A method for simultaneous production and order planning in a cooperative supply chain relationship with flexibility contracts

Tobias Rust, Daniel Brüggemann
University of Paderborn
Department Business Information Systems
Paderborn, Germany
Tobias.Rust@hni.upb.de
Daniel.Brueggemann@hni.upb.de

Abstract

Nowadays a high planning flexibility is required due to customer-driven markets and a rising number of product variants. In context of a supply chain this flexibility is also important for the procurement of input parts from external suppliers. In this work a method for simultaneous production and order planning in a cooperative supply chain relationship is presented. The method is part of an overall approach which tries to coordinate order quantities of an input part using flexibility contracts. Thereby the input part is delivered by two suppliers for which frame contracts with quantity and quota agreements exist. The objective is to guarantee these quantities in the frame contract horizon at minimum costs. A fine planning approach is presented which plans the production of the manufacturer simultaneously with the orders for the considered input part under consideration of priory defined flexibility corridors.

1. Introduction

Customer-driven markets along with a rising number of product variants result in high planning uncertainty and an increasing demand of flexibility for manufacturers. To deliver products in time and at low costs is a challenging task. When essential input parts of the end products have to be purchased this also leads to the need of flexibility in buyer-supplier relationships. In the best-case a relationship is cooperative and regards both buyer and supplier circumstances to reach an overall optimum for the participants.

In traditional production planning first the production is planned and afterwards the required input is calculated and ordered based on the production plan. By disregarding the interdependencies between the single planning steps a potential to save costs is wasted. A saving could be achieved by simultaneous planning which nevertheless also includes some problems. The first problem is that the effort to reach good solutions can become very high due to the increasing complexity of simultaneous planning. The second problem is the unwillingness to share data in a buyer-supplier relationship.

In this work a method for simultaneous production and order planning in a cooperative supply chain relationship is presented. To handle the complexity problem the order planning aspect is limited to one input part and the problem is solved heuristically. The method is part of an overall approach which tries to coordinate quantities with flexibility contracts. Thereby the input part is delivered by two suppliers for which frame contracts with quantity and quota agreements exist. The aim is to guarantee these quantities in the frame contract horizon at minimum costs.

The work is structured as follows. Chapter 2 introduces the considered planning scenario and the reference to an overall approach where the presented work is part of. Chapter 3 ranges the approach in the theory regarding production planning and quantity coordination. Chapter 4 defines the planning as a mixed integer model. The heuristic solution for the planning problem is explained in chapter 5 before results are shown in chapter 6. Finally chapter 7 gives a summary and an outlook on further work.

2. Planning context

2.1. Planning scenario

The supply chain scenario in Figure 1 builds the basis for the overall approach. The customer is a manufacturer which produces finished goods and faces uncertain demand of end consumers. He produces variants on one production line which can have a bottleneck part as input. This part is included in some of the end products, but not in all. All other input parts are disregarded, because it is assumed that they can easily be produced or procured. The bottleneck part is
procured from two suppliers for which frame contracts with quota agreements exist. The problem can easily be extended by adding suppliers or bottleneck parts.

**Figure 1. Considered supply chain scenario**

A practical example for this scenario could be a car manufacturer who orders headlights. These headlights cannot be produced by the manufacturer and have to be procured by two suppliers for which quota agreements exist. For sure a car manufacturer needs other important input parts, but we think that an overall cost reduction can already be achieved by intelligent planning of specific input parts.

2.2 Reference to overall approach

The introduced method is part of an overall approach which aims to result in lower costs for the considered supply chain section.

For this a framework is necessary to plan the procured quantities at lowest possible costs for the customer and the suppliers. The framework contains cooperative arrangements which control the behaviour of the involved parties regarding order quantities and prices. Two of these arrangements are described in [1].

According to the flexibility of orders the customer and its suppliers have competing objectives. The customer faces end customer demand which is usually known approximately for near future, but unconfident for farther time periods. Because of that uncertainty the customer needs the possibility to change its orders according to the demand he has to fulfill. For his purposes this flexibility should be as high as possible, but can decrease when a time period gets nearer and forecasts are replaced by real orders. On the other hand the supplier wants to have as accurate order plans as possible to avoid overcapacities and large stocks.

