in

[9] E. g., demands for periods of weeks may be stable, but highly variable if periods are individual days.
[10] E. g., the demand for a product group may be stable, but demands for different variants of the product can be variable.
[11] E. g., if the capacity is limited by manpower, in Germany it is possible to hire "Leiharbeiter", or temporary staff, to supplement manpower capacity in peak demand periods. They form a capacity reserve that if not used cause no cost.
[12] Cf. e. g. Tempelmeier (2006, pp. 157–164)

more than half of the analyzed cases the demand upstream (sell-in) is twice as variable as the demand downstream (sell-out). Due to the low profit margins and capital intensive resources, capacity dimensioning should aim at a high capacity utilization while at the same time enable a high service level to satisfy customers in a high competition market. Therefore, keeping capacity reserves for high demand periods is usually not viable in the FMCG industry, but capacities need to be utilized highly in all periods. This faces a number of problems as well: building up high inventories can cause significant costs, especially if the shelf life of the products is limited[13], and high production volumes of a single product may use up required capacity for other products which are low on stock due to unexpected high demands.

2.3. Changeover Structure

In the FMCG industry, usually several product variants are produced on the same production line, therefore *changeover activities* (also called setup activities) have to be carried out when switching from one product to another. While process activities usually consume most of the available capacity for productive activities, changeover activities form an unproductive capacity requirement which can often consume a highly relevant amount of capacity, making it unavailable for revenue generating activities.

Since the capacity requirement of process times is usually given by deterministic demands and constant processing times, they cannot be influenced by the planning. In production planning in the mid to long term, the effects of the actual sequence in which products are produced can often be neglected and average setup requirements based on experience can be used, often directly reducing the available capacity. On the other hand, as the number of changeovers depends on the number of lots and the required changeover time can depend on the sequence in which the lots are produced, there is a potential to safe expensive capacity time by proper methods in short term production planning. The complexity of this problem is highly dependent on the changeover structure. Changeover structures can be categorized into the following five basic types:

- In the simplest cases, changeover activities are *sequence independent*. The exact sequence of the scheduled order is of no importance in regard to changeover requirements, and can be neglected in big bucket models, which at most regard the last setup state of a big bucket period for setup carryover. An extreme case is where the changeover cost and times are negligible and therefore assumed to be zero, in which case the problem of lot-sizing and scheduling is reduced greatly and simple methods like prioritization regarding to the stock run out time can be employed.
- Another type can be referred to as *family changeover*, in literature sometimes referred to as *coordinated setup*. The machine is set up and can then produce a part of the product portfolio, the product family, with only small or no setup requirements for a changeover between different products belonging to the same

[13] For considering of shelf life, see e.g. Lütke-Entrup et al. (2005) or Bilgen and Çelebi (2013).

family. In some environments, the setup activity can be done off-line, minimizing the setup time on the production resource, e. g. in PCB-Assembly using trolleys which are set up with a set of components required to produce a set of products. The changeover activity on the production capacity is then simply changing the trolley, which interrupts production only for seconds. Other environments require a production stop while the changeover takes place, e. g. in beverage industry when the moulding device for forming the bottles in a make-and-pack production line is changed.

- With *limited changeover* environments are described, where changeovers for some combinations of predecessor product and successor product are prohibited. This can be the case for example in chemical industry, where properties of the production process allow only for changes between similar products, or in pharmaceuticals, where small quantities of a previous product could contaminate the succeeding product.

- A *natural sequence* exists, when technically or from an economical point of view only a certain sequence is feasible respectively reasonable and can be obtained in advance. This is quite common in some process industries, especially food and beverage industry, e. g. as described in Bilgen and Günther (2010) or Farahani et al. (2012). For each product that is to be produced within its natural sequence, there is a constant or product dependent minor changeover time, while changing out of the sequence (going back in sequence or changing to a product which is not part of the previous sequence at all) incurs a major changeover time. In today's industry, this concept is often found under the name Production Wheel and has been regarded in several other publications, e. g. Lütke-Entrup et al. (2005). Note that this is to some extend the opposite of approaches where first the lot sizes are determined in advance and in a succeeding step the scheduling, and therefore also the sequencing, is carried out.

- The most general type of changeover structure is the *full sequence flexibility*. Just like in the *independent* case, the sequence of products can be chosen arbitrary, but here the changeover times and/or costs depend on the sequence, i. e. both the predecessor product and the successor product. While this type of sequence can be used to reflect all other type of changeover structure, it is the most complex one, requiring a quadratic increasing number of binary variables for reflecting changeover activities with increasing number of products.

These basic types do not necessarily exclude each other, but can be combined to form combined types. For example, a product families with a full sequence flexibility changeover structure when changing between product families and a natural sequence changeover structure for changeovers within a product family. In today's FMCG industry one can find very different kinds of setup structures.

Note that changeovers often do not only cause costs, e. g. for cleaning materiala, but in many cases also capacity loss because the capacity cannot continue processing while the changeover activity takes place. While in literature most regarded problems incur changeover costs, they often neglect changeover times. This is because including changeover times into the models makes them a lot more complicated to solve, therefore changeover costs are included to put some penalty to prevent unnecessary changeover activities. In most cases, out-of-pocket costs for changeover activities can be neglected. However, in cases where they are relevant for the production planning they can just as well as changeover times be highly dependent on the changeover structure.

2.4. Production Plan Assessment

As a plan that cannot be executed or violates major goals of the planner is hardly of any use, the paramount aspect of a production plan is its feasibility. The most important hard constraints which must be met by a plan to be feasible are demand satisfaction, regarding capacity limits and technical practicability. In today's FMCG industry, in can be that no feasible plan exists. This can then be met by relaxing the former hard constraints to soft constraints as long as it does not affect the technical practicability. E. g., in some cases capacities may be extended by adding extra shifts, delaying maintenance activities to later periods, adding more personnel, in other cases demands may be partly split up into several deliveries with some being postponed. This is usually not possible without additional costs. For the remainder of this thesis, it is assumed that capacity dimensioning is sufficient and feasible production plans exist.

Then, in most cases not only one but several feasible plans exist, making it necessary to define a criterion to assess the plans, to chose the one best suited for achieving the companies goals. This is done by defining an *objective function* which assesses the plan. Possible objective functions can be distinguished in *multi-objective functions* and *single-objective functions*:

- In multi-objective functions, two or more different objectives are taken into consideration when evaluating the plan, e. g. minimizing the costs composed of changeover costs and inventory holding costs and maximizing the service level. The is problematic, as in most cases the different objectives are interdependent on each other. This can be treated by either defining some way of a trade-off possibility, or by settling for finding a Pareto-optimal solution where no part of the objective function can be improved without worsening at least one other part. In integrated lot-sizing and scheduling problems, Pareto-optimal solutions are only of minor usability, since still it would be necessary for the many possible Pareto-optimal solutions to decide which would be best. However, it would still be possible to construct a Pareto-optimal front. For example in the fresh-food industry, it can be advantageous to use multi-objective functions, see e. g. Arbib et al. (1999) or Amorim et al. (2012). Defining trade-offs between different feasible solutions is often hard in itself, even more so that usually such trade-offs would not be linear and therefore even if definable, models using such objective functions are hardly solvable.

- In single-objective functions, there is only one objective that is taken into consideration when evaluating the plan. In literature for lot-sizing and scheduling, the most common objective function is the cost minimization. A cost value is assigned to changeover activities and to inventory levels, the sum of those costs is to be minimized. Costs assigned to changeover activities can reflect out-of-pocket costs, like for cleaning material or lost residue raw materials and/or intermediate products. Holding costs can impute real out-of-pocket costs for storage, but since storage facility costs rarely depend on the inventory level those are rarely relevant for production planning, or a decreasing value of the products over time, e. g. due to the selling price being dependent on remaining shelf life, if shelf life is restricted[14]. However, in most cases costs in such objective functions are more dependent on imputed costs. This is done if using a multi-objective function or modeling additional restrictions would increase the model complexity beyond applicability for real industry scenarios. For example, changeover costs are often included to circumvent modeling of capacity usage by changeover activities. Some factor is determined by the human planer by which the available capacity may be utilized for production, reserving some for changeover activities, and another factor penalizes changeover activities to keep capacity usage by changeovers low. However, this can easily lead to infeasible plans overloading capacities in some periods, or to leaving free capacity unused which could be used for an advantageous changeover activity.

In most of FMCG industry, to achieve a high capacity utilization and high service level it is highly recommendable to explicitly model capacity usage by changeovers to neither waste capacity nor to cause stock-outs due to capacity bottlenecks. The inventory turnover rate is usually high, products are not kept on stock for a long time but shipped in short time, inventories are merely regarded as buffers between the manufacturing and distribution stage of the supply chain[15]. Production technology is usually designed to minimize out-of-pocket changeover costs, e. g. the cleaning material for a bottling line is relatively cheap. Therefore, the imputed costs of capacity usage are not relevant since capacity usage should be directly modeled, while capital bonding costs and out-of-pocket costs are of minor importance for short term production planning. Of much more importance are goals like maximizing effective capacity utilization.

While future demands are assumed to be deterministically known for the planning horizon in regard to production planning, in practical application demands can depend on the planning result. For example, if capacity utilization in the planning horizon is low, this opens opportunities to increase demands by marketing activities like promotions, accepting additional orders or taking over production volumes from the market for trade brands. In some cases, it may be advantageous to have a longer production stop than several shorter capacity wastes, e. g. for doing time intensive maintenance or setting manpower free. In such situations, it is advantageous to use a different objective function like the Makespan minimization. The Makespan is defined in this thesis as the time until all capacity consuming activities are finished, given production is sufficient to match

[14]Cf. e. g. Lütke-Entrup et al. (2005)
[15]Cf. Günther (2013)

all demands. By preferring plans with a low Makespan, the company retains maximum flexibility in regard to acquiring new economically effective capacity requiring activities, maximizing the capacity utilization, and a degree of flexibility if real demands exceed forecasted demands to be able to alter the production plan for achieving a high service level.

2.5. Conclusions in Regard to Production Planning in FMCG Industry

In this chapter, an overview was given regarding the relevant aspects in production planning in the FMCG industry. While production usually consists of several production steps, it was described that in many cases in regard to lot-sizing and scheduling only a single Make-And-Pack production stage needs to be considered. It was pointed out that the capacity dimensioning should focus on allowing for a high capacity utilization in all periods while allowing demands to be met in time. In short-term production planning and scheduling, capacity usage is distributed over the periods, while the capacity usage by changeover activities is dependent on the generated production plan. Changeover structures can be of varying types in the FMCG industry, featuring changeover times dependent on the following product, times depending on both the predecessor und the successor product, structures differentiating between product variants and product groups, and/or prohibition of specific predecessor-successor relationships. Finally, ways to assess feasible production plans are discussed, concluding the Makespan being a sound objective which is of high practical relevance in today's FMCG industry, at least where deteriorating products are not of an issue.

In the remainder of this thesis the production planning in the FMCG industry as outlined in this chapter shall be further explored. In the following chapter, methods which can be applied in the short term production planning in the FMCG industry shall be outlined, with focus on the lot-sizing and scheduling problem. Then, three case studies from the FMCG industry are given, for which models are developed to perform the integrated lot-sizing and scheduling. Those models are then compared in the fifth chapter. The final chapter will give a conclusion regarding the importance of the setup structure in integrated lot-sizing and scheduling in the FMCG industry.

3. Methods in Production Planning

In last decades there has been a vast research regarding the development and implementation of methods which can help planners in solving lot-sizing and scheduling problems for real industry. In this chapter, an overview about these methods shall be given and discussed regarding their applicability in today's FMCG industry.

3.1. Overview of Planning Concepts

The problem of production planning and control treats the planning, control and monitoring of production activities in regard to the location of production, the timing of production and the quantities of production. Since the real problem is too complicated to be treated as a whole, it is usually divided in strategic decisions, tactical decisions and operative decisions[1]:

- Strategic decisions cover a long time horizon and very highly aggregated problems, e. g. decisions regarding the network design of production locations and the long term goals of the enterprise. They are usually the responsibility of the top management of the enterprise.
- Tactical decisions are based on the decisions of the strategic planning and cover a mid-term time horizon, usually from a couple of months to a couple of years, on a highly aggregated view, e. g. which type of production resources should be acquired, how the capacities in a plant should be dimensioned or how the layout of the given production plants should be designed. The top management of the respective plant is usually responsible for these decisions.
- On the operative level, detailed decisions are made, usually having a short time horizon of a couple of weeks to months at most. They cover decisions like the determination of the short term production plan, the usage of production factors and production processes and more. They can be as detailed as to which product should be produced in a specific time window on a specific machine, using a specific production process and specific input materials. The problem of lot sizing and scheduling is a planning problem typical for the operative level.

On the operative level, the *Manufacturing Resource Planning*, or MRP II, concept, which is illustrated in figure 3.1, is the perhaps still most used planning concept in industry. It evolved from MRP, or *Material Requirements Planning*, which is based on a commercial database management package developed by IBM in the 1960s. While MRP

[1] Cf. e. g. Drexl et al. (1994)

focused on determining quantities and timing of raw material purchases, MRP II extended it to integrate all aspects of the manufacturing process, being included in many ERP systems which were developed in the 1980s. However, neither the hardware nor software nor database technology of that time was capable to run these systems in real-time and the cost were prohibitive for most businesses[2]. Advances in these technologies allowed a more wide spread of this technology in the last decades.

It is based upon the *Push-Principle*, where a central planning instance in each production plant generates goals which are "pushed" into the production process of this plant. In the Master Planning step, planned production quantities for final products or product groups are determined based on customer orders and/or forecasted demands and perhaps an aggregated mid-term production program to level seasonal demand fluctuations under consideration of current inventory levels. Based on the previously determined production plan, in Quantity Planning quantity requirements for input materials are determined based on the *Bill of Materials* and requirement dates are determined by using experience based *lead times*, while production quantities are grouped into production orders using usually uncapacitated lot-sizing heuristics. The result are production orders which are roughly assigned to time periods and resource groups according to their production process requirements. In Schedule Planning those production orders are scheduled to specific time windows on specific resources, usually using simple uncapacitated algorithms. If resources are overloaded in some time windows, the Capacity Leveling tries to reschedule production orders as a whole or parts of them to other time windows or resources to generate a feasible plan. The result should be a feasible production plan which can then be transferred to the production control level to execute and monitor the production process.

However, industry practitioners found the results of MRP II based technology often lacking, as capacity bottlenecks were common, leading to an insufficient service level while having a high work in progress at the same time binding high amounts of capital. In research this problem has been addressed, e. g. Drexl et al. (1994) point out that, among other issues, the main problem of the MRP II concept is its lack of considering the availability of resources.

Pochet and Wolsey[3] argue that methods applied in the MRP and MRP II concepts and the ERP systems are of heuristic nature and consider mathematical programming as a means to generate better plans, though for problems of realistic size without abstraction available computing power still can be a limiting factor. The major drawbacks mentioned by these and other authors are:

- A decision level for a Supply Network Planning is not included in the MRP II concept at all. While this may not have been necessary widely by the time the MRP II concept was developed, in the last decades business environment evolved, causing enterprises to have several production plants with a complex network of supplier/consumer-relationships. Additional methods for assigning production

[2] Cf. Shum and Lin (2003)
[3] Cf. Pochet and Wolsey (2006, pp. 46–68)

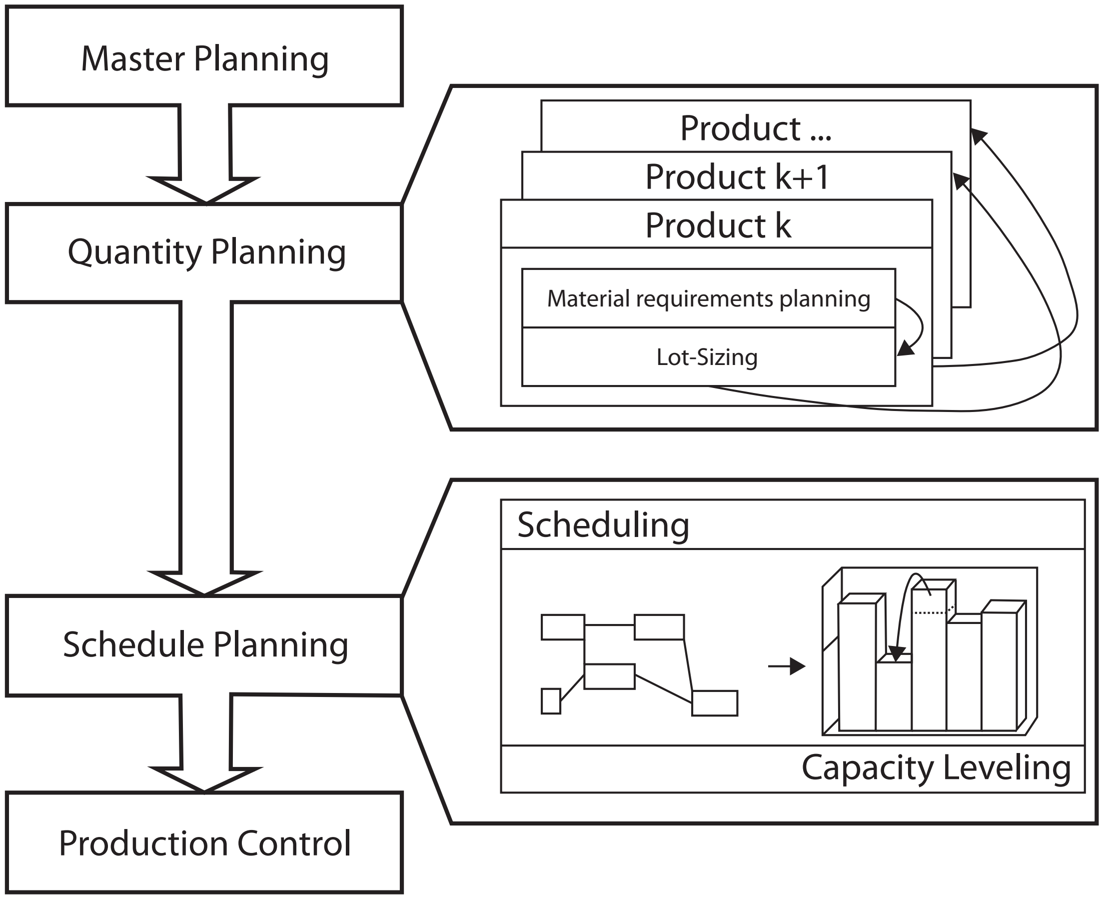

Figure 3.1.: *MRP II planning concept, (cf. e. g. Günther and Tempelmeier (2012), p. 334).*

quantities to plants and for sourcing decisions within the Supply Network are of high importance.

- Due to the top-down sequence of decisions without a systematical consideration of feedback from lower planning levels, in the MRP II planning concept goals defined by superior planning levels for subsequent planning levels may not reachable. E. g., in Quantity Planning capacities are aggregated and a production plan is generated while the amoutn of capacity that is consumed by changeover activities is not known yet. To account for changeover activities, the theoretical available capacity is reduced by some reduction factor based on experience, limiting the capacity usage by production processes. If this reduction factor is to high, capacity may be wasted or suboptimal lot-sizes may be generated, if it is to low the subsequent level may have difficulties finding even a feasible plan without shortages when considering the changeover capacity requirements. This problem is of course even more important if capacities are not considered at all until the final scheduling planning level, which is common in MRP II based planning due to a single-item decomposition, meaning the products are planned independently of each other, even if they share a scarce production resource.

- In material requirements planning, *lead times* (the time from starting an order until it is completed and outputs are available for further use) are based on experience values from the past, while the actual required lead time is strongly dependent on the generated plan. Often more than 85% of the lead time stem from waiting times like e. g. queue times while waiting for a resource to become available, and are therefore highly variable. Especially in multi-level production, when output products of one process are required as input for the next process, it is common to assume very pessimistic lead times for intermediate products to prevent later productions steps to be postponed due to unavailability of the required input material. While this may be necessary to achieve a reasonable utilization rate of resources later in the production process and reasonable service levels, it leads to triggering production of intermediate products even if the process requiring those as input materials cannot be started for a significant time due to other production orders or otherwise caused unavailability[4], causing high *work in process*[5] and therefore high unproductively bound capital and often additional logistical issues regarding the storing and handling of the work in process material.

- Lot-sizing is carried out for each (intermediate or final) product independently of other (intermediate or final) products, even if they are connected to each other by requiring scarce capacity of the same resources or cost interdependencies. Also, generated lot sizes may be disadvantageous in regard to the scheduling, e. g. two lots requiring 6 units of capacity, with only 10 capacity units available per period, would make it impossible to utilize a periods full capacity unless the lot-sizes are changed which can have impact on the optimality of the then changed lot-sizes.

- It is a common problem that in scheduling capacity overloads are generated, which are tried to be removed by the *capacity leveling* step, by *forward rescheduling* (i. e. production orders are moved to an earlier time window) or *backward rescheduling* (i. e. production orders are moved to a later time window). However, forward rescheduling is often not possible due to lacks of available free capacity and especially if a multi-level production process has to be considered forward rescheduling can be limited by availability of the input products for the process. Backward rescheduling is limited by the requirement dates. Often the only way to generate a feasible production plan is to accept backorders, reducing the service level and therefore leading to undesired effects on customer relationships.

- Methods applied in MRP II based concepts and ERP systems are often simple heuristics, which do not guarantee an economically optimal plan even for the relatively simple models. Using true optimization techniques can therefore even be useful if the aforementioned issues are not relevant for a production environment,

[4] See e. g. Günther and Tempelmeier (2012, p. 338)
[5] Also often called *work in progress*, *goods in process* or *in-process inventory*

while using modern technology such relatively simple problems can often be solved in very short time to optimality.

These problems of the MRP II approach, especially high inventories and/or low service level in capacity tight situations, led to several developments which will be discussed briefly in the following.

One approach which is aimed mostly at reducing the high inventories is to replace the Push-Principle based MRP II concept by a concept following the *Pull-Principle*. In the Pull-Principle, production control is decentralized, e. g. production orders are not generated by a planning instance, but by defined triggers in the work shop. Well known examples are the KANBAN system and the CONWIP system. For example, in basic KANBAN, the planning instance defines for each work station a number of containers holding the required input materials for production and quantities of those per container. When one of the containers is emptied in the production process, an information is given to a source location that a new container needs to be provided, triggering the production process of the input material. By this, production is only started if a specific process demands for new input material as its stock is running low, therefore the total work in process in limited. While this approach to production control can be very efficient in some production environments, it should be noted that they have tight requirements, most notably a high reliability of the capacities, a stable demand rate on intermediate products and low capacity usage due to changeover activities.

Another answer to the problems of the MRP II based planning concept came from the software industry. There, the MRP II concept is implemented in the *Enterprise Resource Planning Systems* (ERP-Systems), and they usually employ additional methods based on the Pull-Principle. For many environments, the results were not satisfying and customers demanded for more advanced planning support tools. This led to the development of the *Advanced Planning Systems* (APS), which are usually used to supplement the ERP-Systems, e. g. for planning of important products in capacity tight situations. They feature a couple of extensions compared to the ERP-systems, like additional modules for e. g. Supply Network Planning, Collaborative Planning and Capable-to-Promise, but also enabled the usage of capacitated planning methods and true optimization or more advanced heuristics. For example, the SAP AG developed the *Advanced Planner and Optimizer*, which features a capacitated MILP for Supply Network Planning and a Genetic Algorithm (GA) for short term sequencing considering the available capacities (the methods of GA and MILP are briefly described in subsection 3.4.1).

Paralelly to the develpment of the APS, the problems of the MRP II concept were addressed in literature. Drexl et al. (1994) developed the so called *Hierarchical Planning Concept* as a concept for operative capacity-oriented planning, still neglecting supply network issues, which it was later extended to (see figure 3.2)[6]. One can find many similarities to the APS, which to some extent incorporate the principles of the Hierarchical Planning Concept.

[6]See e. g. Günther and Tempelmeier (2012, pp. 340–342)

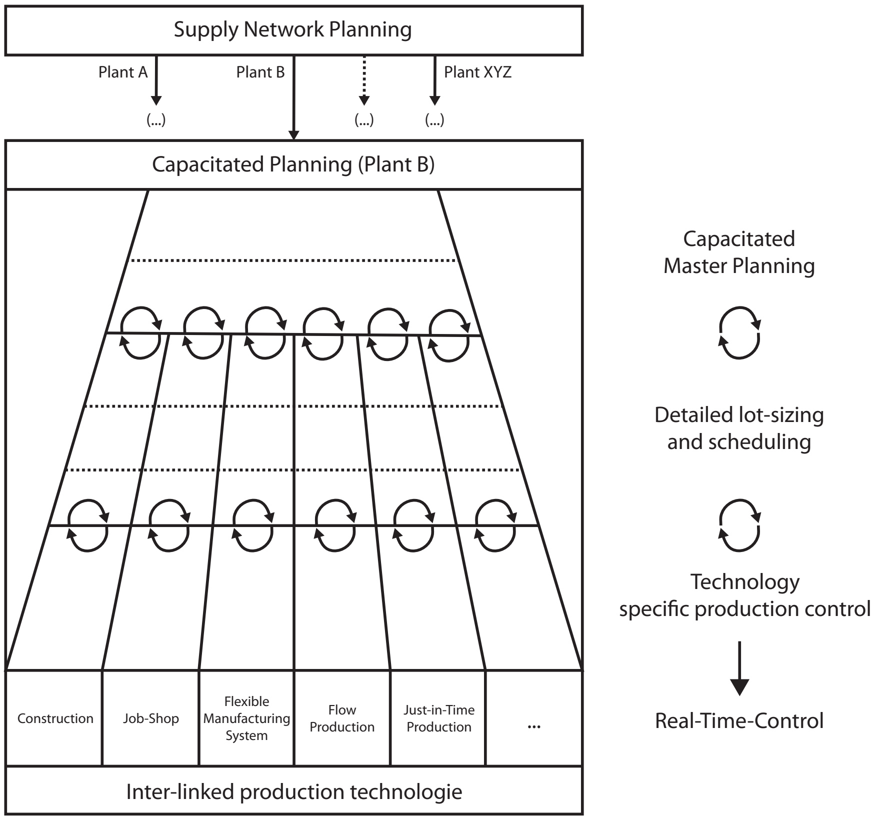

Figure 3.2.: *Hierachical planning concept (based on figures in Drexl et al. (1994) and Günther and Tempelmeier (2012, p. 342)).*

In the aggregated Supply Network Planning, the production quantities for *product types*[7] are assigned to the different production plants and, if necessary, demand fluctuations over a relatively long time horizon are considered by building up stocks in lower demand periods to be able to satisfy higher demand periods without overloading the capacities. In the Master Planning of each plant, the required quantities for products are assigned to more detailed periods and resource groups. Then in the detailed lot-sizing and scheduling part, exact lot-sizes are determined and scheduled on specific machines. In the hierarchical planning concept, it is pointed out that the applied methods are highly dependent on the type of production and utilized production systems. This is also true for the next level, the detailed planning and -control. The main differences between the Hierarchical Planning Concept and the MRP II are two: first, the necessity for applying methods considering the special aspects of each production system is pointed out, second and perhaps even more importantly is that on all levels the limited capacities are taken into

[7] A product type consists of different products with similar cost and demand structures and similar production processes. The combining of those into product types is usually necessary due to a very high number of individual products and product variants (see e. g. Meyr (1999, p. 23))

consideration. Since at higher planning levels the exact amount of required capacity is not known yet, e. g. because sequence dependent changeover times cannot be determined before the number and sequence of lots are determined, a bi-directional information flow is an important feature of the concept, giving higher planning levels information about capacity usage, while in the MRP II concept the information flow is top-down only.

As the FMCG industry requires a high capacity usage and high service levels, MRP II based approaches are not suited for it. Pull-based concepts like KANBAN can hardly be applied, since in most cases the prerequisites for them to work are not given. Therefore, more advanced methods should be applied, considering capacities on all planning levels and considering the special properties of the FMCG industry. This will be further explored in this thesis for the lot-sizing and scheduling problem.

3.2. Classification of Lot-Sizing Models

Lot-sizing problems in production planning are highly dependent on the given production infrastructure and process characteristics. To support the production planning process, quantitative optimization models are a useful tool by giving support in finding and selecting feasible plans under consideration of an objective function and given constraints[8]. Defining an appropriate model to support decision makers is a challenging task, in which the real problem needs to be abstracted to apply solution procedures which allow evaluating the impact of decisions for the real problem. For lot-sizing and scheduling models usually a mathematical model is defined as a minimization respectively maximization problem for some objective function under consideration of constraints. Transforming the real problem into a complete mathematical model is theoretically possible, but usually only applied for describing the underlying problem, since developing and applying of solution methods for these complete models is hardly possible due to the extremely high complexity of such models. To make the mathematical model usable for decision support, it is usually necessary to focus on a set of the most important characteristics.

Bahl et al. (1987) classified lot-sizing problems into four categories based on type of production (single-level or multi-level) depth and presence of resource constraints (limited capacity or unlimited capacity). These basic classification criteria's were extended later, e. g. by Domschke et al. (1997). The criteria relevant for this thesis are:

- Static / dynamic models: Static models reflect the simplifying assumption that the demand level is static, i. e. forecasted demand quantities are constant in the planning horizon and thereafter. This simplifies the modeling a lot, allowing usually for continuous time lines where production can start at any chosen moment. In some industries a rather static demand, besides random fluctuations, can be observed at least in the typical planning horizon for production planning of at most a few months. However, even for those industries effects like the bullwhip effect can lead to fluctuations in demands, which in case of a make-to-order production or production orders to refill safety stocks can lead to dynamic demands as well

[8] Cf. Scholl (2008)

(compare section 2.2). Therefore, for the remainder of this thesis dynamic models are assumed which can reflect changing demand quantities in the planning horizon.

- Deterministic / stochastic models: While deterministic models assume exactly given parameters, e. g. demand figures or capacities, stochastic models include random fluctuations like unreliable demand forecasts or resource breakdowns. Stochastic models gained in scientific attention in the last decades[9], however in spite of giving a more accurate representation of reality, they are scarcely used in industry due to the very high increase in complexity due to the stochastic effects in regard to solvability of the models in comparison to a relatively small gain in accuracy. In industrial practice, stochastic effects are considered by building up reserves like safety stocks or some reserved capacity based on previous experience regarding the expectable breakdown time. Therefore the deterministic modeling approach is still dominant in most cases and will be further explored in the remainder of this thesis.

- Single-level / multi-level models: the production process can consist of only one operation (single-level) or several operations (multi-level). For means of production planning, these operations usually do not reflect each single required operation, but only those that can have an impact on the plan, e. g. if they require a bottleneck resource. For single-product problems, usually a single bottleneck stage can be determined, leading to a single-product and single-level model. For multi-product problems, if different products require different amounts of capacities, it may be that depending on the production plan the bottleneck stage cannot be determined in advance, sometimes even having different bottleneck stages in different time periods (this is also known as a *shifting bottleneck*). As stated in section 2.1 a Make-and-Pack-Production environment is common in today's FMCG industry. Therefore, while technically several opterations are required in the production process, here it is assumed that the production plan is determined by one limiting production operation on the most expensive resource, while the production plans for the other operations follow this plan. In this thesis, such a Make-and-Pack production is considered which is a type of single-level production.

- Single-product / multi-product models: A single-product production takes place if there either only one product is produced in a production system or the products are not interfering with each other, e. g. by technical connections (one product being the input product for another product) or by competition for scarce resources (where, if several products would be planned without regarding each other, each would use up some of this capacity and all together could overload that capacity). As in today's FMCG the production of several products and variants using the same resources is common, for the remainder of this thesis multi-product problems are assumed.

- Capacitated / uncapacitated models: In uncapacitated production planning, it is assumed that the capacities are not a limiting factor in the generation of a plan.

[9] Cf. e. g. Herpers (2009), Tarim and Kingsman (2004) or Sox (1997)

This can be the case e. g. for static demands with sufficient capacities or a Master Production Scheduling considering only a limited set of important products in a first planning step which requirements do not exhaust available capacity in any case. Because Make-and-Pack production in the FMCG industry usually requires expensive resources, capacities are considered to be limited in the remainder of this thesis.

Summarizing, for the remainder of this thesis deterministic, dynamic, multi-product, single-level and capacitated models are considered. In the following subchapters, furthermore a differentiation of successive lot-sizing and scheduling and integrated lot-sizing and scheduling is given, as well as solution procedures for the integrated lot-sizing problem and some of the most renowned models for this problem.

3.3. Successive Lot-Sizing and Scheduling

The first method for lot-sizing was developed in by Harris (1913) and is known as the *Economic Order Quantity* (EOQ) model. It gives the optimal lot-size as the square root of the quotient of two times the stable demand level multiplied with the changeover cost and the holding cost per unit and period. In it, the basic trade-off between higher lot-sizes to save changeover costs and more production lots to save on inventory holding cost can already be identified. The method requires a lot of assumptions, most importantly stable demands and unlimited capacities. The lack of computing technology then hindered further development in this area, until Wagner and Whitin (1958) formulated the *Single-Item Uncapacitated Lot-Sizing Problem* (SIULSP) and developed an algorithm for treating dynamic demands based on modeling of *shortest-route-problems*. For appliance in real industry, employing the Wagner/Whitin Algorithm still required too much computing power if a bigger number of products or a longer time horizon were to be considered[10]. Heuristics like the Silver-Meal-Heuristic[11] or Groff-Heuristic[12] were developed. In today's ERP-Systems, such relatively simple methods are still employed which do not consider limited capacities.

