MASTER

Optimizing the multi-item aggregate production planning problem under minimum order quantities

van Boven, T.I.

Award date:
2020

Department of Industrial Engineering & Innovation Sciences
Operations Planning Accounting & Control Research Group

Optimizing the Multi-Item Aggregate Production Planning Problem under Minimum Order Quantities

Master Thesis

T.I. van Boven

Supervisors
Dr. V.J.C. Lurkin (TU/e)
Dr. C. Fecarotti (TU/e)
Dr. A.M. Florio (TU/e)
L.K. Carspecken (NIKE, Inc.)

Utrecht, March 2020

Keywords: Aggregate Production Planning, Tactical Production Planning, Multi-Objective, Mixed Integer Programming, Minimum Order Quantities, Lost Sales, Priority Bucketing, Short Lifecycle, Inventory Reductions, Tardiness, Penalties.

Abstract

This thesis presents the study, conception and employment of an aggregate production planning solution for one of Nike's footwear production plants. In order to have the right products in the right place at the right time, the production of those products must be planned in advance. This must be done in a highly volatile environment with short product life cycles, long lead times, minimum order quantities and tight production resource capacity. Currently, a fully rule-based method is used which fills the planning on a per-period basis for each Priority Class of products. This approach does not search to optimize any objectives, merely to create a feasible planning. An MIP solution which looks to minimize multiple objectives is presented. These objectives are the minimization of lost sales, demand lateness and demand earliness, which all show improvements.

In order to be able to compare the proposed solution to the current situation, a benchmark has to be made. Because not all data that is used in the day-to-day planning is available, a simulation of the CTM logic has been made which takes the same input as the proposed solution. Through the comparison of the results of this benchmark and the proposed MIP solution, it is shown that there is big room for improvement. This possible improvement lies mostly in the reduction of lost sales and the increase in on-time deliveries, while minimizing growth in inventory levels. All by better utilizing the available capacity and letting an optimization model make some smart trade-offs.

Acknowledgements

It has been over a year since I first started at Nike to finish my Masters in Operations Management and Logistics. The company had a great assignment for me: to find out what the Root Causes were of changes in the Production Plan as a result from changes in the Planning Parameters.

After some weeks to find solid ground at the company and some talks with Has Sanger about the direction we wanted to go with this research, we concluded that the academic challenge in finding these causes might not have been in line with what I was looking for. Together, we looked at other opportunities to investigate and before long, my eye fell on the Planning Algorithm employed by Nike.

Time went by and although the internship was only supposed to last six months, summer came around and no research proposal had been sent in yet. Has went on a stretch to another function. I was busy working on a lot of business-related things, and not so much on thesis-related things. I spent a lot of time working on solutions for helping the main files used by the Supply and Inventory Team more user-friendly. I also spent time doing planning work in the Skateboard and Basketball Categories – though this went on a bit longer than it should have.

Laura Carspecken took over the supervision of my thesis from Has and helped getting me back into gear. Thanks to Laura, I got into contact with Ryan Warr, Andrew Reynolds and Lee Anne Marion.

Both gentlemen were happy to take some time to tell me about the Planning Logic I was so interested in. After talking to both, I got especially energized and decided to pursue this path. After consultation with my supervisor from the University, Virginie Lurkin, we decided that a big data project like that would be great to sink my teeth in. Once I had my research proposal approved, I had trouble getting through the next part though.

It was late in 2019 when Virginie pulled the alarm bell and made it clear to me that if I wouldn't start making progress, I would probably not graduate. After that, we met every two weeks and I learned a lot from her and Claudia Fecarotti about optimization programming. Towards the end of the year, I started building the MIP logic presented in this research. The data pull of which I managed with a lot of support from Lee Anne, big thanks for that! Now, another quarter further, here we are. It's finished, the results are in and it's safe to say that Ryan and Andrew were absolutely right: there is a lot of progress to be made on the planning logic side of things. Hopefully, I can be a part of that improvement – let this thesis be my application letter.

I want to explicitly thank Laura and all my colleagues at Nike for their unwavering support, even when I thought I wouldn't be able to ever finish. Virginie and Claudia, I want to thank you for pulling me through this, the incredible patience you both have shown and all of the support – both in the shape of knowledge and encouragement

Thijs van Boven, Utrecht, March 2020

Executive summary

Business Problem

Most companies which are active in the mass fashion industry sell products with highly volatile demand, short selling-windows, long production and transportation lead times. The typical product lifecycle (PLC) of the products sold is only three to six months, both due to rapidly changing fashion and due to the changing weather. The total lead time, including transport, is often in excess of three months, especially if products have to be shipped from half-way across the globe. Therefore, production has to start long before the customer knows which products will be available for purchase.

To meet customer demand, forecasts have to be made of the demand. These forecasts are then used to acquire enough production capacity for the season in question. These forecasts are also used to make a preliminary tactical production schedule. This is used as an indication of whether or not it will be possible to produce the forecasted quantities at the right time.

Through the exploration of historical literature, state-of-the-art solutions to this problem and logic applied by the author, a Proposed Solution to the Problem is presented.

The problem researched in this thesis is that of the resource capacity allocation, planning all these products with their highly volatile demands, in tight capacity, with short lead times, no options for safety stock and MOQs to adhere to. All while trying to keep inventory, lost sales and lateness to a minimum.

Research Question

Due to the nature of the Capable-to-Match (CTM) planning algorithm being sequential as explained before, the planning is not optimized to, for example, minimize the number of late productions or unplanned units. As shown in the examples in the previous chapter, the CTM does not pull a material forward if needed, nor does it prioritize materials which have low coverage. The planning logic therefore seems to be subject to a big improvement.

An alternative algorithm should be able to better utilize opportunities which can optimize the production planning. However, there are many ways to go about finding a solution to the planning problem described here. These will be explored in part in the literature review and in part by exploring different options in this research. This will give answer to the main research question which is formulated as follows.

MRQ. How can the production planning be improved by using a different planning algorithm, under the same constraints?

The wording of the Main Research Question above is in fact purposefully chosen to include subjectivity through the use of the word ‘improved’, rather than to include something objective such as 'mathematically optimal'. This is done because there are many objectives which may be optimized and some of these are conflicting with each other.

To be able to define whether or not a different planning algorithm improves the production planning, Key Performance Indicators (KPIs) must be defined. To do so, KPIs must be chosen and/or defined where necessary. These can then be used to compare the resulting Buy Plans from the proposed improvement and the CTM output. The first sub question is therefore:

SQ1. Which Key Performance Indicators should be chosen and/or defined to guide and test the performance of a different planning approach?

It might seem obvious that, for example, having higher numbers of just-in-time delivery or lower numbers of Unplanned units would be better by default. However, it could be possible that a planning algorithm lets products with high Demand Plans cannibalize on those with low Demand Plans. This would then in turn mean that smaller demand size products might not get planned at all, even though the larger demand size products are getting fully planned. This might not be a big problem for bigger categories, which have plenty of large demand products. However, smaller categories only have smaller demand size products and might lose a substantial part of their business to the bigger ones.

This has to be taken into consideration while answering SQ1, in order to have a comprehensive representation of the different stakeholders' interests. This is done by exploring not only the individual KPIs' performances, but also the interactions thereof. The second sub question is therefore as follows.

SQ2. How should the chosen Key Performance Indicators' performance and the interactions thereof be interpreted, to reflect the quality of the planning approach?

Answering SQ2 will prevent ambiguity in the case of scenarios such as the one described right after the formulation of SQ1. These KPIs and the valuation of those can also be used to help define the mathematical objectives of the proposed solution. This is important as some objectives are contradictive. Take for example 'Minimize Inventory Levels' versus 'Minimize Product Lateness'. The first of these objective would require that all products are produced either too late or at the very last minute, to be able to ship out products immediately after production and not have them sitting in inventory. However, the second of the objectives would require pre-producing and thus create inventory to utilize the low utilization of the factories in low-demand periods. A solution in which one of these objectives is optimal, will therefore not be optimal for the other objective – unless the demand is perfectly uniform over time and within capacity.

Results

However, it should be noted that the Benchmark is of course not the same as the CTM logic that is currently in place. It is, after all, an approximation which uses the same input as the Proposed Solution, such that they can be compared better. In fact, maybe the comparison isn't really fair, considering the age of the CTM logic and its simple, yet effective approach which can be understood by all.

In reality, there might be more complications which the Proposed Solution might run into. This is in part because of the computation time of all the constraints and setting up the model. If more constraints are added to this model, the computation time will of course also increase.

Table 1: Comparison of Key Performance Indicators of the CTM Logic and Proposed Solution, colors are the same as used in Figures 4.1 and 4.2.

	CTM LOGIC		PROPOSED SOLUTION			
TOTAL DP	**5,805,547**	100%	**5,805,547**	100%	**Difference (% of CTM)**	
>3 WKS EARLY	371,785	6.40%	405,217	6.98%	+33,432	+9.0%
1-3 WKS EARLY	535,623	9.23%	610,899	10.52%	+75,276	+14.1%
IBW	4,133,747	71.20%	4,456,728	76.77%	+322,981	+7.8%
1 WK LATE	199,868	3.44%	235,031	4.05%	+35,163	+17.6%
>1 WK LATE	111,249	1.92%	83,822	1.44%	-24,427	-24.7%
UP	**453,275**	**7.81%**	**13,850**	**0.24%**	**-439,425**	**-96,9%**
PIFOT	90.3%		98.3%		+8.9%	
UNITS*WKS OF INVENTORY	3,760,561		4,003,962		+6.4%	
CAPACITY UTIL DEV	6.9%		5.5%		-20.3%	

Of course, considering the performance of the method, I would recommend implementing it in the business. But I do see the implications thereof: the CTM method works in conjunction with SAP, the Materials Planning system used throughout the entire company. Having something that works with that is of course very convenient.

However, the performance improvements that are implied throughout this research are quite severe. For years, the two main headaches of the business are (1) getting the hugely growing demand in the right place in the right time, struggling with tight resource capacities and (2) the constant rise of inventories. In order to solve the first problem, Nike has taken all kinds of precautions in order to pull production in earlier, such as through the use of Target Days Shipping (TDS, explained in Chapter 3.2.1) and Blind Buys (explained in Chapter 3.1.4) which in turn have caused increments in the Inventory Levels.

Table of Contents

List of Figures

List of Tables

1. Introduction

1.1. Company Background

On January 25th, 1964, runner Philip Hampson "Phil" Knight and running coach William Jay "Bill" Bowerman shook hands and pledged $500 each to produce the first 300 pairs of Tiger running shoes under the improvised business name "Blue Ribbon Sports". In the summer of 1971, the end-date of Blue Ribbon Sports' contract with their Japanese distributor, Onitsuka Tiger, was closing in and Knight and Bowerman turned to a Mexican manufacturer to produce the first shoe under the NIKE brand name, after the winged Greek goddess of victory. After a heated lawsuit battle over contract breaches, Blue Ribbon Sports emerged victorious and won exclusive rights to the footwear names.

Fast forward to the end of fiscal year 2018: Nike just had a $36.4 billion revenue year (NIKE, Inc., 2019) and employs over 74,000 employees worldwide. NIKE, Inc. (henceforth: Nike) has been able to stay on top of the sporting goods market through strong brand support from its customers, state-of-the-art running technology and well-made strategic business choices such as acquiring Converse, Hurley and Umbro - and subsequently letting go of the latter. Compared to the number 2 through 5 apparel brands in terms of brand value, Nike is in a league of its own, as shown in Figure 1.1.

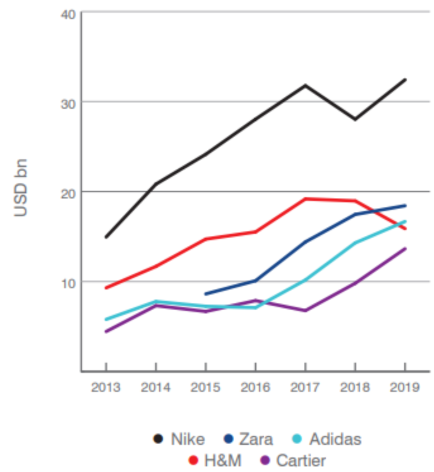

Figure 1.1 Brand value over time for the top 5 apparel brands in US$B (Brand Finance, 2019)

1.2. Organizational Structure

As of the 2018 fiscal year, Nike and its Jordan brand, as well as its subsidiaries Converse and Hurley, are governed from four business units known as Geographies (GEOs). These are North America (NA), Europe, Middle East & Africa (EMEA), Asia Pacific and Latin America (APLA) and Greater China (GC). NA and APLA are directly governed by World Headquarters (WHQ) in Beaverton, Oregon in the USA. EMEA sits under Europe Headquarters (EHQ) in Hilversum, the Netherlands. Greater China Headquarters (GCHQ) is located in Shanghai, China.

The Demand Supply Management (DSM) team consists of seven subdivisions as shown in Figure 1.2. The highlighted team in this structure is the Supply & Inventory Planning (S&IP) team. Within Nike, there are three so-called Product Engines (PEs): Apparel (APP), Equipment (EQP) and Footwear (FTW). The Apparel PE consists of all clothing, with the exception of socks. The Equipment PE contains socks, all types of balls, sticks, protective gear, gloves, among other materials. Footwear are sneakers, flipflops and other shoes.

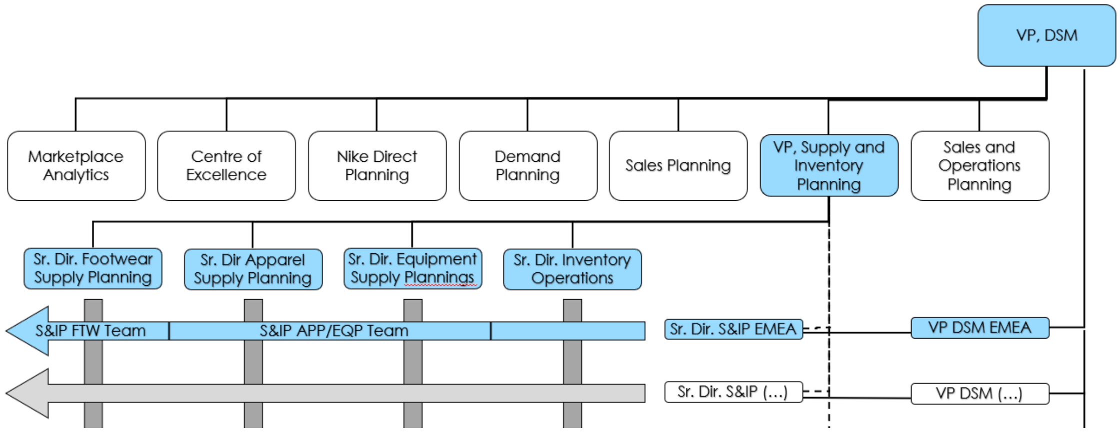

Figure 1.2: Organizational structure of the Demand Supply Management (DSM) Team

As shown in the blue boxes on the left, each of the three Product Engines (PE) has its own Senior Director, who monitors each of the PEs on a global level. There is a Senior Director Inventory Operations as well, who monitors all inventory related business globally. The first blue box to the right of the blue arrow going across shows the Senior Director S&IP for EMEA. There is one of these for each of the four Geographies (GEOs). All these Senior Directors report into the VP S&IP. The Geography specific Senior Directors also report into the VP DSM for their respective GEOs. Both the VP S&IP and the four VP DSMs report into the global VP DSM. The structure for the EMEA S&IP teams is shown in Figure 1.3.

The (Senior) Category Planners take place in the 11 Category teams: Football, Kids, Running, Men's Training, Women's Training, Jordan, Skateboard, Basketball, Tennis, Golf and Nike Sportswear – which is Nike's lifestyle line. Depending on the number of SKUs and volume of demand of these categories, a planner can take place in more than one of them, if needed.

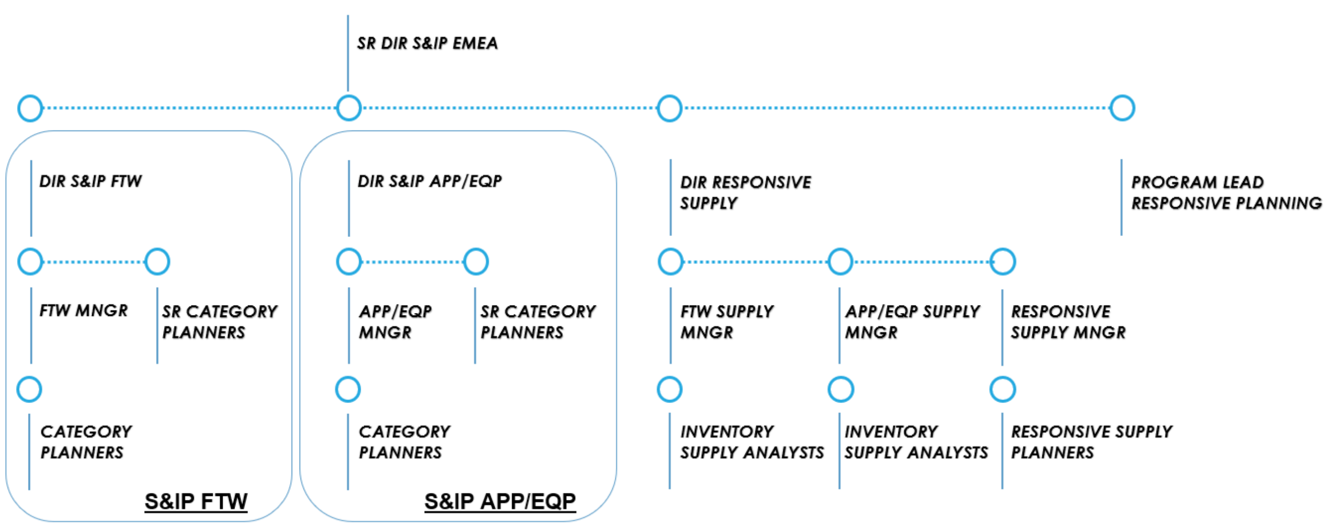

Figure 1.3: Structure of the S&IP Teams in EMEA

These category teams are cross-functional, so they also include a Demand Planner (DP) who forecasts demand on SKU level, a Category Sales Director (CSD) who is the main contact for third party retailers, a Category Sales Manager (CSM) who is the right-hand of the CSD, a Merchandiser who negotiates with global which products will go into the season and a Launch Coordinator who keeps track of all SKUs with important launch dates.

The factories which produce Nike products are not a part of Nike. They do however have contracts with Nike in which production and demand minimums are agreed upon. A single factory never focuses on one single Category of products and one single Category's products are never all in one factory. By doing this, risks are spread for both the factories and the categories.

If, for example, there are delays in a factory, it does not completely ruin a single Category's season. The other way around, if a Category is having a bad season, this reduction in demand is spread over different factories. This is important because Nike has to live up to the agreed production demand, or it could lead to monetary claims or discontinuation from the factories' sides.

To get all the right products in the right time at the right place, a lot of processes have to take place. After the products are designed and sampled, the regions have to decide which products to take into the assortment. To make this less abstract, let's use the most sold shoe in Nike: the 'Air Force 1' (AF1) as example. Every three-month period, also known as a season, a certain range of products are on offer. The AF1 has different designs, materials or colorways mostly every season, and keeps some of the old ones in the assortment as well. In the following chapters, the AF1 will be used for example reasons more often.

1.3. Planning Activities

The Supply and Inventory Planning (S&IP) team has to review and fix the production planning for Nike where necessary. Before they can do so, a few other processes have to be completed. In order to fully understand the work that the S&IP team does, those processes which are done beforehand are described in this chapter.