To handle this problem flexibility corridors are introduced (see Figure 2). The figure shows an example of flexibility corridors for a specific time segment and its changing in time. In this example the predicted order quantity does not change in time, so the customer can just concretize the corridors around the predicted quantity. If this quantity would change the customer would try to concretize corridors around the predicted quantities, but inside the old corridors.

**Figure 2. Flexibility corridors**

This concept should increase the suppliers’ safety in planning. Thereby the suppliers should be able to produce at lower costs which they can give back partly to their customers by offering lower prices in some time periods. At the end all parties could have a benefit.

The customer should be encouraged to communicate his future orders early in time at high precision. For that he has to pay penalty costs for leaving the planned corridors in a next rolling-horizon planning. Thereby two objectives are achieved:

- The supplier gets paid for unexpected adaption of quantities
- The customer is willing to plan and coordinate the flexibility corridors as good as possible

More information about flexibility corridors and a method which plans the corridors can be found in [1].

The planning is done in a rolling way and consists of a rough planning and a fine planning which concretizes the rough planning (see Figure 3).

**Figure 3. Rolling rough and fine planning**

The calculation of flexibility corridors is part of the rough planning and plans quantities for time segments which could be for example months. The corridors are aggregated quantities which are an input to the fine planning which plans concrete orders for periods which are for example days or weeks.

The approach presented in this work is placed in the fine planning part of the overall approach. The customer tries to plan his production and orders simultaneously regarding the prior defined flexibility
corridors and other influences with the objective of cost minimization.

These orders are the input for the last step in the approach, a profit-sharing step where the suppliers are able to propose alternative delivery schedules and balance payments.

3. Classification in theory

3.1. Production planning

In classic production planning and control systems the planning is done successively. In a first step the production is planned under the assumption of unlimited capacity and the availability of all input products. Consecutively the capacities are checked and the production is planned. According to this production plan orders are determined for in-house production or for external suppliers. A good overview of production planning can be found for example in [2] and [3].

Some Advanced planning systems (APS) use a simultaneous approach to regard the different dependencies. The problem of simultaneous planning is the high complexity of real world problems which make a solution in appropriate time often impossible. One possibility to solve this complexity problem is to reduce or divide the overall model into partial models. A detailed description of APS systems can be found in [4].

In context of this work an approach is presented which does a simultaneous planning for the production and the ordering of one input factor necessary for the production. To reduce the complexity this is done in a big bucket model where larger time periods are used so that some events can be planned in a single time period without planning the sequence of these events. For further information about big bucket models see for example [5].

3.2. Coordinating mid-term quantities

Frame contracts are often used to determine rights and responsibilities of supply chain partners. A lot of different contract types deal with price and capacity agreements, buy back responsibilities, quantity flexibilities and other topics. An overview of the different contract types can be found in [6] and [7].

Quantity flexibility (QF) contracts are widely used in praxis and build an interesting concept for the customer-supplier-relation [8]. They give the supplier a higher planning assurance by defining quantities at least within a given bandwidth. The customer is for example a manufacturer who faces uncertain demand and provides the parts supplier with a replenishment schedule which becomes the parts suppliers release schedule. QF contracts are defined between each node. They are parameterized by \((\alpha, \beta)\) which places bounds on how the manufacturer can revise the replenishment schedule. All parameters are defined over time whereby the flexibility intervals should get progressively smaller the nearer a period gets. A rolling horizon planning with updated estimates is repeated each period.

According to [9] both parties will get a benefit and the QF contracts can be treated as an answer to the well-known bullwhip effect. The supplier has to cover any requests up to the maximum of the flexibility bounds giving the customer a safety buffer. On the other side the customer agrees on the lower bound as a minimum quantity and therefore gives the supplier a higher safety in planning.

The flexibility contracts used in the approach of this work are very similar to the QF contracts. Some additional rules were defined to coordinate the quantities according to different time areas. These time areas are described in [1].