After the lot-sizes have been determined, the production orders are scheduled to time windows on the available capacities in the successive lot-sizing and scheduling approach[13]. Common in industrial practice is to execute a scheduling without consideration of limited capacities and then, in a second step, to execute the so-called *Capacity Leveling*, in which the capacity utilization of the resources are checked, and if a resource is overloaded production orders are rescheduled. It is differentiated in *forward or backward rescheduling*. In the first case, production orders are rescheduled to an earlier time window, if capacities are still available, which is often not the case. Also other problems can arise, e. g. in

[10]Note that Evans (1985) reformulated the Wagner-Whitin Model for a more efficient implementation in micro computers. Wagelmans et al. (1992) developed an algorithm that runs in linear time, enabling a quick solving of very large problem instances.
[11]Silver and Meal (1973)
[12]Groff (1979)
[13]For an overview of scheduling methods see e. g. Allahverdi et al. (2008)

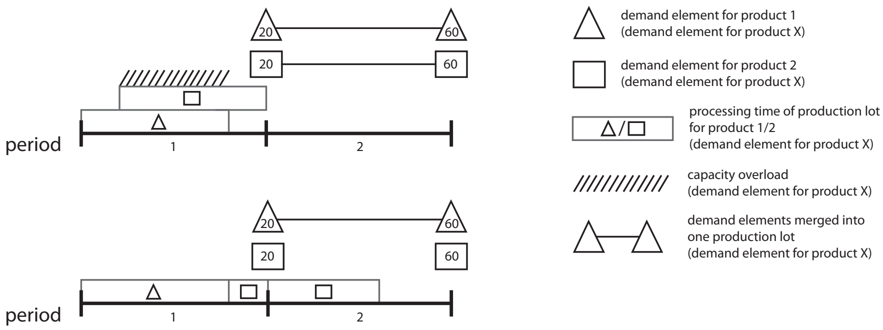

Figure 3.3.: *Successive lot-sizing and scheduling: infeasibility caused by large production lots.*

multi-level production the input materials may not be ready to start production earlier, so their production orders would have to be rescheduled as well, again limited by the available capacity. In the second case, production orders are postponed to periods with low capacity usage. This is usually limited by the date of the real or forecasted customer orders, but as often no low capacity usage periods are available in time, this restriction is violated leading to backorders and stock-outs. This can cause costs by lost sales or by reducing the price to keep the customer content. Additionally it needs to be considered that even if by this rescheduling a feasible plan can be found, it can easily be that it is far worse than a different possible but unknown feasible plan, since the rescheduling procedures are usually focusing on feasibility, not (cost) optimality.

The alternative to uncapacitated scheduling and capacity leveling are optimizing procedures considering limited capacities in the sequencing step. However, since capacities were not considered in the lot-sizing step, the lot-sizes may prohibit the finding of optimal or even feasible solutions. A small example with two products where the lot-sizes prohibit the finding of a feasible solution is given in figure 3.3. In the upper Gantt-Chart of the figure, the lot-sizing procedure combined the demand elements for both products into one production lot each. As it is impossible to schedule both to the first period, no feasible schedule exists, that enables the satisfaction of both product's demands in the first period. As shown in the lower Gantt-Chart of the figure, a different lot-sizing would enable a feasible schedule. Another example is given in figure 3.4. Here, in the upper Gantt-Chart the combination of demand elements into lots made it impossible to find a schedule which exhausts the first periods capacity. It is obvious that even when considering only two products, the result may be unsatisfactory as the production finish time is much later than possible, as can be seen in the lower Gantt-Chart which illustrates a different lot-sizing result. This becomes even more significant, if a third product is added, for example an additional unexpected customer order which is not known until the end of the first period. Then, the wasted capacity of the first period makes it impossible to satisfy this new demand without overloading the capacity, while a different schedule that would not waste some of the first periods capacity retained the flexibility to satisfy the new order.

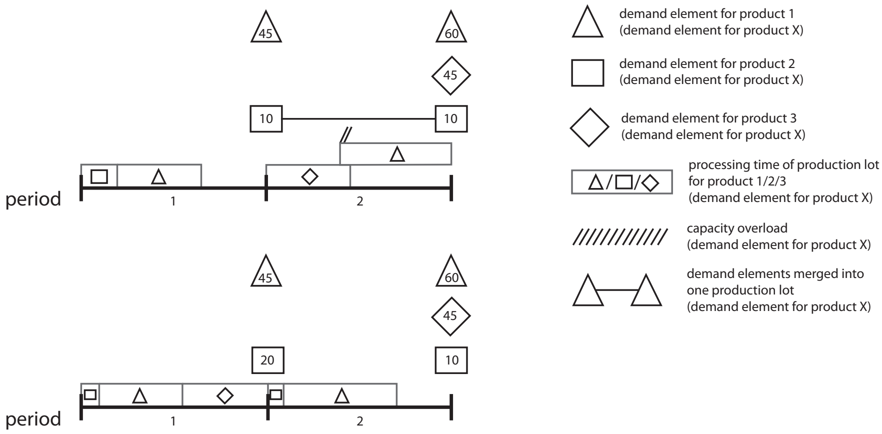

Figure 3.4.: *Successive lot-sizing and scheduling: infeasibility caused by waste of capacity.*

These issues make it necessary to change the previously determined lot sizes, hoping the new lot sizes will allow feasible solutions, which is far from certain. Also, even if the new lot sizes allow a feasible schedule, it can easily be that the so-found schedule is far from optimal.

If in the lot-sizing step the limited capacities were already considered, then an optimizing procedure in the scheduling step should lead to optimal solutions. This is true if the changeover activities are independent on the sequence. If the changeover costs and/or times depend on the sequence, carrying out an optimal lot-sizing procedure is not possible since the values for changeover costs and/or times are not known yet. This is usually treated by generating changeover costs based on previous experience or, to account for changeover times, reducing the available capacity for the lot-sizing step by an experience based value. Then, the scheduling is carried out using those lot-sizes. If the generated changeover times were too low, it may be that no feasible schedule exists. If they were too high, capacity utilization is bad. Regarding the costs, if the scheduling step generates a sequence that substantially differs from those which formed the base for generating the costs, it may be that the lot-sizes combined too many demand elements (generating high holding costs) or too few (generating to high changeover costs). It is up to the human planner to decide if the generated costs/times should be altered, restarting the lot-sizing step and carrying out another scheduling step to find a hopefully better plan. This try and error approach is likely to generate good solutions if the procedure used to generate the costs/times are reasonable, the current planning situation is comparable to past experience, and the human planner is highly competent. It is implemented in many of today's Advanced Planning Systems (e. g., SAP SCM-APO).

In environments which require a high capacity usage and a high service level like today's FMCG industry, applying a successive lot-sizing and scheduling approach is likely to lead to the aforementioned problems which can have a significant impact on the competitiveness and therefore the economical success of the enterprise. This close relationship between lot-sizing and scheduling makes it imperative that both decisions are made simultaneously in order to use the available capacity efficiently[14]. Therefore, for the remainder of this thesis the focus will be on integrated lot-sizing and scheduling methods.

3.4. Integrated Lot-Sizing and Scheduling

In integrated lot-sizing and scheduling, the generation of production lots and the scheduling on time windows of the available capacities is done in one step. This guarantees that the available capacities are not overloaded as well as preventing that the solution of the non-integrated lot-sizing step is limiting capacity utilization (and therefore optimality) or even feasibility. However, the drawback is that the interdependencies make finding of feasible or optimal plans more complicated.

Since even in successive lot-sizing and scheduling the scheduling step is known to be *NP*-hard[15], meaning the difficulty to solve it is exponentially rising with the problem size, and the integrated lot-sizing and scheduling being even more difficult, there were several approaches developed to cope with lot-sizing and scheduling for industrial size problems. For real applications with limited time for generating plans, it should still be considered that in some cases a successive approach may, with all its drawbacks, still get better results in a limited time than integrated approaches may in the same time. This can be of high relevance especially in situations where generated plans cannot be carried out, e. g. due to unexpected long machine-breakdowns, and a new plan has to be obtained in short time. Such problems are then treated in the so called reactive-planning[16] theory and are not in the scope of this thesis.

The integrated lot-sizing and scheduling problem has gained a lot of attention in research, especially in regard to sequence-dependent setups[17], and different solution procedure have been proposed in literature and industry, which will be discussed in the following.

3.4.1. Solution Procedures

Procedures for generating solutions for specific problems in Operational Research can be divided into methods that guarantee to find an optimal solution (though in practical application the time needed for executing it may be prohibitively large) and *Heuristics*. Heuristics in Operational Research are "'seeking' method[s], as [they] cannot guarantee to find anything"[18]. Heuristics are often problem-specific algorithms which do not follow a general structure. However, a number of general heuristic approaches have been developed which can be applied to solve a multitude of different problems.

When it comes to finding optimal solutions, usually a *Mathematical Programming* approach is chosen. However, one can find many connections between methods to find

[14] Cf. Clark et al. (2011)
[15] Cf. e. g. Allahverdi et al. (2008, p. 989)
[16] Cf. e. g. Schöpperl (2013)
[17] Cf. e. g. Zhu and Wilhelm (2006)
[18] Reeves (1995, p. 6)

optimal solutions and general heuristics. Mathematical Programming is based on a mathematical representation of the problem and general heuristic methods are often based on the same or similar mathematical problem representations. While optimizing approaches try to solve the model optimally, heuristics may aim at finding a sufficiently good solution, while requiring considerably less time than a true optimization. On the other hand, optimizing procedures often make use of heuristics to improve the time to find an optimal solution, like the relaxation that is used in the *Branch-And-Bound* algorithm. Also, it is possible to combine different basic heuristic approaches into a new heuristic, e. g. the *Fix-and-Optimize* heuristic as developed by Sahling (2009).

The basics of mathematical programming and some of the most commonly applied heuristic methods will be described briefly in the following.

Mathematical Programming

In Mathematical Programming, a mathemtical model is developed that reflects the relevant aspects of the real problem in an objective function and a number of constraints. This model can then be implemented into a solution software that tries to find values for the decision variables that meet all constraints while achieving an optimal or sufficiently good objective value[19].

The methods applied to solve the mathematical model depend on characteristics of the model. In *Linear Programming* (LP), advanced methods of the *Simplex-Algorithm* are applied. Basically the Simplex-Algorithm takes advantage of that it is known that the optimal solution of a linear model has to be on the crossing of two or more constraints, or in some cases that the optimal solutions lie on one constraint. Using modern hard- and software, solving even quite complex linear programs is possible in short time.

In *Mixed-Integer Linear Programming* (MILP), some of the variables are not linear but discrete. Since today, such models tend to be highly complicated in terms of solving, depending on the number and kind of non-linear variables. Therefore it is recommendable to keep the number of non-linear variables as low as possible. However, some logical constraints have to be non-linear. One common case are variables that reflect the changeover activity. In a specific period, a changeover happens or not, and by this either production of the respective product is allowed or not and the changeover costs or times have to be regarded or not. To reflect this in a mathematical model, a binary variable is used that takes the value 1 if the changeover happens or 0 if not, all other values are prohibited. To solve such mixed-integer programming problems, usually a Branch-and-Bound algorithm[20] is applied, which aims at systematically searching the solution space for the optimal values of those binary variables.

The algorithm contains two steps, the Branching and the Bounding. In the Branching part, the solution space of the problems binary variables is divided in disjunctive sub-problems. The binary variables of such a sub-problem are then fixed to 0 or 1. Then, the

[19] If only a sufficiently good solution is aimed for, this method can also be seen as a heuristic. However, while other heuristics usually cannot give information about how good a specific solution is, in this approach this is usually possible.

[20] Cf. Dakin (1965)

remaining sub-problem is solved, usually using LP-Relaxation, allowing the remaining binary variables to take on continuous values between 0 and 1 (transforming the MILP model into an LP Model for this step), or Lagrange-Relaxation. If the objective value of the solution of the relaxed problem (the "lower bound" since no solution which does not relax the binary restriction can be better) is worse (i. e. for a minimization problem "higher" or "lower" for a maximization problem) than the solution of the best found feasible solution so far, then it is certain that the optimal solution cannot be obtained using the current values for the fixed binary variables and this combination can be discarded. If it is better, by systematically adding or removing binary variables which should be fixed to a value, the set of relaxed variables is narrowed down until either all combinations have to be discarded or a feasible solution is found, which, if it is better than the best found feasible solution so far, is the new reference (the "upper bound").

If the algorithm is given enough time and at least one feasible solution exist, it can always find a feasible optimal solution. While searching the solution space systematically, for bigger sized problems the required time can be, depending on the structure of the problem and parameters for the algorithm, prohibitively large[21]. To improve the performance of the Branch-And-Bound algorithm, several methods have been developed to strengthen the lower bounds, the *Cutting Plane*[22] procedure, the *Branch&Cut* procedure and the *Cut&Branch* procedure, which add additional constraints in the Branch-And-Bound algorithm allowing quicker disposal of inferior solution space[23]. While reducing the time that is be required to solve a model to optimality, in many cases it still is prohibitively large. It is therefore common in Branch-and-Bound applications to define one or several stopping criteria, usually the execution time is limited and/or the algorithm is stopped when a pre-defined MIP Gap, being the relative difference of the current lower and upper bound, is reached.

Heuristics aim at generating good or potentially optimal solutions, but the generated plans are not guaranteed to be optimal or even close to optimality. Heuristics can be of very different kinds. In many cases, heuristics are very specific to a problem. However, most follow a basic concept. In the following, some of the most used basic concepts are described briefly. They can be modified or combined with other heuristics and/or mathematical modeling in many applications.

Decomposition Based Heuristics

Decomposition based heuristics divide the problem into several sub-problems and then solve the sub-problems independent of each other or successively by use of other heuristics or mathematical modeling. E. g., the customers might be differentiated into different groups, and in a first run only the top customers demands are considered, then in a second run the remaining customers demands with the remaining capacity. However, to ensure a feasible plan for the second run still exists, it might be necessary to include

[21] Cf. Domschke and Drexl (2007, pp. 133–138)
[22] Cf. Gomory (1958)
[23] Pochet and Wolsey (2006, pp. 101–107)

some penalty onto to high capacity usage in the first run. The two plans would then be combined into one plan for all orders.

The advantage is that by reducing the problem size the complexity is reduced, lowering the time required to find solutions, so that in a given time window good solutions may be found where a full model would perhaps be unable to find any feasible solution. The drawback is that, while perhaps the plans for each single decomposition step are optimal, the generated plan for the complete problem can be far from the true global optimum.

LP Relaxation

In LP Relaxation, a number of constraints of a mathematical model is "relaxed", that means some of the logical constraints may be violated. For example, changeover operations can happen or they do not happen, which in a mathematical model is reflected by binary variables. The used binary variables which reflect if a changeover happens ($= 1$) or happens not ($= 0$) can be relaxed to continuous variables which may get values between 0 and 1. This improves the solvability of the model substantially.

A common version is the Relax-And-Fix heuristic. A subset of the binary variables is relaxed and the model is solved. Then, those binary variables that meet some criteria, e. g. they were set to 0 or 1 even though other values would have been feasible for the relaxed model, are fixed to that value. Then, the set of variables that are relaxed is changed and a new planning run is started. This procedure is repeated by changing the set of relaxed variables or constraints and/or changing the criteria to fix a variable to a given value until the model is solvable without needing to relax any non-fixed binary variables anymore.

Here, by changing hard constraints to soft constraints, the solvability of the model can be drastically improved. While it is necessary to solve the model several times with different subsets of relaxed constraints, due to the NP-hardness of the basic model the time for executing the heuristic can be significantly reduced in comparison to the complete model. The drawback is, that it is not guaranteed that the fixed variables would have the same value in the optimal solution. Also, it can be that the fixing of variables leads to infeasibilities in later iterations, unless a procedure is introduced that "un-fixes" some binary variables.

Lagrange-Heuristics

In Lagrange-Heuristics, the solution is based on three steps: the *Lagrange-Relaxation*, the updating of the *Lagrange-Multipliers*, and the generation of a feasible plan[24]. By using optimal Lagrange-Multipliers the solution of the relaxed model should be close to the true feasible optimum, which can then be constructed from the infeasible solution within a few steps[25]. The three steps of one iteration of the Lagrange-Relaxation are:

1. In a first step, critical constraints are relaxed in that they may be violated. For example, it is allowed that the required capacity to carry out production in some

[24] Cf. e. g. Reeves (1995, pp. 243–304)
[25] Cf. e. g. Geoffrion (1974), Fisher (1985) or Beasley (1995)

period is bigger than the available capacity. To limit the amount of violation, the amount of violation is multiplied with the Lagrange-Multiplicator and added (for minimization problems) respectively subtracted (for maximization problems) to the objective function. The relaxed problem should be fast to solve, generating a plan that is close to the optimal plan, but may be infeasible.

2. In the second step (which does not strictly to have be carried out in each iteration), the infeasible plan is altered to make it feasible. This can be a critical step, since the procedure to change an infeasible plan to make it feasible can be difficult and can have a large impact on the objective value of the generated plan. This is also influenced by the parameters of the Lagrange-Multipliers used in the solving of the relaxed model, therefore in the next step the Lagrange-Multipliers are adjusted.

3. In the third step, the Lagrange-Multipliers are modified using *subgradient optimization*, which gives directions to change the Lagrange-Multipliers (increasing or decreasing them) based on current constraint violations to achieve improvements in the objective value[26].

Solving the relaxed model with the new Multipliers generates a new, usually infeasible plan, which can then be made feasible again. Ideally, the newly generated plan has fewer infeasibilities and is easier to be changed into a feasible plan with a better objective value of the feasible plan.

The procedure of defining and updating the Lagrange-Multipliers, solving the relaxed model and generating feasible plans can be carried out several times, usually until some stopping criteria (e. g., no more improvements on the objective value of the best generated feasible plan for n iterations) is reached.

Local Search and Meta-Heuristics

Local Search and Meta-Heuristics usually contain two steps. In a first step, one or several starting solutions are constructed. Depending on the underlying problem, the focus can be on generating feasible start solutions, but this is not strictly necessary. Some constraints can be relaxed and the violations are then penalized in the objective value of the generated solutions. Those solutions are the basic set of reference solutions.

The second step is the improvement of the found solutions. New solutions are generated based on the solutions in the reference set and a heuristic-specifc procedure and checked for feasibility and the goodness of this solution according to the objective value. If the new solutions are accepted, the set of reference solutions is updated and the next iteration of the improvement step is started.

The procedures usually end when a stopping criteria is met, e. g. if for n iterations no improvement for the best found objective value is found, no more new solutions can be generated or if a given time limit is reached.

A basic example are the problem specific Greedy Heuristics. Greedy Heuristics can be for constructing new solutions from an empty starting solution and/or to improve given

[26] Cf. e. g. Goffin (1977) or Sandi (1979)

solutions. Construction heuristics start with an empty solution space and iteratively add decisions based on some priority index. After a starting reference solution has been generated, in improvement a neighboring solution is defined, which is usually a solution that differs only in one decision and checks it for feasibility and the goodness of this solution according to the objective value. The solution is then chosen as the new reference if it is feasible and, according to the objective value, better than the previous one. If not, it is discarded and a different neighboring solution is evaluated. If a new reference solution is found, the procedure of generating neighbors and evaluating them is repeated until all neighbors of the reference solution are infeasible or worse than the reference solution. This can lead to the procedure getting stuck in a local minimum with no chance of finding the true optimum. Which local optimum is reached depends on the starting solution and the parameters of the heuristic. It is not possible to evaluate how good a found solution is, therefore the local minimum found may be far worse than the true global optimum. Several problem specific greedy heuristics have been developed for the dynamic lot-sizing problem, e. g. the Silver-Meal-Procedure[27].

Depending on the structure of the underlying problem, Greedy Heuristics can be sufficient, if no local optima exist that are significantly worse than the global optimum. Since this structure is not given for many problems, different ways to prevent getting stuck in a local minimum have been developed, usually as so-called *Meta-Heuristics*.

Meta-Heuristics are based on two concepts, *intensifying* and *diversification*. Intensifying is responsible for searching a promising part of the solution space, also called the *exploitation*. Diversification is responsible for extending the solution space, especially if no improvement of the best found objective value could be found for some time, also called *exploration*[28]. Those Meta-Heuristics can be differentiated into *Local Search based Algorithms* and *Population Based Algorithms*.

Local search based Meta-Heuristics, like e. g. *Simulated Annealing* (SA) or *Tabu-Search* (TS), are similar to the aforementioned Greedy Heuristic. In the SA algorithm introduced by Kirkpatrick et al. (1983)[29], after a neighbor solution is defined and evaluated, there is a probability that the solution is accepted as the new reference solution to allow for an escape out of a local minimum. The probability of accepting worse solutions is decreased over time of the execution of the algorithm so it converts to a greedy algorithm over time. The TS algorithm was introduced by Glover (1986) and is named after the implemented tabu list. In a very basic form of TS, starting from a reference solution a set of neighboring solutions is generated and those solutions which are on the tabu list are discarded. Then the best neighboring solution is selected as the new reference solution and added to the tabu list. Then, a new set of neighours is generated and the procedure repeated until some stopping criteria is met. In the procedure, old entries from the tabu list are eliminated from it, allowing them again to be chosen. Depending on the length of the tabu list, the reference solution can escape a local minimum. It is also possible to

[27]Cf. Silver and Meal (1973)
[28]Cf. Fink and Rothlauf (2006)
[29]See also Henderson et al. (2003)

alter the length of the tabu list over time, expanding it increases the likelyness of leaving a local minimum, decreasing it intensifies the search of the current (local) optimum.

Evolutionary Algorithms[30] like the *Genetic Algorithms*[31] (GA) belong to the population based algorithms, the population being a set of solutions. It is based on the principles of Charles Darwin's natural selection. New child solutions are generated by combining parts of two parent solutions (also called recombination) and can be randomly mutated. The new child solutions are then checked for their fitness value (usually based on feasibility and the objective value). The population for the next iteration (or generation) is then chosen based upon the fitness value of the solutions. The procedure continues until a stopping criterium is fulfilled, e. g. a maximum number of iterations or if for a given number of iterations no improvements regarding the objective value could be achieved. One main difficulty in using GA is to establish methods which make it likely to have child solutions with good fitness values with a reasonable likelihood, especially regarding the feasibility.

Another example would be the *Ant Colony Optimization* (ACO). It is based on the swarm intelligence of ants when providing food to select the shortest route possible. The basic concept here is that several ants choosing different ways, the one with the shortest way will be back first. Since it leaves pheromones on the way, therefore the shortest route has double the pheromone concentration early, making it for later ants starting likely to chose it due to their preference of following the pheromone marked paths based on the concentration. This leads to an exponential increase in the pheromone concentration in the shortest path. Applications for this type of Meta-Heuristic can be found e. g. in the *Vehicle Routing Problem* (VRP)[32]

Combined Heuristics

The aforementioned basic heuristic are not mutually exclusive, but can be combined to achieve better results and/or performance. For example, the *Fix-and-Optimize* heuristic is a relatively new heuristic which is a combination of the ideas underlying the Decomposition and the Relax-and-Fix heuristics and was developed by Sahling et al. (2009) for the multi-level capacitated lot-sizing problem. In it, the binary variables are divided into sub-sets, and the mathematical model is then solved to optimize one of those sub-sets, while keeping the other binary variables at an initial value. Then, the values determined for this sub-set are fixed, replacing any initial value, and the set of variables to be optimized is changed.

3.4.2. Mathematical Modeling

In literature, there has been a vast development on defining mathematical models for lot-sizing, many of which are integrating the sequencing. A current overview of models for dynamic lot sizing problems can be seen in Karimi et al. (2003), Jans and Degraeve

[30]For other algorithms belonging to the class of evolutionary algorithms, cf. e. g. Nissen (1994, p. 13)
[31]Cf. e. g. Nissen (1994) for a detailed description of the Genetic Algorithm procedure
[32]Cf. e. g. Yu et al. (2009)

(2008) or Buschkühl et al. (2010). In the following, a short description of a selection of these formulations is given.

CLSP

The *Capacitated Lot-Sizing Problem* (CLSP) is the extension of the *Single Item Uncapacitated Lot-Sizing Problem* (SIULSP) to reflect multiple products and capacity limitations. It belongs to the so-called *Big Bucket Models*, which are characterized by dividing the time horizon into macro periods in which several production lots for different products can be produced as long as the capacity is sufficient. The sequence of production orders within a macro period is not determined, therefore it is not an integrated lot-sizing and scheduling model. Due to the non-considering of the sequence, setup states are lost at the end of each macro period. Haase (1994) gives an extension to model the setup state at the start and end of each macro period to allow for multi-period setup carryover. Haase and Kimms (2000) extended the CLSP to the *Lot-Sizing and Scheduling-Problem with Sequence-Dependent Setup Costs and Times* (LSPSD) by defining a set of possible efficient sequences in a first step and then extending the CLSP to select one of the pre-defined sequences in each macro period, resulting in an integrated lot-sizing and scheduling model. Almada-Lobo et al. (2007) give two possible extensions of the CLSP which incorporate sequence dependent setups and setup carryovers without the need to pre-define possible sequences.

In practical application, when solving the CLSP in its standard formulation using mathematical modeling and the Branch-And-Bound algorithm[33], it faces the problem of having very weak lower bounds, causing relatively high run times as inferior parts of the solution space cannot be identified and discarded quickly. For this, several reformulations of the CLSP were developed which allow for stronger lower bounds but make it more challenging to adept the CLSP to reflect more complex problems[34]. The standard CLSP-model is as follows:

Indices:

$j \in J$	set of products
$m \in M$	set of macro periods

Data:

B	sufficiently large number
C_m	capacity of macro period m
d_{jm}	demand of product j at the end of macro period m
h_j	holding cost per period and per unit of product j
pt_j	processing time unit of product j
sc_j	cost for a changeover to product j

[33] Cf. Gelders et al. (1986)
[34] Cf. Barany et al. (1984)

Variables:

$I_{jm} \geq 0$	inventory of product j at the end of macro period m
$q_{jm} \geq 0$	quantity of product j to be produced in macro period m
$y_{jm} \in \{0; 1\}$	$= 1$, if product j is set up in macro period m, else 0

$$\min \sum_{j \in J} \sum_{m \in M} (sc_j \cdot y_{jm} + h_j \cdot I_{jm}) \tag{3.1}$$

subject to

$$I_{jm} = I_{j,m-1} + q_{jm} - d_{jm} \qquad \forall\, j \in J, m \in M \tag{3.2}$$

$$\sum_{j \in J} q_{jm} \cdot pt_j \leq C_m \qquad \forall\, m \in M \tag{3.3}$$

$$q_{jm} \leq B \cdot y_{jm} \qquad \forall\, j \in J, m \in M \tag{3.4}$$

DLSP

The *Discrete Lot Sizing and Scheduling Problem* (DLSP) was introduced by Fleischmann (1990) and aimed at reducing the required time to find an optimal production plan in comparison to the CLSP. In the DLSP, instead of macro periods which allow for n products to be produced each period, the DLSP uses micro periods which allow only one product to be produced per period. It has an all-or-nothing assumption, by that if a product is produced in a period, then all of the periods capacity is used for production. This is reasonable in environments with limited capacity, as due to the assumption of only one product per period capacity would be wasted otherwise. One drawback is that it can cause unnecessarily high inventories, depending on the relative length of the micro periods to the required time to produce the quantities for a demand element. The *Continuous Lot-Sizing and Scheduling Problem* (CSLP) is a variation of the DLSP which gives up the all-or-nothing assumption of the DLSP. Another aspect of the DLSP is that the setup state cannot be carried over idle periods.[35]

Fleischmann (1994) extended the DLSP to include sequence-dependent setup costs. Salomon et al. (1997) developed an approach for solving the DLSP with sequence-dependent set-up costs and set-up times by reformulating it to a *Travelling Salesman Problem with time windows*. The basic DLSP-model is as follows:

Indices:

$j \in J$	set of products
$t \in T$	set of micro periods

[35] Cf. e. g. Drexl and Haase (1995)

Data:

B	sufficiently large number
c_t	capacity of micro period t
d_{jt}	demand of product j at the end of micro period t
h_j	holding cost per micro period and per unit of product j
pt_j	processing time per unit of product j
q_j	quantity of product j that can be produced per micro period
sc_j	cost for a changeover to product j

Variables:

$I_{jt} \geq 0$	inventory of product j at the end of micro period t
$y_{jt} \in \{0; 1\}$	1, if product j is set up in macro period t, else 0

$$\min \sum_{j \in J} \sum_{t \in T} (sc_j \cdot \max(0, y_{jt} - y_{j,t-1}) + h_j \cdot I_{jt}) \tag{3.5}$$

subject to

$$I_{jt} = I_{j,t-1} + q_j \cdot y_{jt} - d_{jt} \qquad \forall\, j \in J, t \in T \tag{3.6}$$

$$\sum_{j \in J} y_{jt} \leq 1 \qquad \forall\, t \in T \tag{3.7}$$

PLSP

A further development of the DLSP and CLSP was introduced by Drexl and Haase (1995), the *Proportional Lot-Sizing and Scheduling Problem* (PLSP). Contrary to the DLSP or CSLP, it allows for producing two products in a micro period. It allows for at most one product changeover per micro period, which does not have to be at the beginning of a micro period. The PLSP has no all-or-nothing assumption and is advantageous compared to the CSLP, as it does not have to waste capacity of a micro period if a production lot does not use all of it but remaining capacity can be used for a changeover to a new product and to start production. Still, if production times for the demands are small in comparison to the length of a micro period, it can be that with two products a micro periods capacity cannot be exhausted, causing remaining capacity to be lost.

This leads to a major issue regarding the PLSP, but also the DLSP and the CSLP: there is no generally accepted procedure in literature to determine the length of the micro period, but it is usually assumed to be given by some organization depending charcteristic, e. g. the length of a shift. By selecting a large parameter for the micro periods length, the problem mentioned before is obvious. On the other hand, selecting a small micro period length other issues can arise. An obvious problem is that this results in a high number of micro periods, which can quickly make the problem prohibitively large to be solved using mathematical optimizing, as Drexl and Kimms (1997) point out, while mentioning that for common sense heuristics this is not true.

Another problem is that changeover activities have to be finished in a micro period, they cannot be split up between adjacent micro periods. This can result in wasted capacity or sub-optimal sequences if the remaining capacity after finishing a production lot in a micro period is not sufficient to complete the changeover to the next product of the optimal sequence. This problem arises at most if the changeover times are highly sequence-dependent. In some cases, it can even be that some changeover times would be so high in the (unknown) optimal solution that a complete micro periods capacity would not be sufficient, which would lead to sub-optimal results or even infeasibilities. While Haase (1994) gave an extension to the PLSP to reflect period overlapping changeover times, Suerie (2006) pointed out that Haase's formulation was mathematically incorrect and gave two variants of the PLSP which correctly model period overlapping setup times. However, their mathematical complexity is so high that only very small instances could be solved to optimality in acceptable time. The basic PLSP-model is as follows:

Indices:

$j \in J$	set of products
$t \in T$	set of micro periods

Data:

B	sufficiently large number
c_t	capacity of micro period t
d_{jt}	demand of product j at the end of micro period t
h_j	holding cost per period and per unit of product j
pt_j	processing time unit of product j
sc_j	cost for a changeover to product j

Variables:

$I_{jt} \geq 0$	inventory of product j at the end of micro period t
$x_{jt} \in \{0;1\}$	1, if a changeover to product j takes place in micro period t, else 0
$y_{jt} \in \{0;1\}$	1, if product j is set up at the end of micro period t, else 0

$$\min \sum_{j \in J} \sum_{t \in T} (sc_j \cdot x_{jt} + h_j \cdot I_{jt}) \tag{3.8}$$

subject to

$$I_{j,t-1} + q_{jt} - I_{jt} = d_{jt} \qquad \forall\, j \in J, t \in T \tag{3.9}$$

$$\sum_{j \in J} y_{jt} = 1 \qquad \forall\, t \in T \tag{3.10}$$

$$x_{jt} - y_{jt} + y_{j,t-1} \geq 0 \qquad \forall j \in J, t \in T \quad (3.11)$$

$$B \cdot y_{jt} + B \cdot y_{j,t-1} - q_{jt} \geq 0 \qquad \forall j \in J, t \in T \quad (3.12)$$

$$\sum_{j \in J} pt_j \cdot q_{jt} \leq c_t \qquad \forall t \in T \quad (3.13)$$

GLSP

The *General Lot-Sizing and Scheduling Problem* (GLSP) by Fleischmann and Meyr (1997) can be seen as a next evolutionary step of the aforementioned model formulations. It does not use micro periods of a fixed length, but instead uses a combination of macro periods and micro periods. Demand elements are linked to the macro period and each macro period consists of a given number of micro periods. To each micro period at most one product is assigned, the length of the micro period is determined by the assigned production quantity.