1.3.1. Design to Forecast (Demand Plan)

Let's take the Air Force 1 (AF1) shoe as an example again and say that there are six different types of the AF1 in a certain season which all have their own Stock Keeping Units (SKU). A certain region, such as the Europe, Middle East & Africa (EMEA), might choose to take four out of these six SKUs in their assortment if they do not think there will be enough demand in their markets to launch all of the different types of the AF1.

The EMEA sales team will then communicate to their retailers (Zalando, JD, etc.) which SKUs they are planning to have available for sale. The retailers will then create forecasts for the available SKUs. After communicating these forecasts back to Nike, a general forecast for demand will be created. This is done by combining Nike's own financial goals for the season with the consolidated forecasts from the individual retailers, including their own Nike.com platform. This general forecast is for an entire season of three months and includes the moments in time where the demand will sit. In other words, taking our example from earlier again, it might be that two of the SKUs of the AF1 are entering the market in the first month of the season so those have demand sitting in every month of the season, starting from the very first week. The other two SKUs however might only come onto the market in the second month and therefore have no demand sitting in the first four weeks of the season. This time-bound demand forecast is called the ‘Demand Plan’.

1.3.2. Demand Plan to Buy Plan

Once the Demand Plan is in place, the Global Supply Planning (GSP) team uses the forecasts to allocate which factories will produce which SKUs, usually with different colorways of the same styles in a single factory as they might need the same production tools or raw materials. During this process, the GSP team makes sure that there is enough capacity for all units of demand in the forecasts. Some products might need special raw materials or processes, complicating the production. Examples of this are airbags for in shoes – such as in our AF1 example – or yarn for knit materials, which also take more time to produce as knitting is a more time-consuming process than cutting.

Once the demand plan (forecasts including the required delivery dates), assigned factories and lead times for each product are in the systems, a planning algorithm tries to create a so-called ‘Buy Plan’. This is basically a schedule of how many units of each SKU have to be placed on Purchase Orders (POs) and when, in order for them to arrive on time for the customers. It is possible that the total demand for a product is split up over several buy weeks because of capacity constraints or simply because the full quantity is not needed immediately at the start of its lifecycle. In our example, it could be that 10,000 units are put on PO to meet the demand of the first month of the season, and then 7,500 more are placed a month later to meet the demand of the second month. This allows for excess capacity in month 1 to be utilized by other materials and less stock in the warehouse during the first month.

1.3.3. The Planning Algorithm (CTM)

This planning algorithm that creates the Buy Plan is the SAP extension called ‘Capable-To-Match’ (CTM). Since the implementation of the CTM method in 2016, the amount of manual work that has to be done to plan and put the goods on Purchase Orders has gone down substantially. Furthermore, the performance has gone up in terms of products being planned on time. The detailed workings of the CTM, including a pseudocode, are described in Chapter 3.1. To understand the following parts, it is important to already describe the basic workings, as follows.

For its inputs, the CTM uses the Demand Plan, factory issued capacity constraints, order and delivery minimums, production and transportation lead times and parameters showing by how much the required delivery date can be relaxed. Purchase Orders are placed once per two weeks. This happens automatically for all units that are scheduled to be bought on a certain date. This would mean that only a single Buy Plan is executed per twee weeks. The CTM however actually produces a Buy Plan twice per week, four times as often as necessary.

The reason for these four CTM runs per buy, is such that the Buy Plan output can be reviewed by Planners, and levers can be pulled if necessary. This is done because the CTM is not programmed to optimize its output, resulting in a suboptimal Buy Plan. By pulling the necessary levers, communicating with the Global Supply Plan teams and discussing issues with cross-functional Category teams, the CTM can be pushed to create a better output. In the worst cases, Purchase Orders must be placed manually to be able to meet demand. This does however interfere with the Buy Plan, as capacity is consumed which was intended to be used for other materials.

The CTM makes a feasible production schedule. It does so by using the following to decide the sequence to plan all demand in.

1. Demand Priority Class;
2. Demand Ideal Buy Week (IBW);
3. Demand Type;
4. First Come First Serve (FCFS) Logic.

There are six Priority Classes for Apparel and Equipment products (Figure 1.4), but only four for Footwear products. These classes are Promo demand, Launch demand, Short Leadtime (SLT) demand and three Demand buckets that signify the importance of the timing with regard to the Customer Required Date (CRD). For this research, the focus is on Footwear products though, the reason for which is explained in Chapter 1.6.

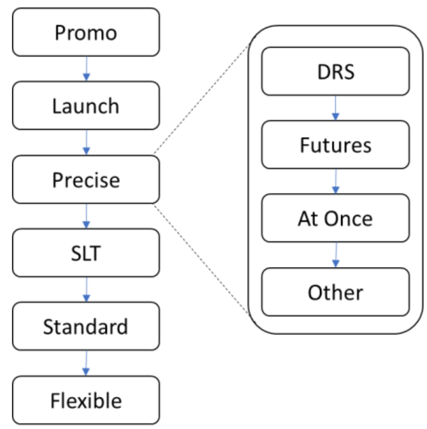

Figure 1.4: Prioritization Classes and Demand Types

Taking the average from all demand from April 2019 to March 2020, the average spread of all demand is found. It is about 0.73% of all demand which sits in the Promo prioritization, 9.9% sits in the Launch prioritization, 30.1% sits in Short Leadtime and the remaining 59.2% is left in the lowest priority scale. That means that the majority of what the CTM is trying to plan for a single buy week is the same in priority. This allows products in the first two priorities to be planned with (almost) 100% certainty, demand in the third priority class are also still relatively safe, but it does not help in the allocation of capacity to the majority of the business.

The second attribute, the IBW, is the week in which the units should be purchased to exactly meet the Customer Required Date (CRD), considering all production and transportation lead times. The calculation of the IBW is further detailed in Chapter 3.4.5 as well. So, for each Priority Class bucket, the demand is sorted by IBW, then it is sorted within each of those IBWs by Demand Type. This Demand Type split is the same for each Priority Class. Finally, the CTM applies a FCFS logic based on the Product Activation Timestamp (PAT) which denotes when the materials' Demand Plans were last altered in the system. This is only used as a tiebreaker for products where all other three attributes are the exact same.

In some cases, units of demand do not get scheduled to be bought. This can be because the CRD cannot be met due to the Lead Time. Another reason can be that production or destination minimums are not met. Yet another reason could be that the factory simply does not have the capacity to produce the units. In any of these cases, the units fall in the planning bucket called 'Unplanned' and are noted in the Buy Plan as such, including the reason why.

Example 1.1

> The Demand Plan for all the products from the 'Running' Category is uploaded in the system before that of the 'Skateboard' Category. If both Categories have units of products in the same Factory, with the same IBWs and Priority 3, the products in question from Running get planned to be bought before the Skateboard ones – given that capacity is enough. This could possibly cause more Skateboard goods to go Unplanned than in the situation where the Demand Plans were uploaded the other way around.

Something else that might happen is fluctuation over time in the Demand quantities as a product might be doing better (or worse) than forecasted and therefore might see more (or less) demand from retailers. It is also possible that the available capacity decreases because of delays in production or raw materials. In both cases, this can lead to delays in the delivery of products, or to units going Unplanned which were in a previous Buy Plan.

1.3.4. Reviewing the Buy Plan

The team of Supply & Inventory Planners (S&IP) in each of Nike's four regions review the Buy Plan that the 'Capable-to-Match' (CTM) outputs. After each CTM run, the planners look at the output and by adjusting planning parameters, they try to make the CTM find a Buy Plan that is an even better fit to what is needed in the market. This includes trying to get units that are Unplanned into the Buy Plan. Examples of these parameters that can be adjusted are the number of weeks that a batch of units of an SKU may come in earlier (or later – that is a separate parameter called "Build Late") than the ideal date. Another way to push the system is by (temporarily) putting a 'Done Buying flag' on an SKU which stops the system from purchasing more units of that product. Finally, it is possible to increase the Forecasted demand in order to meet production or destination minimums.

In practice however, having eleven different categories in each of the four different regions working on all kinds of parameters for a plethora of materials that are sourced in a limited number of factories, often makes for unexpected results. Not only that, it is possible that solving the Unplanned units – those which do not fit in the Buy Plan due to capacity, minimums or lead time – is simply impossible. This can be because a factory is planned full in the foreseeable future, it is seeing big delays in production, or raw materials are missing.

If the period in which materials can be bought to meet the demand of a Season is (almost) over, being able to still get any units at all might be in jeopardy. It is possible to reach out to the Global Supply Planning (GSP) team to discuss if there is any possible way to get the units. This is all manual work though, for example by switching other units which are on a standing Purchase Order out for the required ones.

It should be noted that any tweaks done to get units into the Buy Plan, can cause other units to fall out of it, somewhere down the line. Especially when the constraints are very tight – such as when capacity is fully utilized – relaxing one product's required date may get it to be planned where another product was planned. Thus, resulting in a knock-on effect where other products have to be pushed out to be "Build Late", or might fall out of the Buy Plan and into Unplanned completely. Due to the complexity of the CTM algorithm, it is impossible to see which factor leads exactly to what result. In other words, if a material which was planned fine, goes into Unplanned, it could be any other material that is being produced in the factory that pushed it out.

1.3.5. Exception-Based Planning

The vast majority of the exception-based work that the Planners have to do is to adjust parameters to get units of materials into the Buy Plan. Most of it requires little judgement as long as the solution is within certain thresholds. Such as a product needing to be allowed to get produced two weeks later if there was two weeks of slack time on the production to begin with. Another example where it is not this simple is illustrated in Example 1.2.

Example 1.2

Situation: A batch X of product ABC is planned to be bought such that they arrive one week before (-1) needed but are not converted into purchase orders yet. For all production weeks before the currently planned week, the resources are already fully utilized.

Issue: Due to production delays, the resources for the production in the originally planned week and the two weeks after, are taken by products that have earlier Customer Required Dates (CRD). This basically shifts the production plan forward by three weeks, but only if the products that are shifted allow for the shift. This should cause batch X to be pushed out but unfortunately, its allowed lateness parameter setting is 1 week. Therefore, the batch falls into the 'unplanned' bucket. In other words: unless space frees up, or the parameter settings are changed, it will not be produced.

Possible action: The parameter setting can be increased, such that batch X can be allowed to be produced late (+3). Possible consequences:

- If the action is taken, the retailers get the batch two weeks later than ideal (+2);
- If the action is not taken, the retailers do not get this batch at all.

Taking the action seems to be the obvious choice here since a product coming in two weeks later, is better than not at all. However, this will induce a knock-on effect. In other words, if every batch is allowed to be pushed a bit later, the exact same issue will happen the next week and the next, etc. It could be a better choice to bite the bullet, in order for other orders to still arrive as planned.

Although apparently a simple choice at first, this is not something that can be decided upon automatically. The cons might outweigh the pros. In fact, it could be possible that a new product, or even the exact same product but in a different colorway, is coming out the very next month. In that case, the demand for the previous material might cannibalize on the other product's demand, especially if it gets price reductions to clear out stock. This could mean that not all units are needed, and the batch can be dropped or reduced in size.

These types of issues, where delays cause ripple effects in the supply, can be comprehended and decided upon quite easily. Other issues however are more complex and do not have a very clear cause. To illustrate one such situation, see Example 1.3, which is an extremely simplified situation of reality. A solution to a problem like this can only be found by increasing late/early allowance parameters of whichever batches are falling into the unplanned bucket. It could be prevented though, if the planning logic would allow shifting to less than ideal buy dates, to accommodate other product batches.

Example 1.3

Situation: All demand can be produced up to 2 weeks earlier and 1 weeks later than the Ideal Buy Week (IBW – that is the week that would make sure to get the units at the retailer exactly on their required date, explained further in Chapter 3.4.5). Products A and B are both manufactured in the same Factory and fall in the lowest priority bucket. The available resource capacity and demands for both products are shown in Table 1.1 and Figure 1.5.

Table 1.1: Example capacity and demand

Week	Resource Capacity	Product A Demand	Product B Demand
47	25	20	-
48	25	25	-
49	5	5	-
50	-	-	5
51	-	-	-

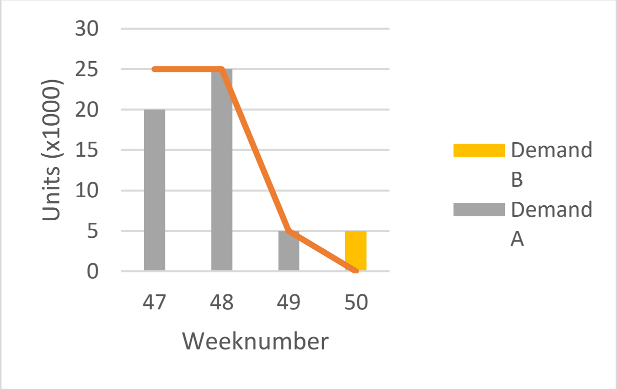

Figure 1.5: Example capacity and demand, graphed

Since the CTM plans per period, all of Product A's demand will be planned in their IBW just fine, as there is enough resource capacity.

Issue: Product B's demand sits outside of available demand though, even when trying to apply its maximum earliness (-2 weeks) or lateness (+1 week) parameters.

Possible actions: Adjust Product B's earliness (lateness) parameter to allow it to be produced earlier (later). Possible consequences:

- If product B gets produced in week 47, it has to be bought a lot earlier than normal, meaning the chance that the demand for the product is subject to change is bigger. Furthermore, it will be sitting in inventory for nearly a month until it is shipped.
- If product B gets produced in week 1 of the next year (if that has resource capacity again), the consequences will be the same as in example 1.2's action-taken scenario.

Producing it early will generate as many extra inventory over time. However, because of the earliness of the buy of product B, any deviations in its demand might render the produced number either insufficient or more than necessary. Producing it late gets rid of that problem but creates the same issues as in example 1.2. There are quite some factors which might make one of the solutions more favorable than the other, such as product-specific holding costs, costs involved for having extra SKUs in the warehouse – which cannot be stored in the same physical location – and whether there's a risk of damage to the goods while in the warehouse, to name a few.

Logically, the best option would be to push 5k units of Product A back to the available capacity in week 47. But there is no way to make that happen using the current logic and parameters, bar playing around with the demand batch information itself. A local search algorithm or a Linear Programming solution could move demand around and find (local) optimums for a certain objective function. This could improve the initial Buy Plan without requiring any intervention from the local planning teams.

Example 1.2 shows that there are planning issues which need assessment from planners to decide on an action and are comprehensible enough to do so by looking at a couple of factors. However, example 1.3 – which, again, is very simplified – shows that there are also planning issues which are very hard to track by hand but can be kept from reaching the planners if the algorithm is able to prevent the problems by itself.

1.3.6. Problem Size

The situation as it is, requires the Supply and Inventory Planning (S&IP) team to figure out which parameters to tweak and how, in order to solve these issues. This is very time-consuming and can – such as is the case in a scenario like that of example 1.3 – be impossible.

Because of this interdependence of planned productions, it is evident that an optimal solution cannot be found in a short time. More specifically and to give more of a sense of size: in the Buy Plan of July 15th, 2019, there are 65,833 batches of materials planned to be bought which sum up to 180.4 million units and still have 16.8 million Unplanned units. If the average batch size of the Unplanned units is comparable, that would mean there are just under 72,000 batches to be produced as per the following calculation:

$$65{,}833\ batches * \frac{16.8\ M\ unplanned + 180.4\ M\ planned}{180.4\ M\ planned} = 71{,}964\ batches$$

Each batch is by default allowed to be planned up to 9 weeks prior, or 1 week later than the Ideal Buy Week (IBW) and can go unplanned as mentioned before. This makes for 12 different states that a batch can be in. There are materials that have different default parameter settings, but those are few, and the ones with shorter horizons cancel out most of the longer horizon materials.

Since each batch can inhibit one of these 12 possible spaces, that means that there are approximately $71{,}964^{12}$ different Buy Plans which can be made with this number of batches. Of course, there is a slight nuance to be made, as capacity and minimum order quantity constraints might not allow some options. However, many products have extended lateness/earliness parameter settings.

Out of these many, many options there is however a big chunk that are qualitatively bad plans. Since the one considered here is created by the CTM, it can be considered a pretty simply achieved Buy Plan. It could be considered that any Buy Plan which has fewer batches planned than this one can be subtracted from the overall number. That would mean that there would 'only' be $65{,}833^{11}$ Buy Plans left that might be better than the current one. After that, it is hard to decide which (set of) Buy Plans is actually better or worse than this one unless performances of both are compared.

Odds are quite probable that better Buy Plan can be found with a logic that aims to optimize towards a certain target, rather than to find a feasible plan with minor logic. However, considering the number of options, it is hard to proof – let alone find – this optimum buy plan. And even if such a solution were found, the definition of a 'good' Buy Plan might differ from person to person as well. Each criterium added to this definition would continue to add exponential complexity to finding the 'best' one. This codependency of possible outcomes and the sheer complexity of both the criteria and possibilities leads to believe that this planning problem is at least in the NP (nondeterministic polynomial) class of computational hardness. As further described in the Literature Review, NP-hardness for a similar problem was claimed by Dixon and Silver (1981) and logically derived by the author from a study by Lagoudakis (1996).

1.4. Problem Statement

1.4.1. Demand Planning Instability

Demand planning is unstable due to the combination of the following properties of this specific production planning problem. See also the timeline in Figure 1.6.

- Most products have short Product Life Cycles (3-6 months);
- The volume of product demand always starts high and goes down throughout the season;
- Production and delivery lead times are long (4-6 months);
- Sales orders come in only 6 months before the Customer Required Date (CRD);
- Most products are new releases of which forecasts are based on similar products, if available;
- These forecasts are used to choose production plants for all products, with tight resource capacity.

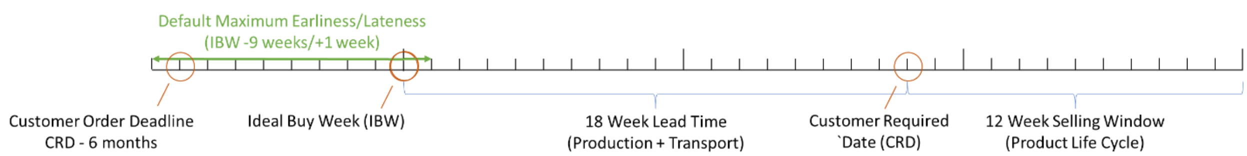

Figure 1.6: Example timeline of a product which might have to resort to Blind Buys if resource capacity is very tight

1.4.2. Maximum Earliness/Lateness Impact

The Maximum Earliness (ME) parameter of a batch of demand is set to 9 weeks by default in the hope to have materials too soon, rather than too late. Mostly, resource capacity is utilized by demand that is planned close to the Ideal Buy Week (IBW). Then, when running into capacity issues later, the demand batches have to be either planned:

- Not at all ('Unplanned'), because the available slot is out of the range of the ME of the demand batch (like Example 1.3), or;
- Very early, since it has to skip all full weeks between it and the available slot which might be months earlier (i.e. an extremer version of Example 1.2). This might even lead to:
    - Blind Buys, orders of which the volume and timing are based fully on forecasts, which are placed at factories before the Customer Order Deadline has passed (see Figure 1.6);

- High inventories, due to materials being (very) early and sitting in DC, waiting idly to be shipped to customers.

This can be caused by insufficient resource capacity as mentioned before but can also be due to not being able to meet Minimum Order Quantities (MOQ). In this case, if a batch of demand can be pulled forward to an earlier date, in order to help meet the MOQ, the CTM does so.