3.3. Coordinating short-term quantities

In the mid-term planning only aggregated quantities are planned which have to be concretized in short-term planning. Even if the mid-term quantities were coordinated well the exact orders can be suboptimal for the suppliers if their situations are disregarded.

In literature an integrated planning and procurement is often proposed to reach a good overall solution. This integrated planning coordinates the production policy of a supplier with the ordering policy of a customer and can either be done simultaneously or successively. Simultaneous approaches can be found for example in [10] and [11]; successive approaches in [12] and [13]. A list of more approaches can be found in [14] and [15]. These approaches reach good results, but are not well applicable for this work.

The difference of the approach of our work to integrated planning and procurement is that the customer plans his production and the required orders simultaneously considering the flexibility corridors. The simultaneous aspect is between the customers’ production and his procurement and not between the customers’ procurement and the suppliers’ production.

In this work the fine planning of the customer is influenced by different prices which are offered by the suppliers for single time periods. So the suppliers can affect the order behaviour of the customer under consideration of their capacity situation. A final profit-sharing step should give the suppliers the possibility to achieve another cost reduction by offering alternative production schedules.
Using this approach the customer is able to plan its production and its procurement simultaneously which should result in cost reduction. At the same time the cost structure of the suppliers is regarded by the prices for single periods and the final step of profit-sharing. This is done in single steps to reduce the complexity of the overall problem.

4. Mathematical model

After introducing some important assumptions the parameters and variables are listed before the model is introduced.

Important assumptions for the model are:
- There are two different kind of products: the bottleneck products (BPR) and the non-bottleneck products, which do not contain this part.
- Every BPA and BPR generates the same storage costs.
- Each BPR contains one identical BPA.
- Delivery times are zero (will be added in future work).
- Setup cost are sequence independent.

Table 1 and Table 2 list the parameters and variables which are considered in the model.

Table 1: Parameters of mathematical model

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T$</td>
<td>Number of time segments</td>
</tr>
<tr>
<td>$P$</td>
<td>Number of periods</td>
</tr>
<tr>
<td>$p^T$</td>
<td>Number of periods per time segment</td>
</tr>
<tr>
<td>$Pr$</td>
<td>Set of products</td>
</tr>
<tr>
<td>$S$</td>
<td>Set of suppliers</td>
</tr>
<tr>
<td>$c_{var}^{p,s}$</td>
<td>Price of supplier $s$ for one BPA in period $p$</td>
</tr>
<tr>
<td>$c_{fix}^{p,s}$</td>
<td>Fixed order costs of supplier $s$ in period $p$</td>
</tr>
<tr>
<td>$max_{p,s}^{BPA}$</td>
<td>Maximum of deliverable BPA units for supplier $s$ in period $p$</td>
</tr>
<tr>
<td>$max_{t,s}^{BPA}$</td>
<td>Maximum of deliverable BPA units for supplier $s$ in time segment $t$</td>
</tr>
<tr>
<td>$LB_{t,s}^{BC}$</td>
<td>Lower bound of flexibility corridor of supplier $s$ in time segment $t$</td>
</tr>
<tr>
<td>$UB_{t,s}^{BC}$</td>
<td>Upper bound of flexibility corridor of supplier $s$ in time segment $t$</td>
</tr>
<tr>
<td>$c_j^{BPA}$</td>
<td>Punishment costs for violating the flexibility corridor of supplier $s$ in time segment $t$ per unit</td>
</tr>
<tr>
<td>$c_D$</td>
<td>Delay costs per period and BPR unit</td>
</tr>
<tr>
<td>$bpr_{p,pr}$</td>
<td>Binary parameter indicating if product $pr$ is a BPR (1) or a non-BPR (0)</td>
</tr>
<tr>
<td>$d_{p,pr}$</td>
<td>Customer demand for product $pr$ in period $p$</td>
</tr>
<tr>
<td>$c_{SI}^{BPA}$</td>
<td>Storage capacity for BPA’s</td>
</tr>
<tr>
<td>$c_{SO}^{BPA}$</td>
<td>Storage capacity for BPR’s</td>
</tr>
</tbody>
</table>