They also show that the GLSP can be transformed into the CLSP, DLSP, CSLP or PLSP by adding additional constraints restricting the time structure of the solution. They state that the solution quality improves from DLSP to GLSP, the CLSP being an exception since it cannot be compared due its lack of modeling the production sequence within a macro period. It is pointed out, that the GLSP with non-zero minimal lot sizes is a very complex combinatorial problem for which even the finding of a feasible solution is NP-complete, and therefore they present heuristic solution procedures. However, modeling of non-zero minimal lot sizes is only necessary if the triangle inequality is not true for changeover costs (the same for changeover times), i. e. that in regard to the objective function it would be preferable to change from product A to B to C without producing any positive quantity of B than to directly change from A to C. As will be seen later in this thesis, without the need of modeling minimal lot sizes, the GLSP is relatively efficiently solvable. The basic GLSP-model is as follows:

Indices:

$j, j' \in J$	set of products
$m \in M$	set of macro periods
$t \in T$	set of micro periods
$t \in T(m)$	set of micro periods that belong to macro period m

Data:

C_m	capacity of macro period m
d_{jm}	demand of product j at the end of macro period m
mls_j	minimum lot size of product j
pt_j	processing time per unit of product j
$sc_{j'j}$	cost for a changeover from product j' to product j

Variables:

$I_{jm} \geq 0$	inventory of product j at the end of macro period m
$q_{jt} \geq 0$	quantity of product j to be produced in micro period t
$x_{j'jt} \in \{0;1\}$	1, if a changeover from product j' to product j takes place in micro period t, else 0
$y_{jt} \in \{0;1\}$	1, if product j is set up in micro period t, else 0

$$\min \sum_{j \in J} \sum_{m \in M} h_j \cdot I_{jm} + \sum_{j' \in J} \sum_{j \in J} \sum_{t \in T} sc_{j'j} \cdot x_{j'jt} \tag{3.14}$$

subject to

$$I_{jm} = I_{j,m-1} + \sum_{t \in T(m)} q_{jt} - d_{jm} \qquad \forall\, j \in J, m \in M \tag{3.15}$$

$$\sum_{j \in J} \sum_{t \in T(m)} pt_j \cdot q_{jt} \leq C_m \qquad \forall\, m \in M \tag{3.16}$$

$$q_{jt} \leq \frac{C_m}{pt_j} \cdot y_{jt} \qquad \forall\, j \in J, t \in T \tag{3.17}$$

$$q_{jt} \geq mls \cdot (y_{jt} - y_{j,t-1}) \qquad \forall\, j \in J, t \in T \tag{3.18}$$

$$\sum_{j \in J} y_{jt} = 1 \qquad \forall\, t \in T \tag{3.19}$$

$$x_{j'jt} \geq y_{j',t-1} + y_{j't} - 1 \qquad \forall\, j' \in J, j \in J, t \in T \tag{3.20}$$

Note that the m in C_m is not defined in equation 3.17 as given by Fleischmann and Meyr (1997), therefore it should be corrected to:

$$q_{jt} \leq \frac{C_m}{pt_j} \cdot y_{jt} \qquad \forall\, j \in J, m \in M, t \in T(m) \tag{3.21}$$

Block Planning

The *Block Planning* approach is a relatively new concept in mathematical modeling, first developed by Günther et al. (2006). It is based on a common practice in industrial application also known under the name of *Production Wheel* (compare section 2.3). Similar to the GLSP, it has two time grids. The first time grid is a discrete representation of macro periods, to which the demand elements are assigned. Within the macro periods is a continuous time grid on which the production lots are scheduled. To each macro period, one *block*[36] is assigned which has to be finished at the end of the corresponding macro period at the latest and depending on the block planning being *rigid* or *flexible*, a block either cannot start before the beginning of the macro period (rigid) or it can

[36] A "block" is a sequence of production orders, while individual orders within a block may be skipped.

start at any time (flexible), providing it does not overlap with another block. In a block, production lots are then sequenced following a given *Natural Sequence*. While it is possible to skip lots in a block, saving the changeover cost/time for this lot, it is generally not possible to go back in sequence within one block. If any production is assigned to a block, a major changeover cost/time is applied.

Similar to the LSDSP, it is possible to define several sequences or blocks and choosing one block for each macro period if not only one specific natural sequence exists. The advantage of defining blocks in advance is that the complexity of the model is significantly reduced, providing the number of choosable blocks is kept low. Modeling sequence-dependent changeovers is however challenging.

Another issue can, especially in flexible block planning, be that inventories produced by production in a block are only added to the inventory at the end of the corresponding macro period. If out-of-pocket inventory holding costs are high this can lead to higher realized costs than expected by the plan. The basic Block Planning model as given by Günther et al. (2006) is as follows [37]:

<u>Sets:</u>

$j \in J$	set of products
$m \in M$	set of macro periods (note that this equals the number of blocks, as to each macro period one block is assigned)
$o \in O$	sequence of production orders over all blocks $(0 = 1, ...\|O\|)$
$o \in O(j, m)$	set of production orders in period (block) m which can produce product j

<u>Parameters:</u>

B	sufficiently large number
d_k	quantity of demand element k
h_j	holding cost per period and per unit of product j
L	length of a macro period
pt^o	variable processing time per unit of production order o
sc^o	changeover cost for production order o
st^f	capacity required for starting a block
st^o	capacity required for a changeover to any production order o

<u>Variables:</u>

$\alpha_o \geq 0$	start time of production order o (α_1 given)
$\delta_o \geq 0$	duration of production order o
$I_{jm} \geq 0$	inventory of product j at the end of macro period m (I_{j0} given)

[37]Note that most later applications of the block model planning forfeited the inventory balance for a direct assignment of production quantities to demand elements, as will be done in the models presented in chapter 4.

$x_o \geq 0$ size of production order o
$y_o \in \{0, 1\}$ 1, if production order o produces a positive quantity, else 0

$$\min \sum_{o \in O} sc_o \cdot y_o + \sum_{j \in J} \sum_{m \in M} h_j \cdot I_{jm} \tag{3.22}$$

subject to

$$x_o \leq B \cdot y_o \qquad \forall\, o \in O \tag{3.23}$$
$$\delta_o = st^o \cdot y_o + pt^o \cdot x_o \qquad \forall\, o \in O \tag{3.24}$$
$$\alpha_o \geq \alpha_{o-1} + \delta_{o-1} \qquad \forall\, o \in O \tag{3.25}$$
$$\alpha_{m \cdot J} + \delta_{m \cdot J} \leq L \cdot m - st^f \qquad \forall\, m \in M \tag{3.26}$$
$$I_{jm} = I_{j,m-1} + x_{o(j,m)} - d_{jm} \qquad \forall\, j \in J, m \in M \tag{3.27}$$

3.4.3. Simulation

Simulation in the context of integrated lot-sizing and scheduling can be used as a means to evaluate available production plans and as a supporting method in generating them. The advantage of simulation is that it is relatively easy to implement a model that is very close to the real production environment, for which a plan would be extremely difficult to find, while keeping the computing requirements for running the simulation at bay. The procedures which were treated before, while trying to find very good or optimal solutions, usually aim at finding solutions in reasonable time with a higher abstracted model of the real production environment than necessary for simulation. Simulation models can then be used to reflect the production environment more detailed and assess the generated plans on it.

E. g., Almeder et al. (2009) give a framework for operational decisions for supply chain networks using a combination of an optimization model and descrete-event simulation. They use a simplified LP model to determine values for parameters that are then given to the simulation, while the simulation gives a feedback regarding the cost parameters that are used in the LP model. This is especially helpful if the cost parameters are non-linear, which would require much more complex MILP models instead of a simple LP model. They show that in reasonable time the cost parameters used in the LP model converge with the costs in the simulation model.

Another possible application of simulation is in the context of *robust planning*. As usually methods to generate production plans are based on determinstic data, while the reality is usually stochastic, it is often necessary to ajust plans to account for differences between the deterministic data and the realized data, like different customer demands or processing times. Such adjustments usually imply costs associated with *system nervousness*, like for starting additional production lots or increasing production capacity by temporary staff[38]. While mathematical modeling is capable of integrating

[38]Cf. e. g. Tunc et al. (2013)

stochastic aspects, the practical applicability of such models is limited, as due to the high complexity only small instances are solvable in realistic time. Simulation can be used to simulate different realizations of such stochastic data and analyze the effect on the generated production plan, assessing for costs caused by the system nervousness.

Additionally, simulation can work as a proof for the correctness of the model, by testing if a production plan with the given parameters could be realized or if the procedure for generating the plan is faulty. Especially in real industry it is common to test new concepts in simulations first, before implementing them in a real production system, as the cost for a faulty implementation in the real production system can be very high.

3.5. Conclusions for Production Planning in the FMCG Industry

Considering the characteristics of the FMCG industry as given in chapter 2, it is recommendable to invest time and effort into the generation of high performance production plans. Therefore, an integrated approach of lot-sizing and scheduling is of advantage. However, several basic model formulations were developed in an evolving process. Nonetheless, and there is no general method for deciding which model formulation is most promising for further development in a more specific production environment and this was not subject to research so far. To close this research gap, in the next chapter the mathematical models that form the last steps of the development, the PLSP, GLSP and Block Planning, are extended to represent the integrated lot-sizing and scheduling problem as it is common in the FMCG industry. In practical applications, it may be necessary to apply heuristic methods as outlined above. However, the scope of this thesis is on comparison of mathematical models for the FMCG industry, since heuristics performance can vary highly depending on the chosen heuristic parameters, the application of mathematical programming to evaluate the performance of different model formulations seems reasonable. Also, it seems reasonable to assume that basic model formulations which are efficient in mathematical programming are likely to be efficient in heuristic approaches as well, though this cannot be said with certainty until further research evaluates the relative performance of heuristics based on the underlying model formulation.

4. Modeling of FMCG Industry Scenarios

As outlined in the previous chapter, the remainder of this thesis is on the usability of the PLSP, GLSP and Block Planning models in the FMCG industry under consideration of different changeover structures. For this, three scenarios are described in the following. They are derived from typical current production environments in the FMCG industry and mainly differ to each other by a specific changeover structure related characteristic. For each scenario, a brief description is given, followed by the extended model formulations to reflect the requirements of the scenario. In the following chapter, the result of numerical investigations are given for each model and scenario.

Since the scenarios shall mainly differ in the type of changeover structure, in the following aspects that should be identical to all of them are outlined:

- For all scenarios, a planning horizon of several weeks is considered. This is reasonable in the FMCG industry since the shelf life of products is often restricted, either by perishable products or because of short life cycles or inaccurate mid-term demand forecasts of the products, e. g. season-only products in food industry, which makes building up stocks to cover high demand periods which lie further in the future economically undesirable.

- Available capacity is generally a limiting factor which has to be addressed. While generally in the FMCG industry a high capacity utilization is important, for some industries with seasonal demand patterns and limited possibilities to built up stock it is possible that the capacity utilization cannot always be high. To reflect this, in the following chapter two capacity load scenarios will be defined for each industry scenario.

- Changeover activities require capacity time, reducing the capacity that is available for production activities. The required time differs depending on the changeover structure and will be addressed more in detail later. The GLSP and BP models restrict changeover times to be less than a macro periods length, which usually is not problematic as macro periods usually consider weeks and are much longer than any changeover time, at least in the FMCG industry. Therefore for modeling purpose it is assumed that changeover times are less than a macro period. However, the PLSP in its standard formulation requires changeover activities to finish in the same micro period as they start. While Haase gives a formulation with long changeover times, Suerie (2006) showed that this formulation was mathematically incorrect and produced suboptimal or infeasible solutions in some cases. While Sürie formulated two new formulations to cope with long changeover times, due to the high complexity of those formulations only very small instances without sequence dependent changeover times could be solved optimally. However, there is no general accepted way of defining the length of micro periods for the PLSP. This will be further addressed to in chapter 5. For this thesis it is therefore assumed that all changeovers times are less than a micro periods length to enable finding of feasible solutions with the PLSP without period overlapping changeover times[1].

- Production cannot carry on over the weekend and all setup states are lost at the end of each week due to cleaning or maintenance activities. This is often true in FMCG industry, but not always. However, it is relatively easy to allow for changeover carryover in the models, since it only requires removing or slight alteration of given restrictions. It does not increase or decrease the complexity of the models to an extent where a substantial difference in relative performance is to be expected.

- As described in chapter 2, in regard to the lot-sizing and scheduling problem the production process can often be reduced to a single-level production. Also, it is assumed that only one capacity needs to be considered, i. e. there are not several parallel machines to which the same production lots can be assigned. This is often true in industry when products are assigned to a specific resource, e. g. due to technical requirements of those products.

- It is assumed that several products exist with demands occurring at the weeks ends. Without limiting the general validity of the models, it is assumed that demand elements are given in required capacity required to produce them instead of physical units with different production coefficients. Those demands have to be covered completely in time, stockouts are not allowed. While the forbidding of stockouts is quite common for many industries to achieve a high service-level, the attaching of the demands to the end of the weeks is not without problem. In industry with very short shelf life, e. g. fresh food industry, it may be infeasible to produce a product which is ordered for Thursday already in the previous week. For such environments it may be necessary to model a finer time grid to attach the demand elements to, e. g. days. While this is possible for all models, e. g. Bilgen and Günther (2010) did this for the BP model, it leads to significant more complex models. The exception here is the PLSP, which is using a fine time grid anyway - however as will be seen later, this leads to considerable problems regarding the required solving time. For such environments with daily demands and more than just a few days of planning horizon, it may be necessary to employ heuristic approaches in the foreseeable future. This thesis focuses on the modeling and solving of MILP models and therefore this is out of scope for the remainder of this thesis.

[1]However, it is not possible to use part of a period for production and start a new changeover to complete it in the next micro period which can in some instances of the numerical investigation lead to infeasibilities if capacities are very tight, as it may cause a waste of capacity. As will be seen in the numerical investigation, even on low load scenarios the PLSP has trouble finding good or even feasible solutions for all but the smallest instances, even though the made assumption here allows for a less complex model.

Many of those aspects can be represented using the basic model formulations of the PLSP, GLSP and Block Planning as given in subsection 3.4.2. The major aspects that need to be incorporated into modified models are:

- the makespan needs to be defined as the new objective to be minimized and constraints are required to determine the makespan,
- the loss of the setup state at the end of the macro periods needs to be modeled, as for the PLSP and the GLSP a setup carry-over is generally allowed. In the Block Planning model, it is not included, however in the so called *flexible block planning*, like the basic model given in chapter 3, production blocks can overlap macro period boundaries. Block planning models that disallow this are called *rigid block planning*,
- the different sequence related aspects of the scenarios need to be implemented. Those will be treated more in detail in the respective sections.

4.1. Full Flexibility Scenario

The most general case in regard to production sequences is the *Full Flexibility Scenario* (FF Scenario). This scenario is able to reflect many different production environments and can be regarded as a general case able to reflect all kinds of setup structures (e. g., for natural sequences or limited changeovers this can be done approximately by setting the changeover times for prohibited changeovers prohibitively high). However, due to the high complexity, finding efficient solutions for this scenario using mathematical modeling is very difficult. Therefore more problem specific modeling approaches should, if possible, be considered, especially in realistic problem sizes, as will be seen in section 5.1.

4.1.1. Scenario Description

The FF Scenario is not defined by a case study with a specific changeover structure, but by the absence of any specific natural sequences and that all changeovers are allowed. Therefore, it can be regarded as a general reference case. The changeover times are sequence dependent and do not necessarily follow any pattern. This is illustrated in figure 4.1 for a small case of 3 Products. For example, st_{12} denotes the changeover time from product 1 to product 2. They are independent of each other, as long as the triangle inequality holds (e. g., $st_{12} + st_{23} \geq st_{13}$). Therefore there is no need to model minimum lot sizes. This is the more common case in the FMCG industry, however if minimum lot sizes have to be modeled, restrictions to include those are presented in the Limited Changeover scenario.

4.1.2. Model Development

In this section possible extensions to the three basic models PLSP, GLSP and Block Planning will be given for the FF Scenario as PLSP-FF, GLSP-FF respectively BP-FF. The FF Scenario models require modifications to represent sequence dependent

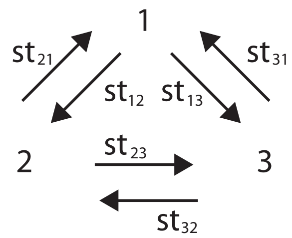

Figure 4.1.: *Changeover structure of the Full Flexibility Scenario.*

changeovers between products, the makespan as the objective to be minimized and the loss of setup state between macro periods require. The loss of setup state over macro period boundaries can be represented by an initial setup state that is defined by including an additional dummy product (Product 0).

PLSP-FF

The major aspects that need to be incorporated into the PLSP to reflect the properties of this scenario are: first, the sequence dependent setup times need to be modeled, this is done here by adapting the formulation given by Haase[2] for sequence dependent setup costs. Second, a new objective function in minimizing the makespan MS is introduced and the constraints necessary to determine it. Third, a set of macro periods is introduced for two reasons: first, an additional dummy product is introduced to reflect the initial setup state at the beginning of each macro period caused by the loss of setup state between different macro periods; second, as demand elements only occur at the macro periods boundaries, the inventory balance needs to be tracked only for those, not for each single micro period.

Sets:

$m \in M$	set of macro periods
$t \in T$	set of micro periods
$t \in T^f$	subset of micro periods that are the first of any macro period
$t \in T(m)$	subset of micro periods belonging to macro period m
$j \in J$	set of products ($j = 1, \dots, J$)
$j' \in J'$	set of products including dummy product ($j' = 0, \dots, J$)

Data:

α_m	start time of macro period m
c	capacity of each micro period
d_{jm}	demand of product j at end of macro period m
$st_{j'j}$	capacity required for a changeover from product j' to j
$\overline{\overline{st}}$	maximum of all setup times

[2]Haase (1994)

Variables:

$I_{jm} \geq 0$	inventory of product j at the end of macro period m
MS	makespan
$q_{jt} \geq 0$	quantity of product j produced in micro period t
$ST_{jt} \geq 0$	capacity required for changeover to product j in period t
$u_m \in \{0;1\}$	1, if macro period m is used, else 0
$y_{jt} \in \{0;1\}$	1, if product j is set up in micro period t, else 0

The objective to be minimized is the makespan.

$$\min MS \tag{4.1}$$

subject to

The makespan is given by the finishing time of the last utilized periods production.

$$MS \geq u_m \cdot \alpha_m + \sum_{j \in J} \sum_{t \in T(m)} (q_{jt} + ST_{jt}) \qquad \forall\, m \in M \tag{4.2}$$

The inventory needs to be tracked only at the macro periods boundaries. To ensure demand satisfaction, it may not become negative. Production in the macro and stock carryover from the previous macro period add to the inventory of a product, while demand is the only decreasing factor.

$$I_{j,m} = I_{j,m-1} + \sum_{t \in T(m)} q_{jt} - d_{jm} \qquad \forall\, j \in J, m \in M \quad (I_{j0} \text{ given}) \tag{4.3}$$

Production in a micro period can only occur if the required setup state exists at the start of the micro period or at the end of the micro period, as in the PLSP only two setup states are allowed per micro period. Due to the loss of the setup state at the macro period boundaries, in the first micro periods of each macro period production can only take place if a changeover to the respective product takes place and therefore the setup state exists at the end of that micro period.

$$\begin{aligned} q_{jt} &\leq c \cdot (y_{jt} + y_{j,t-1}) && \forall\, j \in J, t \in T \setminus T^f \quad (y_{j0} = 0) \\ q_{jt} &\leq c \cdot y_{jt} && \forall\, j \in J, t \in T^f \end{aligned} \tag{4.4}$$

The time required for a sequence dependent changeover activity in a specific micro period can be determined using the following restriction adapted from Haase (1994). It's advantage is that it does not require binary variables with two indices, one for the preceding and one for the subsequent setup state, but only one index for the setup state at the end of a micro period, which greatly saves on model complexity, but identifies the required changeover time directly based on the setup state at the end of the current and the end of the previous micro period.

$$ST_{jt} \geq \overline{\overline{st}} \cdot y_{jt} - \sum_{j' \in J'} (\overline{\overline{st}} - st_{j'j}) \cdot y_{j',t-1} \qquad \forall\, j \in J, t \in T \setminus T^f \quad (4.5)$$

$$ST_{jt} \geq st_{0j} \cdot y_{jt} \qquad \forall\, j \in J, t \in T^f$$

At the end of each micro period, at most one setup state can exist.

$$\sum_{j \in J} y_{jt} \leq 1 \qquad \forall\, t \in T \quad (4.6)$$

Finally, to determine the makespan it is necessary to determine which micro periods have to be utilized. This is the case if the period was used for any production or changeover activity. Additionally, the following restriction ensures that no micro periods capacity is overloaded by changeover or production activities.

$$u_m \cdot \sum_{t \in T(m)} c \geq \sum_{j \in J} \sum_{t \in T(m)} (q_{jt} + ST_{jt}) \qquad \forall\, m \in M \quad (4.7)$$

GLSP-FF

To extend the basic GLSP to reflect sequence dependent setup times, the modification given by Meyr (2000) for a GLSP with sequence dependent setup times for minimizing costs is used as a basis. It is modified by introducing the makespan as the objective and the required constraints to determine it. Also, a modification is necessary as the setup state cannot be carried over between different macro periods. A dummy product is introduced that reflects the initial setup state.

As the GLSP needs a predetermined number of micro periods per macro period, to ensure this does not prohibit the finding of the optimal solution, the number of micro periods per macro periods is set to be the number of products plus one (for the dummy product), allowing theoretically to produce every single product in each macro period.

<u>Sets:</u>

$m \in M$	set of macro periods
$t \in T$	set of micro periods over all macro periods
$t \in T(m)$	set of micro periods belong to macro period t
$t \in T^f$	set of micro periods that are the first of a macro period
$h, j \in J$	set of products ($J = \{1, \ldots, J\}$)
$j' \in J'$	set of products including dummy product ($J' = \{0, \ldots, J\}$)

Data:

α_m	start time of macro period m
C	capacity of each macro period
d_{jm}	demand of product j at end of macro period m
$st_{j'j}$	capacity required to changeover from product j' to j

Variables:

$I_{jm} \geq 0$	inventory of product j at the end of macro period m
MS	makespan
$q_{jt} \geq 0$	quantity of product j produced in micro period t
$u_m \in \{0;1\}$	1, if macro period m is used, else 0
$y_{j'jt} \in \{0;1\}$	1, if product j is set up in micro period t by a changeover from product j', else 0

The objective to be minimized is the makespan.

$$\min MS \tag{4.8}$$

subject to

The makespan is given by the finish of the last utilized macro periods production.

$$MS \geq \alpha_m \cdot u_m + \sum_{t \in T(m)} \Big(\sum_{j' \in J'} \sum_{j \in J} st_{j'j} \cdot y_{j'jt} + \sum_{j \in J} q_{jt}\Big) \qquad \forall\, m \in M \tag{4.9}$$

The inventory needs to be tracked only at the macro periods boundaries. To ensure demand satisfaction, it may not become negative. Production in the macro and stock carryover from the previous macro period add to the inventory of a product, while demand is the only decreasing factor. The inventory balance ensures demand satisfaction.

$$I_{jm} = I_{j,m-1} + \sum_{t \in T(m)} q_{jt} - d_{jm} \qquad \forall\, j \in J, m \in M \quad (I_{j0} \text{ given}) \tag{4.10}$$

In each macro period, a limited capacity is available that may not be overused by changeover or production activities.

$$\sum_{j \in J} \sum_{t \in T(m)} \Big(q_{jt} + \sum_{j' \in J'} st_{j'j} \cdot y_{j'jt}\Big) \leq C \qquad \forall\, m \in M \tag{4.11}$$

Production can only take place if a changeover to the respective product was executed in the corresponding macro period. A setup carryover from the precious macro period is not possible.

$$q_{jt} \leq \sum_{j' \in J'} y_{j'jt} \cdot C \qquad \forall\, t \in T \quad (4.12)$$

In each micro period but the first of a macro period, a specific predecessor-successor changeover can be executed only if in the previous micro period a changeover to the predecessor was executed, the exception being the initial setup state. Note that a changeover from the initial setup state is only possible in the first micro period of each macro period.
Only one product can be produced per micro period.

$$\sum_{h \in J} y_{jht} \leq \sum_{j' \in J'} y_{j',j,t-1} \qquad \forall\, j \in J, t \in T \setminus T^f \quad (4.13)$$

$$\sum_{j \in J} y_{0jt} = 0 \qquad \forall\, t \in T \setminus T^f$$

Macro periods are utilized if a changeover from the initial setup state to any other is executed. At the same time, the following restriction ensures that it is impossible to start a changeover from any other product in the first micro period of any macro period, enforcing the first changeover to use the initial setup state as the predecessor.

$$\sum_{j \in J} y_{0jt} + 2 \cdot \sum_{j' \in J} \sum_{j \in J} y_{j'jt} \leq u_m \qquad \forall\, m \in M, t \in T(m) \cap T^f \quad (4.14)$$

BP-FF

The Block Planning model needs some more important changes to enable it to account for sequence dependent setup times. The major characteristic of the continuous time line can be held onto, but unlike the other Block Planning based models, here the macro periods are explicitly modeled. For each macro period, here it is necessary to define as many blocks as products plus one for the initial setup state for each macro period. The products are assigned to the blocks and depending on this assignment, the sequence dependent setup times are determined. In this special case each block consists only of at most one product, therefore eliminating the need to use binary variables for each block to model its usage to be able to assign one or more products to it. Modeling the macro periods explicitly allows for using binary variables to reflect usage of any block in a macro period, which leads to a smaller number of binary variables than modeling usage of each block.

Other changes are, like for the other models, the introduction of the makespan as the objective to be minimized, constraints to determine it and the loss of setup state at the macro period boundaries, as well as the assignment of production quantities to demand elements instead of an inventory balance..

<u>Sets:</u>

$m \in M$	set of macro periods
$j \in J$	set of products
$j' \in J'$	set of products $(0, \ldots, \lvert J' \rvert)$ including a dummy product 0
$k \in K$	set of demand elements
$i \in I$	set of blocks over all macro periods $(1, \ldots, \lvert I \rvert)$
$i \in I(m)$	set of blocks belonging to macro period m
$i \in I(k)$	set of blocks including those that can be used to satisfy demand element k
$i \in I^f$	subset of blocks that are the first of each macro period

<u>Data:</u>

$\underline{\alpha_m}$, $\overline{\alpha_m}$	end time respectively start time of the macro period m
d_k	quantity of demand element k
$j(k)$	product, to which demand element k refers
$st_{j'j}$	capacity required for a changeover from product j' to product j

<u>Variables:</u>

$\alpha_i \geq 0$	start time of block i
δ_i	duration of block i (not strictly necessary)
MS	makespan
$q_{ik} \geq 0$	quantity of demand element k that is satisfied from production in block i
$u_m \in \{0; 1\}$	1, if macro period m is used, else 0
$y_{ij'j} \in \{0; 1\}$	1, if a changeover from product j' to product j is performed at the beginning of block i, else 0

The objective to be minimized is the makespan.

$$\min MS \tag{4.15}$$

subject to

The makespan is the start time of the last block plus its duration for changeover and processing.

$$MS \geq \alpha_i + \delta_i \qquad \forall\, i \in I \tag{4.16}$$

Instead of using an inventory balance, in the block planning model production quantities are assigned directly to demand elements to satisfy them.

$$\sum_{i \in I(k)} q_{ik} = d_k \qquad \forall\, k \in K \quad (4.17)$$

Production in a block for a products demand element can only take place, if a changeover to the respective product did take place in the respective block.

$$q_{ik} \leq d_k \sum_{j' \in J'} y_{ij'j(k)} \qquad \forall\, k \in K, i \in I(k) \quad (4.18)$$

The following restriction ensures the initial setup state exists only at the beginning of each macro period (note that, if changeover times from this initial setup state are generally high, this restriction may become obsolete).

$$\sum_{j \in J} y_{i0j} = 0 \qquad \forall\, i \in I \setminus I^f \quad (4.19)$$

The first changeover of each used macro period has the initial setup state as its predecessor.

$$\sum_{j \in J} y_{i0j} = u_m \qquad \forall\, m \in M, i \in I^f \cap I(m) \quad (4.20)$$

The following constraint models the changeover structure.

$$\sum_{j \in J} y_{ihj} \leq \sum_{j' \in J'} y_{i-1,j',h} \qquad \forall\, h \in J, i \in I \setminus I^f \quad (4.21)$$

To each block only one product's setup state can be assigned. Also, if to any block of a macro period any product's setup state is assigned, the macro period is used for production.

$$\sum_{j' \in J'} \sum_{j \in J} y_{ij'j} = u_m \qquad \forall\, m \in M, i \in I(m) \quad (4.22)$$

The duration of a block is the time required for the changeover in this block and for production in this block. Note that this restriction is, like in all block planning models, only for convenience.

$$\delta_i = \sum_{j' \in J'} \sum_{j \in J} y_{ij'j} \cdot st_{j'j} + \sum_{k \in K} q_{ik} \qquad \forall\, i \in I \quad (4.23)$$

A block may not start before its earliest start date, that is the start of the corresponding macro period in a rigid block planning model. Note that for macro periods where no production takes place the start date of the block may be set to any low value, so they do not affect the makespan.

$$\alpha_i \geq \underline{\alpha_m} \cdot u_m \qquad \forall\, m \in M, i \in I(m) \quad (4.24)$$

Production blocks may not overlap to prevent overloading of the available capacity.

$$\alpha_i \geq \alpha_{i-1} + \delta_{i-1} \qquad \forall\, i \in I(\alpha_0 = 0) \quad (4.25)$$

Finally, each block has to end at the respective macro periods length at the latest, to ensure that no stockouts happen.

$$\alpha_i + \delta_i \leq \overline{\alpha_m} \qquad \forall\, m \in M, i \in I(m) \quad (4.26)$$

4.2. Limited Changeover Scenario

A common special type of sequence restriction is the *Limited Changeover* (LC), as e. g. in chemicals production or specialty steel production. Small lots of highly customized products are produced. These products can be grouped according to a dominating setup requirements into product families (or *clusters*). The changeover is limited, in that it is not possible or economically undesirable to do a changeover directly from certain families to other specific families. This implies the need to produce at least a minimal quantity of a product family if it is set up for.

4.2.1. Scenario Description

The LC Scenario is derived from a case study featuring a single-stage production of special chemical products. Small quantities of products are produced based on specifications of the customer. Beside the quality of the products, short lead times are of paramount importance. Due to a high number of potential product variants, production on stock for forecasted demands is not reasonable, making this is a typical case for *make-to-order production*. The primary objective is to satisfy the customer demands on time and a secondary objective to be flexible for accepting new orders, while holding costs or changeover costs are by comparison negligible in production planning.

In this case study, the dominant setup requirement is the required temperature range inside the reactor. Products can be grouped into families according to their required processing temperature in the reactor. The changeover between families is limited, e. g. switching from a high temperature directly to a low temperature is not allowed without producing some product (either being one specific product or several products of the respective family) in the middle temperature range, since otherwise a high amount of

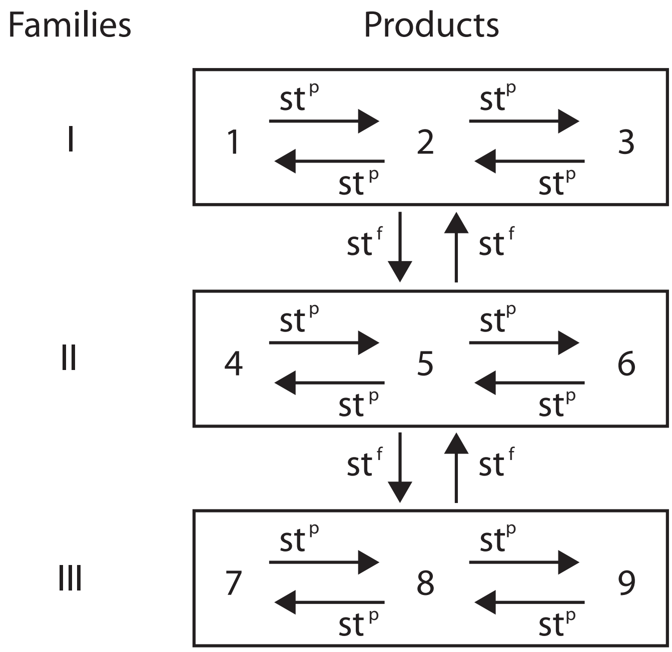

Figure 4.2.: *Changeover sequence structure of the Limited Changeover Scenario.*

waste would be produced in the heating process and production time would be wasted. This implies the need to model minimum lot sizes for the families. Changeovers between products of different families are restricted, allowing changeovers only between products of adjacent families or of the identical family. Since this limited changeover shall be the main aspect of this scenario, it is assumed that changeovers between products of the same family require a standard minor setup time and changeovers between products of different families require an additional major setup time. This is illustrated in figure 4.2 for a case of 3 families with 3 products per family: e. g., a direct changeover from product1 is possible to products 2 or 3 for a changeover time of st^p, to products 3, 4 or 5 for a total changeover time of $st^p + st^f$, and impossible to products 7, 8 or 9.