The Maximum Lateness (ML) parameter of a batch is not used until there is absolutely no capacity left to plan a batch of demand early. In most cases, it makes sense to prefer to have materials too early than too late. In practice however, it seems that the diminishing sales do not really occur until 3-4 weeks lateness. This was researched internally at Nike EHQ.

Resource Capacity issues are common when trying to plan the actual orders for production due to the high volatility of the forecasted demand, the strong seasonality of the products and the tightness of the resource capacity. Furthermore, factories have Minimum Order Quantities (MOQ) per SKU, per week, which have to be met. There are two types of MOQ in place at Nike. One is the so-called Production Minimum (PM) which can be met by combining orders from all over the world, if necessary. The second minimum they use however, called Destination Minimums, are not only per SKU, but also per target location. Unless there are exceptions negotiated beforehand, the Production Minimums are 3000 units, and the Destination Minimums are 240 units for Footwear products. There are some exceptions to this, which will be addressed in Chapter 3.1.3 as well. Furthermore, each batch of demand has predetermined maximum earliness and lateness. These are generally set to maximum 9 weeks early and 1 week late.

The problem researched in this thesis is that of the resource capacity allocation, planning all these products with their highly volatile demands, in tight capacity, with short lead times, no options for safety stock and MOQs to adhere to. All while trying to keep inventory, lost sales and lateness to a minimum.

1.5. Research Question

Due to the nature of the Capable-to-Match (CTM) planning algorithm being sequential as explained before, the planning is not optimized to, for example, minimize the number of late productions or unplanned units. As shown in the examples in the previous chapter, the CTM does not pull a material forward if needed, nor does it prioritize materials which have low coverage. The planning logic therefore seems to be subject to a big improvement.

An alternative algorithm should be able to better utilize opportunities which can optimize the production planning. However, there are many ways to go about finding a solution to the planning problem described here. These will be explored in part in the literature review and in part by exploring different options in this research. This will give answer to the main research question which is formulated as follows.

MRQ. How can the production planning be improved by using a different planning algorithm, under the same constraints?

The wording of the Main Research Question above is in fact purposefully chosen to include subjectivity through the use of the word ‘improved’, rather than to include something objective such as ‘mathematically optimal’. This is done because there are many objectives which may be optimized and some of these are conflicting with each other.

To be able to define whether or not a different planning algorithm improves the production planning, Key Performance Indicators (KPIs) must be defined. To do so, KPIs must be chosen and/or defined where necessary. These can then be used to compare the resulting Buy Plans from the proposed improvement and the CTM output. The first sub question is therefore:

SQ1. Which Key Performance Indicators should be chosen and/or defined to guide and test the performance of a different planning approach?

It might seem obvious that, for example, having higher numbers of just-in-time delivery or lower numbers of Unplanned units would be better by default. However, it could be possible that a planning algorithm lets products with high Demand Plans cannibalize on those with low Demand Plans. This would then in turn mean that smaller demand size products might not get planned at all, even though the larger demand size products are getting fully planned. This might not be a big problem for bigger categories, which have plenty of large demand products. However, smaller categories only have smaller demand size products and might lose a substantial part of their business to the bigger ones.

This has to be taken into consideration while answering SQ1, in order to have a comprehensive representation of the different stakeholders’ interests. This is done by exploring not only the individual KPIs’ performances, but also the interactions thereof. The second sub question is therefore as follows.

SQ2. How should the chosen Key Performance Indicators’ performance and the interactions thereof be interpreted, to reflect the quality of the planning approach?

Answering SQ2 will prevent ambiguity in the case of scenarios such as the one described right after the formulation of SQ1. These KPIs and the valuation of those can also be used to help define the mathematical objectives of the proposed solution. This is important as some objectives are contradictive. Take for example ‘Minimize Inventory Levels’ versus ‘Minimize Product Lateness’. The first of these objective would require that all products are produced either too late or at the very last minute, to be able to ship out products immediately after production and not have them sitting in inventory. However, the second of the objectives would require pre-producing and thus create inventory to utilize the low utilization of the factories in low-demand periods. A solution in which one of these objectives is optimal, will therefore not be optimal for the other objective – unless the demand is perfectly uniform over time and within capacity.

1.6. Scope

1.6.1. Factory

The biggest planning issue is tight capacity constraints, caused by lean capacity procurement, forecasting inaccuracies and production delays. The consequences of this are always on a factory level. Since products from each of the three different product types (apparel, footwear and equipment) are produced in mutually exclusive sets of factories, it is possible to focus on a specific product type and not worry about other product types in the same factories.

Products from a single category (running, training, basketball, etc.) however, are spread over different factories. In other words, focusing on a single category's products will hardly reduce the scope in terms of the number of factories to focus on, as most factories will still be involved. Moreover, if not all products within each of the factories are in the scope of a single category and the consequences for those products would not be considered, that would result in conclusions which do not reflect consequences to other categories' materials. This might then cause the result to be interpreted more positively than it actually is.

If the focus is on a specific factory, rather than a category however, the consequences for each of the products produced in said factory would be considered. This is only the case for footwear factories however, as apparel and equipment products are produced in factories which produce materials for other brands as well. These non-footwear factories do have dedicated capacity for Nike products, but these factories might retroactively change the resource capacity available due to fluctuations in the need for capacity of another company. The other way around, where Nike is not going to utilize all the capacity it initially though, is also possible. This might mean that available capacity data in the system was already adjusted to reflect these changes. Thus, resulting in false utilization rates of the factories, for example.

Therefore, in order to have as many variables in control as possible, it would be better to focus on the footwear factories. This allows for not only independency from other companies, but also for a complete overview of all products produced there. This means that all consequences are taken into consideration for all materials produced in the factory.

1.6.2. Time

Another factor to consider is the moment in time from which to pull the data. This will be further discussed in Chapter 3.4. However, the number of moments to consider and the distribution of the demand over time can already be addressed in order to get a general idea.

Multiple moments in time can be evaluated, in order to see how the volatility of demand has an effect on the performance of the proposed solution. However, a well-chosen moment in time can be as valuable, if not more valuable because it allows a deeper dive in the results for that dataset. An example of how to dive deeper into such a dataset is by splitting it into subsets of demand. Since the demand is volatile over time, it is logical that demand in the system for a moment later in time, will always be subject to more changes than demand which is already due soon. Therefore, prioritizing the earlier demand might be valuable for a good solution. This is especially the case when using a planning algorithm which treats the lateness, earliness and lost sales of any future demand in the exact same way.

As mentioned before, the demand at Nike has strong seasonality. This is not only in the sense of flipflops being sold in spring and summer and outerwear in fall and winter. Products always see a peak in demand at the start of the season and go down significantly towards the end. This makes sense as the market appetite for a product is highest when it is just released. Then, towards the end of the season, the new products are just around the corner and people are less interested in the current season's materials. Not only that, towards the end of the season – and therefore towards the end of a products lifecycle – discounts are often applied to get rid of materials, thus lowering overall margins. Since Wallstreet uses growth in revenue, inventory and margins to calculate stock stability, it is important to keep discounts to a minimum.

One of the dashboards used by the DSM department is called the Planned In Full On Time (PIFOT) dashboard. This dashboard shows the planning status of all materials, on color level, in comparison to the Ideal Buy Weeks. This allows a planner to see how all materials do with regard to earliness, lateness and how many units of demand are unplanned.

This dashboard also has an overview of how materials are 'flowed' throughout a season. In other words: how are sales orders distributed over months within a season? Figure 1.7 shows the overall demand for 2019 graphed against the months in which they are required. The y-axis shows the demand linearly, but the exact numbers are censored. These are however irrelevant to see how much variation there is in product demand. Often resource capacity at the factory is quite constant over time, so that means that early-season products have to be pulled forward into the low-demand part of the previous season, or they are postponed to the low-demand part of the current season.

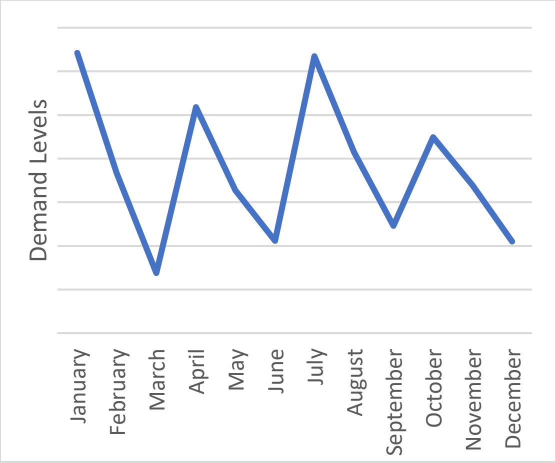

Figure 1.7: Demand over time, plotted against Customer Required Dates (CRD)

In order to give the proposed solution a thorough test, it is important to take a dataset where there are still trying circumstances with regard to the amount of demand to be planned in each week. The demand numbers still shift vigorously throughout the season six months prior, after the Order Entry Deadline (OED). The changes are usually caused by the cutting away of demand that cannot be met and by extending the Maximum Lateness (ML) settings in an attempt to pull the demand into the Buy Plan late. A good dataset would therefore be at the very start of a season, since it will have the least changes made to the demand of the demand two seasons later but will have the actual demand numbers in the books already.

2. Literature Review

2.1. Economic Optimization

The origins of the production planning problem and historical methods that have been used to tackle it are researched first. Many articles take minimizing costs associated with production and inventory as objective (Dannerstedt, 1955; Charnes, et al., 1955; Spurrell, 1967; Hausman, Peterson, 1972; Lambrecht, Vander Eecken, 1978) This is an expected starting point for the research into the topic as minimizing costs maximizes profits if all else is kept equal.

Overhead costs can theoretically be left out since these remain unchanged for each execution of the optimization. Inventory costs are a bit ambiguous then though, as it is not clear what costs are included in these. Running warehouses evidently costs money because of labor, machine usage and depreciation and expansions if needed. Some of these are also overhead though. So, should these costs be included in the objective function then? As a matter of fact, if not all costs are included, the costs calculated might be under- or overstated for either or both costs factors.

In some articles, extra costs are added to the models to find an optimal solution, such as setup, changeover or lost sales costs (Wagner, Whitin, 1958; Wagner, Shetty, 1962; Eppen, et al., 1969; Florian, Klein, 1971; Jannagathan, Rao, 1973; Walker, 1980; Ruth, 1981) If the costs route is taken, this is a step forward as it depicts reality better. However, margins might differ from product to product just as production costs might differ. This raises the question if revenue should be included in the objective function as well. Some literature takes this route, maximizing the profit by minimizing costs as well as maximizing revenue in one function (Lee, Aronofsky, 1958) This seems to be the most comprehensive method when going down the financial route.

In the case of this research however, it is not a matter of reducing costs or increasing profits. All demand must be covered as much as possible, timeliness is the only reason to not have demand in the system. This does translate to penalties for earliness and tardiness though, as cumulative inventory levels require enormous expansions to warehouses. This is impossible to translate directly to a product-specific holding cost, not in the least because of the difference in storage volume consumption, say between a down filled winter jacket and a pair of flip-flops for example.

2.2. NP-Hardness

In the research done by Dixon and Silver (1981) into a heuristic procedure for the lot-sizing problem, the computational complexity of the problem is briefly discussed. They mention that it, the Dynamic Lot-Sizing Problem with two commodities and constant capacity, is NP-hard. Unfortunately, the research they reference for this is Dixon's own unpublished graduation thesis.

The computational complexity raises the question whether the subject of this research is also NP-hard. The problem in question has multiple items, a single level (no interdependent products), one constrained resource (production capacity), due dates, minimum order quantities and lost sales. This makes it similar to a Resource Allocation Problem (RAP) where each time period has limited resources which can be used to meet demand. According to Darmann, Pferschy and Schauer (2010), the RAP is NP-hard as it can be reduced to a 0-1 Knapsack problem.

The 0-1 Knapsack problem has a single capacity constraint, usually referred to as a maximum 'weight' which the knapsack can contain. There are some items $I \in \{i_1, i_2, \dots, i_n\}$, each with weight $W \in \{w_1, w_2, \dots, w_n\}$ and profit $P \in \{p_1, p_2, \dots, p_n\}$. The objective is to maximize the profit without going over the maximum weight of the knapsack. Each item can either be included (1) or excluded (0) from the knapsack, hence the name of the problem. The 0-1 Knapsack problem was proven to be NP-hard by Lagoudakis (1996). Since the problem at hand translates to a further constrained set of knapsack problems with non-integer solutions – i.e. a proportion of a demand batch can be allocated, it is obvious that this problem is also NP-hard. Because of the NP-hardness of the problem, there is no absolute method to solve the problem. The best that can be done approximates optimality using heuristics and/or mathematical programming.

2.3. Timeliness optimization

Starting with Orlicky's documentation of the use and inception of the Material Resource Planning (MRP) system (1975), optimization in pursuit of pure timeliness to meet demand became more and more common. This is how many companies still work today and is without a doubt a fundamental part of tracking inventory, orders, production and coverage of demand to this day.

Some literature concerning the freezing of the Material Production Schedule (MPS) that follows from the MRP is discussed. This concept seems useful for the splitting up of the future demand into chunks, which can individually be optimized. By doing so, the schedule of the earlier demand can remain unaltered, while the rest of the demand can still be planned to optimality. Unfortunately, this is not the type of 'freezing' the authors (Sridharan, et al., 1987; Xie, et al., 2003; Diaz-Madroñero, et al., 2014) mean. Instead, they consider freezing from optimization over optimization. This would mean that part of the future orders that are yet to be placed, are already set in stone for a new optimization run. Following, these will be unalterable and possibly cause over- or underbuying of a commodity due to this. They discuss the usefulness of this, because it allows the production plants to prepare the job sequencing in advance.

2.4. Exploration of current state

The different production strategies are reviewed next. The strategies shown in Figure 2.1 are the strategies commonly used in the industry as discussed by Rudberg and Wikner (2004). The Make-to-Stock (MTS) strategy and Make-to-Order (MTO) strategies specifically require some extra focus as they are most commonly used. Segal (2019) discusses the elements an MTS strategy requires, such as safety-stock possibilities which are not possible with Nike's materials in

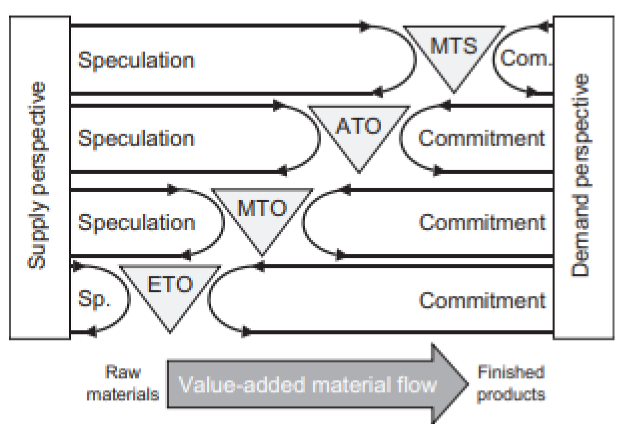

Figure 2.1. The four most common different production strategies (Rudberg, Wikner, 2004)

the scope of this research. The MTO strategy are discussed by Köber and Heinecke (2012). They explain that MTO strategies are usually used for highly customizable materials, which allow users to customize their products. The materials in the scope of this research cannot be produced using this strategy because the lead times are too long.

Because of the specific characteristics that demand has at Nike (long lead times, short lifecycle, etc.), the method explained in Nike founder Phil Knight's biography, "Shoe Dog" (Knight, 2016), still seems to be the best option to produce the demand. This method requires business customers to place orders six months in advance. By doing so, a production planning can be finalized and buy plans can be converted to orders once the Ideal Buy Week (IBW) for products comes around.

The Capable-to-Match planning logic which is currently in place is described in detail by Kallrath and Maindl (2006) who explain that it was originally developed for the Semiconductors industry. The CTM logic is not mentioned in many articles. However, in one article, it is critiqued for being complicated to set up (Kepczynski, Ghita, Jandhyala, Sankaran, Boyle, 2019). On the other hand, it is noted that a plus for the logic is that it is not a black box like many other automated planning systems (Guimaraes, Klabjan and Almada-Lobo (2015). In other words, the system follows a pre-defined set of rules which can be logically followed. Knowing how the input is changed for this system, technically allows the user to determine what repercussions that will have on the output of it.

2.5. Specific MIP approaches

Further applications of Mixed Integer Planning (MIP) logic are explored to solve the issue at hand. Mathematical programming is the most mathematically feasible solution offered in the literature for this problem, considering the size of the problem, as discussed in Chapter 1.3.6. Among the mathematical programming options, Mixed Integer Programming is the most mathematically logical option to pursue because it allows penalties and all necessary constraints to be included in the model.

The other solution methods found in Chapter 2.1 include enumeration by mapping out all possible decisions (Florian, Klein, 1971), heuristics (Eppen, Gould, Pashigian, 1969; Lee, Shaikh, 1969; Hausman, Peterson, 1972) and even game theory (Shubik, 1983) – albeit this last one is not used for the exact purpose this research is pursuing. These other methods are tedious and require building models from the ground up, whereas mathematical programming can be done by employing solvers, which have been optimized for efficiency for years and years.

Looking specifically at MIP solutions which use the quantity of each product in each period to be produced as a decision variable, some relevant articles are found. Just as before, these focus on the minimization of costs. This, again, is common and logical, but raises the question what is and what is not included and, more importantly, what should and should not be included. The article by Hung and Hu (1998) does include revenue as well in its calculation, to be more complete.

All articles discussed in this chapter have constraint on the amount of available production time. The paper by Omar and Teo (2007) also includes a maximum inventory level though. This creates a maximum earliness, due to limited capacity. In this research, there is no such hard maximum, but inventory levels should be minimized as much as possible. It would be interesting to include such a constraint but considering the time between ordering and delivery at the warehouses, preventive measures can be made well in time to store the extra inventory.

Though not exactly used as a Minimum Order Quantity (MOQ), Hung and Hu (1998) do use a binary variable in the constraints which allows production of a product if that binary variable is 1. This can be used to create MOQ constraints as well, as it would be able to set a flexible minimum for production, depending on whether a product is produced in that time slot at all. Omar and Teo (2007) use this method and formulate the constraint in such a way that a product is never unnecessarily setup if it is not produced in a time period. Now this last part is logical if there are setup costs. In the case of this research however, there are no setup costs, so this is essential to include.

Finally, lost sales are included in one of the setups by Sung and Maravelias (2008). There’s not much to say about it, they simply let products fall out of the schedule against a certain cost. This allows for the MIP to choose whether or not to include a product if the benefits outweigh the lost sales. This is necessary for this research as well.

2.6. Resource Utilization Smoothing

Finally, resource utilization smoothing is explored. Some alternative solutions are presented in the work by Lovgren and Racer (2000). They describe the use of an MIP, three different heuristics and combinations thereof to find the best method to optimize the contradictive objectives of minimizing tardiness and resource utilization deviation.

Out of the heuristics they present, it seems that the Border Swap technique is the most applicable to the research. This is because they start from an optimal solution and only make switches based on the highest reward for a swap. It is questionable whether this is feasible with a large dataset though, as the computation time would require the calculation of every single possible swap of demand batches.

3. Methodology

In order to be able to compare any results at all, a benchmark has to be used. Not all historical data is available. Therefore, the proposed solution cannot take all the same factors into account as the original data. Thus, it would be unfair to compare the proposed solution to an actual CTM run outcome. Therefore, a simulation has to be made which uses the same logic as the CTM and the same input which will be used for the proposed solution. This will be described in Chapter 3.1.

As became apparent through the literature study, the best way to approach the problem at hand is the Mixed Integer Programming (MIP) method. However, in order to set up an MIP solution, objectives should be known. In order to find out which objectives to pursue, the S&IP Leadership Team had to be consulted. The results thereof are described in Chapter 3.2.