Table 2: Variables of the mathematical model

<table>
<thead>
<tr>
<th>Variable</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>$l_{s,p,pr}$</td>
<td>Lot size of product $pr$ in period $p$</td>
</tr>
<tr>
<td>$d_{p,pr}$</td>
<td>Number of delivered units of product $pr$ in period $p$</td>
</tr>
<tr>
<td>$d_{p,pr}^{delay}$</td>
<td>Number of delayed units of product $pr$ in period $p$</td>
</tr>
<tr>
<td>$o_{p,s}$</td>
<td>Ordered BPA amount at supplier $s$ in period $p$</td>
</tr>
<tr>
<td>$s_{BPA}^{upper}$</td>
<td>Number of BPA’s stored in period $p$</td>
</tr>
<tr>
<td>$s_{BPR}^{upper}$</td>
<td>Number of stored units of product $pr$ in period $p$</td>
</tr>
<tr>
<td>$p_l^{p,s}$</td>
<td>Binary variable indicating if an order at supplier $s$ is placed in period $p$ (1) or not (0)</td>
</tr>
<tr>
<td>$d_{l,s}^{UB}$</td>
<td>Amount of BPA’s exceeding the upper bound of the flexibility corridor of supplier $s$ in time segment $t$</td>
</tr>
<tr>
<td>$d_{l,s}^{LB}$</td>
<td>Amount of BPA’s fall below the lower bound of the flexibility corridor of supplier $s$ in time segment $t$</td>
</tr>
<tr>
<td>$s_{e,p,pr}$</td>
<td>Binary variable indicating if a setup is required to produce product $pr$ in period $p$ (1) or not (0)</td>
</tr>
</tbody>
</table>

The aim of the described method is the cost minimization of the customer. The goal function is as follows:

\[
\text{Minimize } Z = \sum_{p=1}^{P} \left( c_{SI}^{BPA} \cdot s_{p,pr}^{BPA} + \sum_{p \in Pr} \left( c_{SO}^{BPA} \cdot s_{p,pr}^{BPA} + c_D \cdot d_{p,pr}^{delay} + c_{SC} \cdot \sum_{s \in S} (c_{var}^{p,s} \cdot o_{p,s} + c_{fix}^{p,s} \cdot p_l^{p,s}) \right) + \sum_{t=1}^{T} \left( \sum_{s \in S} (d_{l,s}^{UB} \cdot c_{l,s}^{p} + d_{l,s}^{LB} \cdot c_{l,s}^{p}) \right) \right) \]

(1)

The following costs are regarded:
- Storage costs for BPA’s and BPR’s
- Costs for delayed delivery
- Setup costs
- Variable and fixed order costs
- Costs for violating the flexibility corridors

To get a valid solution the following constraints have to be fulfilled:
Constraint (2) initializes the recursively defined variables with zero and the following two constraints limit the storage utilization to the capacity in each period. Constraint (5) and (6) limit the amount of BPA’s ordered at the particular supplier to the maximum limit of the periods respectively the time segments. Constraint (7) assures that the required production capacity never exceeds the available capacity in any period. The next two constraints are important for satisfying the demands. The first one enforces that each demand can be satisfied by the product units which have been produced up to the particular demand period, if delayed delivery is forbidden. Otherwise this constraint is ignored. The other constraint has a similar structure but considers the whole planning horizon. Thereby it is guaranteed that every demand is satisfied at the end of the last time segment if a valid solution can be found. Constraint (10) satisfies the similar function for the delivery variables. The next constraint forces the value of the binary place-variable according to the orders. Constraint (12) ensures that the available volume of BPA units is high enough for the planned production in each period. The next two constraints are the stock balance constraints for both storage types. Constraint (15) and (16) are important for determining the variance of the order amount from the flexibility corridors. Constraints (17) and (18) are necessary for setting the value of the variables counting the production delay to the accurate values. Constraint (19) sets the binary setup variable to one, if at least one product is produced and sets an upper bound for the lot size.