At the end of weeks, the resources are cleaned and maintenance activities can be performed. It is assumed that the reactor can be prepared to allow a changeover to any first product family at the beginning of each week, requiring the standard family changeover time.

4.2.2. Model Development

In this section possible extensions to the three basic models PLSP, GLSP and Block Planning will be given as models PLSP-LC, GLSP-LC respectively BP-LC to be able to reflect the special requirements of the LC Scenario such as the existence of product families, the limitation of changeovers between product families and the minimum lot sizes for families if a family is set up. Also, the makespan as the new objective function is introduced and the loss of setup state between macro period boundaries is implemented.

PLSP-LC (Limited changeover)

In the basic PLSP, no product families exist, therefore those have to be added here. There is also no limitation on the set of possible changeovers, which is added to the model here as well. Then, macro periods are required to model the loss of setup state between macro period boundaries and to limit the inventory balance to consider those micro periods that can have demand elements assigned to. Also, the makespan as a new objective function has to be formulated as well as constraints to determine it.

Sets:

$m \in M$	set of macro periods
$t \in T$	set of micro periods
$t \in T^f$	set of micro periods that are the first of any macro period
$t \in T(m)$	set of micro periods belonging to macro period m
$f \in F$	set of families
$\varphi \in P(f)$	set of families which can be predecessor of family f
$j \in J$	set of products
$j \in J(f)$	set of products belonging to family f

Data:

α_t	start time of micro period t
c	capacity of each micro period
d_{jm}	demand of product j at the end of macro period m
mls	minimum family lot size
st^f	capacity required for a changeover to any family
st^p	capacity required for a changeover to any product

Variables:

$I_{jm} \geq 0$	inventory of product j at the end of macro period m
MS	makespan
$q_{jt} \geq 0$	quantity of product j produced in micro period t
$u_t \in \{0;1\}$	1, if micro period t is used, else 0
$X_{ft} \in \{0;1\}$	1, if a changeover to family f takes place in micro period t, else 0
$x_{jt} \in \{0;1\}$	1, if a changeover to product j takes place in micro period t, else 0
$Y_{ft} \in \{0;1\}$	1, if family f is set up at the end of micro period t, else 0
$y_{jt} \in \{0;1\}$	1, if product j is set up at the end of micro period t, else 0

The objective to be minimized is the makespan.

$$\min MS \tag{4.27}$$

subject to

The makespan is given by the finish of the last utilized periods production.

$$MS \geq u_t \cdot \alpha_t + \sum_{j \in J} (q_{jt} + st^p x_{jt}) + \sum_{f \in F} st^f X_{ft} \qquad \forall\, t \in T \quad (4.28)$$

The inventory needs to be tracked only at the macro periods boundaries. To ensure demand satisfaction, it may not become negative. Production in the macro and stock carryover from the previous macro period add to the inventory of a product, while demand is the only decreasing factor.

$$I_{jm} = I_{j,m-1} + \sum_{t \in T(m)} q_{jt} - d_{jm} \qquad \forall\, j \in J, m \in M \quad (I_{j0} \text{ given}) \quad (4.29)$$

Production in a micro period can only occur if the required setup state exists at the start of the micro period or at the end of the micro period. Due to the loss of the setup state at the macro period boundaries, in the first micro periods of each macro period production can only take place if a changeover to the respective product takes place and therefore the setup state exists at the end of that micro period.

$$q_{jt} \leq c \cdot (y_{jt} + y_{j,t-1}) \qquad \forall\, t \in T \setminus T^f, j \in J \quad (4.30)$$
$$q_{jt} \leq c \cdot y_{jt} \qquad \forall\, t \in T^f, j \in J$$

At most one product setup state can exist at the end of each micro period.

$$\sum_{j \in J} y_{jt} \leq 1 \qquad \forall\, t \in T \quad (4.31)$$

Also, at most one family setup state can exist at the end of each micro period.

$$\sum_{f \in F} Y_{ft} \leq 1 \qquad \forall\, t \in T \quad (4.32)$$

A product setup state can only exist at the end of a micro period if a changeover to it has been executed or the setup state existed at the end of the previous micro period. For the first micro period of each macro period, the setup state cannot be carried over from the previous micro period.

$$x_{jt} - y_{jt} + y_{j,t-1} \geq 0 \qquad \forall\, t \in T \setminus T^f, j \in J \quad (4.33)$$
$$x_{jt} - y_{jt} \geq 0 \qquad \forall\, t \in T^f, j \in J$$

The same is true for the family setup state.

$$X_{ft} - Y_{ft} + Y_{f,t-1} \geq 0 \qquad \forall\, t \in T \setminus T^f, f \in F \quad (4.34)$$
$$X_{ft} - Y_{ft} \geq 0 \qquad \forall\, t \in T^f, f \in F$$

Here, product and family setup state are linked together.

$$\sum_{j \in J(f)} y_{jt} \leq Y_{ft} \qquad \forall\, f \in F, t \in T \quad (4.35)$$

The limited changeover between the product families needs to be considered.

$$Y_{ft} \leq \sum_{\varphi \in P(f)} Y_{\varphi,t-1} \qquad \forall\, f \in F, t \in T \setminus T^f \quad (4.36)$$

The capacity limitation of each period has to be considered, and if capacity is used the micro period is considered as being utilized for determination of the makespan.

$$c \cdot u_t \geq \sum_{j \in J} (q_{jt} + st^p x_{jt}) + \sum_{f \in F} st^f X_{ft} \qquad \forall\, t \in T \quad (4.37)$$

Finally, the minimum lot sizes for a family are modeled here.

$$\sum_{t \in T(m)} \sum_{j \in J(f)} q_{jt} \geq \sum_{t \in T(m)} X_{ft} \cdot mls \qquad \forall\, f \in F, m \in M \quad (4.38)$$

GLSP-LC (Limited changeover)

As for the other models, the basic GLSP requires modifications to account for the aspects of this scenario. Product families are not represented in the basic GLSP and need to be added, together with the constraints restricting the changeover possibilities. The makespan needs to be introduced as the objective functions with constraints to determine it. Lastly, the loss of setup state between macro period boundaries has to be considered.

Sets:

$m \in M$	set of macro periods
$t \in T$	set of micro periods

$t \in T^f$	set of micro periods that are the first of any macro period
$t \in T(m)$	set of micro periods belonging to macro period m
$f \in F$	set of families
$\varphi \in P(f)$	set of families which can be predecessor of family f
$j \in J$	set of products
$j \in J(f)$	set of products belonging to family f

Data:

α_m	start time of macro period m
C	capacity of each macro period
d_{jm}	demand of product j at the end of macro period m
mls	minimum family lot size
st^f	capacity required for a changeover to any family
st^p	capacity required for a changeover to any product

Variables:

$I_{jm} \geq 0$	inventory of product j at the end of macro period m
MS	makespan
$q_{jt} \geq 0$	quantity of product j produced in micro period t
$u_m \in \{0;1\}$	1, if micro period m is used, else 0
$X_{ft} \in \{0;1\}$	1, if a changeover to family f takes place in micro period t, else 0
$Y_{ft} \in \{0;1\}$	1, if family f is set up in micro period t, else 0
$y_{jt} \in \{0;1\}$	1, if product j is set up in of micro period t, else 0

The objective to be minimized is the makespan.

$$\min MS \tag{4.39}$$

subject to

The makespan is given by the finish of the last utilized periods production.

$$MS \geq u_m \alpha_m + \sum_{t \in T(m)} (\sum_{f \in F} X_{ft} st^f + \sum_{j \in J} (y_{jt} st^p + q_{jt})) \qquad \forall\, m \in M \tag{4.40}$$

The inventory needs to be tracked only at the macro periods boundaries. To ensure demand satisfaction, it may not become negative. Production in the macro and stock carryover from the previous macro period add to the inventory of a product, while demand is the only decreasing factor. The inventory balance ensures demand satisfaction.

$$I_{j,m} = I_{j,m-1} + \sum_{t \in T(m)} q_{jt} - d_{jm} \qquad \forall\, j \in J, m \in M \quad (I_{j0} \text{ given}) \tag{4.41}$$

In each macro period, a limited capacity is available that may not be overused by changeover or production activities.

$$\sum_{j \in J} \sum_{t \in T(m)} (q_{jt} + st^p y_{jt}) + \sum_{f \in F} \sum_{t \in T(m)} st^f X_{ft} \leq C \qquad \forall\, m \in M \quad (4.42)$$

A setup to a product in a micro period is required to enable production of the respective product in this micro period.

$$q_{jt} \leq C \cdot y_{jt} \qquad \forall\, j \in J, t \in T \quad (4.43)$$

Only one product can be produced per micro period.

$$\sum_{j \in J} y_t \leq 1 \qquad \forall\, t \in T \quad (4.44)$$

Only one product family can be assigned to a micro period. At the same time, this constraint identifies which macro periods are utilized.

$$\sum_{f \in F} Y_{ft} = u_m \qquad \forall\, m \in M, t \in T(m) \quad (4.45)$$

A family setup state can be carried over from the previous micro period if no macro period boundary causes loss of the setup state, or can result from a changeover to this family in the respective micro period.

$$Y_{ft} \leq Y_{f,t-1} + X_{ft} \qquad \forall\, f \in F, t \in T \setminus T^f \quad (4.46)$$
$$Y_{ft} \leq X_{ft} \qquad \forall\, f \in F, t \in T^f$$

A setup state for a product can only exist, if the family setup state matches that products family. Note that this also ensures that only one product can be produced per micro period like equation 4.44, however implementing both restrictions led to a better performance of the GLSP-LC model.

$$\sum_{j \in J(f)} y_{jt} \leq Y_{ft} \qquad \forall\, f \in F, t \in T \quad (4.47)$$

The following constraint ensures that only changeovers between families that are allowed take place.

$$X_{ft} \leq \sum_{\varphi \in P(f)} Y_{\varphi,t-1} \qquad \forall\, f \in F, t \in T \setminus T^f \quad (4.48)$$

Finally, the minimum lot sizes for a family are modeled here.

$$\sum_{t \in T(m)} \sum_{j \in J(f)} q_{jt} \geq \sum_{t \in T(m)} X_{ft} \cdot mls \qquad \forall f \in F, m \in M \quad (4.49)$$

Block Planning-LC (Limited changeover)

Due to the necessity of modeling minimum lot sizes here, it can be that a positive inventory is held for some products at the end of the planning horizon which needs to be included in the model. Also, the makespan, product families and the limited changeover need to be included, as well as the assignment of production quantities to demand elements instead of an inventory balance. Note that one block needs to be defined for each family and each macro period.

Sets:

$k \in K$	set of demand elements
$i \in I$	set of blocks over all macro periods
$k \in K(i)$	set of demand elements which may be satisfied from production in block i
$i \in I^f$	subset of blocks that are the first of any macro period
$i \in I(k)$	set of blocks which can be used to satisfy demand element k
$f \in F$	set of families
$\varphi \in P(f)$	set of families which can be predecessor of family f
$j \in J$	set of products
$j \in J(f)$	set of products belonging to family f

Data:

$\underline{\alpha_i}$,$\overline{\alpha_i}$	start respectively end time of the macro period of block i
d_k	quantity of demand element k
$j(k)$	product, to which demand element k refers
mls	minimum family lot size
st^f	capacity required for a changeover to any family
st^p	capacity required for a changeover to any product

Variables:

$\alpha_i \geq 0$	start time for block i
$\delta_i \geq 0$	duration of block i
$e_{ij} \geq 0$	inventory of product j at the end of the planning horizon built up by production in block i
MS	makespan
$q_{ik} \geq 0$	quantity of demand element k satisfied by production in block i
$Y'_i \in \{0; 1\}$	1, if all blocks until block i are used, else 0

$y_{ij} \in \{0;1\}$	1, if product j is set up in block i, else 0
$z_{fi} \in \{0;1\}$	1, if family f is assigned to block i, else 0

The objective to be minimized is the makespan.

$$\min MS \tag{4.50}$$

subject to

The makespan is the start time of the last block plus it's duration for changeover and processing.

$$MS \geq \alpha_i + \delta_i \qquad \forall\, i \in I \tag{4.51}$$

Instead of using an inventory balance, in the block planning model production quantities are assigned directly to demand elements to satisfy them.

$$\sum_{i \in I(k)} q_{ik} = d_k \qquad \forall\, k \in K \tag{4.52}$$

Production in a block for a products demand element can only take place, if a changeover to the respective product did take place in the respective block.

$$q_{ik} \leq d_k y_{i,j(k)} \qquad \forall\, i \in I, k \in K \tag{4.53}$$

A setup state for a product can only exist, if the family setup state matches that products family.

$$\sum_{j \in J(f)} y_{ij} \leq z_{fi} \cdot |J(f)| \qquad \forall\, i \in I, f \in F \tag{4.54}$$

The following constraints gives information about which blocks are within the makespan.

$$\sum_{f \in F} z_{fi} \leq Y_i' \qquad \forall\, i \in I \tag{4.55}$$

$$Y_i' \leq Y_{i-1}' \qquad \forall\, i \in I(Y_0' = 1) \tag{4.56}$$

The duration of a block is the time required for the changeover in this block and for production in this block. Note that this restriction is, like in all block planning models, only for convenience.

$$\delta_i = \sum_{f \in F} st^f z_{fi} + \sum_{j \in J} (st^p \cdot y_{if} + e_{ij}) + \sum_{k \in K(i)} q_{ik} \qquad \forall\, i \in I \quad (4.57)$$

A block may not start before the start time of the corresponding macro period due to the loss of the setup state at the macro period boundaries. Blocks that are outside the production horizon are not considered so they do not affect the makespan.

$$\alpha_i \geq \underline{\alpha_i} Y_i' \qquad \forall\, i \in I \quad (4.58)$$

Blocks may not overlap.

$$\alpha_i \geq \alpha_{i-1} + \delta_{i-1} \qquad \forall\, i \in I(\alpha_0 = 0) \quad (4.59)$$

The end time of a block may not exceed the end of the corresponding macro period.

$$\alpha_i + \delta_i \leq \overline{\alpha_i} \qquad \forall\, i \in I \quad (4.60)$$

The following constraint ensures that only changeovers between families that are allowed take place.

$$z_{fi} \leq \sum_{\varphi \in P(f)} z_{\varphi, i-1} \qquad \forall\, i \in I \setminus I', f \in F \quad (4.61)$$

Finally, the minimum lot sizes for a family are modeled here.

$$\sum_{k \in K(i)} x_{ik} + \sum_{j \in J} e_{ij} \geq \sum_{f \in F} z_{fi} \cdot mls \qquad \forall\, i \in I \quad (4.62)$$

4.3. Natural Sequence Scenario

The *Natural Sequence Scenario* (NS Scenario) will explore an environment typical for example in beverage production. The product portfolio can be grouped into product families which resemble in the major production characteristic in regard to setup requirements. Changeovers within such a family require only a minor changeover time as long as they follow a given natural sequence[3].

[3] In literature, the following of a pre-defined production sequence is often described using the term *"Production Wheel"*.

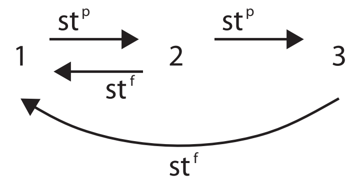

Figure 4.3.: *Changeover sequence structure of the Natural Sequence Scenario.*

4.3.1. Scenario Description

This scenario is derived from the beverage industry. Like in many other real applications, the capacity of the production line is limited by one expensive bottleneck resource (e. g. in beverage industry the combined bottle molding and filling machine). Often, the production lines are independent of each other due to technical specifications (e. g. the basic type of bottle that can be filled). Therefore, each line produces only one product group and can be planned independently of the others.

The product portfolio for each line can be furthermore grouped into product families, determined by the necessity of a major setup activity when changing between product families (e. g. changing the bottle size). In each product family, sequence dependent setups can be observed, however following a certain natural sequence. This sequence should be followed, for reversing the sequence is unattractive due to high setup activities (e. g. producing light-tasting variants before stronger variants, and diet variants before standard variants). By following this sequence, only a standard setup activity (e. g. a standard cleaning procedure) is required as long as the product family is the same. This changeover structure is illustrated in figure 4.3 for a small case of 3 products. This is also true if a product in sequence is skipped.

Since at the end of weeks extensive cleaning operations are necessary, setup states are lost at the end of macro periods. Unless the shelf-life of the products is very short, demand elements are usually on a weekly time scale. Therefore, the families can be regarded as being in a natural sequence as well for modeling - changing the sequence of families of the models result is possible if required without effect on any variables.

4.3.2. Model Development

In this section three possible extensions to the three basic models PLSP, GLSP and Block Planning will be given as models PLSP-NS, GLSP-NS respectively BP-NS to be able to reflect the special requirements of the Natural Sequence Scenario, the product families, the natural sequence within families, the loss of setup state between macro period boundaries and the makespan as the new objective function.

PLSP-NS

The PLSP has to be extended to be able to reflect changeover times within a family or between families and additionally the natural sequence needs to be enforced. Note that for the following notation the setup times need to be shorter than the length of any micro period. Also, macro periods are introduced to reflect the loss of setup states and to limit the necessity of inventory balance to those micro periods where demands can occur. Finally, the new objective of makespan minimization requires a couple of modifications.

Sets:

$m \in M$	set of macro periods
$t \in T$	set of micro periods
$t \in T^f$	subset of micro periods that are the first of any macro period
$t \in T(m)$	set of micro periods belonging to macro period m
$f \in F$	set of families
$j \in J$	set of products in the natural sequence, 1 being the first, $\lvert J \rvert$ the last
$j \in J(f)$	set of products belonging to family f

Data:

α_t	start time of micro period t
c	capacity of each micro period
d_{jm}	demand of product j at the end of macro period m
st^f	capacity required for a changeover to any family
st^p	capacity required for a changeover to any product

Variables:

$I_{jm} \geq 0$	inventory of product j at the end of macro period m
MS	makespan
$q_{st} \geq 0$	units produced using the setup state s in micro period t
$u_t \in \{0;1\}$	1, if micro period t is used, else 0
$X_{ft} \in \{0;1\}$	1, if a changeover to family f takes place in micro period t, else 0
$x_{jt} \in \{0;1\}$	1, if a changeover to product j takes place in micro period t, else 0
$Y_{ft} \in \{0;1\}$	1, if family f is set up at the end of micro period t, else 0
$y_{jt} \in \{0;1\}$	1, if product j is set up at the end of micro period t, else 0

The objective function to be minimized is the makespan.

$$\min MS \tag{4.63}$$

subject to

The makespan equals the last used periods start time plus the required changeover and processing time of that period.

$$MS \geq u_t \cdot \alpha_t + \sum_{j \in J}(q_{jt} + st^p x_{jt}) + \sum_{f \in F} st^f X_{ft} \qquad \forall\, t \in T \tag{4.64}$$

The inventory balance has been adapted to reflect that demands only occur at those micro periods that mark the end of any macro period.

$$I_{j,m} = I_{j,m-1} + \sum_{t \in T(m)} q_{jt} - d_{jm} \qquad \forall\, j \in J, m \in M \quad (I_{j0} \text{ given}) \tag{4.65}$$

The following constraint links production to the setup state while considering the loss of the setup state at the end of each macro period.

$$q_{jt} \leq c(y_{jt} + y_{j,t-1}) \qquad \forall\, t \in T \setminus T^f, j \in J \tag{4.66}$$

$$q_{jt} \leq c \cdot y_{jt} \qquad \forall\, t \in T^f, j \in J$$

It needs to be ensured that the capacity of a period is not overloaded by production and changeover activities, and that the capacity is unavailable if the period shall not be used at all.

$$c \cdot u_t \geq \sum_{j \in J} (q_{jt} + st^p x_{jt}) + \sum_{f \in F} st^f X_{ft} \qquad \forall\, t \in T \tag{4.67}$$

At most one product setup state can exist at the end of a period, however allowing micro periods to have no product setup state, which is relevant at the macro period boundaries.

$$\sum_{j \in J} y_{jt} \leq 1 \qquad \forall\, t \in T \tag{4.68}$$

At most one family setup state can exist at the end of a period, however allowing micro periods to have no family setup state, which is relevant at the macro period boundaries.

$$\sum_{f \in F} Y_{ft} \leq 1 \qquad \forall\, t \in T \tag{4.69}$$

The following constraint links product changeover and product setup states, modified to reflect the loss of the product setup state at the end of macro period boundaries.

$$x_{jt} - y_{jt} + y_{j,t-1} \geq 0 \qquad \forall\, t \in T \setminus T^f, j \in J \tag{4.70}$$

$$x_{jt} - y_{jt} \geq 0 \qquad \forall\, t \in T^f, j \in J$$

The following constraint links family changeover and family setup states, modified to reflect the loss of the family setup state at the end of macro period boundaries.

$$X_{ft} - Y_{ft} + Y_{f,t-1} \geq 0 \qquad \forall\, t \in T \setminus T^f, f \in F \quad (4.71)$$
$$X_{ft} - Y_{ft} \geq 0 \qquad \forall\, t \in T^f, f \in F$$

The link between product setup state and family setup state is introduced in the following constraint.

$$\sum_{j \in J(f)} y_{jt} \leq Y_{ft} \qquad \forall\, f \in F, t \in T \quad (4.72)$$

To enforce the natural sequence, the following constraint is added, forbidding a step back in the natural sequence unless a family changeover activity is performed, which would start a new sequence. Note this restriction only exists for micro periods that are not first of any macro period – for those, no previous setup state exists and it is assumed that the sequence can start with any setup state.

$$x_{jt} \leq 1 - \sum_{\substack{i \in J(f) \\ i > j}} y_{i,t-1} + X^f_{ft} \qquad \forall\, f \in F, j \in J(f), t \in T \setminus T^f \quad (4.73)$$

GLSP-NS (natural sequence)

Each macro period can produce the products of several product families. When a family is set up, all products within this family can be produced, but only in a given sequence. However, it is possible to skip products of a product family. The (small) changeover in the same family from one product to a later one in sequence is constant, as is the (big) changeover between product families.Due to the given sequence of products within a family and the implicit sequence of the families, the assignment of products to micro periods by using variables as made in the standard GLSP is not necessary. Each macro period is split into one micro period per product (for all families), so each micro period can be fixed to one specific product.

Other changes to the basic model are the restrictions used to reflect the family setups and family setup states, the introduction of the makespan and the modeling of the loss of setup states at the end of macro periods.

Sets:

$j \in J$	set of products
$m \in M$	set of macro periods
$f \in F$	set of product families
$t \in T$	set of micro periods
$t \in T^f$	subset of micro periods that are the first of any macro period
$t \in T(f)$	set of micro periods belonging to family f

$t \in T(f, m)$	set of micro periods belonging to family f and macro period m
$t \in T(j, m)$	set of micro periods that can produce product j and belong to macro period m
$t \in T(m)$	set of micro periods belonging to macro period m

Data:

C	capacity of each macro period
d_{jm}	demand of product j at the end of macro period m
st^f	capacity required for a changeover to any family
st^p	capacity required for a changeover to any product

Variables:

$I_{jm} \geq 0$	inventory of product j at the end of macro period m
MS	makespan
$q_t \geq 0$	quantity produced in micro period t
$u_m \in \{0; 1\}$	1, if macro period m is used, else 0
$X_{ft} \in \{0; 1\}$	1, if a changeover to family f takes place in micro period t, else 0
$Y_{ft} \in \{0; 1\}$	1, if micro period t is set up to family f, else 0
$y_t \in \{0; 1\}$	1, if micro period t is set up to the corresponding product, else 0

The objective to be minimized is the makespan.

$$\min MS \tag{4.74}$$

subject to

The makespan is defined be the finish time of the last used macro periods production.

$$MS \geq u_m \cdot \alpha_m + \sum_{t \in T(m)} (st \cdot y_t + \sum_{f \in F} X_{ft} SF + q_t) \qquad \forall\, m \in M \tag{4.75}$$

The inventory needs to be tracked only at the macro periods boundaries. To ensure demand satisfaction, it may not become negative. Production in the macro and stock carryover from the previous macro period add to the inventory of a product, while demand is the only decreasing factor. The inventory balance ensures demand satisfaction.

$$I_{j,m} = I_{j,m-1} + \sum_{t \in T(j,m)} q_t - d_{jm} \qquad \forall\, j \in J, m \in M \quad (I_{j0} \text{ given}) \tag{4.76}$$

In each macro period, a limited capacity is available that may not be overused by changeover or production activities.

$$\sum_{t \in T(j,m)} (q_t + st^p \cdot y_t) + \sum_{f \in F} \sum_{t \in T(m)} X_{ft} \cdot st^f \leq C \qquad \forall\, m \in M \quad (4.77)$$

A setup to a product in a micro period is required to enable production of the respective product in this micro period.

$$q_t \leq C \cdot y_t \qquad \forall\, t \in T \quad (4.78)$$

The following constraint links the product setup state to the family setup state.

$$y_t \leq Y_{ft} \qquad \forall\, f \in F, t \in T(f) \quad (4.79)$$

The family setup state can be carried over from the previous period, unless a macro period boundary prevents the setup carry over, or set anew.

$$\begin{aligned} Y_{ft} &\leq Y_{f,t-1} + X_{ft} && \forall\, f \in F, t \in T \setminus T^f \quad (4.80) \\ Y_{ft} &\leq X_{ft} && \forall\, f \in F, t \in T^f \end{aligned}$$

Only one family setup state can exist in each micro period.

$$\sum_{f \in F} Y_{ft} \leq 1 \qquad \forall\, t \in T \quad (4.81)$$

Finally, a macro period is used if one or more families are assigned to micro periods of that macro period, as this is only required if production takes place.

$$u_m \cdot |J| \geq \sum_{f \in F} \sum_{t \in T(f,m)} Y_{ft} \qquad \forall\, m \in M \quad (4.82)$$

BP-NS

The BP-NS model is extended for reflecting product families, the makespan and restrictions to determine it, the loss of setup state at the end of macro periods and the direct assignment of production quantities to demand elements. To each macro period, one block per family is assigned, with each of them starting earliest at the beginning of the macro period and ending at the latest at the macro periods end. Since it is one optional block for each family and demand elements are assigned to the end of macro periods, modeling a dynamic assignment of families to the blocks is not necessary here.

Sets:

$i \in I$	set of blocks over all macro periods
$j \in J$	set of products
$k \in K$	set of demand elements
$i \in I(k)$	set of blocks which can be used to satisfy demand element k (preceding and matching the family of the product)
$j \in J(i)$	set of products belonging to the family of block i

Data:

$\underline{\alpha_i}, \overline{\alpha_i}$	start respectively end time of the macro period of block i
d_k	quantity of demand element k
$j(k)$	product to which demand element k refers
st^f	capacity required for a changeover to any family
st^p	capacity required for a changeover to any product

Variables:

$\alpha_i \geq 0$	start time for block i
$\delta_i \geq 0$	duration of block i
MS	makespan
$q_{ik} \geq 0$	quantity of demand element k satisfied by production in block i
$Y_i \in \{0; 1\}$	1, if block i is used, else 0
$y_{ij} \in \{0; 1\}$	1, if product j is set up in block i, else 0

The objective to be minimized is the makespan.

$$\min MS \tag{4.83}$$

subject to

The makespan is the start time of the last block plus its duration for changeover and processing.

$$MS \geq \alpha_i + \delta_i \qquad \forall\, i \in I \tag{4.84}$$

Instead of using an inventory balance, in the block planning model production quantities are assigned directly to demand elements to satisfy them.

$$\sum_{i \in I(k)} q_{jk} \geq d_k \qquad \forall\, k \in K \tag{4.85}$$

Production blocks may not overlap to prevent overloading of the available capacity.

$$\alpha_i \geq \alpha_{i-1} + \delta_{i-1} \qquad \forall\, i \in I (\alpha_0 = 0) \quad (4.86)$$

Production in a block for a products demand element can only take place, if a changeover to the respective product did take place in the respective block.

$$q_{ik} \leq d_k y_{i,j(k)} \qquad \forall\, i \in I, k \in K \quad (4.87)$$

A block may not start before the start time of the corresponding macro period due to the loss of the setup state at the macro period boundaries. Blocks that are outside the production horizon are not considered so they do not affect the makespan.

$$\alpha_i \geq \underline{\alpha_i} Y_i \qquad \forall\, i \in I \quad (4.88)$$

The end time of a block may not exceed the end of the corresponding macro period.

$$\alpha_i + \delta_i \leq \overline{\alpha_i} \qquad \forall\, i \in I \quad (4.89)$$

The duration of a block is the time required for the changeover in this block and for production in this block. Note that this restriction is, like in all block planning models, only for convenience.

$$\delta_i = st^f \cdot Y_i + \sum_{j \in J} st^p \cdot y_{ij} + \sum_{k \in K} q_{ik} \qquad \forall\, i \in I \quad (4.90)$$

Finally, the following constraint ensures that changeover (and therefore production) activities can only occur in blocks that are within the production horizon.

$$\sum_{j \in J(i)} y_{ij} \leq Y_i \cdot |J(i)| \qquad \forall\, i \in I \quad (4.91)$$

5. Numerical Evaluation

In this chapter, the model formulations developed in the previous chapter shall be evaluated using numerical tests. For this, a general experimental design is outlined, followed by subchapters for each scenario (Full Flexibility, Limited Changeover and Natural Sequence) which extend this design where necessary and present the results of the adopted models.

The main research question which should be answered by these numerical tests is, how different changeover structures in integrated lot sizing and scheduling can be modeled using the three given basic model formulations. For this, three subordinate questions are to be answered:

- How efficient are those model formulations in regard to generating an economically advantageous production schedule and in regard to computational effort to determine these schedules?
- How strong is the influence of the problem size, e. g. the number of products, on the economical and computational performance of the given model formulations?
- How strong is the influence of the capacity load on the economical and computational performance of the given model formulations?

5.1. Experimental Design

To answer these questions, three problem sizes Small (S), Medium (M) and Large (L) are defined which give general parameters for all three changeover structure scenarios and differ mainly by having different number of products to be planned (4/9/25 products). While in industry the number of product variants that are to be produced on a single line often exceed these numbers by far, in such cases it is common that very similar product variants are merged for production planning purposes into product groups with a pre-optimized sequence within each of this product groups. The planning horizon is assumed to be four weeks long.

Two capacity load scenarios will be given, reflecting a total capacity requirement by production and changeover activities to be 70% (90%) for the low (high) capacity load scenario. From industrial experience it is known that commonly five to twenty percent of production time is consumed by changeover activities. It seems reasonable that usually more products mean more changeover activities and therefore a higher capacity consumption by changeovers. Therefore, for the three problem sizes S/M/L, total capacity consumption by changeover activities is assumed to be 5/8/15% of available production time (20% for an additional scenario in section 5.4). As the number of products increases

Table 5.1.: *Basic scenario parameters.*

Problem Size	S		M		L	
Planning Horizon (weeks)	4		4		4	
Products	4		9		25	
Average Changeover Time Per Product (CU)	2.5		1.778		1.2	
Capacity Load	Low	High	Low	High	Low	High
Total Capacity Usage (%)	70	90	70	90	70	90
Capacity Usage by Changeovers (%)	5	5	8	8	15	15

relatively more than the capacity consumption by changeover between the scenarios, the average changeover time between products is smaller for scenarios with more products, which matches the reasonable assumption that more products mean a higher similarity and that if changeovers are more frequent the used technology is more focused on reducing changeover times. Under the assumption that on average half of the product portfolio is produced per week, the average changeover time (ACT) per product is then for example for scenario S:

$$ACT = \frac{100\% \cdot 0.05\,[\text{total changeover requirements}]}{2\,[\text{average number of changeovers}]} = 2.5\%$$

of available capacity per week. With a normalized capacity of 100 capacity units (CU) per week, this leads to 2.5 CU as average setup time per product for this problem size. The complete values for all problem sizes are given in table 5.1.

To generate demand elements, a product is selected randomly, the demand element size (in capacity units needed for production) are drawn from the uniform distribution d in $[0.5 \cdot D, 1.5 \cdot D]$, with D being the total available capacity over all weeks multiplied with the capacity usage for production (without changeovers) and then divided by the number of product/week combinations, and the demand element is randomly assigned to one of the weeks. If the generated demand element belongs to an in advance randomly determined half of the products[1] and the demand element would be assigned to the first week, it is assumed to be fulfilled from starting inventory and therefore removed. If the created demand element would cause an exceeding of the effective capacity load limit, it is resized to the maximum size where it does not exceed the capacity load limit[2]. This procedure of generating demand elements is repeated until the effective capacity load limit until the end each of the weeks is reached.

[1] In the Limited Changeover Scenario and the Natural Sequence Scenario, not single products are randomly selected but whole product families.