After mapping the objectives to pursue, Chapter 3.3 will describe the approach to actualize these. This chapter will include a motivation for the choice of the approach and a description of the workings behind Mixed Integer Programming and the solver used for this research.

In Chapter 3.4, the exact data which will be used for the research will be discussed next. This will include both the selection criteria and descriptive statistics of the data in question. The method of data extraction and a sample of the data will also be shown, in order to create a sense of understanding of what is available and how it can be used for the proposed solution approach.

Chapter 3.5 will describe how all the different parts of the problem are incorporated into the solution approach. Finally, Chapter 3.6 will present the final mathematical model, including all variables, the objective function and all constraints necessary for solving the problem.

3.1. Benchmark

3.1.1. CTM Logic

As described in Chapter 1.3.3, the current logic is the SAP APO extension called the Capable to-Match (CTM) logic. This logic is a rule-based sequential planning logic. As described in that chapter, the rules consist of a bucketing of the different demand prioritizations, sorting each of those buckets according to the Ideal Buy Week (IBW), all the demand within each IBW by demand type and finally by the Product Activation Timestamp (PAT) – which acts as a tiebreaker.

The exact input data that the CTM logic uses is not available. To still be able to compare the proposed solution to the CTM logic, a simulation of it had to be made. The description of this simulation is also a great opportunity to dive deeper into the exact workings of the CTM logic. To this end, Chapters 3.1.1 through 3.1.5 explain all the details of the logic and includes examples and a pseudocode of the CTM benchmark.

3.1.2. Demand Batches

The Demand that needs to be produced at the factories has certain attributes. Examples of these attributes are the unique Stock-Keeping Unit (SKU) for each style-color combination, the demand season, the Customer Required Dates (CRD), Lead Times (LT) and demand destination (for a complete list, see the pseudocode in Chapter 3.1.5)

As mentioned before, Nike works with four seasons per year, each consisting of three months. Materials from different seasons can technically not be combined. However, sometimes production for a certain material is so late, that the units can be used in the season after. The CTM logic will however never combine different Seasons' demands.

For convenience, it is possible to group together Demand that has the exact same parameters into batches. These batches of demand will be referred to as **Demand Batch (DB)** from here on out. Note that the CTM might split these batches up if necessary. They are, after all, only grouped together for convenience. More on why and when these are split up and what the benefits and downsides of doing this are, is explained in Chapter 3.1.4.

Another vital attribute that is tied to all demand, is the **Customer Required Date (CRD)**. By deducting the Lead Times from this CRD, the **Ideal Buy Week (IBW)** can be calculated. The IBW reflects the week in which a DB order has to be placed at the factory for it to arrive exactly in the week in which the customer needs it – given that the actual lead times are exactly the same as those in the system, of course.

For feasibility's sake however, there is a possibility to deviate from the IBW if necessary. How many weeks earlier or later than this IBW an order may be placed is dependent of the **Maximum Earliness (ME)** and **Maximum Lateness (ML)** parameters in the system. The author decided to name this range in which the materials can be planned the **Planning Horizon (PH)** for convenience. The calculation of this PH is illustrated in Figure 3.1 with arbitrary values for the lead times, ME and ML. The IBW is also used to sort the DBs into the order in which they are planned in the actual algorithm, which is discussed in detail in the pseudocode in Chapter 3.1.5 as well.

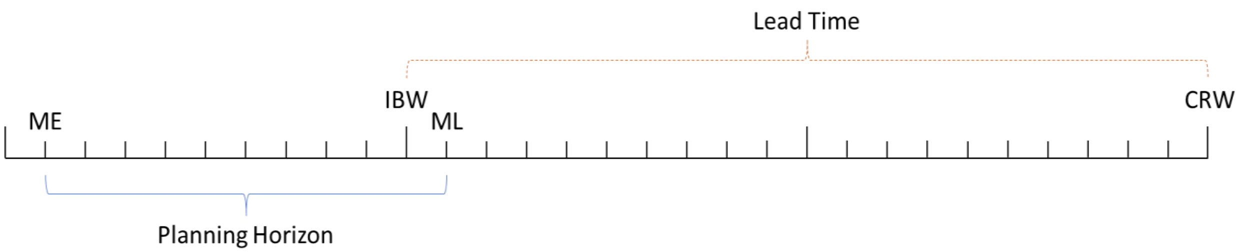

Figure 3.1: Example PH calculation where ME = 9 weeks, ML = 1 week and Lead Time = 20 weeks

3.1.3. Planning Constraints and Unplanned

There are certain constraints that must be met for the Demand to be planned for production. If there is no possibility to plan the DB in the PH that is in the system, it is labeled as Unplanned. This means that these units will never be made, unless a (Global) Supply Planner takes countermeasures to get the DB into the buy planning. Countermeasures that can be taken include changing the BE and BL parameters, negotiating extra Capacity with the factory in question or changing the volume of the demand. When which of these actions can be taken to solve issues is further discussed in the chapters handling Unplanned reasons up next.

3.1.3.1. Unplanned due to Capacity

The first and foremost constraint is that there must be **Capacity** in the factory where it is to be produced. For footwear materials, a single type of product is always sourced from one factory. This means that once a factory's capacity is fully planned with other materials, it is possible that any more demand simply does not fit. When this happens, the Demand Batch is labeled as 'Unplanned due to Capacity'.

To solve this issue, Supply Planners can widen the product's allowed Planning Horizon by tweaking the ME and ML parameters. If this is done, the CTM might be able to plan the units for purchase in the next iteration of the run. Global Supply Planners are often working with factories to increase capacity for certain products if there is a high need for it. Because of this, sometimes, it is better to contact them and see if there are possibilities to increase capacity, than to add several weeks of allowed lateness immediately.

A consequence of increasing allowed lateness is that the product might face lost sales due to being unavailable when the market wants it. Another possible consequence is that the window for selling the product becomes too small as the final offer date is predetermined for all products. This selling window can be extended, often going hand in hand with discounts, but doing so might cannibalize on sales of other products which are meant to be sold at that moment for full price.

3.1.3.2. Unplanned due to Lead Time

When capacity is not added, but some weeks have gone by waiting for this, it is possible that the PH has fully passed for the DB in question. In this case, a DB falls into 'Unplanned due to Lead Time'. If this is the case, the only way to still get the units, is by increasing the ML such that the PH is extended past the current date. Of course, this does not guarantee that the order can be placed, as the other constraints still also have to be met.

3.1.3.3. Unplanned due to Minimums

Earlier, it was mentioned that Demand Batches may be split by the CTM if necessary. Sometimes, this occurs when Capacity constraints do not allow for the full batch to be produced. In this case, part of the batch can be produced in one week and another part in another. In cases of very high demand, this can be split even further.

However, there are Minimum Order Quantities (MOQ) to be considered as well. There are two of these quantities: **Production Minimums (PM)** and **Destination Minimums (DM)**. Both are weekly quantities and can differ between SKUs and destinations. Each DB has to adhere to these predefined MOQs. DBs may be combined to meet the PM and DM. This process is also shown in the pseudocode in Chapter 3.1.5. There are some exceptions to these MOQs. Namely, any DB which is in the Promo Priority Class does not have to adhere to either the PM or the DM but will always be planned if Capacity allows it. Materials that fall in the Direct Ship (DRS) Demand Type do not have to adhere to the Destination Minimums either. This is because these materials ship directly to the client. Besides, a client cannot place a DRS order for a small amount, so this should technically not be a problem. It does mean however, that it cannot help other DBs meet the DM either, exactly because of this property.

Production Minimums are on SKU-level, this means that demand from all over the world can be combined to meet them. It is a different story for Destination Minimums though. These must be met for all demand of a single SKU for each Destination separately.

Example 3.1

> To illustrate, imagine demand for one SKU of 1,500 units for the EU, demand for the same SKU of another 1500 units for the USA and a Production Minimum (PM) of 3,000 units. Separately, the demands do not meet the PM and thus are impossible to be planned separately. However, if the Planning Horizon (PH) for both demands overlaps, they might be combined to meet the PM.

The CTM can combine future DBs with the one it is trying to plan, if the volume helps reach the Production Minimums, if necessary. It does this by pulling all of the DBs back to the earliest of the DBs' IBW – or as close to that as possible. In the case that the overlapping date is later than the earliest IBW, the DB with the earliest IBW is moved out to that same earliest possible date. It should also be noted that if non-Promo DBs of the same SKU are already planned in the current week, the PM has already been met and the DB in question can be added without problems.

Example 3.1 (cont.)

> Let's say that there is one week that overlaps for the DBs mentioned before, as shown in Figure 3.2. In this case, the EU DB has to be moved out 1 week and the NA DB has to be moved back 9 weeks for them both to meet the PM and get planned.

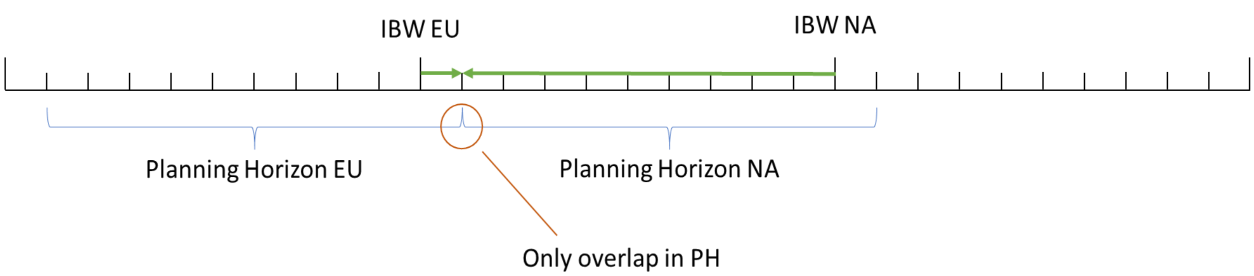

Figure 3.2: How two batches might be moved to meet the PM

In other words, as long as demand can be combined to meet the PM, irrespective of the destination of the goods, those units can be produced. In the cases where the PM cannot be met because the PHs are not overlapping, there is not enough capacity in the overlapping period to meet the PM or simply because of insufficient demand, the DBs in question are labeled <u>'Unplanned due to Production Minimums'</u>.

The case for Destination Minimums is very similar, but the DBs can only be combined if they have the same Destination, as explained before. Example 3.2 looks at a hypothetical situation.

Example 3.2

Let's take a PM of 3,000 again and a DM of 500 for all destinations. Now imagine that EU has a DB of 400 units of a certain SKU and NA has a DB of 2,600 for the same SKU.

In this case the PM can still be met by combining the DBs, however EU is 100 units short to meet the DM. Those 400 units will therefore fall into the Unplanned bucket. They will then in turn no longer be able to help NA out to meet the PM. This will cause those 2,600 units to also fall into the Unplanned bucket.

If Europe artificially increases their DB by 100 units to meet the DM, the batches can be combined to meet the overall PM again. If North America wants their units without being dependent of Europe's choice to increase or not, they can choose to artificially increase their demand by 400 units to meet the PM as well. This will leave EU without supply though.

The extra units that come from these artificial inflations will evidently not be linked to customer orders but will be labeled as **Available to Promise (ATP)** inventory. Customers can see what is available in ATP from their ordering systems and place **At Once (AO)** orders for these units such that they can receive them when they arrive – or immediately if the units have already shipped to the destination.

3.1.3.4. Miscellaneous Unplanned reasons

The Unplanned reasons discussed so far are the main drivers of all Unplanned. Taking the average from a couple of random samples, Unplanned due to Capacity and due to Lead Time are the most prevalent, responsible for 36% of the total Unplanned each. Unplanned due to Minimums comes in at about 12%. The remaining 16% of Unplanned are due to six smaller causes. These are, from most common to least:

- Done Buying (4.4%) – the Done Buying Flag (DBF) is a date set by planners which prevents the CTM from planning remaining demand for the SKU-Destination-Season combination from that date onwards;
- Marketing Line Plan (4.2%) – caused by a material not being taken into a season's purchasable products but somehow still getting orders placed by accounts;
- Planning Master Data Exception (3.3%) – caused by incomplete data, such as missing factory information, lead times or the like;
- Master Data (2.2%) – Any problem that has to do with missing data/untimeliness of demand that is not captured by the previous two reasons;
- Raw Material (1.5%) – caused by constraints set in the system where raw material for production of goods is limited (e.g. knit products);
- Initial Capacity Consumption Week (0.5%) – caused by a hard-set first possible consumption date before which no orders can be placed for a product.

Upon each iteration of the CTM run, all the latest data from the system is taken to create a buy plan. It does, however, not take previous buy plans that have been generated into account. That means that anything which was planned in a previous CTM run and has not been converted into a Purchase Order (PO), is completely disposed of upon the new CTM run. Due to this property of the CTM logic, it is possible that something that was in the planning before without any issues, now shows up in one of the 'Unplanned' buckets.

3.1.4. CTM Runs

In reality, one 'CTM Run' consists of several instances of an algorithm being run. This algorithm is called the Sequential Algorithm (SA). Each instance of the SA runs for a single Priority Class. The resulting Buy Plan from one SA is stored in the system before the next instance. The products that are planned to be bought from previous SA runs take up capacity which is considered in subsequent SA runs. With the exception of the Promo Priority Class, DBs which are planned in earlier SA runs can be used by subsequent SA runs' DBs to meet Production and Destination Minimums – if the SKU and, in the case of DM, Destination matches of course.

3.1.4.1. DB Sequence

The Priority Classes, from high to low, are Promotion, Launch, Precise, Short Lead Time, Standard and Flexible. The Benchmark solution includes all of the Priority Classes, but the scope of the research renders three out of these six useless. As the scope only includes Full Lead Time (FLT) Footwear materials, for which the Precise, Standard and Flexible timing buckets are not set up.

These three timing buckets (Precise, Standard, Flexible) are a recent expansion of the Priority Classes and are currently experimented with in the Apparel and Equipment Product Engines. The idea is that products in the Precise bucket are more important to have on time. These therefore have tighter default Maximum Earliness (ME) and Maximum Lateness (ML) settings. However, due to their high prioritization, this generally does not pose an issue for getting the materials planned. The Flexible bucket is another story though. These products are allowed to arrive even earlier or later than the Default settings of 9 weeks early or 1 week late. However, as they are the very last Priority Class to be planned, these are currently seeing a lot of Unplanned issues due to Capacity.

In the Footwear Product Engine, only the Standard bucket of this three is in use. Since this simply means that the other Classes will be empty, the Benchmark CTM simulation can remain the same. This means that the Sequential Algorithm (SA) is run for Promotion Materials first, then, taking the results of that run, it runs for Launch products, etc. From one run to the next, the planned products and remaining capacity are updated as they will impact the next SA run.

Within each of these runs, the demand is sorted to a specific sequence before it starts the actual creating of the buy plan. Within each Priority Class, the demand is sorted according to the Ideal Buy Week (IBW). This used to be done in order of the Customer Required Date. This seems fairer at first, because each destination wants to have the same products at the same time – at the start of the season mostly. However, destinations close to the factories ran into big problems with their begin-of-season demand losing out in priority to the (very) late end-of-season demand from further away destinations. Of course, the last demand of a season is less profitable than the demand for the new season's materials. Therefore, it was changed to the IBW instead.

Then, within each Priority Class, in each IBW, the DBs are once again sorted. This time they are sorted according to the Demand Type. These are – again from high to low – **Direct Ship (DRS)** demand, standard demand, **At Once (AO)** demand and lastly, all other demand.

As briefly mentioned before, DRS orders are demand for products that are shipped directly to the accounts (Zalando, Intersport, etc.), rather than to the DC. Standard orders are normal Sales Orders (SOs) which are placed by accounts, At Once (AO) orders are placed for products that are already in stock at the warehouse, so they are irrelevant to the production planning. Other orders are DBs that do not have one of the other labels. This might be because the Demand is not on any order but is only a forecast that has not been converted, a so-called 'Blind Buy'.

In case that multiple DBs are of the exact same Priority Class, the demand has the same Ideal Buy Week and the Demand Type is the same, the **Product Activation Timestamp (PAT)** is used to break the tie. The PAT is almost certainly unique to each DB, as it is the timestamp of when the Demand Plan for those exact units was last altered. The odds that those are also exactly the same for different DBs are nearly impossible. However, if they are, the multiple DBs are handled in an unknown random order.

3.1.4.2. The Steps of the Benchmark

For convenience, the Benchmark CTM Simulation, which consists of multiple runs of the Sequential Algorithm (SA) will be called 'the Benchmark' from here on out. It is programmed fully in Microsoft's Visual Basic programming language and uses Microsoft Excel tables to receive its input and paste its output. The Benchmark simulates the actual CTM logic, but one big difference is that the Benchmark does all separate SA runs in one. It simply continues from the end of one PC to the next. All of the steps described here correspond to the steps in the pseudocode in Chapter 3.1.5.

<u>Step 0.</u> The first thing which the Benchmark does is bucketing all DBs into their respective Priority Class (PC). Then, it sorts the DBs in each PC according to the IBW first, then the Demand Types (DT) order discussed before and finally the Product Activation Timestamps (PAT). It then calculates the Available Factory Resource Capacity (AFRC) by subtracting any DBs which are already converted to Purchase Orders (POs) in the system from the total Factory Resource Capacity (FRC). This is technically preprocessing and could arguably be done before running the Benchmark, but for convenience is included into the code.

<u>Step 1.</u> The Benchmark starts by checking the AFRC for the specific Capacity Type (CT) in the IBW of the first DB. If there is enough capacity to produce the DB in question, the Benchmark moves to step 2. However, if there is not enough capacity in this week, the available capacity is compared to the Production Minimum (PM) of the DB in question. If there is enough capacity to produce the PM of the DB in question, the DB can be split in two separate DBs. The first of these will then be set equal to the remaining AFRC in the current week and will move on to step 2. The other DB will move into the queue of DBs to be planned.

If there is not enough capacity to produce the PM, there is no use looking in this week any further for this DB. To continue, t is set to the next week and the Benchmark goes back to the start of Step 1. The sequence of the weeks to plan a DB in is as follows. It starts with the IBW, then moves back one week at a time until the Maximum Earliness (ME) parameter is maxed out (IBW-ME), it then moves to one week later than the IBW and from there moves forward one week at a time until the Maximum Lateness (ML) parameter is maxed out (IBW+ML).

If no week with available capacity for the CT of the current DB is found, the DB is labelled 'Unplanned due to Capacity' and will not be planned during this run of the Benchmark.

<u>Step 2.</u> After finding available capacity to produce the DB, the Benchmark goes on to check if the Production Minimums (PM) and Destination Minimums (DM) are met. These can also be met by combining the current DB with existing POs in the same week. If they are not met however, the Benchmark tries to find future DBs of the same SKU (and Destination for DM) to add to the current DB. It should be noted that DBs that are combined to meet Minimum Order Quantities are not combined into one DB in the system. This is because the attributes of the DB (such as its IBW, PC and DT) can still differ between them. They are simply planned at the same moment.

Promo materials do not have to adhere to either the PM or DM constraints, because these materials are considered critical for business and need to be produced no matter what. Direct Shipment (DRS) orders do not have to adhere to DM constraints on paper – though they always do because accounts cannot place small DRS orders. DRS orders cannot be combined with regular orders to meet DMs of those regular orders due to the different destinations.

If no possible combination of DBs is found to meet the PM and/or DM, the DB is labelled 'Unplanned due to Minimums' and will not be planned during this run of the Benchmark.