5. The method

5.1. Main idea

The main objective of the introduced method is to find a feasible and good solution concerning the resulting overall costs by following iterative steps. The algorithm consists of a constructive part and several improving methods whereas some of these methods are executed during the constructive calculations and some of them try to improve the result after a first solution for the given problem has been found. Due to page restrictions only the constructive part is described.

The idea of the method is to start with an empty plan and always add the best lot to the production plan until all demands are fulfilled. Thereby the
determination of the lot considers all relevant information like the supplier and the supply period in case of a bottleneck product.

To clearly define the information which is needed to describe the manufacturing of one product the term production is introduced. A production has to contain the following information:

1. Supply period of the BPA’s
2. Production period
3. Demand period
4. Produced product
5. Supplier for the BPA’s
6. Lot size

The operations are always the same: In each step the method tries to find a production which results in minimal costs in the particular situation. As it has been described in part four, there are several different costs which have to be considered simultaneously, like for example storage, delay or setup costs. To afford the presented process of choosing the locally best production in an iterative way, there must be a way to estimate the costs per product which will expectedly occur by adding the chosen tuple to the resulting plan. Furthermore there must be a data structure to generate and store the possible productions and to determine the particular best one in an efficient way. This data structure is a periodically updated dice which contains all available productions and the cost estimations. The next section will describe this structure in detail.

5.2. Data structure

The data structure to store and analyse the possible productions in the particular situation is a three-dimensional structure, called cost dice (see Figure 4).

Figure 4. Cost dice

The dimensions are the three different temporal factors, which influence the planning and implementation of the production plan:

1. Supply period
2. Production period
3. Demand period

Thereby the structure can store a production and its relevant values for each combination of the three different timestamps. The dice contains dice elements, which store the remaining items of the production tuple. To keep the cost calculation simple the produced product is not determined and stored during the calculation process but during the production choice in the constructive algorithm. Therefore the dice element simply stores if the particular estimated cost value does assume the production of a BPR or a non-BPR. This simplification to increase the efficiency of the method is valid because the costs of all BPR’s and non-BPR’s are similar according to the assumptions in chapter 4.

Planning a large amount of periods consequently leads to a large and complex data structure, which might be inefficient to handle. Because of this the dice is cut to a manageable size by removing illegal or useless elements.

5.3. Calculation and update of cost dice values

To always determine the best production in the particular situation the values of the dice elements must be always up to date. The following sections will explain how the estimated costs of a dice value are calculated and when they have to be updated.

5.3.1 Calculation of cost dice values

The cost value of the dice elements always represents the costs per product which might occur if a single element of the particular production is added to the result. Before calculating the costs the algorithm has to perform some checks to determine if a production of the dice element is still possible. This is necessary because some elements can become impossible, because other productions were planned before which influence for example storage or production capacity. If at least one condition is violated the estimated cost value can be set to $\infty$ without performing any more calculations. Otherwise all different costs must be examined for the particular production.

In the following explanations the calculation of the costs for the supply period $s$ lying in time segment $t$, the production period $p$ and the demand period $d$ is assumed. The different costs are considered:

1. Punishment costs for violating the flexibility corridor of the supplier

   The punishment costs $c^p_{s,p,d}$ are the minimum of the punishment costs $c^p_{s,p,d,sup}$ of all known suppliers in the particular time segment:

   \[
   c^p_{s,p,d} = \min\left\{ c^p_{s,p,d,sup} \mid sup \in S \right\}
   \]  \hspace{1cm} (20)
\( c_{s,p,d,\text{sup}}^P = \{ \)
\[
-c_{t,\text{sup}}^P, \text{if } \sum_{s=(t-1)}^{t \cdot P} (o_{s,\text{sup}}) < Lb_{t,\text{sup}}^{FC} \\
-c_{t,\text{sup}}^P, \text{if } \sum_{s=(t-1)}^{t \cdot P} (o_{s,\text{sup}}) \geq Ub_{t,\text{sup}}^{FC} \\
0, \text{otherwise}
\]

The punishment costs are a saving if an order would help to reach the flexibility corridor. Otherwise the costs are zero if the flexibility corridor has already been reached or they are positive if the upper bound of the sector has been exceeded.