[2] For example, the effective capacity load limit until the end of the second macro period in the S scenario with low capacity load is $2\,[\text{number of macro period}] \cdot (70\,[\text{Total capacity usage}] - 5\,[\text{capacity usage by changeovers}]) = 130\,[CU]$. Therefore, the demand elements assigned to the first two weeks should not exceed 130 CU. How those are distributed between those two weeks does not matter here.

For each changeover structure scenario, problem size and capacity load, ten sets of demand elements and, where needed, changeover structure parameters are generated and solved using the respective model formulations[3].

A specific issue with the PLSP models is the length of the micro periods, which has to be determined in advance. In literature, there is no generally accepted way to determine those, therefore a practical approach is chosen here, setting the micro period length to half of a standard shift time of 8 hours, leading to 4 hours or 5 CU. By this, it is ensured that changeover times never exceed the length of one micro period[4], while having a moderate number of micro periods (80 for 4 weeks of 100 CU each).

For the numerical investigation, for each combination of scenario, problem size and capacity load, ten instances of demand elements and, where applicable, changeover times have been generated and solved using OPL Studio 6.3 with CPLEX 11. Computers with an AMD Athlon Dual Core Processor 4050e (2,1 GHz) and 2 GB RAM, using Windows XP were used. In OPL, a MIP Gap of 1% were tolerated, otherwise the standard configuration was used. The optimization run times were limited to one hour (3600 seconds).

In each of the following subchapters, for each scenario the required additional procedures and the numerical results are given, structured as follows:

- The procedures for generating the changeover parameters are outlined.
- Then, a table showing the performance indicators of the models for the different problem sizes by averaging the values for the 10 instances of different demand values and, where applicable, setup structure parameters, is given. The percentage of instances in which a feasible solution was found is presented, as well as the percentage of found and proven optimal solutions (a feasible solution that is proven to be less than 1% worse than the true optimum is sufficient for being considered optimal in this evaluation) and for instances solved to optimality the average run time ("n/A" if no optimal solutions are found). For those instances that were not solved to optimality, the average MIP Gap between the best found solution and the corresponding non-integer solution is given ("n/A" if no non-optimal feasible solutions were found). Finally, to compare the solution quality, for instances where all models found at least one feasible solution, the average increase over the best found objective value is given as the performance indicator "Avg. derivation from best (%)" (taking the value of "n/A" if for no instance at least one feasible solution was found by each model).
- To support analyzing of the performance of individual models, in specific problem size/capacity load combinations graphical representations of the results are given if they are of specific relevance. A complete set of graphical representations for all scenario/size/capacity load combinations as well as tables with the detailed results

[3] The procedure for determining the changeover time parameter depends on the scenario and is therefore described in the respective subhcapters.

[4] In the procedures for determining the changeovers described in the following subchapters, the changeover time never exceeds 5 CU.

Table 5.2.: Performance indicators of the Full Flexibility Scenario.

	Problem Size	S		M		L	
Model	Capacity Load	70%	90%	70%	90%	70%	90%
PLSP-FF	Optimal Solutions Found (%)	40	10	0	0	0	0
	∅ Optimal Solution Time (s)	2830	64	n/A	n/A	n/A	n/A
	Feasible Solution Found (%)	100	100	100	100	100	100
	∅ MIP Gap (%)	2.0	2.3	12.2	7.3	33.3	23.4
	∅ Derivation From Best (%)	<0.1	<0.1	0.6	1.1	22.2	n/A
GLSP-FF	Optimal Solutions Found (%)	100	100	80	100	0	0
	∅ Optimal Solution Time (s)	<1	<1	35.8	291.4	n/A	n/A
	Feasible Solution Found (%)	100	100	100	100	70	30
	∅ MIP Gap (%)	n/A	n/A	1.3	n/A	10.0	11.1
	∅ Derivation From Best (%)	<0.1	0.1	<0.1	<0.1	<0.1	n/A
BP-FF	Optimal Solutions Found (%)	100	100	100	100	0	0
	∅ Optimal Solution Time (s)	1	1	89	89	n/A	n/A
	Feasible Solution Found (%)	100	100	100	100	90	30
	∅ MIP Gap (%)	n/A	n/A	n/A	n/A	8.8	5.4
	∅ Derivation From Best (%)	0	0	0	<0.1	1.5	n/A

for all instances is given in the appendix A. Also, an explanation on the design of the figures is given in the appendix.

- The major aspects of the numerical results are outlined and an interpretation of the numerical results is given for reach problem size for both capacity loads.
- Finally, a summary is given and the effect of the problem size on the models performance is discussed.

5.2. Full Flexibility Scenario

For the Full Flexibility Scenario, product changeover times between different products and from an initial setup state are generated by drawing values from the uniform distribution d in [0.5;1.5]. Then, triangle inequality is ensured by lowering direct changeovers from a product to another product to the minimum of all potential ways of changeover, if such exist. Finally, the resulting values are normalized to an average value of 1 and then multiplied by the average changeover time per product as given in table 5.1.

The results are summarized in table 5.2 by averaging the values for 10 instances of different demand values and changeover times. In the following, for each scenario size a detailed comparison of the results of the three model formulations shall be given for the Full Flexibility Scenario. At the end of this subchapter, a summary of the results for the Full Flexibility Scenario is given.

Small problem size (4 products)

As can be easily seen, the BP-FF and GLSP-FF had no difficulties for the small problem size with low capacity load, finding optimal solutions within a neglectable run time of 1 to 2 seconds. The PLSP-FF model encountered problems in finding an optimal solution in most of the instances, and even in cases where it found an optimal solution the run time needed to find this solution was on average 2830 seconds. As can be seen in the detailed results shown in the appendix, for no instance the run time was less than 1950 seconds (cf. appendix A). Even for instances where all models found an optimal solution, the PLSP-FF models objective value is often higher, e. g. in instance 10, than for the other two models. However, the difference is minor and might not be caused by a problem of the model formulation itself, but by an effect of the MIP Gap[5]. This will not be true for other scenario, problem size and capacity load combinations and be discussed at that point more in detail.

For the small problem size, the capacity load only notably effects the PLSP-FFs solution time, which interestingly improves with the higher capacity load. It is likely that this effect is caused by a smaller number of possible feasible combinations, so the Branch-and-Bound algorithm can cut of parts of the solution space more quickly.

Medium problem size (9 products)

In the medium problem size, the PLSP-FF still was able to find feasible solutions in all instances, however with a significant MIP Gap in most times. When comparing the BP-FF and the GLSP-FF model, while both models almost always found an optimal solution, there is no clear advantage of one model over the other that can be observed. The GLSP-FF found in the low capacity load instances optimal solutions in shorter time than the BP-FF, though in two instances it did was not able to find an optimal solution in two instances within the time limit of 3600 seconds[6]. In the detailed results as given in the appendix it can be seen that in the majority of instances it found an optimal solution in slightly shorter time than the BP-FF. On the other hand, the BP-FF always found an optimal solution for each instance in 325 seconds in the low capacity load instances respectively 346 seconds in the high capacity load instances at most. In

[5]For explanation, let's assume a true optimal (MIP Gap = 0%) value would be 297. With a MIP Gap threshold of 1%, the solution process might be stopped as soon as the objective value reaches a value of 300, if the linear relaxation gave a lower bound that equals the true optimal value. Two different models could therefore stop at different objective values, which is not necessarily a problem of the models, but resulting from the MIP Gap threshold.

[6]Note that due to the serious problems in solving the GLSP-FF in these two instances while other instances were solved quickly, several runs were executed for further testing (only the results from the first run were included in this thesis results). When setting "parallel mode" in CPLEX to deterministic, leading to an identical search of the solution tree, the results were identical. Without this, the run time was most of the time much shorter and optimal solutions could be found within a couple of minutes. So here it seems likely, that the solution space in the GLSP-FF can have a structure that, depending on minor parameters of the mathematical solver, causes the solver to be searching in an inferior solution space for a relatively long time before identifying it as inferior and therefore discarding it.

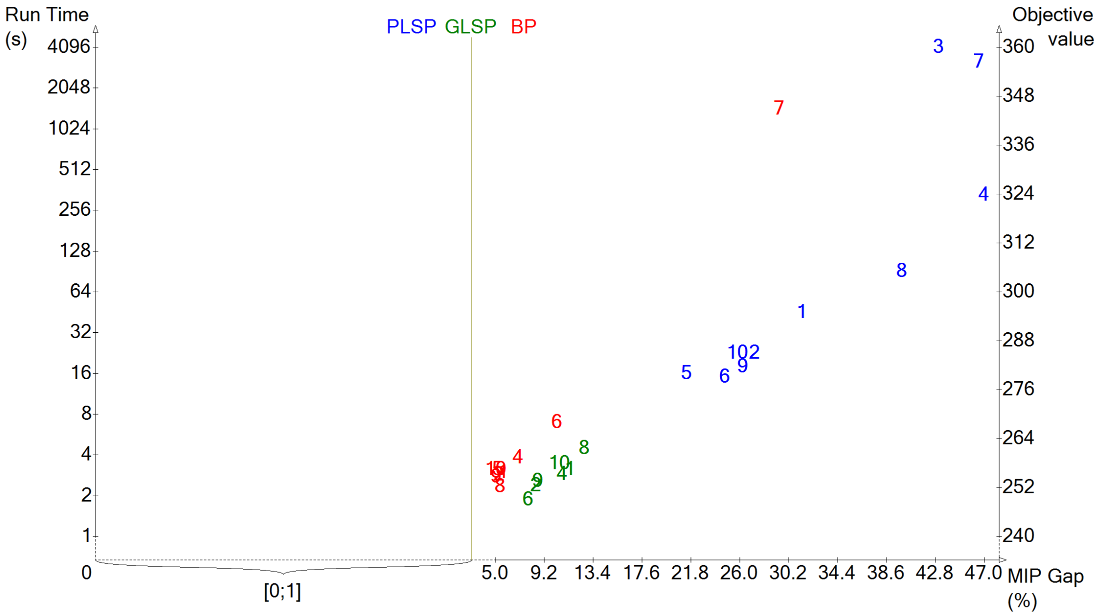

Figure 5.1.: *Detailed results of the Full Flexibility Scenario, Size L, Low Capacity Load.*

most applications, the higher stability of the BP-FF results and only slightly worse run time in some instances than the GLSP-FF is likely to be of higher importance than a chance of a slightly shorter run time, which depends on the specific demand distribution.

Regarding the effect of the capacity load, it is noteworthy that the BP-FF models results are not affected by it for the given two different loads. For the GLSP-FF results, the higher capacity load increased the duration to find optimal solutions significantly. The PLSP-FF had a lower MIP Gap for the high load capacity, however in both capacity loads it failed to find optimal solutions.

Large problem size (25 products)

In the large problem size, the limits of the given model formulations can be observed. For this problem size, a more detailed analysis is given, based on figure 5.1 and figure 5.2, which show the detailed results for both capacity loads.

While the PLSP-FF is still able to find feasible solutions for all instances, which the other two models fail to, it can be seen that the MIP Gaps are very high, being on average 33.3% in low capacity load instances and 23.4% in the high capacity load instances. As can be seen in the detailed results in the appendix, for no iteration the MIP Gap was less than 21.4%. In the low capacity load instances, the BP-FF found in nine instances a feasible solution, whereas the GLSP-FF found a feasible solution only in 7 instances. Additionally, the MIP Gap of the BP-FF solutions was generally smaller. On the other hand, as can be seen in figure 5.1, for those instances where both models found an optimal solution, the GLSP-FF required a little less time and, in instances where all models found a feasible solution, the objective values of the GLSP-FF were slightly better than

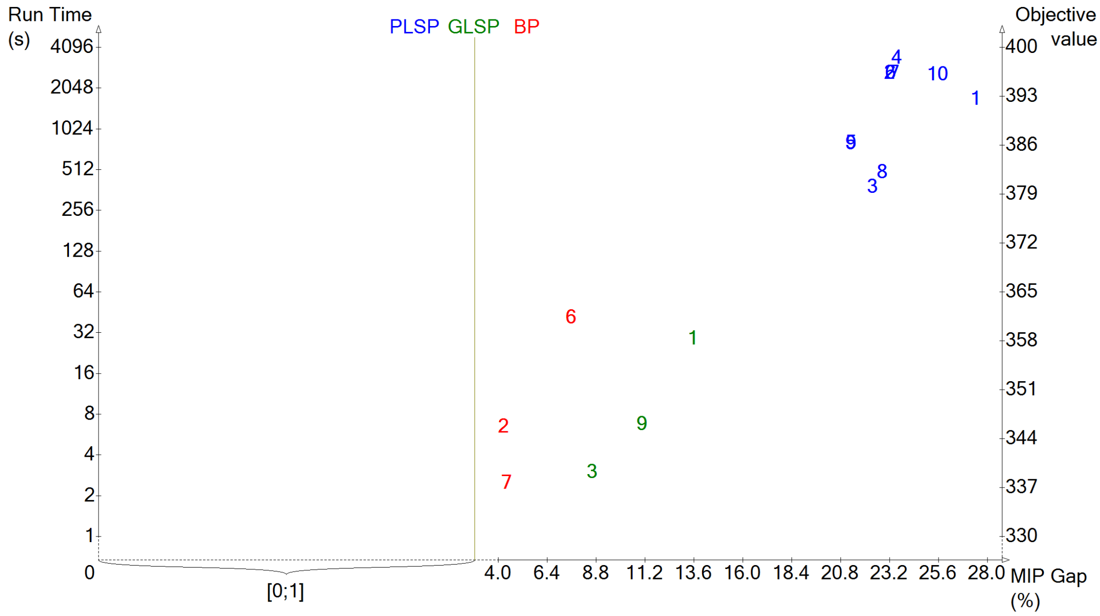

Figure 5.2.: Results of the Full Flexibility Scenario, Size L, High Capacity Load.

those of the BP-FF. This is interesting, since both models should, given enough time, be able for each instance to find the same optimal solution, and a higher MIP Gap usually indicates a higher potential for further improvement of the best found solution so far. In this comparison, obviously the potential for improvement up to the true optimum is of course higher if the best found solution so far is worse, so the BP-FF with worse objective values should be expected to have higher MIP Gaps. This is not the case here, which indicates a numerical advantage of the BP-FF: as a higher MIP Gap indicates a higher required additional time to close the Gap and find an optimal solution, even though the best found solution objective value at the time limit of 3600 seconds, there is a reasonable possibility that with longer run times the BP-FF gains an advantage over the GLSP-FF in terms of the objective value. However, to explore this additional numerical tests would be necessary. For the high capacity load scenario (illustrated in figure 5.2), a comparison of the BP-FF and the GLSP-FF is not possible, as for no instance both models found a feasible solution. However, for those instances where a feasible solution was found, it was better than the PLSP-FFs best found solution, while the PLSP-FF has as already mentioned the advantage of having found feasible solutions for all instances.

The capacity load for the PLSP-FF had a similar effect as for the medium problem size, reducing the MIP Gap for the high load instances. For the GLSP-FF, difficulties in finding a feasible solution increased strongly, while the MIP-Gap was not increased significantly. The BP-FF however, while having been relatively unaffected in the medium problem size, did encounter high problems with finding a feasible solution in the high load instances as well. This leads to the conclusion that the BP-FF ability to find optimal solutions is not strongly influenced by the capacity load, however when finding feasible solutions gets problematic, the BP-FF is similarly affected like the GLSP-FF.

Summary

As could be seen, the Full Flexibility Scenario puts a challenge to the mathematical modeling approach, as the difficulty of finding feasible or optimal solutions rises exponentially with the problem size. However, in the small and medium problem sizes the BP-FF and GLSP-FF models still achieved very good results (with the BP-FF model being slightly ahead of the GLSP-FF model), while the PLSP-FF model performed only mediocre in the small problem size and struggled with the medium problem size. For the large problem size, results are a little bit more complicated. The BP-FF and GLSP-FF models faced difficulties in finding feasible solutions for some instances, especially in the high capacity load instances. For those, where feasible solutions were found, the MIP-Gap was still relatively high after the solution time limit was exhausted. The PLSP-FF model interestingly found feasible solutions for all instances. However, the found solutions were far from optimal, as can be seen by the MIP Gap and even more definitive in those instances where all three models found feasible solutions, the best solutions found by the PLSP-FF model were significantly worse in quality than the best solutions found by the other two models. Therefore, in applications where finding a feasible solution is sufficient, the PLSP-FF model might be the best to choose. However, it seems possible that this advantage of the PLSP-FF is less caused by an inherent advantage of the model, but by the method employed by the default CPLEX solver to search the discrete solution pattern. Developing and applying better suited search patterns might nullify this advantage of the PLSP-FF model.

5.3. Limited Changeover Scenario

The limitation of changeover is here motivated by a dominating property of the production process like a required temperature for processing a product family. The number of product families for the three scenarios S/M/L is set to 3/4/5, with the products being evenly distributed between the families. For this numerical investigation, it is assumed that the products are split evenly across the families (e. g. 2/1/1 for 4 products and 3 families). Due to technical limitations, it is not possible to directly change from a low temperature family to a high temperature family, or it would incur very high costs (e. g., stop of production while the processor is in the heating/cooling process). Therefore, here it is assumed that the families are in an order and only changeovers to direct neighbors are allowed. E. g., for the S-Scenario, from family 1 only family 2 can be reached, from family 2 both other families can be reached, and from family 3 only family 2 can be reached. To allow to go from family 1 to family 3, it is necessary to make a changeover to family 2 and produce a minimal quantity of products belonging to family 2. This minimal quantity is assumed here require 10% of the available weekly capacity divided by the number of families, as a larger number of families makes it reasonable that the differences of the dominating property of the production process like temperature is more similar between the families if there are more product families.

The duration of a changeover between different families is, if it is feasible, assumed to require 2 CU additionally to the product changeover time. This leads under consideration

Table 5.3.: Performance indicators of the Limited Changeover Scenario.

	Scenario Size	S		M		L	
Model	Capacity Load	70%	90%	70%	90%	70%	90%
PLSP-LC	Optimal Solutions Found (%)	50	40	0	0	0	0
	∅ Optimal Solution Time (s)	795.2	1452.3	n/A	n/A	n/A	n/A
	Feasible Solution Found (%)	100	90	100	40	40	0
	∅ MIP Gap (%)	3.1	3.3	15.8	12.7	49.2	n/A
	∅ Derivation From Best (%)	1.3	1.1	13.8	10.4	40.5	n/A
GLSP-LC	Optimal Solutions Found (%)	100	100	100	70	0	0
	∅ Optimal Solution Time (s)	1.4	2.0	503	1088	n/A	n/A
	Feasible Solution Found (%)	100	100	100	100	100	80
	∅ MIP Gap (%)	n/A	n/A	n/A	1.6	5.2	6.0
	∅ Derivation From Best (%)	0	0	0	0	4.7	n/A
BP-LC	Optimal Solutions Found (%)	100	100	100	100	100	100
	∅ Optimal Solution Time (s)	<1	<1	1.5	2.9	14.4	35.7
	Feasible Solution Found (%)	100	100	100	100	100	100
	∅ MIP Gap (%)	n/A	n/A	n/A	n/A	n/A	n/A
	∅ Derivation From Best (%)	<0.1	<0.1	0	<0.1	0	n/A

of the average changeover times for the S/M/L scenarios and assuming production of averagely half of the portfolio per period to capacity requirements for a changeover between different products of the same family of 1/0.889/0.5 CU. E. g. for the S Scenario:

$$\frac{5\,[\text{total changeover}] - 0.5 \cdot 3 \cdot 2\,[\text{required for family changeover}]}{0.5 \cdot 4\,[\text{average number of product changeovers}]} = 1$$

The results are summarized in table 5.3 by averaging the values for 10 instances of different demand values.

Small problem size (4 products)

The PLSP-LC struggles with the small problem sizes instances much more than the PLSP-FF for similar sized instances. It was not able to find optimal solutions in more than half of the instances, and if it did it required a relatively long time, almost 800 seconds on average for the low capacity load instances and about 1500 seconds for the high capacity load instances. In comparison, both the GLSP-LC and the BP-LC models achieved very good results by finding optimal solutions in two or less seconds (for the BP-LC even less than one second).

The results worsen with increased capacity load for the PLSP-LC and GLSP-LC models. The GLSP-LC still has very good results, however by comparison it requires an about 33% longer run time to find an optimal solution. The PLSP-LC found one less optimal solution than in the low capacity load instances, and even failed to find a feasible solution for one instance. The run time for the PLSP-LC also worsens, requiring almost double the time on average to find an optimal solution. In those instances where it did not find an optimal solution, the MIP Gap stays roughly the same. For the BP-LC, no effect of the capacity load could be identified for this problem size.

Medium problem size (9 products)

The BP-LC model shows clear advantages here compared to both other models. It found optimal solutions for all instances in less than three seconds for all instances of both capacity loads. The GLSP-LC did in the low load capacity instances still show a good performance, requiring on average less than 10 minutes to find an optimal solution. However, in the high capacity load instances it was not able to find an optimal solution within the time limit for three instances, and where it found an optimal solution, the required time doubled. The PLSP-LC is struggling, while in the low capacity load scenario it was at least able to find feasible solutions for all instances, in the high capacity load it was not able to find those in all instances anymore. Also, the MIP Gap to the optimal solution was high with over 12% for all instances. Comparing the best solutions found by the PLSP-LC to the best of all three models, its objective value was more than 10% higher on average.

The effect of the capacity load is pretty similar for all models here. Where optimal solutions were found, the required time to find those roughly doubled. Also, for the GLSP-LC finding optimal solutions became much more difficult, failing three times, and for the PLSP-LC finding of even feasible solutions was becoming difficult, failing six times while failing zero times in the low capacity load instances.

Large problem size (25 products)

Even in the large problem size scenarios, the BP-LC still was able to find optimal solutions in perfectly reasonable time, of about 15 seconds on average in the low capacity load instances and a little more than 30 seconds on average in the high capacity load instances. The GLSP-LC was not able to cope with this scenario size anymore in a satisfying way, as no optimal solutions could be obtained anymore within the time limit. The MIP Gap and the difference to the optimum as determined by the BP-LC was more than 5%, and in the high capacity load instances it failed two times to find an even feasible solution. The PLSP-LC is hardly able to find any feasible solutions anymore, only in the low capacity load instances it found four of them, and for those the MIP Gap and the relative makespan increase was more than 40%, making these plans very poor in regard to the objective value.

Comparing the two capacity loads, it can be seen that the BP-LC did just like in the medium problem size require roughly double the time to find optimal solutions in the high capacity load instances. The GLSP-LC and PLSP-LC both had more problems finding feasible solutions, failing more often. In the high capacity load instances, the PLSP-LC was not able to find a single feasible solution anymore.

Summary

The results show that the PLSP-LC is not suited to solve problems of industry relevant size. Even in small problem instances it found an optimal solution in only about half of the instances, and required a relatively high running time. For the medium size scenario, in the low capacity load scenario, it was still able to find feasible solutions for all instances, however with a high MIP Gap and, where a comparison was possible, considerable suboptimal objective value. In the high capacity load instances it did struggle to find feasible solutions, as well as in the large scenario sizes. The GLSP-LC and the BP-LC models were both able to cope with scenarios of medium or smaller size. However, the BP-LC outperformed the GLSP-LC clearly. While the GLSP-LC required a relatively high running time to find optimal solutions in the medium sized problems, in the large sized problems it was only able to find feasible solutions, and in the high capacity load scenario in a few instances not even a feasible solution could be found. The BP-LC was able to find optimal solutions in all problem sizes and capacity load instances in a very reasonable time of averagely no more than 36 seconds. This makes the BP-LC a viable approach to cope with industry sized problems and is, considering the BP model was originally developed for a more specific changeover structure, a very good result.

Regarding the effects of the capacity load, it can be seen that all models require about double the run time to find optimal solutions, in those instances where optimal solutions could be found. Additionally, the models did find less optimal solutions in the high capacity load instances. An influence on the MIP Gap of best found feasible solutions cannot be identified clearly. These results show that the capacity load has an important influence on the performance of the given model formulations, however the dominating factor is the problem size.

5.4. Natural Sequence Scenario

In the natural sequence scenario, similar products, which can be grouped into product families, follow a predefined sequence. Changeover times when changing between products following the natural sequence are relatively small and can be considered constant in many cases, e. g. a standard cleaning procedure. For a changeover to a product of another family, or to go back in sequence, a high changeover time is needed, e. g. changing the equipment. The number of product families for the three scenarios S/M/L is set to 2/3/5, with the products being evenly distributed between the families. The duration of a changeover between different families is assumed to require 4 CU additionally to the product changeover time. This leads under consideration of the average changeover times for the S/M/L scenarios and assuming production of averagely half of the portfolio per period to durations for changeovers between different products of 0.5/0.444/0.4. E. g. for the S Scenario:

$$\frac{5\,[\text{total changeover}] - 0.5\cdot 2\cdot 4\,[\text{required for family changeover}]}{0.5\cdot 4\,[\text{average number of product changeovers}]} = 0.5$$

Table 5.4.: Performance indicators of the Natural Sequence Scenario (S, M, L sizes).

	Scenario Size	S		M		L	
Model	Capacity Load	70%	90%	70%	90%	70%	90%
PLSP-NS	Optimal Solutions Found (%)	90	30	0	0	0	0
	∅ Optimal Solution Time (s)	687	1587	n/A	n/A	n/A	n/A
	Feasible Solution Found (%)	100	100	100	0	0	0
	∅ MIP Gap (%)	2.0	2.2	24.8	n/A	n/A	n/A
	∅ Derivation From Best (%)	0.6	0.8	14.6	n/A	n/A	n/A
GLSP-NS	Optimal Solutions Found (%)	100	100	100	100	100	100
	∅ Optimal Solution Time (s)	<1	<1	<1	<1	1.6	3.4
	Feasible Solution Found (%)	100	100	100	100	100	100
	∅ MIP Gap (%)	n/A	n/A	n/A	n/A	n/A	n/A
	∅ Derivation From Best (%)	0	0	0	n/A	n/A	n/A
BP-NS	Optimal Solutions Found (%)	100	100	100	100	100	100
	∅ Optimal Solution Time (s)	<1	<1	<1	<1	<1	1.4
	Feasible Solution Found (%)	100	100	100	100	100	100
	∅ MIP Gap (%)	n/A	n/A	n/A	n/A	n/A	n/A
	∅ Derivation From Best (%)	0	<0.1	0	n/A	n/A	n/A

The results are summarized in table 5.4 by averaging the values for 10 instances of different demand values.

Small problem size (4 products)

In the small problem size, both GLSP-NS and BP-NS achieved very good results, finding an optimal solution in less than a second. The PLSP-NS however did struggle even in this small scenario. In the low capacity load instances, it was not able to find an optimal solution in one test instance, and in the high capacity load instances it failed three times. If it found an optimal solution, it required a relatively long time of roughly 10 to 20 minutes.

The influence of the capacity load cannot be measured here for the GLSP-NS and BP-NS model. In the PLSP-NS model however, the required run time to find an optimal solution doubles.

Medium problem size (9 products)

The GLSP-NS and BP-NS did also for the medium problem size achieve very good results, not notably worse than for the small problem size. The PLSP-NS did already fail to find optimal solutions anymore, in the high capacity load instances it did even fail to find feasible solutions. Obviously, the PLSP-NS does not benefit from the trivial sequencing in this scenario, but results are even slightly worse than for the PLSP-LC.

Table 5.5.: *Performance indicators of the Natural Sequence Scenario L, XL and LHV.*

Model	Scenario Size Capacity Load	L 70%	 90%	XL 70%	 90%	LHV 70%	 90%
BP-NS	∅ Optimal Solution Time (s)	<1	1.4	1.2	2.0	<1	1.2
GLSP-NS	∅ Optimal Solution Time (s)	1.6	3.4	9.6	16.9	2.8	2.9

The influence o the capacity load is not measurable for the GLSP-NS and the BP-NS in the medium sized problem. For the PLSP-NS, the higher capacity load made it impossible to find any feasible solutions anymore, while in the lower capacity load feasible, though no optimal solutions were still found for all instances.

Large problem size (25 products)

Even for the large problem size, both GLSP-NS and BP-NS achieved very good results. An advantage of the BP-NS over the GLSP-NS can be seen, however the results are not clear enough to identify a pattern so far. The PLSP-NS failed to find any feasible solutions and is obviously unsuited for larger problem sizes.

A first impact of the capacity load for the GLSP-NS and the BP-NS can be identified, as solution times increase with the higher capacity load.

Regarding the GLSP-NS and the BP-NS, for the Natural Sequence Scenario the results were very good, instances of all sizes and capacity loads were solved to optimality within a few seconds. To allow for a comparison between these two models, two additional subscenarios XL ("eXtra Large") and LHV ("Large with High demand Variation") are defined based on the L size scenario. For XL, the number of products is increased to 60 and the total capacity consumption by changeover activities is increased to 20%. For LHV, the variation of the demand elements size is increased: Demand elements are drawn with a likeliness of 40% from uniform distribution d^a in $[0.5 \cdot D; 1.5 \cdot D]$, with a likeliness of 40% from d^l in $[0; 0.5 \cdot D]$ and with a likeliness of 20% from d^h in $[1.5 \cdot D; 3.5 \cdot D]$.

The results are summarized in table 5.5, including the L scenario size as reference. As all instances found an optimal solution, only the time until it was found is given. Note that, as the PLSP-NS was not able to find any feasible solutions in the L scenario, it was not considered in this additional tests.

Extra large problem size (60 products)

The GLSP-NS and the BP-NS still achieved a very good performance in this problem size. Especially the BP-NS was still able to find optimal solutions in just two seconds on average (at most three seconds in some instances). By comparison, the GLSP-NS performed now significantly worse, requiring about six times the run time. Perhaps even more noteworthy is that in the L scenario size, the run of the GLSP-NS was at most about 2.5 times (on average) that of the BP-NS, meaning that apparently the relative advantage of the BP-NS over the GLSP-NS gets higher the bigger the problem size is. This might put a limit to the applicability of the GLSP-NS in bigger industry sized problems where the number of products is a medium three digit value or more, whereas the BP-NS is likely to be able to achieve a very good performance in any realistic problem size. However, as the XL scenario already has a realistic number of products for most of FMCG industry, the GLSP-NS is also a viable choice, as long as no additional constraints, like e. g. a multi-level production or several parallel capacities, which impact the difficulty of the problem have to be considered. For those, it seems more promising to base a model extension on the BP-NS than on the GLSP-NS, as long as the aspect of the given Natural Sequence holds.

The high capacity load instances require a little less than double the time of the low capacity load instances to be solved to optimality for both the GLSP-NS and the BP-NS. This is in compliance with most of previous results.

High demand variability (25 products)

In the high demand variability model, the opportunity was taken to identify how strong those two model formulations react to a higher or lower variation of demand element sizes. However, for the given higher demand variability, no clear effect on the performance of the models could be identified. The perfomance is similar to the large scenario with standard demand variation. Only the GLSP-NS in the low capacity load instances performed significantly worse in the LHV than in the L subscenario. The combination of a relatively loose capacity, meaning more different solutions being feasible, and larger demand element variation put therefore an additional challenge to the GLSP-NS in finding an optimal solution and closing the MIP Gap to prove it. However, results are still very good so this does not limit the applicability of the GLSP-NS in relatively large problem sizes.

Summary

From these results, it is clear that the PLSP-NS is inferior to the GLSP-NS and BP-NS. Only in the small size scenario, it was able to find the optimal solution in the given time limit, however it required even in these small instances significant time, whereas the other two models solved it practically instantly. In the medium size scenario, it was able to find feasible solutions if the capacity load was low, in the high load instances it failed to even find a feasible solution. It is noteworthy that the PLSP-NS performed worse than the PLSP-LC, though the Natural Sequence Scenario should be easier as the sequencing practically does not exist in it. The PLSP-NS differs from the PLSP-LC in three points: the constraints of the PLSP-LC which limited the family changeover and the constraint to ensure minimum lot sizes were removed, while the constraint for ensuring the natural sequence was added. The likely explanation for the relatively bad performance is, that the products still had to be assigned to the single micro periods. The BP-NS and the GLSP-NS performed very good in all scenario sizes and demand variations. As can be seen, the increase in demand variability had no clear effect on the solution time for both the GLSP-NS and BP-NS. This shows that variability of the demand quantities is not a critical issue for both models and make them usable in real industry environments with different demand fluctuation levels. All together, the BP-NS outperformed the other two models, which shows its strength in the environment where it was originally developed for – the existence of a natural product sequence.

Similarly to the Limited Changeover Scenario, the high capacity load instances increased the run time for finding an optimal solution to about twice that of the low capacity load instances, making the capacity load an important factor, which is however less important than the problem size.

6. Concluding Remarks and Outlook

In this thesis, specific characteristics regarding the production planning in the Fast Moving Consumer Goods industry have been discussed.

In chapter 2, the Fast Moving Consumer Goods Industry was characterized in comparison to other types of industry. It was outlined that most models in literature for lot-sizing and scheduling are still focusing on minimizing of planned costs, while those costs are rarely out-of-pocket costs, but disputed costs of questionable relevance in a fast paced environment like the FMCG industry. Therefore, the Makespan as an objective to maximize more relevant goals like retaining flexibility of the production system has been introduced. It was shown that in many industries of the FMCG industry, for means of lot-sizing and scheduling the production process can be simplified to a single-stage production, known as Make-And-Pack production. The high relevance of considering a limited capacity for production planning and high utilization was discussed. A specific focus was put on the effect of different setup structures which can be identified in the FMCG industry. Five general types of changeover structures were identified, the independent changeover, family changeover, limited changeover, natural sequence and full flexibility.