<u>Step 3.</u> This is where the actual finalizing of the planning of the DB takes place. The DB is marked as planned, the AFRC is reduced by the number of items in the DB and the Benchmark moves to the next DB to plan it.

3.1.4.3. Benchmark extra notes

It should be noted that there are two input parameters which might cause a DB to move into an unplanned bucket during step 0. These are the Initial Capacity Consumption Week (ICCW) parameter, and the Done Buying Date (DBD) parameter. Both parameters reflect a date: the ICCW gives the first possible week in which a SKU may be produced, the DBD gives the last possible date on which a SKU may be bought. The DBD is Destination-specific. These parameters do not need to have a value but can be set by (Global) Supply Planners in order to prevent buys from happening.

Including the logic of these parameters in step 0 of the pseudocode made it too convoluted as this would cause too many checks on different levels. They are however used in the actual programmed version of the Benchmark though.

3.1.5. CTM Pseudo-code

```
Input:
 - Demand Batches (DBs), each containing information on its:
   o Product Type (SKU)
   o Destination
   o Priority Class (PC)
   o Demand Type (DT)
   o No. of units in Batch (Demand)
   o Customer Required Week (CRW)
   o Lead Times (LT)
   o Ideal Buy Week (IBW) [= CRW – LT]
   o Maximum Earliness (ME)
   o Maximum Lateness (ML)
   o Initial Capacity Consumption Week (ICCW)*
   o Done Buying Date (DBD)*
   o Capacity Type (CT)
   o Product Activation Timestamp (PAT)
 - Production Minimum (PM), per SKU
 - Destination Minimum (DM), per SKU, per Destination
 - Factory Resource Capacity (FRC), per CT, per week
 - Planned DBs, per week

Step 0.
 - 0a. Split up the data according to Priority Class (high to low: Promo,
   Launch, Precise, Short Lead Time, Standard, Flexible);
 - 0b. Sort the DBs within each Priority Class (PC), according to:
   o Demand Type (high to low: Direct Shipments, Standard, At Once, Other);
   o Ideal Buy Week;
   o Product Activation Timestamp – tiebreaker if the other two are the same.
 - 0c. Calculate the Available FRC (AFRC), per CT, per week by subtracting
   the Planned DBs' Demand per CT, per week, from the respective FRC;
 - 0d. Select the first DB in the first PC & Go to Step 1.

Step 1. Initialize by setting t = Ideal Buy Week (IBW) for the current DB
 - 1a. If the AFRC in t for the CT of this DB is greater than or equal to the
   Demand in this DB:
   ➢ Go to Step 2.
   ➢ Otherwise, go to Step 1b.
 - 1b. If the AFRC in t for the CT of this DB is greater than or equal to the
   PM of this DB:
   ➢ Split the DB into one part which equals the AFRC in t for the CT of
     this DB and one part which equals the rest;
   ➢ Set the current DB to the first of the two new DBs, set the next DB in
     line to the other DB & Go to step 1.
 - 1c. If t equals the IBW or is between IBW and the lower limit (IBW-ME):
   ➢ Set t = t - 1 & Go to step 1a.
   Else, if t is equal to the lower limit (IBW-ME):
   ➢ Set t = IBW + 1 & Go to step 1a.
   Else, if t is greater than the IBW but lower than the upper limit (IBW+ML):
   ➢ Set t = t + 1 & Go back to step 1a.
   Else if t is equal to the upper limit (IBW+ML):
   ➢ Set the DB status to 'Unplanned due to Capacity'.
 - 1d. If there is another DB in the current PC:
   ➢ Select it & Go back to Step 1.
```

Else, if there is another PC:
- Select it and its first DB & Go back to Step 1.

Else, End Code.

Step 2.

- 2a. *If* the DB Demand is less than its PM** AND there is no other non-Promo DB planned in t with the same SKU:
  - Go to Step 2b.
  - Otherwise, Go to step 2c.
- 2b. If there are any future DBs of the same SKU in this or any lower PC that have a planning horizon (IBW-ME to IBW+ML) which includes t:
  - If possible, add as many of those DBs to the current selection to meet the PM, starting with the closest IBW to t & Go to step 2c.
  - Otherwise, set DB status to 'Unplanned due to Production Minimums' & go to 1d.
- 2c. If the DB Demand is less than its DM*** AND there is no other non-Promo and non-DRS DB planned in t with the same SKU AND Destination:
  - Go to Step 2d.
  - Otherwise, go to Step 3.
- 2d. If there are any future DBs of the same SKU AND Destination in this or any lower PC that have a planning horizon (IBW-ME to IBW+ML) which includes t:
  - If possible, add as many of those DBs to the current selection to meet the DM, starting with the closest IBW to t & Go to step 3.
  - Otherwise, set DB status to 'Unplanned due to Destination Minimums' & go to 1d.

Step 3.

- Set the current DB(s) to 'Planned';
- Reduce AFRC for the CT, for t by the Demand of the DB(s);
- Go to step 1d.

* *Usage explained in Chapter 3.1.4.3*
** *Not applicable for Promo PC*
*** *Not applicable for Promo PC and Direct Shipment DT*

3.2. Optimization Objectives

3.2.1. Leadership Team Input

In order to be able to construct an optimization solution, the objectives of the product planning have to be mapped. To do so, the S&IP leadership team was asked the question which objectives they would like to see optimized. This led to a substantial list of items as seen in Table 3.1. Most of the concepts in the table, such as Unplanned, should be familiar by now. However, there are some new metrics and abbreviations in this table which have not been used before. These will be explained briefly.

The given objectives can be split up in roughly three groups. The first group are concrete values which can be quite easily measured given the availability of the values necessary to calculate them, of course. The second group are metrics which can all be expressed in a direct percentage of the Demand Plan which is in place. The third group are meta-measurements, which look at deviations and stability of certain factors.

Table 3.1: List of Proposed Objectives

Objective Name	Measure	Function
Unplanned	% of DP	Minimize (-)
Lateness	% of DP	Minimize (-)
Earliness	% of DP	Minimize (-)
Expected Inventory	Units*weeks	Minimize (-)
Revenue	$	Maximize (+)
Profit Margins	%	Maximize (+)
PIFOT	% of DP	Maximize (+)
Target Days Shipment	% of DP	Minimize (-)
Airfreight necessary	% of DP	Minimize (-)
UP due to Production Mins	% of DP	Minimize (-)
UP due to Destination Mins	% of DP	Minimize (-)
UP due to Cap	% of DP	Minimize (-)
UP due to LT	% of DP	Minimize (-)
PIFOT Deviation	SD of PIFOT between runs	Minimize (-)
Buy Plan Deviation	Aggregate SD of Buy Plan volumes per SKU between runs	Minimize (-)
Parameter Deviation	SD of parameter settings between runs	Minimize (-)
Parameter Utilization	% of SKUs with non-standard parameter settings	Minimize (-)
Production Capacity Utilization Variance	SD of Cap. Util.	Minimize (-)
Customer Service Level Variance	SD of Destination Service Levels	Minimize (-)
Minimums Shortage	% increase of DP necessary	-

Of **group 1**, colored yellow in Table 3.1, the first three objectives are quite self-explanatory: unplanned are materials which have not been included in the buy plan due to possible reasons described in Chapter 3.1.3.4. Lateness and Earliness are measures of how many weeks demand is produced before/after the Customer Required Date (CRD). The Expected Inventory is the sum of the time which products spend in the warehouse due to early production. Revenue is the sum of the expected selling price of the produced goods. Profit Margin is the average expected cut of the Revenue which is higher than the costs of the production of the goods. PIFOT stands for Planned In Full On Time and is a measure of how many units are not late for the CRD.

Since PIFOT is the main metric used to evaluate the quality of the Buy Plan now, it might be good to explain it a bit further. It is calculated by taking the portion of the Demand Plan (DP) which has been planned to be delivered in time for the CRD. All demand that has been planned to be produced more than three weeks early is labeled ‘Built Early’ (BE), demand that has been planned to be produced up to three weeks early, or up to one week late is labeled Just In Time (JIT) and demand that is planned to be produced more than one week late are labeled ‘Build Late’ (BL). The calculation for PIFOT for any given time t is therefore as follows. Note that this can be done for any time interval, such as a week, an entire season or even an entire year if needed. This is because all parts of the calculation can be measured on this same timescale.

$$PIFOT_t = \frac{BE_t + JIT_t}{DP_t} * 100\% = \left(1 - \frac{BL_t}{DP_t}\right) * 100\%$$

The objectives in **group 2**, colored blue in Table 3.1, can be measured fairly straightforwardly as they can all be expressed as a portion of the total Demand Plan. Target Days Shipment (TDS) is a parameter setting which allows the S&IP team to artificially pull demand earlier than necessary. By doing this, the Ideal Buy Week (IBW) of the demand is changed, causing it to be prioritized earlier in the CTM run. However, this will result in earlier deliveries as well. Airfreight necessary can be calculated by taking the portion of the DP which has Lead Time issues, but still falls within the time window that it can be delivered on time in order to meet the CRD. This can only be done for Launch and Promo products though.

The following five metrics are all specific instances of the Unplanned metric. All of these can be expressed as a percentage of the total DP as well. These different Unplanned reasons are explained in detail in Chapter 3.1.3.4. The miscellaneous Unplanned reasons which are described in that chapter are missing from this objective list. This is because these are independent of the planning logic. For example, if a Done Buying Date (DBD) is entered in the system, it does not matter which logic is used: if the DBD is passed, it cannot be planned anymore.

Group 3, colored green in Table 3.1, could be described as a set of meta-objectives. These are all measurements of deviations and/or stability over time. PIFOT Deviation should reflect the Standard Deviation (SD) in PIFOT scores when comparing two optimization runs. Buy Plan Deviation should reflect the aggregate SD of the weekly Buy Plan output when comparing two optimization runs.

Parameter Deviation is not caused by the Planning logic itself, but actually happens in between optimization runs. It is however a reaction to the Planning logic’s output as the parameters are adjusted to remedy Unplanned issues as discussed in Chapter 3.1.3.4 as well. It can be measured by taking the SD of the input parameter settings between two optimization runs.

Parameter Utilization, Production Capacity Utilization Variance and Customer Service Level Variance are metrics that can be measured within one optimization run, unlike the others. Parameter Utilization measures the portion of Demand Batches (DBs) which have parameter settings that are different from the default settings. Capacity Utilization Deviation measures the SD of the weekly Capacity Utilization at a factory. Customer Service Level Variance measures the aggregate SD of product Coverage between different destinations. Product Coverage is the portion of the Demand Plan that is planned overall, irrespective of timing.

Grouped separately from the three main groups is the factor Minimums Shortage. The idea is to measure how many units needed to increase the Demand of the materials that are Unplanned due to Production Minimums (PM) or Destination Minimums (DM). However, this is independent of the planning algorithm used. This is because the minimums requirements have to be met no matter the logic. This one is therefore immediately disregarded as a possible objective of the optimization.

3.2.2. Finalized Objectives

The same Leadership Team was asked to prioritize the aforementioned list from most to least important objectives. Of this prioritized list, a feasibility check was done by the author. This resulted in the list shown in Table 3.2. A lot of the proposed objectives were excluded from the final model. A brief explanation per exclusion reason is given. The finalized list of objectives is shown in Figure 3.3.

The first reason in order of appearance is self-explanatory: no accurate data of prices and margins was available for the goods. This is partly because actual selling prices can still be altered before the season. Furthermore, products can be sold for different prices, depending on the account, the timeliness of the order and whether it was sold at full price, was discounted or liquidated. All in all, there are too many uncertain factors to be able to use this in the final model.

Both PIFOT related objectives were excluded because their values can be completely derived from the Unplanned, Lateness and Earliness objectives. It could be argued that the inter-run compared PIFOT scores could be useful, but that would hardly be an optimization objective, but rather a performance indicator of the stability of a combination of the Demand Plan and the Planning logic.

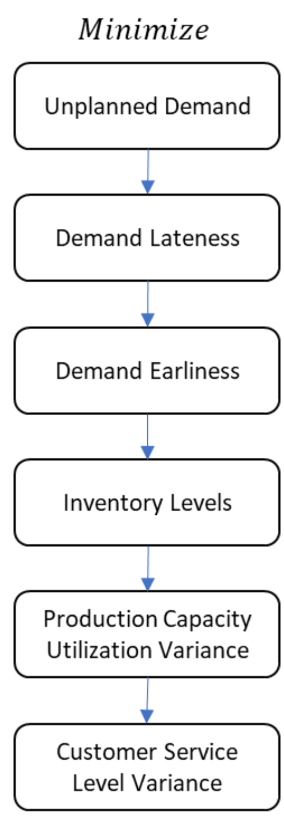

Figure 3.3: Final set of optimization Objectives

Table 3.2: Prioritized list of objectives, including decision to include or not

Objective Name	**Priority**	**Inclusion in model**
Unplanned	1	Yes
Lateness	2	Yes
Earliness	3	Yes
Expected Inventory	4	Yes
Revenue	-	No; missing accurate pricing data.
Profit Margins	-	No; missing accurate pricing data.
PIFOT	-	No; fully dependent of other metrics.
Target Days Shipment (TDS)	-	No; too circumstantial.
Airfreight necessary	-	No; too marginal.
UP due to Production Mins	-	No; specified below.
UP due to Destination Mins	-	No; specified below.
UP due to Cap	-	No; specified below.
UP due to LT	-	No; specified below.
PIFOT Deviation	-	No; fully dependent of other metrics.
Buy Plan Deviation	-	No; too circumstantial.
Parameter Deviation	-	No; too circumstantial.
Parameter Utilization	-	No; too circumstantial.
Production Capacity Utilization Variance	5	Yes
Customer Service Level Variance	6	Yes
Minimums Shortage	-	No; as explained before.

The TDS, Buy Plan deviation and Parameter related objectives were excluded because of two reasons. First, they are not caused by the Planning logic. It could be argued that they are reactions to the Planning logic output, but even then, it would take time for Planners to adapt their use of these levers to a new method. Therefore, it would be impossible to measure without actual implementation over a long period of monitoring. The second reason is that they are Planner-dependent. Some Planners might have a specific approach to applying these levers. For example, some planners increase the Maximum Lateness parameter for many, many weeks of a single product's demand, such that they only have to do it once, rather than only in response to when it does not get planned. All of these objectives were therefore deemed too circumstantial.

The Airfreight necessary was deemed to marginal a measure. Though its impact on transportation costs and carbon emissions is not to be dismissed, this will be improved by reducing Lateness as much as possible if anything. Since Airfreight is only applicable for Promo and Launch materials, it is important that these are still prioritized in the final planning logic though. This will be further explained in Chapter 3.5.2.

Finally, the zoomed in Unplanned reasons were all excluded because they are all dependent of other factors, plus it does not matter to anyone what the reason is why something is Unplanned in terms of performance. The only reason these are important, is such that the planners know how to resolve them. If all Unplanned is caused by Capacity, or it is caused by Lead Time, yet the numbers are the exact same, the performance will be deemed equal.

This final list of prioritized objectives was shared and discussed with the S&IP Leadership Team and was unanimously agreed upon as a good set of objectives to pursue. However, there were doubts of whether the two variance objectives were to be pursued directly. Reason for this is that it is thought that these would be improved automatically by pursuing the other four objectives. For now, these are left in the final objective list, but could be subject to change later.

3.3. Optimization Approach

3.3.1. Decision for MIP

As found in the literature review and specifically in Chapter 2.2, mathematical programming is a preferred option for dealing with production planning problems. Mathematical programming includes, but is not limited to, Linear Programming, Integer Linear Programming and Mixed Integer Programming. The application for each of these is specific to the problem at hand. Some of these will be discussed in order to provide a basic knowledge for the Mixed Integer Programming method applied in this research.

3.3.1.1. Linear Programming (LP)

The most fundamental of these mathematical programming methods must be the Linear Programming Model. It can be used to optimize a mathematical objective under certain constraints, as long as all of these functions are linear.

An example of this could be an objective function which maximizes a formula with two Decision Variables ($\max Z = 10x - 2y$), two Constraints ($x \leq 4;\ y - x \geq 1$) and a non-negativity constraint ($x, y \geq 0$). These constraints limit the values that the variables can take to specific ranges. Such a mathematical programming problem has an optimum solution for at least one combination of values of x and y (in this example, $x = 4,\ y = 5,\ Z = 30$). Given the constraints and the objective, there is no other solution that will yield a higher result. In Linear Programming, the variables can take any value in $\mathbb{R}$, the set of real numbers, for the solution.

3.3.1.2. Integer Linear Programming (ILP)

In Integer Linear Programming, the values which the decision variables can take are limited to $\mathbb{Z}$, the set of integer values, including negative numbers and zero by default. In practice however, non-negativity is applied, leaving only $\mathbb{W}$, the set of integer positive values and zero as possible values for the variables. This setup can be useful when the variables represent units which cannot be split, such as a number of goods to produce, trucks to buy or people to employ.

Take for example the number of people that work during a shift. Though an employee can work half shifts technically speaking, this might be contractually forbidden. Therefore, the solution must be a whole integer number of employees working on a shift.

To solve this optimization, the problem is often 'relaxed' to exclude the integrality constraint. When disregarding this constraint however, the optimal answer might be a fraction (e.g. $x = 3.7$) – as it often is. To go from that fraction to an optimal integer solution, new constraints are implemented ($x \leq 3$; $x \geq 4$) to exclude the fraction. Often this results in one of the limits being the optimal solution but can result in other solutions of course. In such cases, the adding of constraints is repeated until an integer solution is found. This method is called the **Branch and Bound** method (https://gurobi.com/resource/mip-basics/).

3.3.1.3. Mixed Integer Linear Programming (MILP)

Moving along, there are sometimes combinations of integer value variables and continuous value variables. These can be combined into a mathematical program and are called Mixed Integer Linear Programs.

An example of this could be the aforementioned factory which produced integer numbers of products but allows the employer to send home an employee if there is not any work to do anymore. Thus, allowing shifts that are non-integer values. It is quite common to see MILP implementations. The optimal solution of an MILP is found through similar methods as an ILP, as the Integer variables must still adhere to integrality, which can be found using the Branch and Bound method described before.

3.3.1.4. Quadratic Programming (QP)

Quadratic Programming gives rise to many new possibilities. The QP model can be used to solve optimizations that include interdependent variables and non-linear objectives. This can be further expanded to Quadratically Constrained Quadratic Programming (QCQP), which also allows for non-linear constraints to be used in the optimization.

The methods of solving these types of programming become severely more complicated, compared to the previously discussed mathematical programming methods. These methods include the use of relaxation of constraints, as discussed before, heuristics to find local optima and vast enumeration methods.

3.3.1.5. Mixed Integer Programming (MIP)

The Mixed Integer Programming method is slightly different from the MILP method discussed before. Where MILP requires all of its constraints and objective function to be fully linear, the MIP allows a combination of linear and non-linear constraints.

In the proposed solution, there was seemingly no need to include Quadratic Constraints or a Quadratic Objective Function. In fact, the proposed solution is practically a MILP solution. However, in order to be able to include the Minimum Order Quantity (MOQ) constraints, quadratic functions had to be used where Binary variables were multiplied with decision variables. The exact workings of these constraints are explained in Chapter 3.6.4.

3.3.2. Gurobi Solver and R Studio

To solve the Mixed Integer Programming solution approach, the best option is using a solver such as CPLEX or Gurobi. The author chose to use Gurobi due to experience with the solver in the past, as well as expansive online documentation on the workings, compatibility with many different software programs and, of course, the fact it has free student licenses.

Once downloaded, the Gurobi installation folder contains many example codes which come in handy to help understand the way the program works. These examples are written in many different coding languages, such as Python, C++, R Studio, Java and C#. Again, because of the author's experience with the program, the choice was made for R Studio. R studio is a free, open-source piece of data analysis software. It does currently support Python as well, but the R programming language was used for this research.