2. Storage costs
The ordered BPA’s as well as the produced BPR’s have to be stored which leads to the storage costs \( c^S \) of the production:
\[
c^S = (p-s) \cdot c^{SI} + \max(0, (d-p)) \cdot c^{SO} \tag{22}
\]

3. Fixed order costs
The fixed order costs \( c_{s,p,d}^F \) are general costs and not applied to one single product unit. Therefore the fixed order costs are unitized to one product unit by dividing them through the estimated amount \( AM_{s,p,d} \) of produced product units. How this amount is determined will be explained in section 5.4.2.
\[
c^F_{s,p,d} = \frac{c_{s,\text{sup}}^{\text{fix}}}{AM_{s,p,d}} \tag{23}
\]

4. Variable order costs
The variable order costs are the prices of the BPA’s. Because of the goal of cost minimization the cheapest supplier is chosen and the variable costs \( c^V_{s,p,d} \) are consequently the minimum of all available prices:
\[
c^V_{s,p,d} = \min\{c^\text{var}_{s,\text{sup}} | \text{sup} \in S\} \tag{24}
\]

5. Delay costs
If delayed delivery of the products to the customers is permitted a production might lead to delay costs \( c^D_{s,p,d} \):
\[
c^D_{s,p,d} = \max(0, (d-p)) \cdot c^D \tag{25}
\]

6. Setup costs
As it has been already mentioned the setup costs \( c^S_{s,p,d} \) are sequence independent and therefore easy to quantify. The only problem is that the algorithm does not know which product will be produced when this production is chosen so the estimated setup costs are determined by calculating the average setup costs of all products:
\[
c^S_{x,p,d} = \frac{\sum_{\text{Pr} \in \text{Pr}} c^S_{x_{\text{pr}},p,d}}{|\text{Pr}|} \tag{26}
\]

After calculating the value of the different costs the estimated cost value \( c_{s,p,d} \) of the dice element can be determined:
\[
c_{x,p,d} = c^P_{x,p,d} + c^S + c^F_{x,p,d} + c^V_{x,p,d} + c^D_{x,p,d} + c^S_{x,p,d} \tag{27}
\]

5.3.2 Update of cost dice values after adding a production to the result

After adding a production of the cost dice to the result there might be the need to update several dice elements. This update depends on the following conditions.

1. Used dice element
   The constructive algorithm determines the producable amount by using several restrictions. Therefore at least one of the several different bounds has been reached and an update is consequently necessary for this element.

2. First order of the supplier
   If the added production has opened a new order at a particular supplier all dice elements with the supply period \( s \) have to be updated because the fixed order costs can be removed from the calculation. Of course this assumes that the order limit of that supplier has not been reached yet.

3. All demands are satisfied
   If the added production has satisfied the complete demand in period \( d \) of all products of the used bottleneck property all fields with the demand period \( d \) have to be updated because the estimated costs will change due to the change of the examined product set.

4. Order limit of the supplier has been reached
   If the added production has the consequence that the order limit of the supplier in the examined period or time segment has been reached, all dice elements with the used supply period, or even all elements with a supply period lying in the affected time segment, have to be updated. This is necessary because another supplier with different costs has to be chosen.
5. Storage capacity has been reached
If the added production has reached the capacity limit of either the product or the part storage the costs of the dice elements, which use this bottleneck period in their production, can be set to \( \infty \).

6. Production capacity has been reached
If the added production has decreased the production capacity in period \( p \) to a value which is not as high as at least one of the products of the examined bottleneck property demand, the dice elements of the production period \( p \) have to be updated. The update will either choose the other product set or set the cost value to \( \infty \).

If a production is removed which is the case in an iterative step the update procedure is similar.

5.4. Constructive algorithm

Figure 5 shows the general structure of the constructive algorithm for finding a first feasible solution for the given problem.

![Diagram of the constructive algorithm](image)

**Figure 5. Constructive algorithm**
The following two subsections explain the main task of the constructing algorithm in detail.