In chapter 3, the problem of production planning in the FMCG industry was discussed. The development of concept for tackling this problem has been outlined, and a focus was set on the issue of lot-sizing and scheduling, showing the advantages of integrating these two often successively treated planning steps. This was followed by an overview of different approaches to solve the integrated lot-sizing and scheduling problem, with a focus on the evolution of mathematical models.

Then, in chapter 4, the most current models of PLSP, GLSP and Block Planning have been modified and extended to be able to reflect the integrated lot-sizing and scheduling problem for three different changeover structure scenarios which incorporate different aspects of the five mentioned general changeover structures. More detailed aspects, like e. g. minimum lot size or product groups have been regarded in the scenarios where they are thought to be most appropriate. However, modeling techniques to regard those can, if necessary, be applied to other scenarios as well.

A quantitative comparison of the different modeling techniques for the different scenarios has been carried out in chapter 5 to compare the different modeling techniques to each other in regard to their performance. For each scenario, three (for the natural sequence scenario five) scenario sizes were defined which increased in difficulty, mainly differing by the number of products to be planned. For each scenario size, additionally two capacity load variations have been given to represent different planning situations, e. g. different seasons in seasonal demand patterns, fluctuations in available capacity due to holiday times or regular long-time machine maintenance and so on. To ensure that the models performance is measured in a couple of different planning situations, ten instances of different realistic demand and, where appropriate, changeover values were determined by using of a specifically designed algorithm.

The results of the numerical tests allowed for a number of conclusions in regard to the research questions as formulated in chapter 5:

- Some major differences in the performance of the three model formulations could be observed. The most clear result is that the PLSP based models performed relatively poorly. Only in the smallest test instances it was able to find optimal solutions. In most more realistically sized test instances, it had problems finding feasible solutions, and where it did they were far from the optimal solutions (if those have been found by the GLSP and/or Block Planning based models) and had high MIP Gaps. However, interestingly in the theoretically most complex scenario, the Full Flexibility Scenario, it was able to find feasible solutions for all instances, whereas the Block Planning and GLSP based models had problems finding feasible solutions in the large sized instances. In the two scenarios with more limitations in regard to sequence possibilities, it struggled more than the Block Planning and GLSP based models, while the Block Planning based model was slightly superior to the GLSP based models. Therefore, for problems where finding feasible solutions is paramount and having no limitations on possible sequences, the PLSP might be a reasonable model. However, the quality of the found solutions is hard to judge, especially since the PLSP model requires much time to close the MIP Gap. The smaller size scenarios result hint that the Block Planning and GLSP based models are able to close the MIP Gap much quicker than the PLSP. So the likely conclusion is, that in the Full Flexibility scenario the PLSP is superior in finding a feasible solution, whereas the other two models are superior in closing the MIP Gap to find good or optimal solutions. This result might be of use in further research, as for sequence dependent changeovers and realistic problem sizes, it is likely that in the foreseeable future heuristic methods will have to be applied to generate sufficiently good solutions. For example, local search based heuristics might make use of a feasible solution generated by a PLSP model and improve it iteratively, decomposition based heuristics might make use of the BP or GLSP based models to solve relatively small subproblems to optimality (results from this research would hint that the GLSP based model is more efficient in scenarios with a mediocre capacity load, while the Block Panning based model is superior in models with a high capacity load).

  In the scenarios with limitations on the possible sequences, the PLSP-NS was outperformed clearly by both other models. In the Natural Sequence scenario, the BP-NS and GLSP-NS models were so efficient in all sizes S, M and L, that additional problems were defined to allow for a comparison of the models, one increasing the size significantly, the other changing the demand element structure to have a higher variance. It could be observed that the higher demand variance only had a mediocre negative impact on the efficiency on the GLSP-NS model in the high capacity load, while for the low capacity load and for the BP-NS model in general no negative impact was observed. In all problem sizes, the BP-NS and GLSP-NS model proved to be very efficient, with the BP-NS model outperforming the GLSP-NS by requiring up to eight times less run time than the GLSP-NS. This is not surprising, as the BP model was originally developed to exploit this specific changeover structure, however it shows some significant advantage by designing models to exploit specific problem structures. On the other hand, the GLSP showed to be highly efficient as well – in industry it will be necessary to decide whether the advantage of more specific models outweighs the cost to develop those, or adaption of existing models is to be preferred.

More surprisingly, in the Limited Changeover Scenario the BP-LC model outperformed the GLSP-LC model clearly. This hints at that the Block Planning model, while originally developed to follow natural changeover sequences, has some more deep advantages. Concluding from this research, it is likely that the Block Planning model is very strong if changeover times do not depend on the sequence of production orders (which otherwise would be contrary to the scenario assumptions the Block Planning model was originally developed for), making the scheduling of production orders within a macro period trivial as long as demand elements are assigned to the end of macro periods, due to its direct assignment of production quantities to demand elements. An interesting further research opportunity would be to develop different model formulations which are not specifically designed for a natural sequence but refrain from using inventory balances for reflecting demand fulfillment but direct assignment of production orders to demand elements. Also, the effect of allowing stockouts on the relative performance of formulations using an inventory balance versus those assigning production directly to demand elements might prove an interesting research opportunity.

Another interesting further research opportunity would be to compare model formulations in environments where the time grid to reflect demand elements needs to be more precise. Here, the PLSP might have an advantage in that it already is designed to consider micro periods for its inventory balance, so as long as the time grid for the demand elements is not more detailed as the time grid of micro periods, no substantial modifications would be necessary, while the GLSP and BP model would need adjustments (e. g. Bilgen/Günther[1] introduced in a Block Planning model so called Heavy-Side variables to track the completion of production orders on a time grid finer than the production blocks). This can be also interesting in regard to the analyzing of the BP-LC model, as the combination of non-sequence dependent changeover times and the demand elements being at the end of macro periods leads to a trivial scheduling problem within a macro period which might be of essence for its relative strong performance.

Summarizing it can be said, that the PLSP is not very promising for further usage as a mathematical model to find optimal solutions in most environments, especially when taking into account the inherent problems of the PLSP model caused by the

[1] Bilgen and Günther (2010)

limitation of one changeover per micro period which also has to be completed in that period and the problem of choosing an "appropriate" micro period length[2], to which no generally accepted approach exists up to today to the best of the author's knowledge. More flexible models like the GLSP or Block Planning are likely to achieve better results, being it in true optimization or as optimizing procedures for subproblems in heuristic approaches.

However it should be noted that the model adaptions were designed with relatively straight-forward extensions. It may well be that more sophisticated model adaptions exist, that increase the performance of the models in certain scenarios compared to the adaptions presented here. Especially the PLSP comes to mind here, as in the theoretically most complex case, the Full Flexibility Scenario, it performed better than in the simpler Limited Changeover Scenario and in the most simple case, the Natural Sequence Scenario, it performed worst – a contrary development compared to the GLSP and BP models. However, it is unlikely that, given similar effort is invested in improving the model formulations, models that perform poorly now would outperform the other models. Additional research would be required to strengthen this point.

- As can be seen, for all scenarios and models the problem size is of major importance for the efficiency of the given model formulations. In the more complex scenarios Full Flexibility and Limited Changeover, the computational requirements increase exponentially with the problem size. This is in compliance with many previous research results. However, in the Natural Sequence Scenario, the computational requirement scales approximately linearly with the number of products for the Block Planning and GLSP based models, while for the PLSP based models the results hint to an exponential increase for this scenario as well. As in real industry applications problem sizes will be more likely to be similar to the larger problem sizes L and XL, the PLSP based model is therefore not practically usable. The Block Planning and GLSP based models are likely to be usable in many real industry environments which match the basic scenario assumptions. Also, for real cases where additional aspects need to be considered in the modeling, it is much more likely to achieve satisfying results using one of these models, with the Block Planning based model offering the best chances. This illustrates the advantage of using problem specific characteristics to simplify the underlying planning problem, like a natural sequence in both relevant steps: the formulation of relevant aspects of the real problem that needs to be considered, and the effort to develop specific modeling approaches to make best use of those aspects.

- As for the last question regarding the impact of the capacity load on the performance of the given model formulations, the results are varying. The PLSP based model for the Full Flexibility Scenario was, contrary to what could be expected, increasing in performance with the capacity load. This may be explainable by the effect, that a higher capacity load causes more tree nodes in the Branch-and-Bound algorithm

[2] Cf. subsection 3.4.2

to be cut off due to proven infeasibility in the high capacity load scenario, allowing the PLSP-FF model to search the tree faster. As the other results shown issues of the PLSP based models in searching the tree quickly and closing the MIP Gap, this seems plausible. The GLSP-FF behaved more like expected, in that with higher capacity load the performance decreased (though still being much better than the PLSP-FF model), while the BP-FF's efficiency was unaffected by the capacity load. In the Limited Changeover and Natural Sequence scenarios, the impact of higher capacity load was as expected, increasing the difficulty with higher capacity load. It can be seen that in most cases the Block Planning based models were less affected than the other models. Additionally it can be seen that the increase in capacity load increases the problem difficulty especially in finding not only feasibly but good solutions, however the impact of a higher problem size in regard to the number of products was significantly higher, as could be expected from previous research results.

A. Appendix

In this appendix, the results of the numerical tests are given more in detail for futher reference. First, tables are given which show the detailed results of all tested instances. *Run Time* is the time after the solver finished, either by closing the MIP Gap to less than 1% or by exhausting the run time limit of 3600 seconds, whichever is first. *Objective Value* gives the best found objective value. If no feasible solution was found, it is given as *None*. *MIP Gap* gives the relative difference between upper and the lower bound on the objective value. If it was less than 1%, the solution was conisdered to be optimal and the solution process was stopped. If no feasible solution was found, the MIP Gap is given as *n/A*.

Second, figures with graphical representations of the detailed results are given. Each figure displays the results for a specific scenario size (S/M/L for the Full Flexibility scenario and Limited Changeover scenario, S/M/L/XL/LHV for the Natural Sequence scenario) and capacity load (low/high) combination. The left parts of the figures show for which instances (1 to 10) the three scenario specific model formulations found an optimal solution (i. e. the MIP Gap was 1% or less) and how much run time was needed to find it. For example, the red "5" in figure A.2 gives that the BP-FF model required 2 seconds of time to find an optimal solution (problem size small, high capacity load) for instance 5. The right parts of the figures show for each model and each instance if a feasible (but no optimal) solution was found, the objective value of the best found solution and the remaining MIP Gap. For example, the blue "8" in figure A.2 gives that the PLSP-FF models best solution for instance 8 had an objective value of 361.6 while still having a MIP Gap of 1.8% (problem size small, high capacity load). Note that instance/model combinations that did not find a feasible solution at all are not included in the figures.

Table A.1.: *Results of the FF Scenario, problem size S.*

	Capacity Load (%)		70			90	
	Model	PLSP	GLSP	BP	PLSP	GLSP	BP
Iteration	Performance Indicator						
1	Run Time (s)	3569	2	2	3600	<1	<1
	Objective Value	276.8	274	274	364	359.5	359.5
	Mip Gap (%)	2.5	<1	<1	2	<1	<1
2	Run Time (s)	3599	<1	1	3600	2	2
	Objective Value	276.6	271.9	271.9	362.1	360.4	360.2
	Mip Gap (%)	2.4	<1	<1	2	<1	<1
3	Run Time (s)	3595	<1	2	3600	1	1
	Objective Value	277.6	275.8	275.5	359.8	358.1	356.4
	Mip Gap (%)	2.3	<1	<1	2.1	<1	<1
4	Run Time (s)	3248	<1	<1	3600	<1	2
	Objective Value	273.7	273.7	273.7	362.1	354.9	355.1
	Mip Gap (%)	<1	<1	<1	3.3	<1	<1
5	Run Time (s)	3572	1	2	3600	2	2
	Objective Value	273.8	273.5	273.5	365.1	357.6	357.7
	Mip Gap (%)	1.3	<1	<1	3.4	<1	<1
6	Run Time (s)	3600	1	<1	3600	<1	1
	Objective Value	273.8	273	271.9	360.5	354.1	354.1
	Mip Gap (%)	1.4	<1	<1	2.4	<1	<1
7	Run Time (s)	3166	1	1	3600	1	<1
	Objective Value	272.9	271.9	271.9	358.4	357.4	357.4
	Mip Gap (%)	<1	<1	<1	1.3	<1	<1
8	Run Time (s)	2949	<1	1	3600	<1	1
	Objective Value	271.3	270.7	270.7	361.6	358.4	358.4
	Mip Gap (%)	<1	<1	<1	1.8	<1	<1
9	Run Time (s)	3599	<1	1	64	<1	<1
	Objective Value	277.5	275.3	275.3	351.5	352.4	350.8
	Mip Gap (%)	2.2	<1	<1	<1	<1	<1
10	Run Time (s)	1956	<1	<1	3600	1	1
	Objective Value	272.8	271.8	271.8	364.6	359.5	359.1
	Mip Gap (%)	<1	<1	<1	2.6	<1	<1

Table A.2.: Results of the FF Scenario, problem size M.

	Capacity Load (%)		70			90	
	Model	PLSP	GLSP	BP	PLSP	GLSP	BP
Iteration	Performance Indicator						
1	Run Time (s)	3600	6	28	3600	249	28
	Objective Value	301.4	263.9	264	362.4	348.5	348.5
	Mip Gap (%)	16.4	<1	<1	7.8	<1	<1
2	Run Time (s)	3600	22	14	3600	1910	82
	Objective Value	301.9	267.6	267.6	362.9	350.1	350.1
	Mip Gap (%)	16.3	<1	<1	7.7	<1	<1
3	Run Time (s)	3600	10	12	3600	542	346
	Objective Value	274.3	265.5	265.1	358.4	348.1	348.1
	Mip Gap (%)	7.9	<1	<1	6.5	<1	<1
4	Run Time (s)	3600	48	195	3600	35	57
	Objective Value	278.1	264.2	264.2	369.2	351.9	351.7
	Mip Gap (%)	9	<1	<1	9.3	<1	<1
5	Run Time (s)	3600	3600	78	3600	41	53
	Objective Value	277.8	264.3	264.2	363.6	352.9	353
	Mip Gap (%)	9.2	1.2	<1	7.9	<1	<1
6	Run Time (s)	3600	96	72	3599	16	20
	Objective Value	306.7	266	265.8	359	348.1	348.6
	Mip Gap (%)	17	<1	<1	6.7	<1	<1
7	Run Time (s)	3600	3600	325	3600	20	38
	Objective Value	272	265.2	264.8	356.8	345.7	346.5
	Mip Gap (%)	6.7	1.5	<1	6.7	<1	<1
8	Run Time (s)	3600	16	46	3600	24	78
	Objective Value	273.8	265.5	265	356.6	348.4	348.6
	Mip Gap (%)	7.8	<1	<1	6.3	<1	<1
9	Run Time (s)	3600	15	49	3600	55	35
	Objective Value	302.4	265.1	264.6	360.4	350.4	350.2
	Mip Gap (%)	16.2	<1	<1	7	<1	<1
10	Run Time (s)	3600	73	66	3600	22	152
	Objective Value	300	264	264	358.2	347.9	347.2
	Mip Gap (%)	15.9	<1	<1	6.6	<1	<1

Table A.3.: Results of the FF Scenario, problem size L.

	Capacity Load (%)	70			90		
	Model	PLSP	GLSP	BP	PLSP	GLSP	BP
Iteration	Performance Indicator						
1	Run Time (s)	3600	3600	3600	3600	3600	3601
	Objective Value	295	256.5	255.7	392.6	358.3	None
	Mip Gap (%)	31.3	11.4	5.7	27.4	13.6	n/A
2	Run Time (s)	3600	3600	3600	3600	3600	3601
	Objective Value	285	252.4	255.1	396.3	None	345.7
	Mip Gap (%)	27.3	8.5	5.3	23.2	n/A	4.3
3	Run Time (s)	3600	3600	3600	3600	3600	3601
	Objective Value	360	None	None	380	339.2	None
	Mip Gap (%)	43.1	n/A	n/A	22.4	8.6	n/A
4	Run Time (s)	3600	3600	3600	3600	3600	3600
	Objective Value	323.8	255.3	259.3	398.5	None	None
	Mip Gap (%)	47	10.8	7	23.6	n/A	n/A
5	Run Time (s)	3600	3601	3600	3600	3600	3600
	Objective Value	280	None	256.4	386.3	None	None
	Mip Gap (%)	21.4	n/A	5.2	21.3	n/A	n/A
6	Run Time (s)	3600	3600	3600	3600	3600	3600
	Objective Value	279.1	249	267.9	396.3	None	361.3
	Mip Gap (%)	24.7	7.8	10.3	23.3	n/A	7.6
7	Run Time (s)	3600	3600	3601	3600	3600	3600
	Objective Value	356.4	None	344.8	396.3	None	337.6
	Mip Gap (%)	46.5	n/A	29.4	23.4	n/A	4.4
8	Run Time (s)	3600	3600	3600	3599	3601	3600
	Objective Value	305	261.6	252.2	382.1	None	None
	Mip Gap (%)	39.9	12.6	5.4	22.8	n/A	n/A
9	Run Time (s)	3600	3600	3601	3600	3600	3601
	Objective Value	281.6	253.5	254.6	386.2	346	None
	Mip Gap (%)	26.3	8.7	5.1	21.3	11.1	n/A
10	Run Time (s)	3600	3600	3600	3600	3600	3600
	Objective Value	285	257.8	256.5	396.1	None	None
	Mip Gap (%)	25.8	10.5	5.1	25.6	n/A	n/A

Table A.4.: *Results of the LC Scenario, problem size S.*

	Capacity Load (%)		70			90	
	Model	PLSP	GLSP	BP	PLSP	GLSP	BP
Iteration	Performance Indicator						
1	Run Time (s)	3597	1	<1	1579	1	<1
	Objective Value	279.7	276	277	364.3	364	364
	Mip Gap (%)	1.7	<1	<1	<1	<1	<1
2	Run Time (s)	232	<1	<1	3592	1	<1
	Objective Value	277.8	276	276	None	369	369
	Mip Gap (%)	<1	<1	<1	n/A	<1	<1
3	Run Time (s)	1135	<1	<1	3600	2	<1
	Objective Value	276	276	276	363	362	363
	Mip Gap (%)	<1	<1	<1	1.3	<1	<1
4	Run Time (s)	3599	1	1	824	<1	<1
	Objective Value	306.3	282	282	360	360	362
	Mip Gap (%)	9.3	<1	<1	<1	<1	<1
5	Run Time (s)	3477	7	2	3461	2	<1
	Objective Value	278.9	278.9	278.9	367	360	360
	Mip Gap (%)	1.4	<1	<1	2.7	<1	<1
6	Run Time (s)	521	1	<1	3462	3	1
	Objective Value	277.4	277	277	363	362	362
	Mip Gap (%)	<1	<1	<1	1.6	<1	<1
7	Run Time (s)	3228	2	<1	963	<1	<1
	Objective Value	277.9	276	276	359.8	359.8	359.8
	Mip Gap (%)	1	<1	<1	<1	<1	<1
8	Run Time (s)	1101	<1	<1	3453	1	1
	Objective Value	277.8	276	276	375	366	366
	Mip Gap (%)	<1	<1	<1	4	<1	<1
9	Run Time (s)	987	<1	<1	3600	8	1
	Objective Value	277.6	276	276	387.5	369.1	369.1
	Mip Gap (%)	<1	<1	<1	7.1	<1	<1
10	Run Time (s)	3600	2	<1	2443	2	<1
	Objective Value	281.1	279	279	359.2	359	359
	Mip Gap (%)	2.2	<1	<1	<1	<1	<1

Table A.5.: *Results of the LC Scenario, problem size M.*

	Capacity Load (%)		70			90	
	Model	PLSP	GLSP	BP	PLSP	GLSP	BP
Iteration	Performance Indicator						
	Run Time (s)	3596	245	1	3595	3063	3
1	Objective Value	305.4	272.7	272.7	None	363.6	363.6
	Mip Gap (%)	14.9	<1	<1	n/A	<1	<1
	Run Time (s)	3595	248	1	3595	2605	11
2	Objective Value	307.1	274.7	274.7	386.6	359.3	359.3
	Mip Gap (%)	15.3	<1	<1	11.1	<1	<1
	Run Time (s)	3600	646	1	3597	632	1
3	Objective Value	335	273.6	273.6	None	368	368
	Mip Gap (%)	22.4	<1	<1	n/A	<1	<1
	Run Time (s)	3594	150	1	3596	3597	1
4	Objective Value	360.4	272.7	272.7	None	373.6	373.6
	Mip Gap (%)	27.6	<1	<1	n/A	1.2	<1
	Run Time (s)	3459	565	1	3591	473	1
5	Objective Value	318.4	274.7	274.7	399.5	356.4	356.7
	Mip Gap (%)	16.8	<1	<1	13.9	<1	<1
	Run Time (s)	3460	1967	5	3579	3206	4
6	Objective Value	297.3	272.7	272.7	None	363.3	363.3
	Mip Gap (%)	12.5	<1	<1	n/A	2.1	<1
	Run Time (s)	3458	10	1	3443	14	1
7	Objective Value	309.2	271.4	271.4	390	354.4	355.6
	Mip Gap (%)	15.1	<1	<1	12.1	<1	<1
	Run Time (s)	3458	348	2	3446	221	<1
8	Objective Value	283.3	271.8	271.8	None	360.2	360.2
	Mip Gap (%)	8.2	<1	<1	n/A	<1	<1
	Run Time (s)	3600	269	1	3600	3600	4
9	Objective Value	291.5	274.7	274.7	None	374.3	373.4
	Mip Gap (%)	10.8	<1	<1	n/A	1.5	<1
	Run Time (s)	3600	583	1	3600	607	3
10	Objective Value	305.4	275.6	275.6	398.6	355.6	355.6
	Mip Gap (%)	14.9	<1	<1	13.6	<1	<1

Table A.6.: Results of the LC Scenario, problem size L.

	Capacity Load (%)		70			90	
	Model	PLSP	GLSP	BP	PLSP	GLSP	BP
Iteration	Performance Indicator						
1	Run Time (s)	3594	3594	3	3594	3594	4
	Objective Value	368.5	259.5	255	None	340.5	339.5
	Mip Gap (%)	57.2	8	<1	n/A	4.2	<1
2	Run Time (s)	3595	3596	27	3594	3594	34
	Objective Value	280.5	252	251.1	None	354	351.5
	Mip Gap (%)	23.4	4.9	<1	n/A	7.8	<1
3	Run Time (s)	3597	3595	7	3596	3591	7
	Objective Value	385	258	256	None	357	353
	Mip Gap (%)	54.4	5.7	<1	n/A	6.4	<1
4	Run Time (s)	3597	3598	5	3595	3595	72
	Objective Value	389.5	250.5	250.5	None	348	348.5
	Mip Gap (%)	61.8	4.6	<1	n/A	6.1	<1
5	Run Time (s)	3448	3482	5	3591	3439	6
	Objective Value	None	250	250	None	None	348
	Mip Gap (%)	n/A	3.8	<1	n/A	n/A	<1
6	Run Time (s)	3458	3448	8	3181	3591	50
	Objective Value	None	250.5	250.5	None	344.5	342
	Mip Gap (%)	n/A	2	<1	n/A	6.3	<1
7	Run Time (s)	3591	3591	65	3438	3424	14
	Objective Value	None	255.9	252	None	347	338.5
	Mip Gap (%)	n/A	6.2	<1	n/A	5.8	<1
8	Run Time (s)	3450	3247	5	3413	3439	59
	Objective Value	None	255	255	None	346	345
	Mip Gap (%)	n/A	5.3	<1	n/A	5.6	<1
9	Run Time (s)	3600	3600	8	3601	3600	51
	Objective Value	None	257.5	250.5	None	None	356.5
	Mip Gap (%)	n/A	6.7	<1	n/A	n/A	<1
10	Run Time (s)	3600	3600	11	3599	3600	60
	Objective Value	None	255.5	255.5	None	345	342.5
	Mip Gap (%)	n/A	5	<1	n/A	5.6	<1

Table A.7.: Results of the NS Scenario, problem size S.

	Capacity Load (%)	70			90		
	Model	PLSP	GLSP	BP	PLSP	GLSP	BP
Iteration	Performance Indicator						
1	Run Time (s)	290	<1	<1	3599	<1	<1
	Objective Value	280	279.5	279.5	370.7	368.5	369
	Mip Gap (%)	<1	<1	<1	2.3	<1	<1
2	Run Time (s)	372	<1	<1	3600	<1	<1
	Objective Value	280	279.5	279.5	383.5	373	373.5
	Mip Gap (%)	<1	<1	<1	4.7	<1	<1
3	Run Time (s)	467	<1	<1	3600	<1	<1
	Objective Value	280	279.5	279.5	373.5	369	369
	Mip Gap (%)	<1	<1	<1	1.3	<1	<1
4	Run Time (s)	3600	<1	<1	736	1	<1
	Objective Value	285.8	279.5	279.5	370	369	369
	Mip Gap (%)	2	<1	<1	<1	<1	<1
5	Run Time (s)	372	<1	1	3479	1	<1
	Objective Value	280.3	279.5	279.5	375	373.5	373.5
	Mip Gap (%)	<1	<1	<1	1.3	<1	<1
6	Run Time (s)	2663	1	<1	1233	<1	1
	Objective Value	282.5	279.5	279.5	364.8	364.5	364.5
	Mip Gap (%)	<1	<1	<1	<1	<1	<1
7	Run Time (s)	197	1	<1	3547	<1	<1
	Objective Value	279.5	279.5	279.5	379	373	373.5
	Mip Gap (%)	<1	<1	<1	2.4	<1	<1
8	Run Time (s)	374	<1	<1	3539	<1	1
	Objective Value	281.9	279.5	279.5	376.3	373.5	373.5
	Mip Gap (%)	<1	<1	<1	2	<1	<1
9	Run Time (s)	748	<1	<1	2793	<1	1
	Objective Value	282.2	279.5	279.5	369.8	369	369
	Mip Gap (%)	<1	<1	<1	<1	<1	<1
10	Run Time (s)	703	<1	<1	3600	<1	<1
	Objective Value	280	279.5	279.5	373.5	373.5	373.5
	Mip Gap (%)	<1	<1	<1	1.5	<1	<1

Table A.8.: *Results of the NS Scenario, problem size M.*

	Capacity Load (%)		70			90	
	Model	PLSP	GLSP	BP	PLSP	GLSP	BP
Iteration	Performance Indicator						
1	Run Time (s)	3594	<1	1	3597	<1	<1
	Objective Value	310	278.2	278.2	None	368.9	368.9
	Mip Gap (%)	23.1	<1	<1	n/A	<1	<1
2	Run Time (s)	3597	<1	1	3598	<1	<1
	Objective Value	316.5	278.2	278.7	None	383.5	385.8
	Mip Gap (%)	24	<1	<1	n/A	<1	<1
3	Run Time (s)	3600	<1	1	3600	1	<1
	Objective Value	334.4	279.1	279.1	None	368.9	368.9
	Mip Gap (%)	28.6	<1	<1	n/A	<1	<1
4	Run Time (s)	3600	1	<1	3600	<1	<1
	Objective Value	291.6	277.8	277.8	None	368.9	369.8
	Mip Gap (%)	18.3	<1	<1	n/A	<1	<1
5	Run Time (s)	3484	1	1	3456	1	<1
	Objective Value	339	277.8	277.8	None	364	364
	Mip Gap (%)	29.6	<1	<1	n/A	<1	<1
6	Run Time (s)	3461	<1	<1	3591	<1	<1
	Objective Value	315.4	273.8	273.8	None	379.1	379.5
	Mip Gap (%)	24.8	<1	<1	n/A	<1	<1
7	Run Time (s)	3531	1	1	3543	1	<1
	Objective Value	304	277.8	277.8	None	364.4	365.3
	Mip Gap (%)	21.8	<1	<1	n/A	<1	<1
8	Run Time (s)	3544	<1	<1	3541	<1	<1
	Objective Value	309.4	277.8	277.8	None	384	384
	Mip Gap (%)	23	<1	<1	n/A	<1	<1
9	Run Time (s)	3600	<1	<1	3600	<1	2
	Objective Value	311.3	277.8	277.8	None	364	364.9
	Mip Gap (%)	23.3	<1	<1	n/A	<1	<1
10	Run Time (s)	3600	<1	<1	3600	1	1
	Objective Value	350.4	279.1	279.1	None	368.9	368.9
	Mip Gap (%)	31.3	<1	<1	n/A	<1	<1

Table A.9.: *Results of the NS Scenario, problem size L.*

Iteration	Capacity Load (%)	70			90		
	Model	PLSP	GLSP	BP	PLSP	GLSP	BP
	Performance Indicator						
1	Run Time (s)	3597	<1	<1	3597	4	1
	Objective Value	None	267.2	267.4	None	362.4	362.4
	Mip Gap (%)	n/A	<1	<1	n/A	<1	<1
2	Run Time (s)	3597	2	1	3597	1	1
	Objective Value	None	265.2	265.2	None	357.2	357.2
	Mip Gap (%)	n/A	<1	<1	n/A	<1	<1
3	Run Time (s)	3600	<1	<1	3600	2	1
	Objective Value	None	266.4	266.4	None	374.8	374.4
	Mip Gap (%)	n/A	<1	<1	n/A	<1	<1
4	Run Time (s)	3600	<1	<1	3600	5	1
	Objective Value	None	260.8	260.8	None	369.2	368.8
	Mip Gap (%)	n/A	<1	<1	n/A	<1	<1
5	Run Time (s)	3591	<1	1	3452	<1	1
	Objective Value	None	270.8	270.8	None	365.2	364.4
	Mip Gap (%)	n/A	<1	<1	n/A	<1	<1
6	Run Time (s)	3187	4	1	3450	4	<1
	Objective Value	None	270.8	270.8	None	384.8	384.8
	Mip Gap (%)	n/A	<1	<1	n/A	<1	<1
7	Run Time (s)	3591	3	1	3354	7	1
	Objective Value	None	264.8	264.8	None	363.2	363.6
	Mip Gap (%)	n/A	<1	<1	n/A	<1	<1
8	Run Time (s)	3591	5	1	3484	9	6
	Objective Value	None	264.4	264.4	None	390	389.6
	Mip Gap (%)	n/A	<1	<1	n/A	<1	<1
9	Run Time (s)	3600	1	<1	3600	1	1
	Objective Value	None	261.2	262	None	380.8	380.8
	Mip Gap (%)	n/A	<1	<1	n/A	<1	<1
10	Run Time (s)	3600	1	<1	3600	1	1
	Objective Value	None	267.2	267.2	None	371.6	371.6
	Mip Gap (%)	n/A	<1	<1	n/A	<1	<1

Table A.10.: *Results of the NS Scenario, problem size XL.*

	Capacity Load (%)	70		90	
	Model	GLSP	BP	GLSP	BP
Iteration	Performance Indicator				
1	Run Time (s)	15	2	18	2
	Objective Value	252.3	252.3	355.3	355.3
	Mip Gap (%)	<1	<1	<1	<1
2	Run Time (s)	3	1	11	2
	Objective Value	248	248	362.3	364
	Mip Gap (%)	<1	<1	<1	<1
3	Run Time (s)	8	1	6	3
	Objective Value	253.6	254.3	350	348.6
	Mip Gap (%)	<1	<1	<1	<1
4	Run Time (s)	11	1	3	3
	Objective Value	248.3	248.3	358.6	362.1
	Mip Gap (%)	<1	<1	<1	<1
5	Run Time (s)	11	2	20	2
	Objective Value	255.6	255.6	361	361
	Mip Gap (%)	<1	<1	<1	<1
6	Run Time (s)	17	2	83	1
	Objective Value	254.3	254.3	372	381.3
	Mip Gap (%)	<1	<1	<1	<1
7	Run Time (s)	8	1	5	2
	Objective Value	254.6	254.6	365.3	353
	Mip Gap (%)	<1	<1	<1	<1
8	Run Time (s)	11	1	3	1
	Objective Value	256	255.6	368.6	369
	Mip Gap (%)	<1	<1	<1	<1
9	Run Time (s)	7	<1	6	1
	Objective Value	250.3	251	362	362
	Mip Gap (%)	<1	<1	<1	<1
10	Run Time (s)	5	1	14	3
	Objective Value	249	249.6	361	361.3
	Mip Gap (%)	<1	<1	<1	<1

Table A.11.: *Results of the NS Scenario, problem size LHV.*

	Capacity Load (%)	70		90	
	Model	GLSP	BP	GLSP	BP
Iteration	Performance Indicator				
1	Run Time (s)	3	1	5	1
	Objective Value	269.2	269.2	362	362.4
	Mip Gap (%)	<1	<1	<1	<1
2	Run Time (s)	2	<1	1	1
	Objective Value	270.4	270.4	357.6	357.6
	Mip Gap (%)	<1	<1	<1	<1
3	Run Time (s)	3	1	4	2
	Objective Value	270	270.4	377.6	378
	Mip Gap (%)	<1	<1	<1	<1
4	Run Time (s)	1	1	3	1
	Objective Value	265.6	266	367.6	368
	Mip Gap (%)	<1	<1	<1	<1
5	Run Time (s)	3	1	4	1
	Objective Value	266.4	266.4	366.8	366.4
	Mip Gap (%)	<1	<1	<1	<1
6	Run Time (s)	2	1	1	1
	Objective Value	260.4	260.8	362	361.6
	Mip Gap (%)	<1	<1	<1	<1
7	Run Time (s)	7	1	4	1
	Objective Value	264	264	361.6	362.4
	Mip Gap (%)	<1	<1	<1	<1
8	Run Time (s)	5	2	1	2
	Objective Value	264.8	264.8	384	384.4
	Mip Gap (%)	<1	<1	<1	<1
9	Run Time (s)	<1	<1	2	1
	Objective Value	260.4	260.4	357.2	357.6
	Mip Gap (%)	<1	<1	<1	<1
10	Run Time (s)	2	<1	4	1
	Objective Value	265.2	265.2	362.8	362.8
	Mip Gap (%)	<1	<1	<1	<1

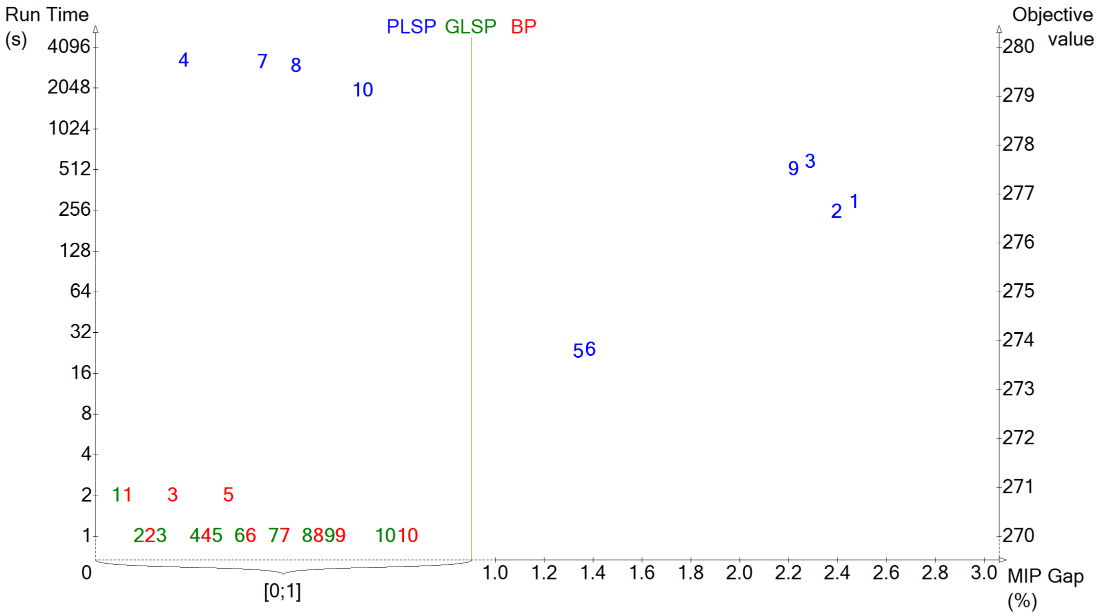

Figure A.1.: Detailed results of the FF Scenario, problem size S, low capacity load.