3.4. Dataset

To be able to design and test the proposed MIP solution, data had to be collected from the system. The decision for Footwear products over Apparel and Equipment products was already touched upon in Chapter 1.6.1.

The choice to take a dataset from the beginning of a season, such that the season 6 months from that point is still as clean as possible was also explained. By doing so, the Maximum Earliness (ME) and Maximum Lateness (ML) parameter settings for that season are still mostly untouched, as well as that demand is not altered to fit the constraints better. Aside from this, picking this moment in time also allows the proposed solution to show whether it can improve the planning for the season that is 3 months from that point in time, which is already heavily adjusted to meet the CTM logic. The different variables which can be used to narrow down all historical data are described in the next subchapters.

3.4.1. Factory Level

As discussed in Chapter 1.6.1, the Footwear Product Engine (PE) has more variables in control than the other two. This is mostly because these factories are not shared with other companies and as a result, the resource capacity cannot be shifted between companies if it becomes available.

The choice for a factory was made based on absolute demand values at the factories, as well as a good representation of diversity in types of products. By doing so, any issues from more difficult to create products (e.g. shoes with Air cushions or Knit outsides) will interfere with performance in the sample. This is a perhaps a more extreme representation of the overall demand, but allows for a demanding performance from both the CTM logic and the proposed solution

The types of factors which are considered to choose the factory for the datasets is:

- Production volumes (absolute);
- Number of different SKUs within the season (absolute);
- Relative product complexity, such as Air or Knit materials (% of total);
- Inclusion of high priority (Promo & Launch) materials (% of total);

At the time of exploring the data, only planning data from the Europe GEO was available. That is why, the comparison of the different factories and seasons is done solely based on these numbers. However, it is likely that the other Geographies have comparable numbers for these factories, as they generally do sell the same products, which need to be produced at the same factories.

3.4.2. Time

Each week has two CTM runs, both with input and output. The midweek CTM run does not take some changes in parameters into account, which the weekend CTM run does. Because of this, the parameters are usually not fully updated for a midweek CTM run. In order to give the CTM a fighting chance, a midweek engine run dataset should not be chosen.

There are 13 weeks within one season, of which typically 6 are buy weeks and 7 are planning weeks. A buy week is different from a planning week in the sense that Purchase Orders (POs) are placed on the Wednesday in the middle of the week. The planning parameters are generally left the same in a buy week. In a planning week however, the CTM engine outputs the new Buy Plan after the initial weekend run. This gives the planner insights into what actions should be taken in order to achieve a better Buy Plan. Using the same logic, the dataset just after a planning week is the most updated and allows for a superior result from the CTM logic when compared to that at the end of a buy week. That would mean that the focus can be put on the weekend CTM runs which precede the buys.

Following the logic posed in Chapter 1.6.2, this means that the best dataset is that just before the first buy week of a new season. Because of the moment in time that the dataset is pulled from the system, a dataset that a good mix of unaltered input variables and variables that have already been adjusted to optimize the CTM logic runs, Summer 2019 is the best option. This means that any demand for Fall 2019 is already optimized, yet the demand for Holiday 2019 is not altered yet. Since (high-level) forecasts until January 2021 are already in the system, this allows for a dataset which is expansive enough as well.

3.4.3. Comparison and Decision

The factors discussed in Chapter 3.4.1 were taken into consideration. All Footwear factories were compared for the Summer 2019 Dataset. The values in Table 3.3 show the relative rankings of all the factories for each factor. By taking the sum of these rankings, a total score is calculated per factory. The factory with the lowest score has the highest average ranking among all factories.

The factory came out as the most interesting in terms of high demand and high variation was factory VO.

Table 3.3: Factory rankings per decision factors, summed to find the best (minimum) factory

2019SU	VO	JJ	VJ	VT	XC	VL	VP	QD	QT	LN	MX	MD
Volume of DP	5	3	2	1	9	4	8	7	6	10	11	12
# of SKUs	3	5	1	2	4	6	7	9	10	8	11	11
Promo (%DP)	3	12	9	8	4	7	11	5	10	6	2	1
Launch (%DP)	6	11	8	5	7	12	10	2	9	1	3	4
Flyknit (%DP)	4	10	7	5	6	11	9	12	3	8	1	2
Air (%DP)	6	7	8	9	5	10	3	4	1	2	12	11
Zoom (%DP)	7	6	5	8	1	2	4	3	9	9	9	9
SUM	**34**	54	40	38	36	52	52	42	48	44	49	50

3.4.4. Dataset Collection and Description

The data was pulled directly from Nike's Teradata environment. Teradata is an SQL User Interface that allows for the same use of SQL queries as usual. Teradata does have an Explorer view of all tables and the columns that exist within each of the tables with the format of the value.

The data which had to be collected was all the input needed for the Benchmark and MIP planning logic. The necessary data consists of the following list. The mathematical sets used are defined in parentheses. The only set that is not defined is time. That is because the optimization logic uses the Ideal Buy Week (IBW) for time. The calculation of this is explained in Chapter 3.4.5.

- Demand Batches (DB; $i \in I$; index number) that needs to be planned, including:
  - Product Type (SKU_i; $j \in J$; alphanumeric code);
  - Destination (Loc_i; $l \in L$; numeric code);
  - Demand Type ($DT_i \in \{\text{DRS, Standard, AO, Other}\}$);
  - Demand Quantity ($D_i \geq 0$; integer pairs of footwear);
  - Customer Required Date (CRD_i; date);
  - Product Activation Timestamp (PAT_i; datetime).

- Placed POs at Factory, including the same data as the DBs, except for CRW and PAT;

- Transportation Lead Time, per Destination (TLT_l; integer days);

- Product attributes, per SKU, including:
  - Manufacturing Lead Time (MLT_j; integer days);
  - Priority Class ($PC_j \in \{\text{Promo, Launch, Precise, SLT, Standard, Flexible}\}$);
  - Initial Capacity Consumption Week ($ICCW_j$; date);
  - Capacity Type (CT_j; $c \in C$; alphanumeric code);
  - Production Minimum (PM_j; integer pairs of footwear);
  - Destination Minimum, per destination (DM_{jl}; integer pairs of footwear);

- Parameter Settings, per SKU, per destination, including:
  - Maximum Earliness, per week (ME_{jlt}; integer days)
  - Maximum Lateness, per week (ML_{jlt}; integer days);
  - Done Buying Date (DBD_{jl}; date).

- Factory Resource Capacity (FRC), per CT, per week
  - Total Factory Resource Capacity (Cap_{ct}; integer pairs of footwear)

3.4.5. Data Preprocessing

The IBW is calculated by taking the Customer Required Date (CRD), subtracting the sum of the Lead Times for that SKU and Location from that and converting it to a week number. The first IBW in the dataset is the week of April 20th, which is week 16 of 2019. This week is set as $t = 1$. All of the IBWs in the dataset together comprise the total time set to which each t belongs: $t \in T$.

All time related variables are converted to week numbers in order to be able to express them in terms of t. Since the week of April 20th, 2019 is set as $t = 1$, some values have to be adjusted. Any ME_{jlt} value that would push the earliest possible order moment to a time before $t = 1$ is set such that the earliest moment is equal to $t = 1$. This is realistic as in practise, it is impossible to order products in the past.

$ICCW_j$ and DBD_{jl} are not filled in necessarily. This means that sometimes, these have date values, and sometimes these are NULL. If an $ICCW_j$ date is earlier than the week of $t = 1$, it is set to 0. If a DBD_{jl} date is earlier than the week of $t = 1$, it is set equal to 1 as well.

The Placed PO data is not used directly in the Benchmark or the proposed solution, but the Available Factory Resource Capacity (AFRC) per Capacity Type (CT) per week is calculated by subtracting the sum of the PO volumes placed in the corresponding weeks from the Total Factory Resource Capacity (TFRC).

The Production Minimum (PM) and Destination Minimum (DM) are pulled out of the system per SKU and SKU-Destination combination respectively. However, these values are converted to vectors for all $t \in T$, because they have to be met on a weekly basis. These values per week are also reduced with the number of demand which is also on PO for the same week and SKU (and SKU-Destination combination for DM). In this case, Promo POs are subtracted from the PM and DM for a week, because they can help meet the MOQs. DRS POs are considered for Production Minimums but not for Destination Minimums.

In order to be able to sort by the Priority Class and Demand Type, the classes were converted to a numerical ranking. This was done in the order of the classes as shown in the list above (so Promo = 1, Launch = 2, etc.). For the MIP model, the usage of the priorities is different than for the Benchmark. This is further explained in Chapter 3.4.2.

Samples of the data can be found in Appendix A.

3.5. Research Problem to Model

To build the actual model, all abstract elements had to be changed to a set of mathematical concepts. These concepts include the decision variables, constants, objective function, constraint functions and the splitting of the overall time horizon into smaller “Timechunks”.

First, the reasoning which led to the objective function is discussed in Chapter 3.5.1. This includes the use of penalties and how the values of these penalties are chosen. The implementation of the prioritization of the demand is also discussed here. This specifies how the Priority Class, Demand Type and the First Come First Serve logic are adapted for the MIP.

Chapter 3.5.2 will focus on the constraints which needed to be implemented for the MIP to function properly. This chapter will include explanations of how the MIP can work with Unplanned demand if needed, the factory resource capacity constraints which need to be considered and the minimum order quantities (PM and DM) which must be adhered to.

The prioritization of DBs with earlier Ideal Buy Weeks (IBWs) should be discussed as well. The nature of an MIP is to move everything around to make space. This is also the main reason why linear programming can find optima which are impossible to find by using a sequential planning logic such as the CTM algorithm. It would therefore be nonsensical to remove this ability from the suggested MIP by implementing extra penalties for demand that is sitting earlier. However, there is still a way to take it into account, all the while reducing the computation time of the MIP as well. Doing this by splitting the full dataset according to the Ideal Buy Weeks (IBWs) and the ensuing logic required to still run all the DBs in the dataset successfully, is described in chapter 3.5.3.

3.5.1. Objective Function

The objectives which came forward out of the communication with the S&IP leadership team which were discussed in chapter 3.2.1 must now be turned into mathematical concepts. In order of priority, these objectives were to minimize:

1. Unplanned Demand
2. Demand Lateness
3. Demand Earliness
4. Inventory Levels
5. Production Capacity Utilization Variance
6. Customer Service Level Variance

In order to implement objectives 1 through 3, penalties or costs had to be connected to demand. Because of the hierarchy of the objectives, these penalties had to be different for each of these objectives. In other words, specific values of penalties had to be appointed to demand which would fall into the Unplanned bucket, to the demand that would be late and to the demand that would be early.

The objective of minimizing inventory levels is slightly different than the aforementioned objectives. However, by minimizing the Demand Earliness, the Inventory Levels are also reduced.

The first two objectives are contradictive to these inventory minimizing objectives though. In order to be able to strike a balance between these objectives, a penalty scale had to be devised for each week of earliness and lateness.

The implementation of the last two objectives is far less straightforward. The values which are described by these objectives are on a completely different level as the first four, as they are both relative, rather than absolute scores. Including these in the same mathematical objective function would therefore be illogical. This is because any absolute penalty value given to the relative values might either completely overthrow the higher priority objectives or have no impact at all.

A method to implement these objectives would be by implementing a heuristic. This would then need to be able to take the output from the MIP optimization as input, allow for some predefined deterioration of the penalty (e.g. 1%) and find local optima for these values. This is very similar to what was done by Lovgren and Racer (2000). Unfortunately, the method they describe which would be best suitable for this implementation, the Border Swap Technique, was already mentioned to have long computation times for small batched of demand. Let alone, for some 10,000s of Demand Batches, encompassing millions of units of demand. Because of these computational limits, it was chosen to use these objectives only as KPIs of the results, rather than to actively pursue the minimization of them.

Aside from the penalties which are necessary to weigh the different objectives, the original CTM approach used Prioritization of demand to decide which to plan first. These can be priorities can be implemented in the MIP as well. This could be done by setting either hard or soft priorities. The choice for this is explained further in Chapter 3.5.1.2.

3.5.1.1. Penalty Selection

The proposal is to create an optimized planning, based on penalties, rather than costs. The reasoning for this is two-fold. First of all, costs optimization never paints the full picture. In this case, we assume that we want to produce as much of the demand as possible and if possible, on time. Holding costs, lost sales costs and costs for missing sales would be too big of an assumption to implement. The second reason is that the costs for holding inventory are different per destination, lost sales due to unplanned might be translated to increased demand for other products and lateness might not have a direct effect on the portion of the sales that can’t be made

The Logic

Penalties are very similar to these costs, but can take roots in logic, rather than monetary value. Especially, when considering that a hierarchy was decided upon by the S&IP leadership team. The logic used to define the values that the penalties must take, is therefore as follows.

- Unplanned Demand must be penalized more than any Earliness or Lateness can cause;
- Lateness should be penalized heavier than Earliness, but:
    - It is not the case that any amount of earliness should be penalized less than any amount of lateness.

This last point is required some further investigation. After further consultation with the S&IP leadership team, it was decided that the penalty for three weeks early production should equal the penalty for one week of late production. It makes sense that this would be the preference as the Just In Time (JIT) borders which are in place at the moment coincide with these preferences.

The fourth objective (Minimize Inventory Levels) is not explicitly included in the mathematical objective function. It can be used in the logic to decide on the increments that the Earliness Penalty should follow though. Since Inventory is used up linearly by materials that are sitting in the warehouse, it would make sense to follow linearity for the Earliness Penalty.

For the lateness penalty, there is a question of whether the proportion of the sales that are lost by lateness are linear or exponential. The second would make more sense, but there is no evidence in the company whether this is the case. As a matter of fact, demand is often taken just as well if it's late. However, there are more discounts applied to the products if they are still there at the end of the season. The relationship of revenue and lateness is not clear. Because there is no clear reasoning to choose otherwise, the Lateness Penalty is chosen to be linear, just as the Earliness Penalty.

Since a season is always 12 to 13 weeks, it was decided that a DB which is 12 weeks late is as valuable as being not planned at all. This makes sense since neither an unplanned batch nor a 12 weeks late batch will be sold, logically speaking. This makes it so that neither a late nor an early batch can ever cause a higher penalty than an unplanned batch. In reality, 12 weeks of Maximum Lateness (ML) parameter is hardly ever set unless there is a specific ask for by the Global office. Since these requests still have to be able to be processed, it is necessary that the proposed solution allows these 12 weeks of lateness, albeit very costly in terms of penalties.

The Penalty Values

Based on the logic above, the Unplanned/Lateness/Earliness penalties were chosen. Since the penalties are not linked to real-life numbers, an arbitrary number can be chosen for each of them. In this case, the following must hold according to the logic:

$$P_U = 12 * P_L = 3 * (12 * P_E)$$

The penalty for Unplanned demand (P_U) is a constant value, this could be 1. The penalty scores (P_L for Lateness, P_E for Earliness) are linear and based on the number of weeks they are late ($t - IBW_i$) or early ($IBW_i - t$), respectively. In this case, the number of weeks that demand must be late to get the same penalty score as being Unplanned is: $t - IBW_i = 12$, were t is the week in which the Demand is planned. Since a result of three weeks of Earliness should have the same penalty as one week of Lateness, $P_E = \frac{P_L}{3}$ must hold.

3.5.2. Constraints

In order to make sure that the values of the decision variable don't assume nonsensical values, constraints had to be put in place as well. The first constraint makes sure that demand is either met or falls into the unplanned bucket. The second constraint ensure that the Factory Capacity is kept to. The third constraint is responsible for the adherence of the demand volume planning to the Minimum Order Quantities. A final chapter is dedicated to the extra constraints that are necessary to tie all the loose ends together. These are however not representations of rules which are designed.

3.5.2.1. Demand Constraint

The demand constraint is fairly straightforward: all demand should either be planned or fall in the unplanned bucket. The Demand within a Demand Batch (DB) can however be split up over several weeks (time buckets). The demand of DB i which is planned to be produced in time bucket t is represented by the decision variable μ_i^t. Any demand of DB i which falls in the unplanned bucket is represented by the decision variable λ_i. The sum of all demand covered in different time buckets of DB i and the unplanned demand must be exactly equal to the total demand in the DB, D_i. Since Demand can not be planned before the earliest limit ($IBW_i - ME_i$) or after the latest limit ($IBW_i - ML_i$), the summation is taken only for that time window. This must hold true for all $i \in I$.

An extra element was added to this constraint for the purpose of adhering to the Minimum Order Quantities later. This element is the binary variable α_i^t which denotes whether or not any demand of DB i is planned in period t. If this is the case, it must take the value 1, otherwise it can take the value 0. If it were to take the value 0 while $\mu_i^t > 0$ though, the total Demand covered in this period would be negated for this constraint. The value of μ_i^t would then only take up Capacity in the system, blocking other demand without any use. Since the MIP will not do this as it harms the objective function, the addition of the α_i^t does not harm this constraint.

This results in the following Demand Constraint.

$$\sum_{t=IBW_i-ME_i}^{IBW_i+ML_i} (\mu_i^t * \alpha_i^t) + \lambda_i = D_i, \ \ \forall i \in I$$

3.5.2.2. Resource Capacity Constraint

Unfortunately, the data for the available capacity per Capacity Type is not kept historically. That means that it is updated from run to run. In other words, since the exact Capacity Type utilization was not extracted on the same date as the dataset's t_0 (April 13th, 2019), it was impossible to include this specific information into the optimization logic.

However, the overall Capacity is available. This means that overall capacity can still be considered for the optimization. The Benchmark was also run without the Capacity Type specific Capacity constraints, only with the overall Capacity. This is extra confirmation that the creation of the Benchmark was necessary to create a fair comparison between the CTM logic and the proposed Optimization Algorithm.

The constraint to make sure that the available Capacity is not surpassed at any given moment is also fairly straightforward. The constraint simply demands that the summation of all demand planned is smaller than or equal to the available Capacity in that week (Cap_t). This must hold true for each week $t \in T$.

This results in the following Capacity Constraint.

$$\sum_{i \in I} \mu_i^t \leq Cap_t, \;\; \forall t \in T$$

It should be noted that for Capacity Type (CT) specific Capacities, an extra constraint would have to be added. This extra constraint would need to include a binary variable which denotes whether or not a DB i is part of that CT. Then the equation would be similar to this one, requiring the sum of all planned demand μ_i^t which sits in each specific CT c to be lower than or equal to the available capacity for that CT in that week.

3.5.2.3. Minimum Order Quantities Constraint

The Minimum Order Quantity (MOQ) constraint was the most complicated out of the three to be implemented. Inspiration was taken from Omar and Teo (2007), where a similar approach was taken. The problem in this constraint sits in the fact that a demand which is to be planned in a certain timeslot, must adhere to the MOQ quantities if, and only if it is planned to be produced in that timeslot at all. If it isn't produced in that timeslot, the demand for that DB should be equal to 0.

This is where the binary variable α_i^t, which was added to the demand constraint, comes in handy. Similar to this variable, the variables γ_j^t and δ_{jl}^t were added. The first of these is used to signify if any demand of SKU j is produced in week t. The second is used similarly but adds the destination to the mix. These are created specifically for the MOQ constraints.

Aside from these binary variables which signify whether Demand is planned in week t for specific SKU j or a SKU-Destination combination (j, l) respectively, two more binary variables had to be added. These variables, β_{ij} and ϵ_{il} signify whether a DB i contains units of SKU j and whether a DB i is going to a Destination l, respectively.