5.4.1. Find the best production in the cost dice

Due to the fact that the minimal costs can be found in every legal dice element the algorithm has to examine every element of the dice. An improvement of execution speed can be achieved by just examining a defined amount of intelligent or randomly chosen dice elements, but this of course can lead to a decreased quality or, in some cases, to the result, that the problem is infeasible although a feasible solution could be found.

Nevertheless the algorithm tries to find the production in the dice with minimal costs per unit of all elements which are taken into account. If there are several productions with the same costs a rule decides about the choice of the production.

The result of this process is the production-6-tuple which contains all information about the next step for determining the product and lot size. This process is described in the next subsection.

5.4.2. Determine the producable product and its lot size

After identifying the production which should be added to the result in the first step of the iteration, the algorithm now tries to determine the product, which should be produced and its lot size. The objective is to satisfy the customer demand as far as possible so the production with the greatest lot size will be chosen for the production. Thereby the product group indicated by the dice element must be considered in this evaluation.

The lot size of a production with supply period \( s \) in time segment \( t \), production period \( p \) and demand period \( d \) has the following upper bounds:

1. Demand
   Because stock holding should be avoided the demand \( d_{d,pr} \) of product \( pr \) in period \( d \) is one limitation for the lot size.

2. Production capacity
   The producible amount depending on the available production capacity in the particular period and the capacity requirements of one product unit is another upper bound for the lot size. The capacity bound \( CB_{p,pr} \) for the product \( pr \) in the production period \( p \) is defined as:
   \[
   CB_{p,pr} = \frac{C_p}{CD_{pr}}
   \]  
   (27)

3. Storage capacity
   The capacity of the BPA stock \( CA_{p,pa}^S \) in the periods between the supply and production period as well as the capacity of the BPR stock \( CA_{p,pr}^S \) in the periods between the production and demand period may not be exceeded. So they represent a limitation for the lot size:
   \[
   CB_{p,a,d}^S = \min \left( \min_{i \in \{s, p-1\}} (CA_{i,pa}^S), \min_{i \in \{p, d-1\}} (CA_{i,pr}^S) \right)
   \]  
   (28)
4. Maximum order limits
The capacities of the cheapest supplier, which has been determined in the first step, may not be exceeded for the supply period \( s \). This applies for the period \( LC_{p,sup}^{O,P} \) and the time segment \( LC_{t,sup}^{O,T} \). So these values represent an upper bound for the lot size:

\[
LC_{p,sup}^{O,P} = \max_P o_{s,sup}^P \tag{29}
\]

\[
LC_{t,sup}^{O,T} = \max_T \left( \sum_{t=(t-1)P^d+1}^t (o_{s,sup}) \right) \tag{30}
\]

5. Flexibility corridor bounds
The flexibility corridor bounds have an influence on the lot size because exceeding the lower or upper bound changes the estimated costs of one product unit and thereby puts the local optimality of the chosen production in question. The order limit \( CS_{s,sup} \) for the time segment \( t \) is defined as:

\[
o^T = \sum_{s=(t-1)P^d+1}^t (o_{s,sup}) \tag{31}
\]

\[
CS_{s,sup} = \begin{cases} 
LB_{s,sup}^{FC} - o^T, & \text{if } o^T < LB_{s,sup}^{FC} \\
UB_{s,sup}^{FC} - o^T, & \text{if } LB_{s,sup}^{FC} \leq o^T < UB_{s,sup}^{FC} \\
\infty, & \text{if } o^T \geq UB_{s,sup}^{FC}
\end{cases} \tag{32}
\]

The lot size \( LS_{s,p,d,pr} \) of a product \( pr \) for the periods determined in the first step of the iteration is the minimum of the calculated limitations:

\[
LS_{s,p,d,pr} = \min(d_{p,pr}; CB_{p,pr}; CS_{s,p,d,pr}; LC_{p,sup}^{O,P}; LC_{t,sup}^{O,T}; T_{s,sup}^{CS}) \tag{33}
\]

The lot size \( LS_{s,p,d} \) of the production and the produced product \( pr \) can now be identified by maximizing this value over all products belonging to the examined product group:

\[
pr = \text{argmax}_{pr} (LS_{s,p,d,pr}) \tag{34}
\]

\[
LS_{s,p,d} = LS_{s,p,d,pr} \tag{35}
\]

All necessary information for the new production has been determined. The second step of the iteration has finished and the new production can now be added to the result set.