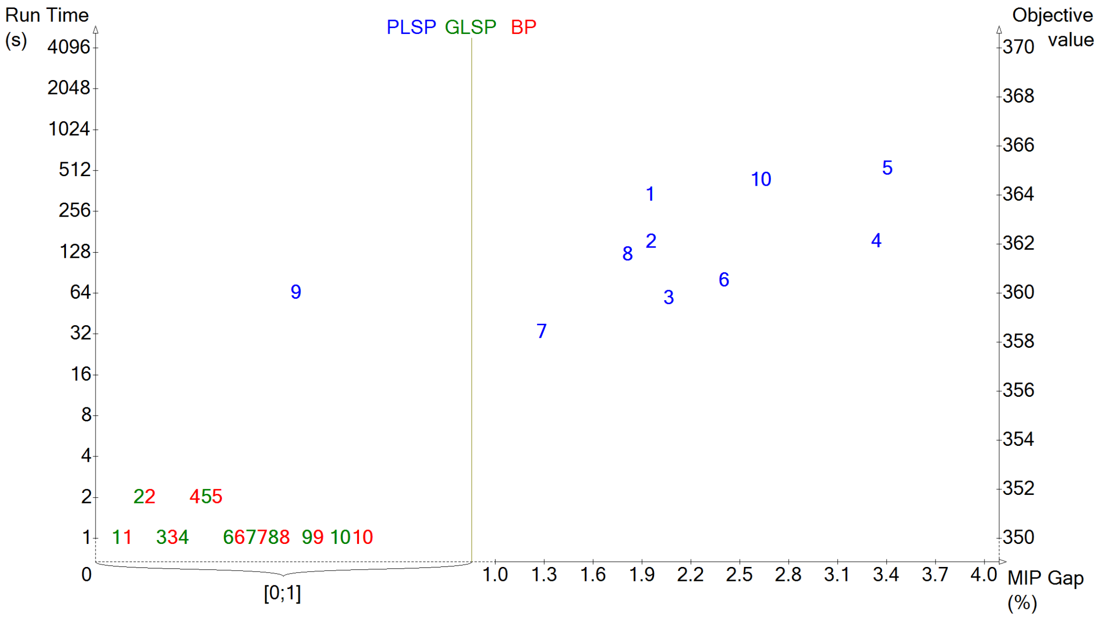

Figure A.2.: Detailed results of the FF Scenario, problem size S, high capacity load.

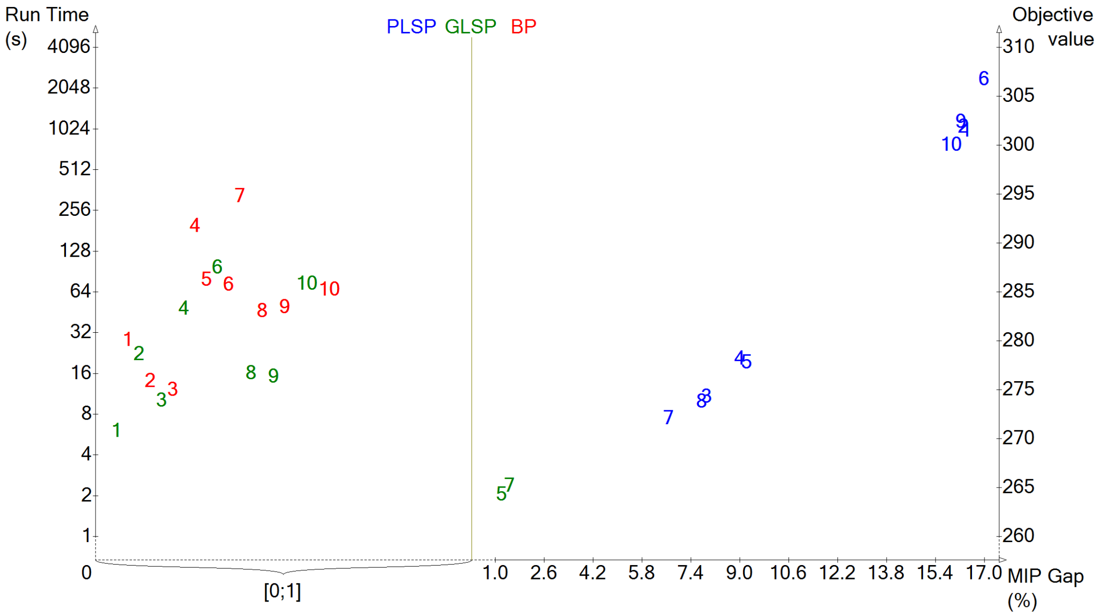

Figure A.3.: Detailed results of the FF Scenario, problem size M, capacity load LCL.

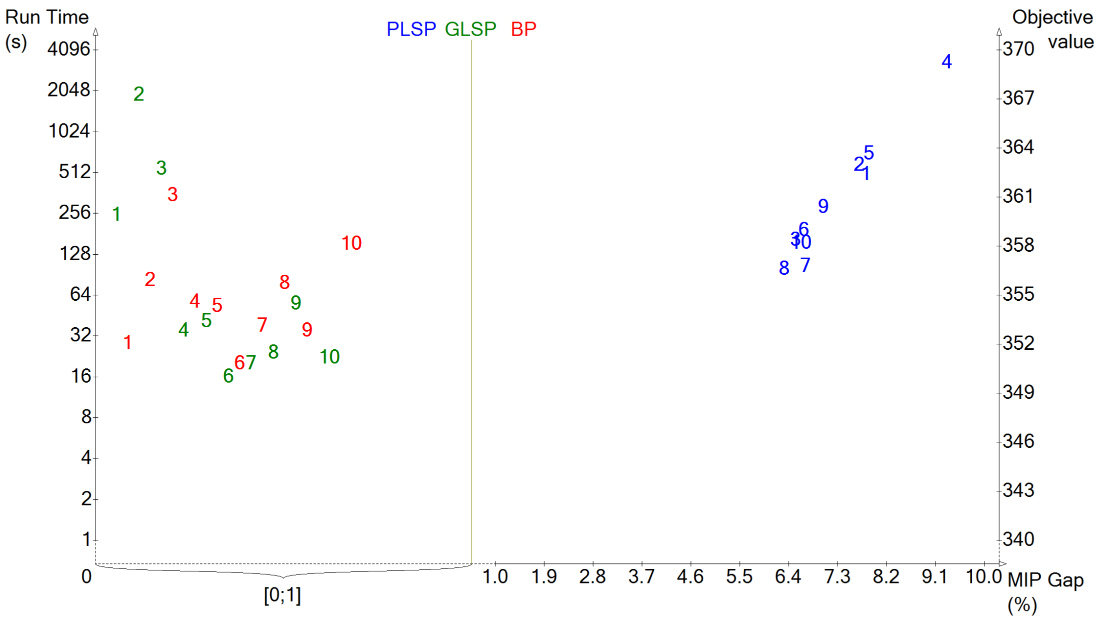

Figure A.4.: Detailed results of the FF Scenario, problem size M, high capacity load.

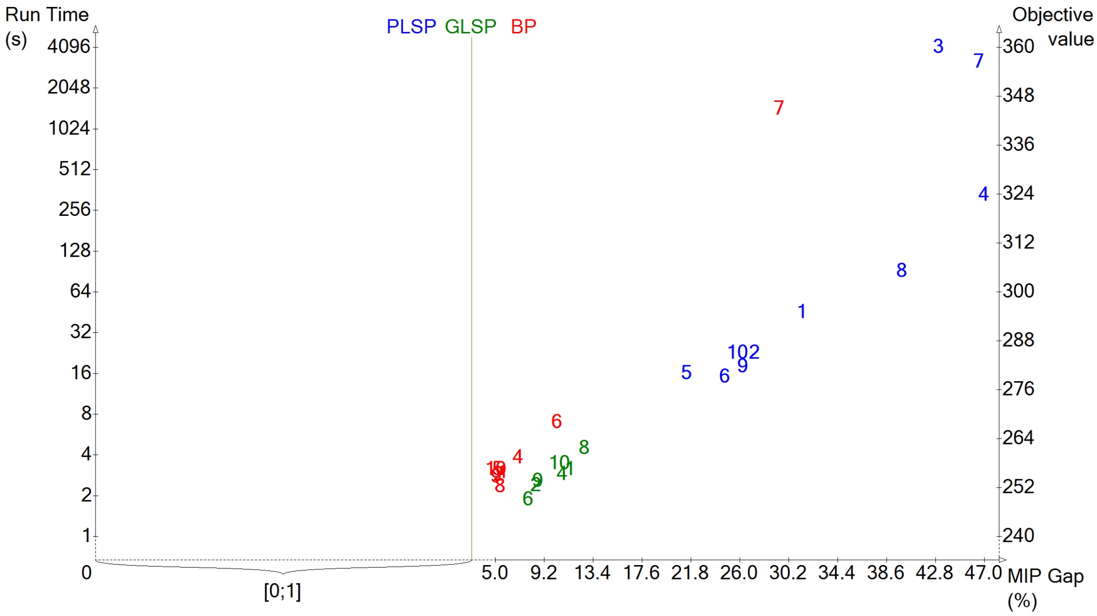

Figure A.5.: *Detailed results of the FF Scenario, problem size L, capacity load LCL.*

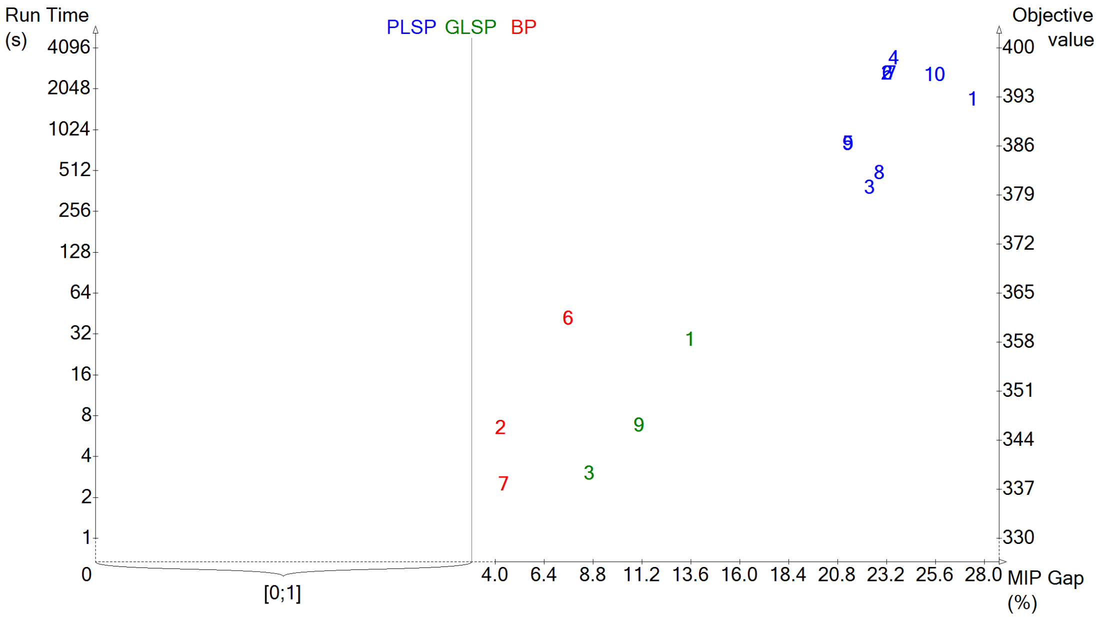

Figure A.6.: *Detailed results of the FF Scenario, problem size L, high capacity load.*

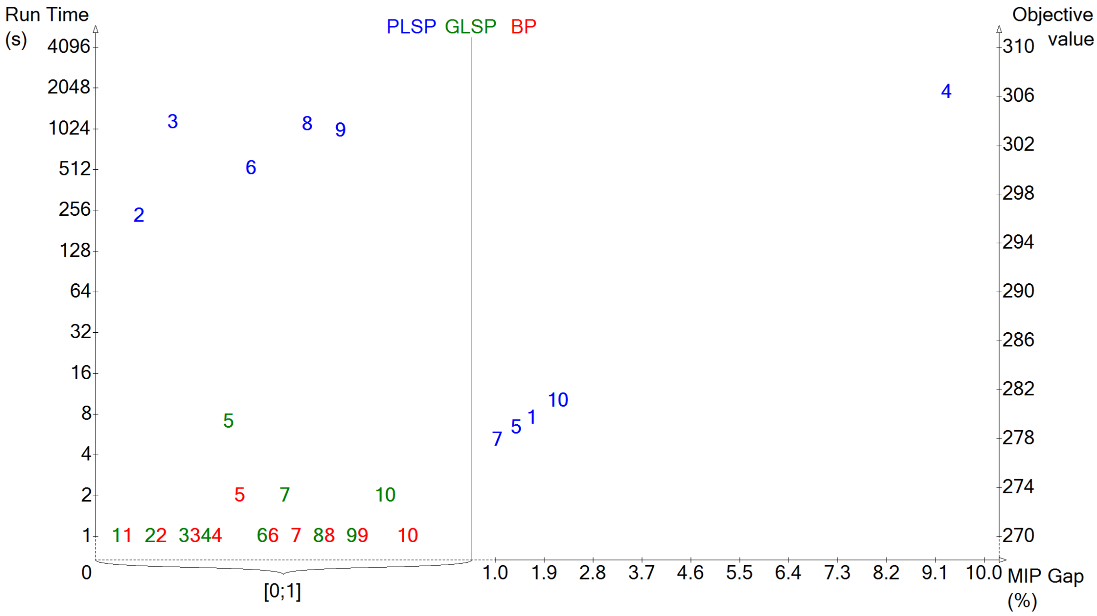

Figure A.7.: *Detailed results of the LC Scenario, problem size S, low capacity load.*

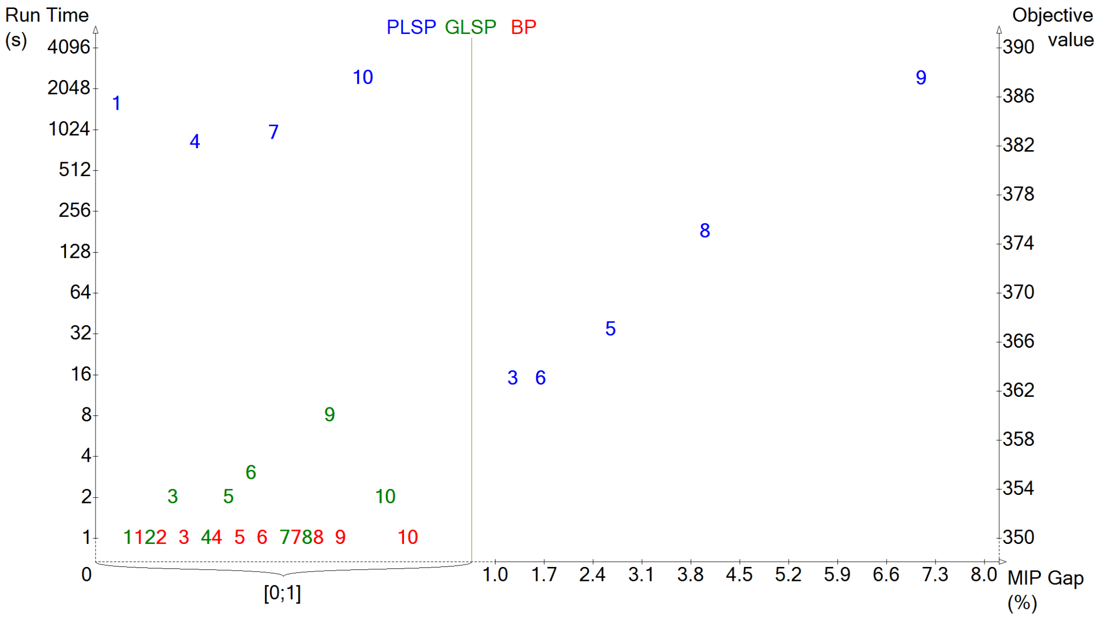

Figure A.8.: *Detailed results of the LC Scenario, problem size S, high capacity load.*

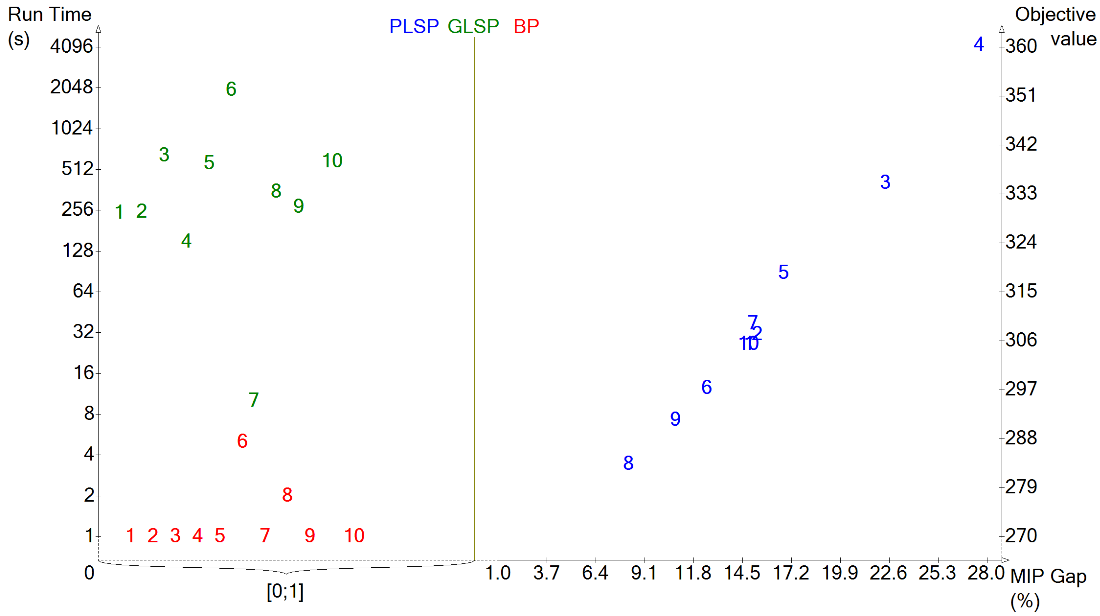

Figure A.9.: *Detailed results of the LC Scenario, problem size M, low capacity load.*

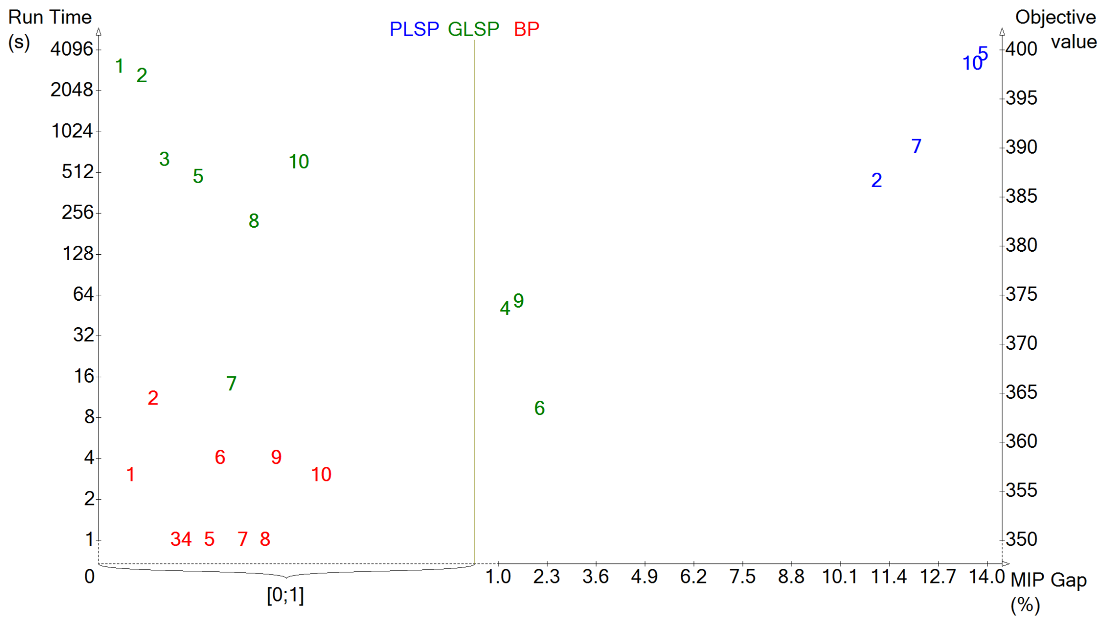

Figure A.10.: *Detailed results of the LC Scenario, problem size M, high capacity load.*

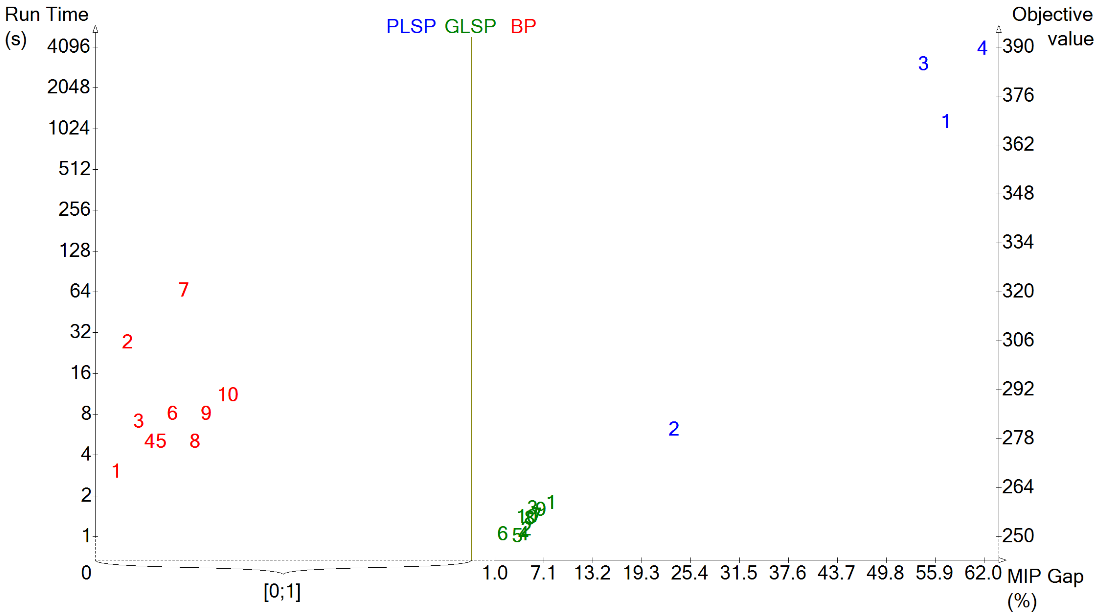

Figure A.11.: Detailed results of the LC Scenario, problem size L, low capacity load.

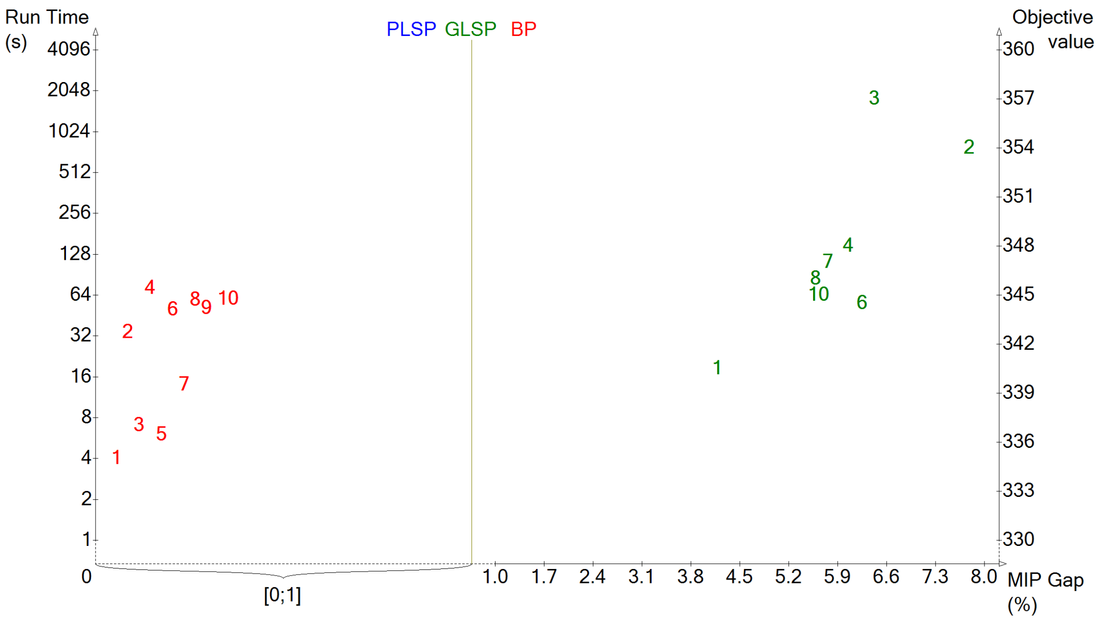

Figure A.12.: Detailed results of the LC Scenario, problem size L, high capacity load.

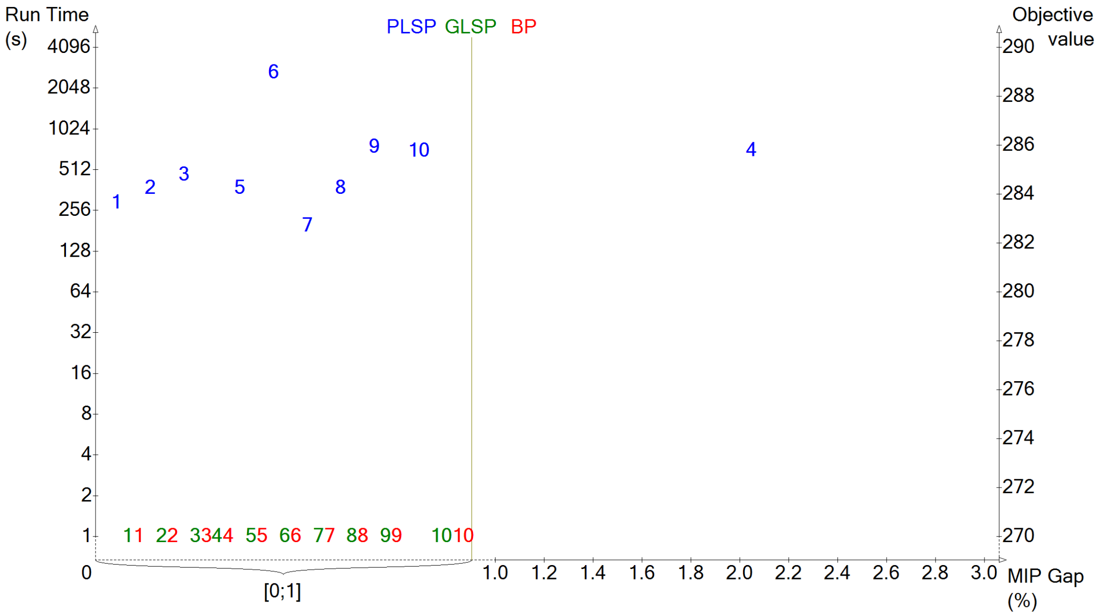

Figure A.13.: Detailed results of the NS Scenario, problem size S, low capacity load.

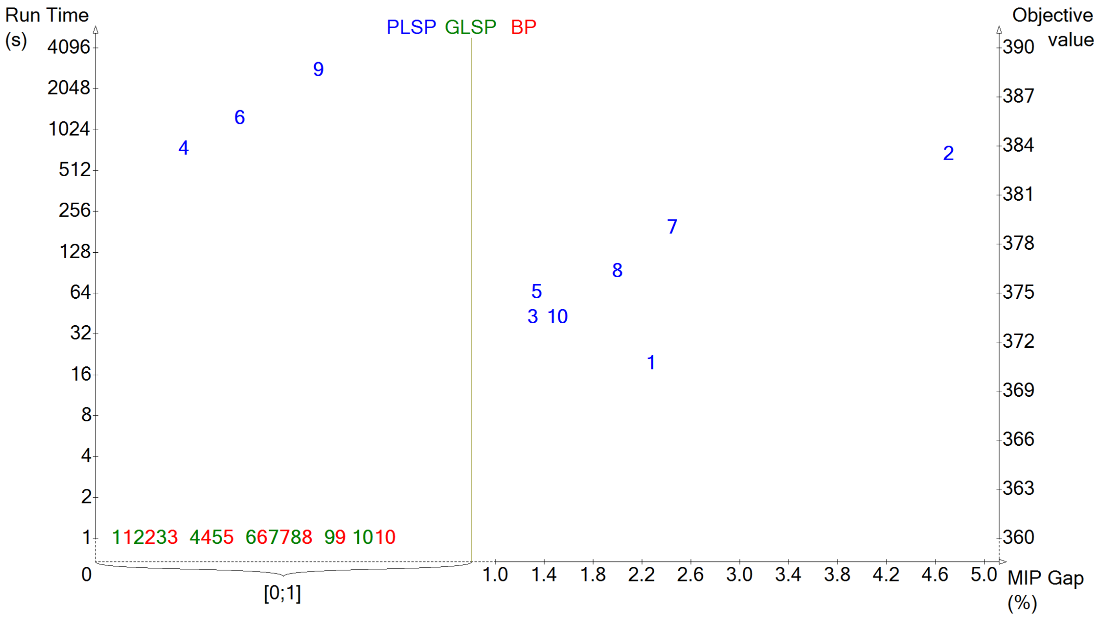

Figure A.14.: Detailed results of the NS Scenario, problem size S, high capacity load.