By using all these variables, the constraint for the Production Minimums can be formulated as follows. The left-hand side of the inequality can never be smaller than the right-hand side of the inequality. If this would be the case, it would mean that (1) there is a production of SKU j in timeslot t (i.e. $\gamma_j^t = 1$), yet (2) the sum of *all* the Demand of the DBs $i \in I$ that in that week t which consist of SKU j would not be bigger than the Production Minimum PM_j^t for that SKU in that timeslot. This is evidently what we try to prevent. This must hold for all SKUs $j \in J$ and timeslots $t \in T$.

$$\sum_{i \in I} \alpha_i^t * \mu_i^t * \beta_{ij} \geq PM_j^t * \gamma_j^t, \ \ \forall j \in J, \forall t \in T$$

A very similar formula is presented for the Destination Minimums DM_{jl}^t. The only difference here is the inclusion of the ϵ_{il} variable on the left-hand side such that only demand which is of the exact SKU-Destination combination for which DM_{jl}^t is in effect is considered. On the right-hand side of the equation, γ_j^t is replaced by δ_{jl}^t for the same reason. This must hold for all SKUs $j \in J$ all Destination $l \in L$ and all time buckets $t \in T$.

$$\sum_{i \in I} \alpha_i^t * \mu_i^t * \beta_{ij} * \epsilon_{il} \geq DM_{jl}^t * \delta_{jl}^t, \ \ \forall j \in J, \forall l \in L, \forall t \in T$$

3.5.2.4. Other Constraints

The rest of the constraints are added simply to make the others work. These are as follows:

1. Each DB $i \in I$ can only contain one type of SKU $j \in J$;
   a. $\sum_{j \in J} \beta_{ij} = 1, \ \ \forall i \in I$
2. Each DB $i \in I$ can only contain demand for one Destination $l \in L$;
   a. $\sum_{l \in L} \epsilon_{il} = 1, \ \ \forall i \in I$
3. The binary variable γ_j^t, which signifies the production of a SKU $j \in J$ at time $t \in T$, must equal 1 in order for any DB $i \in I$, which consists of said SKU ($\beta_{ij} = 1$), to be able to produce in that time period. M signifies an arbitrarily large number, such as not to limit the number of DBs that can be produced in a timeslot;
   a. $\sum_{i \in I} \alpha_i^t * \beta_{ij} \leq M * \gamma_j^t, \ \ \forall j \in J, \forall t \in T$
4. The binary variable δ_{jl}^t, which signifies the production of a SKU-Destination combination of $j \in J$ and $l \in L$ at time $t \in T$, must equal 1 in order for any DB $i \in I$, which consists of said SKU-Destination combination ($\beta_{ij} = 1$ and $\epsilon_{il} = 1$), to be able to produce in that time period. M signifies an arbitrarily large number, such as not to limit the number of DBs that can be produced in a timeslot;
   a. $\sum_{i \in I} \alpha_i^t * \beta_{ij} * \epsilon_{il} \leq M * \delta_{jl}^t, \ \ \forall j \in J, \forall l \in L, \forall t \in T$
5. The decision variables μ_i^t and λ_i may only be integer values, including 0, for all $i \in I$ and all $t \in T$;
   a. $\mu_i^t, \lambda_i \in \mathbb{N}_0, \ \ \forall i \in I, \forall t \in T$
6. All binary variables may only have the values 0 and 1.
   a. $\alpha_i^t, \beta_{i,j}, \gamma_j^t, \delta_{j,l}^t, \epsilon_{i,l} \in \{0,1\}, \ \ \forall i \in I, \forall j \in J, \forall l \in L, \forall t \in T$

3.5.2.5. Priority Classes and Demand Types

In the original Planning Problem, the Priority Class and Demand Type are used as ranked variables which only dictate the order at which the Demand Batches should be handled. However, in practice it is possible that a very small amount of high priority demand blocks a large amount of lower priority demand. This is especially the case when Capacity is very tight.

The Logic

It is evident that higher priority materials should be given preference for being produced closer to their Ideal Buy Week. Especially when it comes to being produced earlier than necessary though, there is no apparent logic why it needs to be planned as close to its IBW as possible. After all, inventory sitting in a warehouse is treated all the same.

Because of the logic of the CTM approach, it is necessary to give buckets and prioritizations for it to work. In the MIP everything can be treated equal. However, the logic that some products are simply more important than others is sound enough to implement. Especially since the classification of the materials is already fully implemented in the system, it's a great opportunity to incorporate it.

The Priority Values

Instead of a hard priority, a soft priority will therefore be used in the objective function. This Priority value will be calculated by sorting all the combinations of Priority Classes (PC) and Demand Types (DT). This is done using the original CTM processing hierarchy. The values assigned to this ranked list are inversely proportional to the ranking. I.e. the lowest PC-DT combination get a Priority Value (PR_i) of 1, increasing by 1 for each step up the ranking.

This PR_i value can then be added to the objective function as a weight which increases the Penalty Values decided in the previous chapter. The sum of the total penalty is thus calculated after the application of the Priority weight.

The resulting objective function is as follows.

$$\min \sum_{i \in I} PR_i \left[\sum_{t=IBW_i-ME_i}^{IBW_i-1} [\mu_i^t * P_E * (IBW_i - t)] + \sum_{t=IBW_i+1}^{IBW_i+ML_i} [\mu_i^t * P_L * (t - IBW_i)] + P_U \lambda_i \right]$$

3.5.3. Iteration over Timechunks

Beside the Objective Function and Constraints as discussed so far, there are some other levers to pull. One of these is the amount of data to process in an MIP run. Considering the instability of future demand, it is far more important to make sure that the demand which must be ordered soon is planned correctly, that the demand further away. To that effect, the original CTM logic also takes the Ideal Buy Week into consideration for the sequence in which the DBs are planned. As briefly mentioned before, it would be unwise to add this to the MIP solution, as it is specifically the point of mathematical optimization that it can weigh its own decisions using the given objective and constraints.

In order to include a prioritization of earlier demand, the choice has been made to split the dataset up in smaller sets, which I call 'Timechunks', based on the Ideal Buy Week (IBW). By doing so, an optimization can be run for one Timechunk and the output of that run can be taken into the next Timechunk's run as input. Some careful considerations must be made though. There are two big issues with this approach:

1. Minimum Order Quantities;
2. Conflicting border cases.

3.5.3.1. Minimum Order Quantities (1)

If the full dataset is not included in the MIP run, MOQ may not be met because Demand Batches (DBs) cannot be combined. To make sure that this does not pose any problem, each Timechunk must include *all* DBs which consist of materials with the same SKUs as the demand in the Timechunk, if the Planning Horizon (PH) overlaps. The PH was discussed earlier, but just in case: this is the range in which a product can be planned, from the earliest moment ($IBW_i - ME_i$) to the latest ($IBW_i + ML_i$).

Example 3.3

> Let's say that a Timechunk contains IBW 1 through 8, and the DBs in this Timechunk contain SKUs A-100, among others. For each SKU in the Timechunk, the outer ranges of the Planning Horizons must be considered. Imagine that the extreme outer ranges of the Planning Horizons for SKU A-100 are $t = 2$ and $t = 12$.
>
> In that case, any other DB that consists of SKU A-100, which has any $t \in [2,12]$ in its own Planning Horizon, must be added to this Timechunk. By doing so, the MIP has the possibility to pull in those DBs if they are needed to meet the MOQs.

3.5.3.2. Conflicting Border Cases

If all the DBs in a Timechunk are planned and carried over to the next Timechunk, the MIP loses part of its functionality. Namely, to be able to make a weighted decision of demand to be planned when it conflicts. This is still done for all the DBs within each Timechunk of course but imagine a DB with an IBW in week 8 and one in week 9. There is no reason why the one in week 8 should be planned without considering the DB with an IBW in week 9 completely.

Therefore, the Timechunk should contain more DBs than the ones it will send over as output to the next MIP run. The choice was made to double the IBWs in a Timechunk with regard to which will be given as input for the next. This means that a Timechunk containing IBW 1 through 8, will only give the planning output of the DBs with IBW 1 through 4 as input for the next. The next Timechunk will then contain all materials with IBW 5 through 12 and will give the planning of the DBs with IBW 5 through 8 to the next MIP run. This will be repeated until the very last Timechunk is run.

3.6. Mathematical Model

3.6.1. Input Variables

- $i \in I$: the index of the full collection of Demand Batches in the data set $\{1,2,..,I\}$
- $t \in T$: the weeks in which demand can be planned for production $\{1,2,\dots T\}$;
- $PR_i \in \mathbb{N}$: a 'soft priority' weight of Demand Batch i;
- P_U: a set penalty score for Unplanned Demand;
- P_L: a set penalty score for Demand planned late;
- P_E: a set penalty score for Demand planned early;
- $IBW_i \in T$: The Ideal Buy Week of Demand Batch i
- ME_i: The Maximum Earliness of Demand Batch i, in weeks;
- ML_i: The Maximum Lateness of Demand Batch i, in weeks;
- D_i: The total demand in Demand Batch i;
- Cap_t: The total integer Available Factory Resource Capacity at time t, in units of Demand;
- M: An arbitrarily large number;
- PM_j^t: The integer Production Minimum for SKU $j \in J$ at time $t \in T$;
- DM_{jl}^t: The integer Destination Minimum for SKU-Destination combination (j, l) for $j \in J$, $l \in L$ at time $t \in T$.

3.6.2. Decision Variables

- μ_i^t: The integer number of units of Demand Batch i which are planned in time t;
- λ_i: The integer number of units of Demand Batch i that are Unplanned;
- α_i^t: The binary variable that signifies production of Demand Batch i at time t;
- γ_j^t: The binary variable that signifies production of SKU j at time t;
- δ_{jl}^t: The binary variable that signifies production of SKU-Destination combination (j, l) at time t;
- β_{ij}: The binary variable that signifies which SKU j is produced in Demand Batch i;
- ϵ_{il}: The binary variable that signifies to which Destination l Demand Batch i goes.

3.6.3. Objective Function

$$\min \sum_{i \in I} PR_i \left[\sum_{t=IBW_i-ME_i}^{IBW_i-1} [\mu_i^t * P_E * (IBW_i - t)] + \sum_{t=IBW_i+1}^{IBW_i+ML_i} [\mu_i^t * P_L * (t - IBW_i)] + P_U \lambda_i \right]$$

3.6.4. Constraints

1. Demand Constraint: For each Demand Batch $i \in I$, the sum of the planned units μ_i^t in the possible weeks they can be planned plus the unplanned units λ_i must always equal the total demand in DB_i: D_i. Another new variable, α_i^t, is introduced, this is a binary that must equal 1 if $\mu_i^t > 0$, because otherwise the constraint can't be fulfilled.

$$\sum_{t=IBW_i-ME_i}^{IBW_i+ML_i} (\mu_i^t * \alpha_i^t) + \lambda_i = D_i, \ \ \forall i \in I$$

2. Capacity Constraint: For each week $t \in T$, the sum of all units planned in that week may not exceed the capacity of that week: Cap_t. For this constraint, α_i^t should not be added because that would allow any μ_i^t to have a value other than 0, as it's simply cancelled out by an $\alpha_i^t = 0$. This would lead to an unnecessarily larger solution space and therefore higher computation time.

$$\sum_{i \in I} \mu_i^t \leq Cap_t, \ \ \forall t \in T$$

3. SKU Constraint: Each Demand Batch can only be of one SKU. To show this, the binary variable β_{ij} is introduced, which equals 1 if a Demand Batch i is of SKU type $j \in J$.

$$\sum_{j \in J} \beta_{ij} = 1, \ \ \forall i \in I$$

4. Destination Constraint: Each Demand Batch can only be linked to one Destination. For this, another binary variable ϵ_{il} is introduced, which equals 1 if a Demand Batch i has been ordered for location $l \in L$.

$$\sum_{l \in L} \epsilon_{il} = 1, \ \ \forall i \in I$$

5. Gamma Constraint: This constraint makes sure that the binary variable γ_j^t is set to 1 if any Demand Batch which consists of product of SKU j, is planned in week t. To do so, M, an arbitrarily large number, is multiplied with γ_j^t on the right-hand side of the equation. This makes sure that γ_j^t must be set to 1 if any α_i^t of an i that consists of SKU j (thus, $\beta_{ij} = 1$) is also set to 1. Because of the size of M, any number of α_i^t for which $\beta_{ij} = 1$ can be active once the corresponding $\gamma_j^t = 1$.

$$\sum\nolimits_{i \in I} \alpha_i^t * \beta_{i,j} \leq M * \gamma_j^t, \ \ \forall j \in J, \forall t \in T$$

6. Production Minimums Constraint: Using the γ_j^t variable from constraint 5, which is set to 1 if any units of a SKU j are planned in week t, the sum of all units of Demand Batches $i \in I$ that are planned in that week t and consists of that SKU j (thus, $\beta_{ij} = 1$) has to adhere to the Production Minimums PM_j^t for that SKU j, for that week t. If $\gamma_j^t = 0$, the left-hand side of the constraint may be equal to 0 as well. This MOQ is time dependent as it should be met in every week.

$$\sum\nolimits_{i \in I} \alpha_i^t * \mu_i^t * \beta_{i,j} \geq PM_j^t * \gamma_j^t, \ \ \forall j \in J, \forall t \in T$$

7. Delta Constraint: Exactly the same concept as constraint 5. In this case however, binary variable δ_{jl}^{t} is introduced, which keeps track of the activation of any Demand Batch i which is of SKU j ($\beta_{ij} = 1$) ***and*** goes to location l ($\epsilon_{il} = 1$). The construction with the arbitrarily large number, M, is the same as in constraint 5 as well.

$$\sum\nolimits_{i \in I} \alpha_i^t * \beta_{ij} * \epsilon_{il} \leq M * \delta_{jl}^t, \ \ \forall j \in J, \forall l \in L, \forall t \in T$$

8. Destination Minimums Constraint: Again, very much the same as constraint 6, but using the δ_{jl}^{t} variable to signify any planned units of any Batch Line i which consists of SKU j **and** go to Location l during week t. Also, the integer variable DM_{jl}^{t} is introduced which contains the Destination Minimums for the particular combination of SKU, Location and week in question.

$$\sum_{i \in I} \alpha_i^t * \mu_i^t * \beta_{ij} * \epsilon_{il} \geq DM_{jl}^t * \delta_{jl}^t, \ \ \forall j \in J, \forall l \in L, \forall t \in T$$

9. Integrality Constraint: For each batch of demand, the number of planned units μ_i^t in each timeslot $t \in T$ and the number of unplanned units λ_i can only take on integer values.

$$\mu_i^t, \lambda_i \in \mathbb{N}_0, \ \ \forall i \in I, \forall t \in T$$

10. Binarity Constraint: All defined binary variables can only take on the values 0 or 1.

$$\alpha_i^t, \beta_{i,j}, \gamma_j^t, \delta_{j,l}^t, \epsilon_{i,l} \in \{0,1\}, \ \ \forall i \in I, \forall j \in J, \forall l \in L, \forall t \in T$$

4. Results

The penultimate chapter of this research contains the comparison of the proposed solution versus the Benchmarked CTM logic. To do so, the KPIs which were used to construct the proposed solution, and some others that were deemed interesting by the S&IP Leadership Team, as discussed in Chapter 3.2.1, are included. An assessment of the calculation time of the proposed solution is also made.

4.1. Key Performance Indicators

The performance of the Benchmark and the proposed solution was measured using several Key Performance Indicators (KPIs) that were derived from the input from Nike. These KPIs are:

1. Unplanned Demand: units which are not planned for production;
2. Build Late Demand: units which are planned later than its IBW;
3. Build Early Demand: units which are planned earlier than its IBW;
4. Inventory Levels: the sum of the number of units that sit in inventory over each week $t \in T$;
5. Capacity Utilization Deviation: The Standard Deviation of the Utilized Resource Capacity over each week $t \in T$;
6. Customer Service Level Deviation: The Aggregate Standard Deviation of the Demand Coverage for all SKUs $j \in J$ for each Destination $l \in L$.

As stated in Sub Question 2 however, not only the raw values of these KPIs should be assessed, but also the interactions of the different KPIs. To that effect, the interaction tables in Appendix B are created. These tables show the implications of interactions of all the KPIs in the case of both being high (B.1), one being high and one being low (B.2) and both KPIs being low (B.3).

4.2. CTM vs Proposed Solution

In order to judge whether the proposed solution performs any better than the CTM engine which is in place, the aforementioned Benchmark and iterative MIP model were run on the same data. Both logics gave a feasible planning. The production which is planned per week for both methods is shown in Figures 4.1 and 4.2 for the Benchmark and the Proposed Solution respectively.

At first glance, the Planning for both logics looks very similar. It is possible to make out some difference when looking at some specific points though. For example, the capacity used in the very first week is clearly different. The CTM logic does not utilize this free capacity at all, whereas the proposed solution almost utilizes it fully. By doing so, the Proposed Solution opens up capacity in the busier weeks for other demand.

Another clear difference is the use of the capacity which is sitting later in the planning. The CTM logic leaves most of that capacity which isn’t used for demand that has its IBW there unutilized. The Proposed solution fills it up more of this capacity with some Demand that is pushed

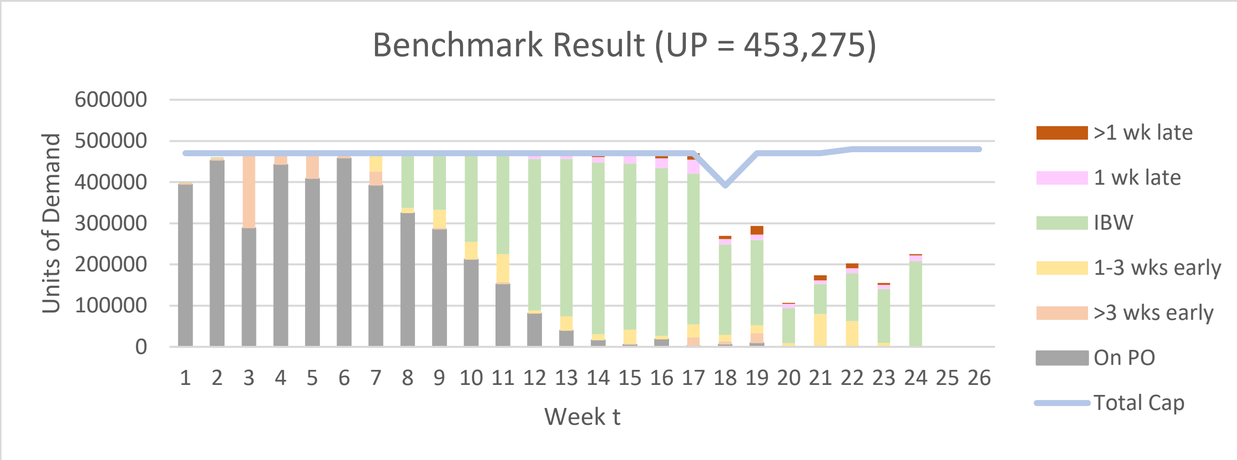

Figure 4.1: Benchmark Result, shown in planned units per week for all $t \in T$

out though. Especially weeks 17 through 19 show that Demand which is only 1 week late is planned there.

Another distinct difference is noticeable not in the graphs themselves, but in the number in the title of the graphs. The Unplanned Demand in the CTM logic is over 450 thousand units, whereas that of the proposed solution is just under 14 thousand. This is a reduction of 96.9% of all Unplanned units. When looking deeper into the Unplanned reasons for the Benchmark results, it shows the following split.