This process will be repeated until a valid solution is found or until the infeasibility of the problem is identified.

6. Results
An implementation of the presented algorithms and processes has been created to demonstrate their ability to find good solutions in many different scenarios. We used the mathematical model presented in section 4 and implemented it in the optimizer MOPS [16] to benchmark the method results with the optimal results. The implementation in the current state has the target to demonstrate the quality of the results and is not yet optimized concerning the runtime.

Table 3 presents the scenarios which have been used to compare the method with the optimal value.

<table>
<thead>
<tr>
<th>Scenario</th>
<th>( S )</th>
<th>( Pr )</th>
<th>( P/T )</th>
<th>Speciality</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>( P=4; T=2 )</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>( P=4; T=2 )</td>
<td>forced punishment</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1</td>
<td>( P=5; T=1 )</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>2</td>
<td>( P=4; T=2 )</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>2</td>
<td>( P=40; T=10 )</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>2</td>
<td>( P=50; T=10 )</td>
<td>storage bottlenecks</td>
</tr>
<tr>
<td>7</td>
<td>3</td>
<td>5</td>
<td>( P=100; T=20 )</td>
<td></td>
</tr>
</tbody>
</table>

Column \( S \) indicates how many suppliers were regarded, column \( Pr \) the number of products, column \( P/T \) the number of periods and time segments and the last column shows if a special demand situation was analyzed.

These scenarios produce the results in Table 4 which presents the computation time and overall costs of the mathematical solver and the method.

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Time (s)</th>
<th>Cost</th>
<th>Time (s)</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>&lt;0.1</td>
<td>710</td>
<td>&lt;0.1</td>
<td>710</td>
</tr>
<tr>
<td>2</td>
<td>&lt;0.1</td>
<td>1860</td>
<td>&lt;0.1</td>
<td>1860</td>
</tr>
<tr>
<td>3</td>
<td>1.5</td>
<td>470</td>
<td>&lt;0.1</td>
<td>490</td>
</tr>
<tr>
<td>4</td>
<td>2.0</td>
<td>2230</td>
<td>&lt;0.1</td>
<td>2240</td>
</tr>
<tr>
<td>5</td>
<td>10.0</td>
<td>16440</td>
<td>8.5</td>
<td>17840</td>
</tr>
<tr>
<td>6</td>
<td>14.0</td>
<td>144510</td>
<td>8.0</td>
<td>145324</td>
</tr>
<tr>
<td>7</td>
<td>360</td>
<td>304112</td>
<td>300</td>
<td>307205</td>
</tr>
</tbody>
</table>

The results demonstrate the following facts:
1. The quality of the overall cost result is very good in comparison to the optimal value.
2. The implementation of the described method was faster, even if the algorithm is not yet optimized for runtime.
7. Conclusion

In this work an approach for a simultaneous production and order planning in a cooperative supply chain relationship was presented. First a description of the planning scenario and the reference to an overall method was given. Afterwards the work was classified in the theory of production planning and quantity coordination. A mathematical model was developed that enables an optimal solution and can be used as a benchmark for the heuristic method which was introduced in detail. First results were presented which show a very good quality of the presented approach.

Additionally to the presented method we have also implemented some improvement methods which try to find a better solution after a valid plan was found. These methods achieve improved results by changing single productions, but were not described in this work due to page restrictions. In context of the overall rolling planning the described method will be used both for new planning and for change planning. In change planning productions which needs to be re-planned will be identified, deleted and planned again according to the values of the cost dice.

At the moment we are working on the conjunction of the rough planning and the fine planning to automatically plan the flexibility corridors, the orders and the production planning in a rolling way. This planning will run for the whole frame contact horizon and will be benchmarked with an optimal solution for which a mathematical model was already developed.

8. References