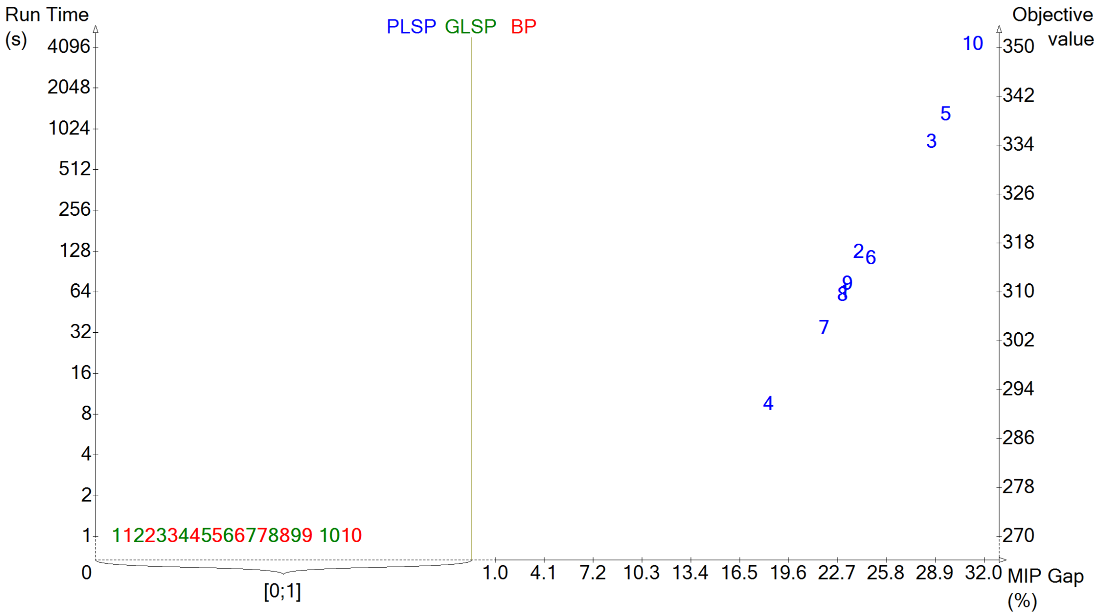

Figure A.15.: *Detailed results of the NS Scenario, problem size M, low capacity load.*

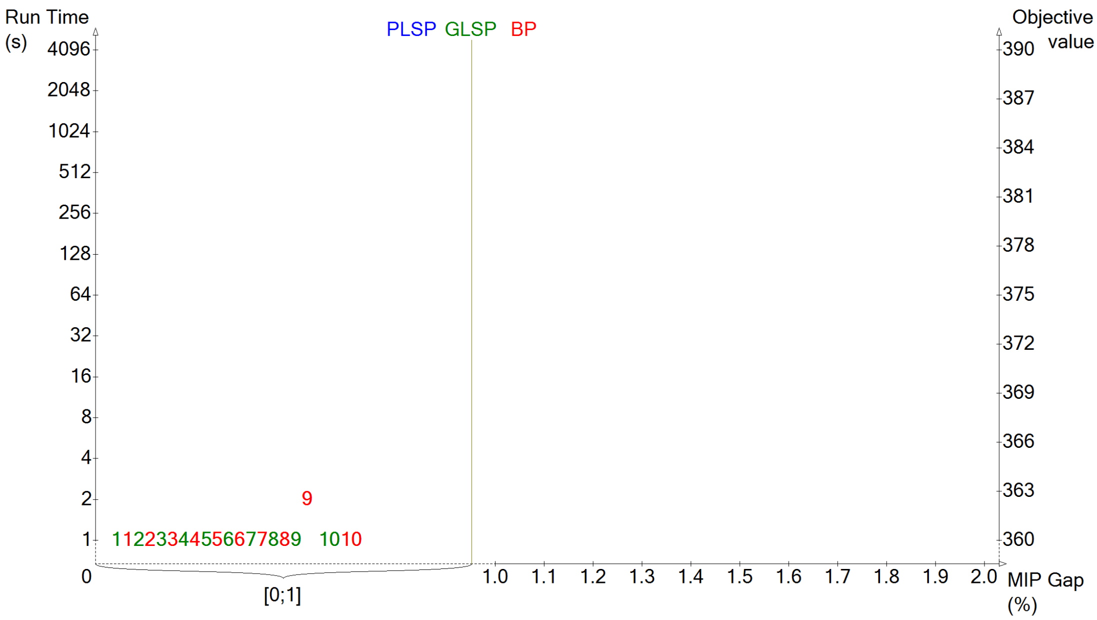

Figure A.16.: *Detailed results of the NS Scenario, problem size M, high capacity load.*

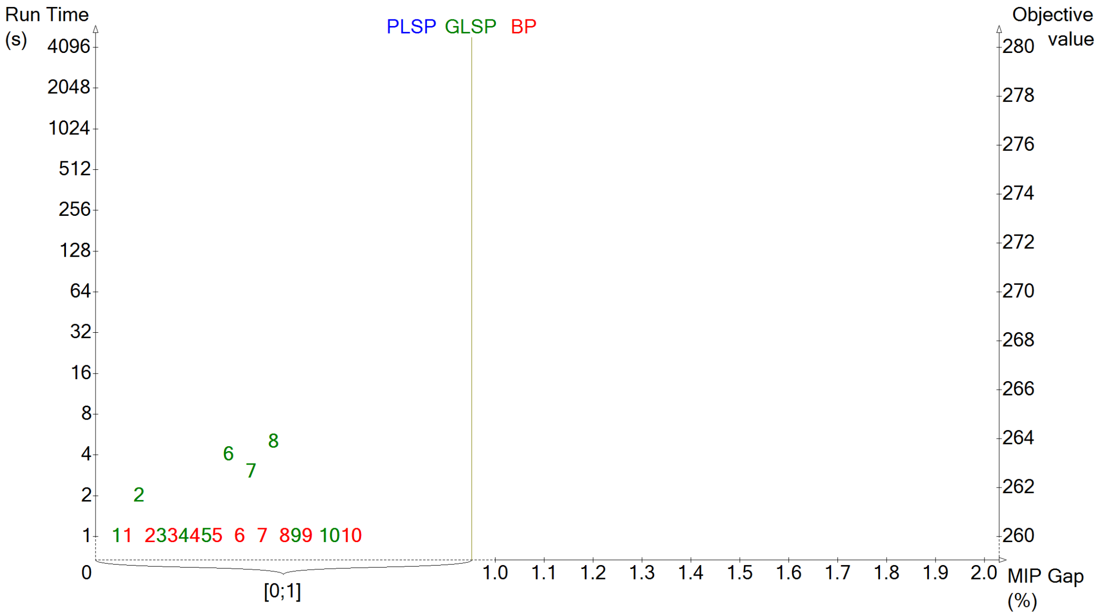

Figure A.17.: *Detailed results of the NS Scenario, problem size L, low capacity load.*

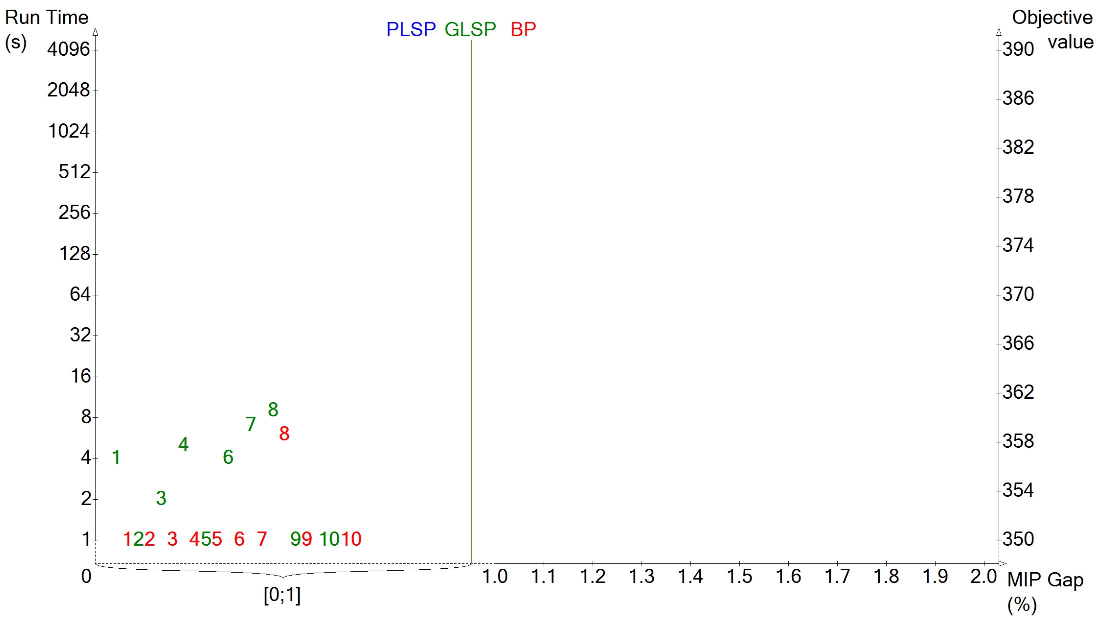

Figure A.18.: *Detailed results of the NS Scenario, problem size L, high capacity load.*

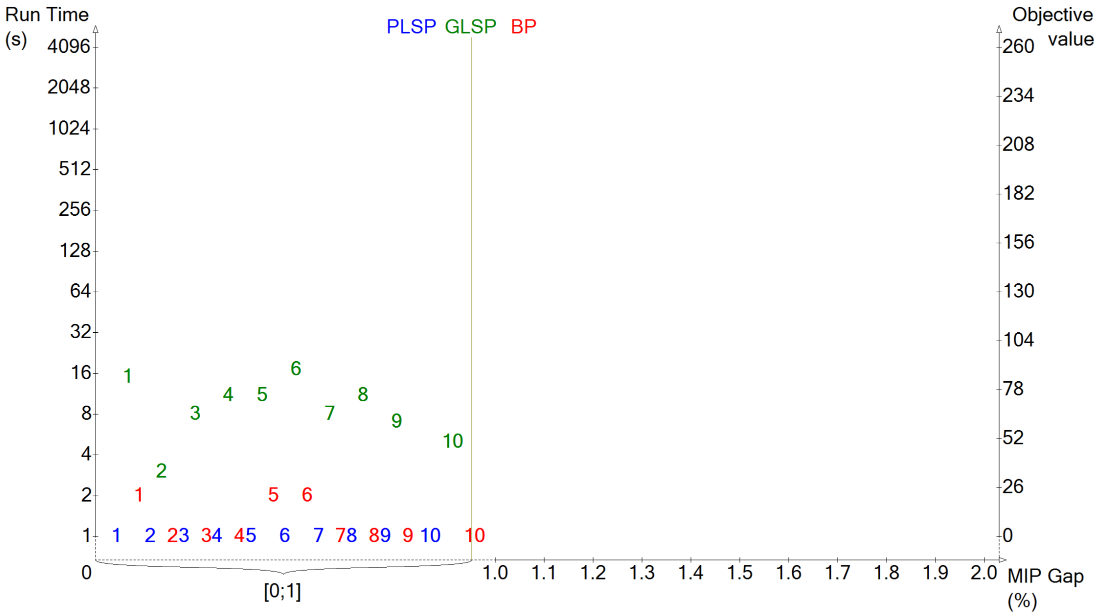

Figure A.19.: Detailed results of the NS Scenario, problem size XL, low capacity load.

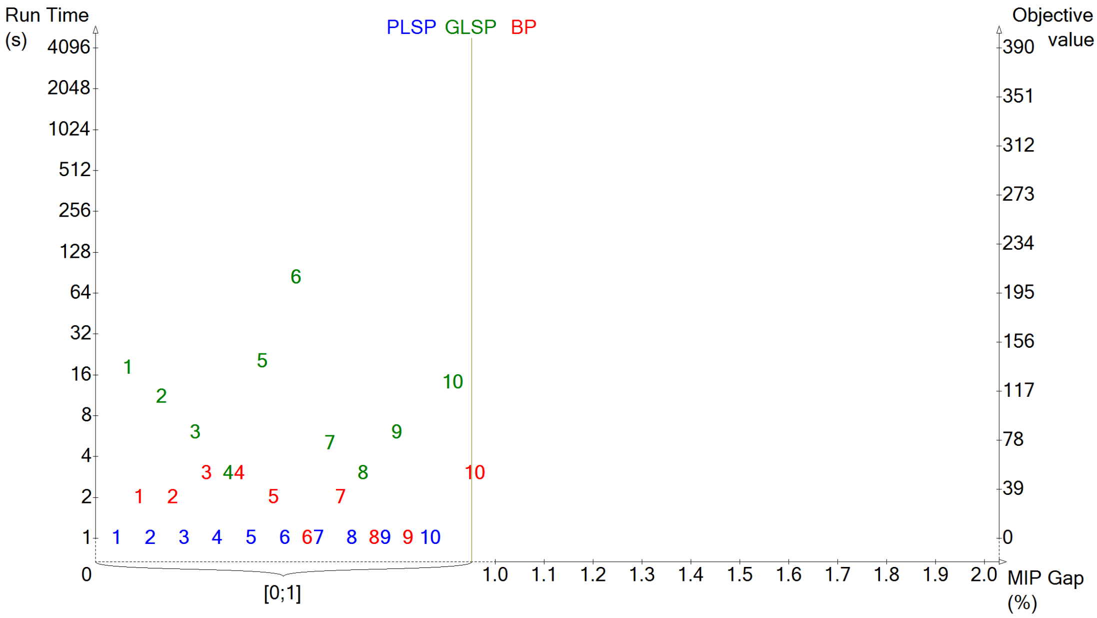

Figure A.20.: Detailed results of the NS Scenario, problem size XL, high capacity load.

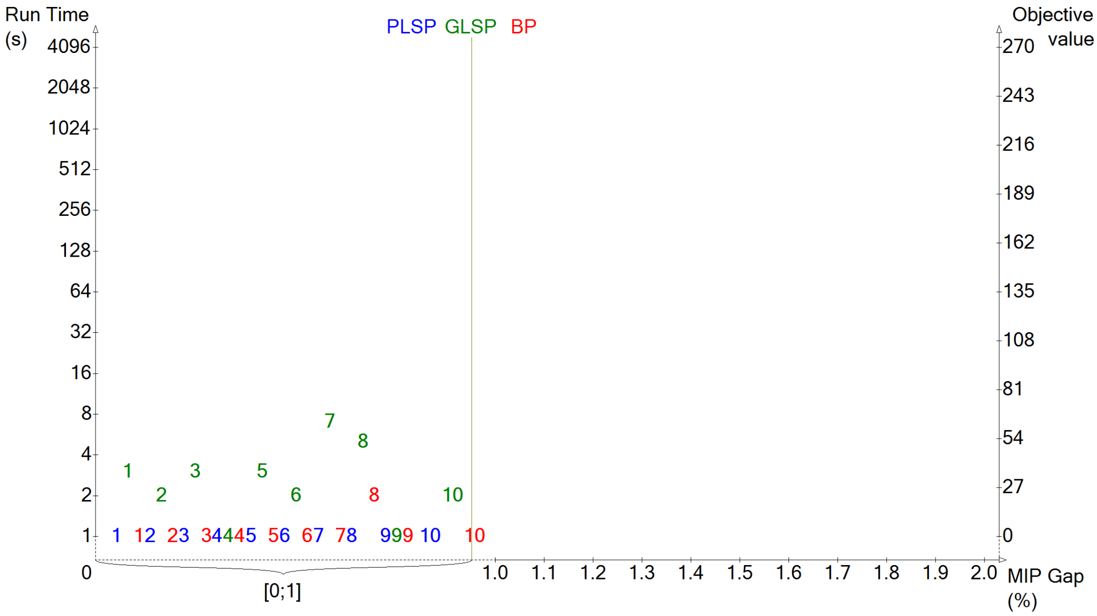

Figure A.21.: *Detailed results of the NS Scenario, problem size LHV, low capacity load.*

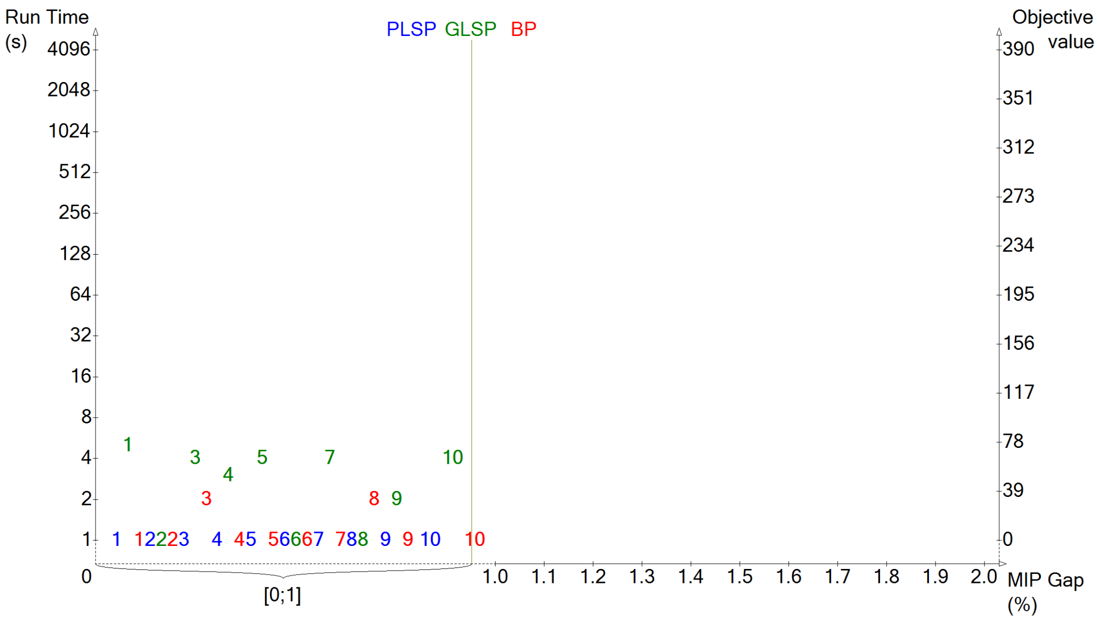

Figure A.22.: *Detailed results of the NS Scenario, problem size LHV, high capacity load.*

Bibliography

A

Akkerman, R. and D.P. van Donk (2008). “Development and application of a decision support tool for reduction of product losses in the food-processing industry”. In: *Journal of Cleaner Production* 16.3, pp. 335–342.

Allahverdi, A., C.T. Ng, T.C. Cheng, and M. Kovalyov (2008). “A survey of scheduling problems with setup times or costs”. In: *European Journal of Operational Research* 187.3, pp. 985–1032.

Almada-Lobo, B., D. Klabjan, M. Antónia Carravilla, and J.F. Oliveira (2007). “Single machine multi-product capacitated lot sizing with sequence-dependent setups”. In: *International Journal of Production Research* 45.20, pp. 4873–4894.

Almeder, C., M. Preusser, and R.F. Hartl (2009). “Simulation and optimization of supply chains: alternative or complementary approaches?” In: *OR Spectrum* 31.1, pp. 95–119.

Amorim, P., H.-O. Günther, and B. Almada-Lobo (2012). “Multi-objective integrated production and distribution planning of perishable products”. In: *International Journal of Production Economics* 138.1, pp. 89–101.

Arbib, C., D. Pacciarelli, and S. Smriglio (1999). “A three-dimensional matching model for perishable production scheduling”. In: *Discrete Applied Mathematics* 92.1, pp. 1–15.

B

Bahl, H.C., L.P. Ritzman, and J.N.D. Gupta (1987). “Determining lot sizes and resource requirements: a review”. In: *Operations Research* 35.3, pp. 329–345.

Barany, I., T.J. van Roy, and L.A. Wolsey (1984). “Strong formulations for multi-item capacitated log-sizing”. In: *Management Science* 30, pp. 1255–1261.

Beasley, J.E. (1995). “Lagrangean relaxation”. In: *Modern heuristic techniques for combinatorial problems.* Ed. by C.R. Reeves. McGraw-Hill, London, pp. 243–303.

Beck, A. (2002). “Automatic Product Identification & Shrinkage: Scoping the Potential”. In: *ECR Europe*.

Bilgen, B. and Y. Çelebi (2013). “Integrated production scheduling and distribution planning in dairy supply chain by hybrid modelling”. In: *Annals of Operations Research*, pp. 1–28. Online available since July 2013, print still pending.

Bilgen, B. and H.-O. Günther (2010). “Integrated production and distribution planning in the fast moving consumer goods industry: a block planning application”. In: *OR Spectrum* 32.4, pp. 927–955.

Bouchriha, H., M. Ouhimmou, and S. D'Amours (2007). "Lot sizing problem on a paper machine under a cyclic production approach". In: *International Journal of Production Economics* 105.2, pp. 318–328.

Bundesvereinigung der Deutschen Ernährungsindustrie (2013). *Annual Report 2012/13*. URL: http://www.bve-online.de. Retrieved September 16, 2013.

Buschkühl, L., F. Sahling, S. Helber, and H. Tempelmeier (2010). "Dynamic capacitated lot-sizing problems: a classification and review of solution approaches". In: *OR Spectrum* 32.2, pp. 231–261.

C

Christou, I.T., A.G. Lagodimos, and D. Lycopoulou (2007). "Hierarchical production planning for multi-product lines in the beverage industry". In: *Production Planning & Control* 18.5, pp. 367–376.

Clark, A., B. Almado-Lobo, and C. Almeder (2011). "Lot sizing and scheduling: industrial extensions and research opportunities". In: *International Journal of Production Research* 49.9, pp. 2457–2461.

Cooper, J., M. Browne, and M. Peters (1994). "European Logistics: Markets, Management and Strategy". In: 2nd ed. Oxford and Cambridge: Blackwell.

D

Dakin, R.J. (1965). "A tree-search algorithm for mixed integer programming problems". In: *The Computer Journal* 8.3, pp. 250–255.

Domschke, W. and A. Drexl (2007). *Einführung in operations research*. 7th ed. Springer, Berlin et al.

Domschke, W., A. Scholl, and S. Voß (1997). *Produktionsplanung: Ablauforganisatorische Aspekte*. 2nd ed. Springer, Berlin et al.

Drexl, A., B. Fleischmann, H.-O. Günther, H. Stadtler, and H. Tempelmeier (1994). "Konzeptionelle Grundlagen kapazitätsorientierter PPS-Systeme". In: *Zeitschrift für betriebswirtschaftliche Forschung* 46.12, pp. 1022–1045.

Drexl, A. and K. Haase (1995). "Proportional lotsizing and scheduling". In: *International Journal of Production Economics* 40.1, pp. 73–87.

Drexl, A. and A. Kimms (1997). "Lot sizing and scheduling: survey and extensions". In: *European Journal of Operational Research* 99.2, pp. 221–235.

E

Evans, J.R. (1985). "An efficient implementation of the Wagner-Whitin algorithm for dynamic lot-sizing". In: *Journal of Operations Management* 5.2, pp. 229–235.

F

Farahani, P., M. Grunow, and H.-O. Günther (2012). "Integrated production and distribution planning for perishable food products". In: *Flexible Services and Manufacturing Journal* 24.1, pp. 28–51.

Ferreira, D., R. Morabito, and S. Rangel (2010). "Relax and fix heuristics to solve one-stage one-machine lot-scheduling models for small-scale soft drink plants". In: *Computers & Operations Research* 37.4, pp. 684–691.

Fink, A. and F. Rothlauf (2006). *Heuristische Optimierungsverfahren in der Wirtschaftsinformatik*. Department of Information Systems 1, University of Mannheim.

Fisher, M.L. (1985). "An applications oriented guide to Lagrangian relaxation". In: *Interfaces* 15.2, pp. 10–21.

Fleischmann, B. (1990). "The discrete lot-sizing and scheduling problem". In: *European Journal of Operational Research* 44.3, pp. 337–348.

Fleischmann, B. (1994). "The discrete lot-sizing and scheduling problem with sequence-dependent setup costs". In: *European Journal of Operational Research* 75.2, pp. 395–404.

Fleischmann, B. and H. Meyr (1997). "The general lotsizing and scheduling problem". In: *OR Spektrum* 19.1, pp. 11–21.

G

Gelders, L.F., J. Maes, and L.N. van Wassenhove (1986). "A branch and bound algorithm for the multi-item single level capacitated dynamic lot-sizing problem". In: Axsäter. *Multistage Production Planning and Inventory Control*. Springer, Berlin et al., pp. 92–108.

Geoffrion, A.M. (1974). "Lagrangean relaxation for integer programming". In: *Mathematical Programming Study* 2, pp. 82–114.

Glover, F. (1986). "Future paths for integer programming and links to artificial intelligence". In: *Computers & Operations Research* 13.5, pp. 533–549.

Goffin, J.L. (1977). "On convergence rates of subgradient optimization methods". In: *Mathematical programming* 13.1, pp. 329–347.

Gomory, R.E. (1958). "Outline of an algorithm for integer solutions to linear programs". In: *Bulletin of the American Mathematical Society* 64.5, pp. 275–278.

Groff, G.K. (1979). "A lot sizing rule for time-phased component demand". In: *Production and Inventory Management* 20.1, pp. 47–53.

Günther, H.-O. (2013). "The block planning approach for lot-sizing and scheduling in make-and-pack production systems". In: *Working paper*.

Günther, H.-O., M. Grunow, and U. Neuhaus (2006). "Realizing block planning concepts in make-and-pack production using MILP modelling and SAP APO©". In: *International Journal of Production Research* 44.18-19, pp. 3711–3726.

Günther, H.-O. and H. Tempelmeier (2012). *Produktion und Logistik*. 9th ed. Springer, Berlin et al.

H

Haase, K. (1994). *Lotsizing and scheduling for production planning.* Springer, Berlin et al.

Haase, K. and A. Kimms (2000). “Lot sizing and scheduling with sequence-dependent setup costs and times and efficient rescheduling opportunities”. In: *International Journal of Production Economics* 66.2, pp. 159–169.

Harris, F.W. (1913). “How many parts to make at once”. In: *Operations Research* 38, pp. 947–950.

Henderson, D., S.H. Jacobson, and A.W. Johnson (2003). “The theory and practice of simulated annealing”. In: *Handbook of metaheuristics.* Ed. by F. Glover and G.A. Kochenberger. Springer, Berlin et al., pp. 287–319.

Herpers, S. (2009). “Dynamische Losgrößenplanung bei stochastischer Nachfrage”. PhD thesis. Kölner Wissenschaftsverlag, Cologne.

J

Jans, R. and Z. Degraeve (2008). “Modeling industrial lot sizing problems: a review”. In: *International Journal of Production Research* 46.6, pp. 1619–1643.

K

Karimi, B., S.M.T Fatemi Ghomi, and J.M. Wilson (2003). “The capacitated lot sizing problem: a review of models and algorithms”. In: *Omega* 31.5, pp. 365–378.

Kirkpatrick, S., D. Gelatt Jr., and M.P. Vecchi (1983). “Optimization by simulated annealing”. In: *Science* 220.4598, pp. 671–680.

L

Liu, X. and Y. Tu (2008). “Production planning with limited inventory capacity and allowed stockout”. In: *International Journal of Production Economics* 111.1, pp. 180–191.

Lütke-Entrup, M. (2005). “Advanced planning in fresh food industries: integrating shelf life into production planning (contributions to management science)”. PhD thesis. Physica-Verlag, Heidelberg, pp. 58–63.

Lütke-Entrup, M., H.-O. Günther, P. van Beek, M. Grunow, and T. Seiler (2005). “Mixed-integer linear programming approaches to shelf-life-integrated planning and scheduling in yoghurt production”. In: *International Journal of Production Research* 43.23, pp. 5071–5100.

M

Meyr, H. (1999). *Simultane Losgrößen- und Reihenfolgeplanung für kontinuierliche Produktionslinien: Modelle und Methoden im Rahmen des Supply Chain Management.* Tech. rep. Darmstadt Technical University, Department of Business Administration, Economics and Law, Institute for Business Studies (BWL).

Meyr, H. (2000). "Simultaneous lotsizing and scheduling by combining local search with dual reoptimization". In: *European Journal of Operational Research* 120.2, pp. 311–326.

N

Nissen, V. (1994). *Evolutionäre Algorithmen: Darstellung, Beispiele, betriebswirtschaftliche Anwendungsmöglichkeiten.* Deutscher Universitäts-Verlag, Wiesbaden.

P

Pochet, Y. and L.A. Wolsey (2006). *Production planning by mixed integer programming.* Springer, Berlin et al.

R

Reeves, C.R. (1995). *Modern heuristic techniques for combinatorial problems.* McGraw-Hill, London, p. 6.

S

Sahling, F., L. Buschkühl, H. Tempelmeier, and S. Helber (2009). "Solving a multi-level capacitated lot sizing problem with multi-period setup carry-over via a fix-and-optimize heuristic". In: *Computers & Operations Research* 36.9, pp. 2546–2553.

Salomon, M., M.M. Solomon, L.N. van Wassenhove, Y. Dumas, and S. Dauzère-Pérès (1997). "Solving the discrete lotsizing and scheduling problem with sequence dependent set-up costs and set-up times using the travelling salesman problem with time windows". In: *European Journal of Operational Research* 100.3, pp. 494–513.

Sandi, C. (1979). "Subgradient optimization". In: *Combinatorial optimization*, pp. 73–91.

Scholl, A. (2008). "Modellierung logistischer Systeme". In: Arnold, D., H. Isermann, A. Kuhn, K. Furmans, and H. Tempelmeier. *Handbuch Logistik.* 3rd ed. Springer, Berlin et al., p. 36. [A 2.1 – A 2.2].

Schöpperl, A. (2013). "Evolutionary production planning and scheduling". PhD thesis. Technische Universität Berlin, Fakultät VII – Wirtschaft und Management.

Shum, P. and G. Lin (2003). “Knowledge and Innovation Culture as Determinants of Financial Performance in New Product Development”. In: *Australia: The International Journal of Knowledge, Culture and Change Management* 6, pp. 95–108.

Silver, E.A. and H. Meal (1973). “A heuristic for selecting lot size requirements for the case of a deterministic time-varying demand rate and discrete-opportunities for replenishment”. In: *Production and Inventory Management* 14, pp. 64–74.

Sox, C.R. (1997). “Dynamic lot sizing with random demand and non-stationary costs”. In: *Operations Research Letters* 20.4, pp. 155–164.

Statista GmbH (2013). *Daten & Fakten zur Konsumgüterindustrie*. URL: `http://de.statista.com/statistik/faktenbuch/363/a/branche-industrie-markt/sonstige/konsumgueterindustrie/`. Retrieved July 31, 2013.

Suerie, C. (2006). “Modeling of period overlapping setup times”. In: *European Journal of Operational Research* 174.2, pp. 874–886.

T

Tarim, S.A. and B.G. Kingsman (2004). “The stochastic dynamic production/inventory lot-sizing problem with service-level constraints”. In: *International Journal of Production Economics* 88.1, pp. 105–119.

Tempelmeier, H. (2006). *Inventory management in supply networks: problems, models, solutions*. Norderstedt: Books on Demand, pp. 157–164.

Tempelmeier, H. (2008). *Material-Logistik*. Springer, Berlin et al.

Tunc, H., A.K. Kullic, A.S. Tarim, and B. Eksioglu (2013). “A simple approach for assessing the cost of system nervousness”. In: *International Journal of Production Economics* 141.2, pp. 619–625.

V

Van Dam, P., G.J.C. Gaalman, and G. Sierksma (1999). “Production scheduling across multi-machine packaging lines for the pre-scheduling level”. In: *International Journal of Production Research* 37.16, pp. 3619–3641.

Van Donk, D.P. (2001). “Make to stock or make to order: The decoupling point in the food processing industries”. In: *International Journal of Production Economics* 69.3, pp. 297–306.

W

Wagelmans, A., S. van Hoesel, and A. Kolen (1992). “Economic lot sizing: An O(n log n) algorithm that runs in linear time in the Wagner-Whitin case”. In: *Operations Research* 40.1, pp. 145–156.

Wagner, H.M. and T.M. Whitin (1958). “Dynamic version of the economic lot size model”. In: *Management Science* 5.1, pp. 89–96.

Y

Yu, B., Z.-Z. Yang, and Y. Baozhen (2009). "An improved ant colony optimization for vehicle routing problem". In: *European Journal of Operational Research* 1996.1, pp. 171–176.

Z

Zhu, X. and W.E. Wilhelm (2006). "Scheduling and lot sizing with sequence-dependent setup: A literature review". In: *IIE Transactions* 38.11, pp. 987–1007.

Zipkin, P.H. (1991). "Computing optimal lot sizes in the economic lot scheduling problem". In: *Operations Research* 39.1, pp. 56–63.

Zotteri, G. (2013). "An empirical investigation on causes and effects of the Bullwhip-effect: Evidence from the personal care sector". In: *International Journal of Production Economics* 143.2, pp. 489–498.