- Unplanned Capacity: 349,175 units
- Unplanned Destination Minimums: 9,956 units
- Unplanned Production Minimums: 94,144 units

Looking at the materials that are marked as unplanned in the Proposed Solution Output gives no further insight in the split according to Unplanned reasons. Though upon closer investigation,

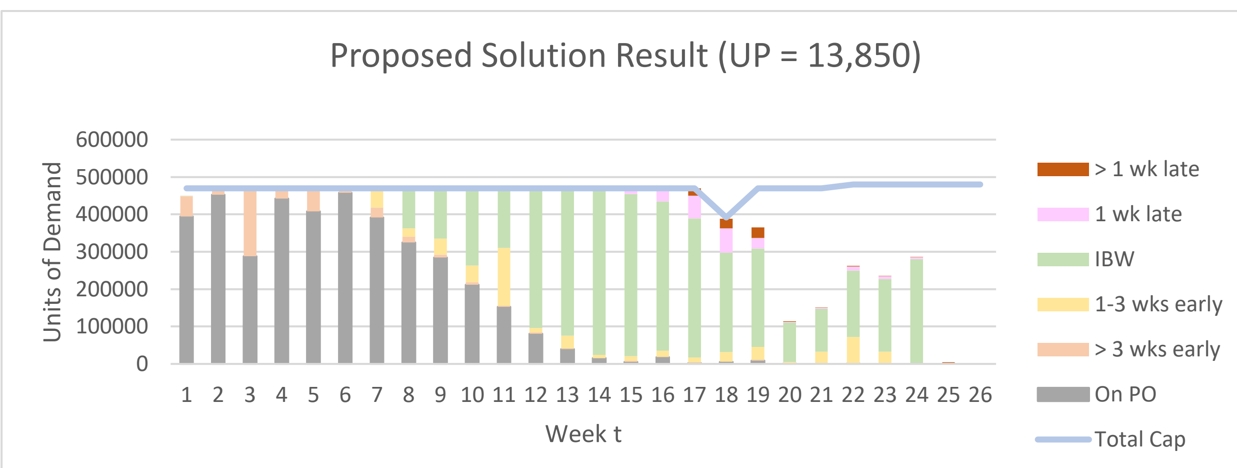

Figure 4.2: Proposed Solution Result, shown in planned in units per week for all $t \in T$

it can be found that 9,294 units of the Unplanned demand in the Proposed Solution have their Ideal Buy Weeks between $t = 20$ and $t = 24$ – weeks with plenty of capacity left. It is safe to assume that these unplanned units are caused because the MOQs can't be met. Considering the quantity is even less than the Unplanned Destination Minimums of the CTM logic, it would be safe to say that these have the same cause. The other 4,556 units cannot be explained easily.

Looking at the exact comparison Table 4.1 confirms the observations made from the previous Figures and shows some extra interesting pieces of information. The main focus of the comparison is the percentual difference. As mentioned before, the Unplanned Demand has seen a drastic decrease. The other buckets – except for the 'very' late one – see increases. This makes sense as they need to accommodate the extra demand that is being planned.

It would be interesting to see what the percentages in each bucket would have been if the extra demand that is being fulfilled by the proposed solution would have been smeared out over the buckets in the ratio that the CTM had originally planned the demand. Calculating this shows that the percentages for each of the buckets would in that situation would have been 6.97%, 10.05%, 77.51%, 3.74% and 2.09% respectively. The first two of these percentages match the Proposed Solution's results very closely. The IBW percentage is slightly lower, the 1 week late bucket slightly higher but the good news is that the 'very' late bucket performs very, very well in this regard.

The increase in PIFOT makes a lot of sense due to the extreme decrease in Unplanned materials. The Inventory Levels are an expected trade-off. This increase is caused by the extra materials that are being planned. The fact that the increase is less than the relative increase in planned units (+8.2%), is a small win. Finally, the Capacity Utilization Deviation. This was calculated over the first 24 weeks. The decrease makes sense logically, due to the extra goods which are produced pushing the overall utilization of the resource capacity up closer to the fully utilized weeks.

Table 4.1: Comparison of Key Performance Indicators of the CTM Logic and Proposed Solution, colors are the same as used in Figures 4.1 and 4.2.

	CTM LOGIC		**PROPOSED SOLUTION**			
TOTAL DP	**5,805,547**	100%	**5,805,547**	100%	**Difference (% of CTM)**	
>3 WKS EARLY	371,785	6.40%	405,217	6.98%	+33,432	+9.0%
1-3 WKS EARLY	535,623	9.23%	610,899	10.52%	+75,276	+14.1%
IBW	4,133,747	71.20%	4,456,728	76.77%	+322,981	+7.8%
1 WK LATE	199,868	3.44%	235,031	4.05%	+35,163	+17.6%
>1 WK LATE	111,249	1.92%	83,822	1.44%	-24,427	-24.7%
UP	**453,275**	**7.81%**	**13,850**	**0.24%**	**-439,425**	**-96,9%**
PIFOT	90.3%		98.3%		+8.9%	
UNITS*WKS OF INVENTORY	3,760,561		4,003,962		+6.4%	
CAPACITY UTIL DEV	6.9%		5.5%		-20.3%	

5. Discussion

5.1. Conclusion

The performance of the Proposed Solution is especially better than the CTM in preventing Unplanned and Late deliveries. The inventory on hand does grow, but less than the increase in planned units. To further prevent this, lateness should be penalized less, as a late delivery can technically move straight to the customer. It should be noted that the Benchmark is of course not the same as the CTM logic that is currently in place. It is, after all, an approximation which uses the same input as the Proposed Solution, such that they can be compared better. In fact, maybe the comparison isn't really fair, considering the age of the CTM logic and its simple, yet effective approach which can be understood by all.

In reality, there might be more complications which the Proposed Solution might run into. This is in part because of the computation time of all the constraints and setting up the model. If more constraints are added to this model, the computation time will of course also increase. To run this model for the 28 periods, in 6 Timechunks (1-8, 5-12, 9-16, 13-20, 17-24, 21-28), cost my Lenovo t480s 11 hours, 45 minutes and 9 seconds for this size dataset. That is a long time, but it gives a half year planning for a single factory. Since the factories are not interdependent, as the SKUs are sourced at a single factory, the optimizations could be run in parallel. What's more, the results for the short term are already output long before the final batch is run thanks to the Timechunks. So, even if the setting up of the model and running the optimization would take a long time, the planners could already start working on the season which is in their scope before the entire optimization finishes.

It is also important to take into consideration that even though the running time mentioned before is considerable (11h45m), it does the setting up for all demand twice (except for weeks 1-4 and 25-28). Running the same dataset without the Timechunks logic, cost the same machine 17 hours, 32 minutes and 14 seconds. Almost all of this computation time comes from the setting up of the constraints (15h34m).

Another weakness of the Proposed Solution is not knowing *why* exactly materials go Unplanned now. Though with the great reduction in Unplanned demand, the Supply & Inventory Team would have more time on their hands to find this out themselves. In fact, it is quite probable that the majority of the Unplanned issues from the Proposed Solution will be caused by Minimum Order Quantity issues.

5.2. Recommendations

Of course, considering the performance of the method, I would recommend implementing it in the business. But I do see the implications thereof. Of course, the CTM method works in conjunction with SAP: the Materials Planning system used throughout the entire company. Having something that works with that is of course very convenient.

However, the performance improvements that are implied throughout this research are quite severe. For years, the two main headaches of the business are (1) getting the hugely growing demand in the right place in the right time, struggling with tight resource capacities and (2) the constant rise of inventories. In order to solve the first problem, Nike has taken all kinds of precautions in order to pull production in earlier, such as through the use of Target Days Shipping (TDS, explained in Chapter 3.2.1) and Blind Buys (explained in Chapter 3.1.4) which in turn have caused increments in the Inventory Levels.

5.3. Further Research Suggestions

There are quite some alterations which can be made to the proposed solution which can be explored. These include, but are definitely not limited to:

- Using different datasets;
- Using more capacity constraints;
- The values of the penalties;
- The values of the priority parameters;
- The use of the Timechunks, in the sense of:
  - Using them at all, or if used;
  - The number of weeks to include in each.
- The addition of a heuristic which allows penalty score deterioration for secondary objective improvement;
- Run over run comparison of results to evaluate Buy Plan stability;
- Include DC inventory constraints (possibly instead of earliness penalties).

And all of these suggestions are made when using this research's Proposed Solution. There are many more methods in (more advanced) mathematical programming, heuristics and combinations of the two which can probably be explored in order to solve the problem at hand.

References

Brand Finance. (2019) *Apparel 50 The annual report on the world's most valuable and strongest apparel brands*. Retrieved from https://brandfinance.com/images/upload/apparel_50_locked.pdf

Charnes, A., Cooper, W. W., & Mellon, B. (1955). A model for optimizing production by reference to cost surrogates. *Econometrica: Journal of the Econometric Society*, 307-323.

Darmann, A., Pferschy, U., & Schauer, J. (2010). Resource allocation with time intervals. *Theoretical Computer Science*, *411*(49), 4217-4234.

Díaz-Madroñero, M., Mula, J., & Peidro, D. (2014). A review of discrete-time optimization models for tactical production planning. *International Journal of Production Research*, *52*(17), 5171-5205.

Dixon, P. S., & Silver, E. A. (1981). A heuristic solution procedure for the multi-item, single-level, limited capacity, lot-sizing problem. *Journal of opérations management*, *2*(1), 23-39.

Eppen, G. D., Gould, F. J., & Pashigian, B. P. (1969). Extensions of the planning horizon theorem in the dynamic lot size model. *Management Science*, *15*(5), 268-277.

Florian, M., & Klein, M. (1971). Deterministic production planning with concave costs and capacity constraints. *Management Science*, *18*(1), 12-20.

Guimaraes, L., Klabjan, D., & Almada-Lobo, B. (2015). Rolling horizon formulations for short-to-medium term production planning.

Hausman, W. H., & Peterson, R. (1972). Multiproduct production scheduling for style goods with limited capacity, forecast revisions and terminal delivery. *Management Science*, *18*(7), 370-383.

Hung, Y. F., & Hu, Y. C. (1998). Solving mixed integer programming production planning problems with setups by shadow price information. *Computers & operations research*, *25*(12), 1027-1042.

Jagannathan, R., & Rao, M. R. (1973). A class of deterministic production planning problems. *Management science*, *19*(11), 1295-1300.

Kallrath, J., & Maindl, T. I. (2006). *Real optimization with SAP® APO*. Springer Science & Business Media.

Kepczynski, R., Ghita, A., Jandhyala, R., Sankaran, G., & Boyle, A. (2019). Improve Your Responsiveness with Operational Planning. In *Implementing Integrated Business Planning* (pp. 375-441). Springer, Cham.

Köber, J., & Heinecke, G. (2012). Hybrid production strategy between make-to-order and make-to-stock–a case study at a manufacturer of agricultural machinery with volatile and seasonal demand. *Procedia Cirp*, *3*, 453-458.

Knight, P. (2016). *Shoe dog: A memoir by the creator of Nike*. Simon and Schuster.

Lagoudakis, M. G. (1996). The 0-1 knapsack problem--An introductory survey.

Lambrecht, M., & Vander Eecken, J. (1978). A capacity constrained single-facility dynamic lot-size model. *European Journal of Operational Research*, *2*(2), 132-136.

Lee, A. S., & Aronofsky, J. S. (1958). A linear programming model for scheduling crude oil production. *Journal of Petroleum Technology*, *10*(07), 51-54.

Lee, E. S., & Shaikh, M. A. (1969). Optimal production planning by a gradient technique I. First variations. *Management Science*, *16*(1), 109-117.

Lovgren, R. H., & Racer, M. J. (2000). Algorithms for mixed-model sequencing with due date restrictions. *European Journal of Operational Research*, *120*(2), 408-422.

Mixed-Integer Programming (MIP) - A Primer on the Basics. (n.d.) www.gurobi.com/resource/mip-basics/.

NIKE, Inc. (2019) *Form 10-K*. Retrieved from https://s1.q4cdn.com/806093406/files/doc_financials/2018/ar/docs/nike-2018-form-10K.pdf

Omar, M. K., & Teo, S. C. (2007). Hierarchical production planning and scheduling in a multi-product, batch process environment. *International Journal of Production Research*, *45*(5), 1029-1047.

Orlicky, J. (1975). MRP Materials Requirements Planning. In *The New Way of Life in Production and Inventory Management*. McGraw Hill.

Rudberg, M., & Wikner, J. (2004). Mass customization in terms of the customer order decoupling point. *Production planning & control*, *15*(4), 445-458.

Ruth, R. J. (1981). A mixed integer programming model for regional planning of a hospital inpatient service. *Management Science*, *27*(5), 521-533.

Segal, T. (2019, May 13). Understanding Make To Stock (MTS), Drawbacks, and Alternatives. Retrieved March 14, 2020, from https://www.investopedia.com/terms/m/make-to-stock.asp

Shubik, M. (1983). Auctions, bidding, and markets: an historical sketch. *Auctions, bidding, and contracting*, 165-191.

Spurrell, D. J. (1967). A simple method of production planning in multi-item situations. *Journal of the Operational Research Society*, *18*(2), 149-159.

Sridharan, V., Berry, W. L., & Udayabhanu, V. (1987). Freezing the master production schedule under rolling planning horizons. *Management Science*, *33*(9), 1137-1149.

Sung, C., & Maravelias, C. T. (2008). A mixed-integer programming formulation for the general capacitated lot-sizing problem. *Computers & Chemical Engineering*, *32*(1-2), 244-259.

Wagner, D. E., & Shetty, C. M. (1962). Solution to a production scheduling problem with fixed costs. *Journal of the Operational Research Society*, *13*(1), 87-94.

Wagner, H. M., & Whitin, T. M. (1958). Dynamic version of the economic lot size model. *Management science*, *5*(1), 89-96.

Walker, W. T. (1980). An investigation of the relationship between aggregate planning and the master production schedule.

Xie, J., Zhao, X., & Lee, T. S. (2003). Freezing the master production schedule under single resource constraint and demand uncertainty. *International Journal of Production Economics*, *83*(1), 65-84.

Appendix A: Data Sample

Table A.1 Demand Data, after pre-processing, used for the Benchmark

PC_i and DT_i are in their non-numerical form, but that is not important for the sorting logic of the CTM, they do start with a numerical value to be able to sort them to.

SKU_i	Destination_i	D_i	IBW	PC_i	DT_i	ME_i	ML_i	LT_i	PAT_i	DBD_i	PM_I	PDM
838938-061	1025	54364	29/07/2019	4 - Standard	2 - Standarc	84	7	0	25/01/2016	31/12/2199	3000	240
315123-111	1025	46847	05/08/2019	4 - Standard	2 - Standarc	63	7	0	26/02/2018	31/12/2199	3000	240
315123-111	1025	45512	29/07/2019	4 - Standard	2 - Standarc	84	7	0	26/02/2018	31/12/2199	3000	240
BQ4420-100	1027	842	15/07/2019	2 - Launch	2 - Standarc	70	21	0	27/05/2019	31/12/2199	3000	240
CD0188-003	1025	38966	02/09/2019	4 - Standard	2 - Standarc	63	7	0	24/06/2019	31/12/2199	3000	240
CQ4810-621	1025	38657	19/08/2019	4 - Standard	2 - Standarc	63	7	0	27/05/2019	31/12/2199	3000	240
BV7427-003	1014	33396	22/07/2019	4 - Standard	1 - DRS	63	7	0	24/06/2019	31/12/2199	3000	240
AA7071-011	1014	30674	22/07/2019	2- Launch	2 - Standarc	77	7	0	26/02/2018	31/12/2199	240	240
315123-001	1014	28402	05/08/2019	4 - Standard	1 - DRS	91	56	0	26/02/2018	31/12/2199	3000	240
315123-111	1014	26740	05/08/2019	4 - Standard	1 - DRS	91	56	0	26/02/2018	31/12/2199	3000	240
315123-111	1025	26065	02/09/2019	4 - Standard	2 - Standarc	91	7	0	26/02/2018	31/12/2199	3000	240
AO4438-008	1025	450	29/07/2019	1 - Promo	2 - Standarc	63	7	0	27/05/2019	31/12/2199	3000	240
CQ4810-111	1025	25498	12/08/2019	4 - Standard	2 - Standarc	63	7	0	27/05/2019	31/12/2199	3000	240
BV7427-002	1014	24648	26/08/2019	4 - Standard	1 - DRS	105	7	0	24/06/2019	31/12/2199	3000	240
BQ4420-100	1027	24196	22/07/2019	4 - Standard	2 - Standarc	77	21	0	27/05/2019	31/12/2199	3000	240
CN9512-002	1014	21540	22/07/2019	4 - Standard	1 - DRS	77	7	0	25/03/2019	31/12/2199	3000	240
CD5079-001	1014	20900	30/09/2019	4 - Standard	2 - Standarc	105	7	0	30/09/2019	31/12/2199	3000	240

Appendix B. KPI Interaction Matrices

Table B.1: High interactions of KPIs; the diagonal contains the single KPI's implication if high

High	1	2	3	4	5	6
1	**Missing many sales opportunities**	Very low on time delivery: will lose business	ML parameter should be used to prevent UP	Too much risk is taken with Blind Buys	Unplanned is mostly due to other reasons than Capacity	Big differences in Demand per GEO
2	-	**Might lose business to competition**	Hardly any UP; low JIT delivery	Focus is on the wrong products	Struggling to catch up with demand	Other GEOs are doing better
3	-	-	**Unnecessarily early production**	Impending storage catastrophe	Not producing lean enough	Might be cannibalizing other GEOs
4	-	-	-	**Storage shortage issues will arise**	Not producing lean enough	Divert shipment requests impending
5	-	-	-	-	**Factory Planning will be hard**	Cross GEO Demand is not aligned
6	-	-	-	-	-	**Manual work is required**

Table B.2: Mixed interactions of high & low KPIs; KPI on the left is low, on top is high

Low\High	*1*	*2*	*3*	*4*	*5*	*6*
(1)	-	Struggling to keep up	Growing inventory	Expect excess inventory	Low total demand	Cross GEO demand is not aligned
(2)	Bad coverage, but could be worse	-	Producing too early overall	Expect excess inventory	Producing close to IBW probably	Cross GEO demand is not aligned
(3)	Bad coverage	Producing too late overall	-	Expect inventory reduction	Producing close to IBW probably	Cross GEO demand is not aligned
(4)	Expect shortages	Expect shortages	Expect inventory increase	-	Lean production	Request divert shipments
(5)	Factories running at max capacity	Struggling to keep up	Getting highly popular demand	Factory struggling, enough buffer	-	High total demand
(6)	Bad coverage for all GEOs	Factories are struggling	Too much early production	Stable inventories	Low total demand	-

Appendix B (Cont.)

Table B.3: Low interactions of KPIs; the diagonal contains the single KPI's implication if low

Low	(1)	(2)	(3)	(4)	(5)	(6)
(1)	**High coverage of demand**	Demand is met on time (high PIFOT!)	Demand might be arriving late	Lots of demand in the pipeline	Factories are managing demand	All GEOs are seeing great coverage
(2)	-	**Timely delivery**	Either lean production, or high unplanned	Either lean production or high unplanned	Factories are either managing demand or are running at full capacity	All GEOs have timely deliveries
(3)	-	-	**Lean production**	Either lean production or not-so-great coverage	Factories are either managing lean production or are struggling to keep up	All GEOs have lean production
(4)	-	-	-	**Very lean; may cause out of stocks when demand surges**	Factories are either producing very lean or are running at full capacity	All GEOS have lean inventories
(5)	-	-	-	-	**Stable factory utilization; might indicate working at full capacity**	Cross GEO demand is aligned
(6)	-	-	-	-	-	**Demand distributed evenly over Geos**