Improvement of Advanced Planning System in Semiconductor Supply Chain System

Behrouz Alizadeh Mousavi

Supervised by Prof Cathal Heavey

A thesis presented for the degree of Doctor of Philosophy

School of Engineering

University of Limerick

February 2021

Dedication

To my lovely spouse, Afrooz. We step side by side in ups and downs. Without her supports, patients, and listening I could not achieve my scientific goals.

To my selfless mother, Nayereh. She taught me how to be strong and how to give without expectation.

To my wise father, Naser. He opens my eyes to enthusiastic looking at life itself.

To my perspicacious brother, Omid. His philosophy of science and dedicated time has always empowered me.

Declaration

I hereby declare that this thesis is entirely my own work, and has not been submitted for any other award at this or any other academic establishment. Where use has been made of the work of other people it has been fully acknowledged and referenced.

Behrouz Alizadeh Mousavi

February 2021

Acknowledgements

Firstly, I would like to express my sincere gratitude to my advisor Prof. Cathal Heavey for the continuous support of my Ph.D. study and related research, for his patience, motivation, and immense knowledge. His guidance, valuable feed-backs, steady supports, and invaluable inputs did not only challenge and motivate me, but also shaped, sharpened, and developed my thinking. I could not have imagined having a better advisor and mentor for my Ph.D. study.

I further want to thank Dr. Radhia Azzouz, a postdoctoral researcher at the University of Limerick for her contributions and countless discussion. Her efforts and suggestions make inevitable impacts on the results of this thesis.

Moreover, I want to thank Mr. Zhikang Tian for his contributions in web-application development. His knowledge supported me regarding the use of proper technologies within ReCAST.

Also, I would like to acknowledge the support of colleagues and staff at the University of Limerick, organizers, and members of project Productive 4.0, and members of the case study company.

Finally, this project has received funding from the Electronic Component Systems for European Leadership Joint Undertaking under grant agreement No 737459. This Joint Undertaking receives support from the European Union's Horizon 2020 research and innovation program and Germany, Austria, France, Czech Republic, Netherlands, Belgium, Spain, Greece, Sweden, Italy, Ireland, Poland, Hungary, Portugal, Denmark, Finland, Luxembourg, Norway, Turkey. Besides, this thesis has emanated from research conducted with the financial support of Science Foundation Ireland (SFI) under Grant Number SFI 16/RC/3918, co-funded by the European Regional Development Fund.

Contents

List of Figures

List of Tables

Abbreviations

AATP	Allocated Available to Promise
AB	Agent Based
ADI	Advanced Demand Information
AF	Allocation Fence
ALM	Allocation Manager
AM-UI	Allocation Maintenance - User Interface
APS	Advanced Planning System
ATO	Assemble to order
ATP	Available To Promise
ATT	Actual Transit Time
ATV	Automotive
BE	Back End
BPMN	Business Process Modeling Notation
BR-Cycle	Batch Rescheduling Cycle
CLM	Customer Logistic Manager
CMAD	Confirmed Material Availability Date
CO	Confirmed On Time
CTO	Configure to Order
CW	Current Week
DB	Die Bank
DC	Distribution Center
DCA	Distribution Center in Asia
DE	Discrete Event
DELS	Discrete Event Logistics System
DF	Demand Fulfilment
DM	Divisional Model
DP	Demand Planning
DREP	Die-Bank Representative

DSL	Domain Specific Language
DSS	Digital Security Solutions
EM	Enterprise Modeling
ERP	Enterprise Resource Planning
EW	Early Warning
FAB	fabrication
FCFS	First Come First Serve
GIT	Goods In Transit
HPS	Hierarchical Planning System
IC	Integrated Circuits
IoT	Internet of Things
IPC	Industrial Power Control
KPI	Key Performance Indicator
LCO	Later Confirmed Order
MA	Manufacturing Product Type
MBSE	Model Based System Engineering
MIP	Mixed Integer Programming
MRP	Material Requirement Planning
MSP	Master Schedule Planning
MTF	Make to Forecast
MTO	Make to Order
MTS	Make to Stock
nEWs	Negative Early Warnings
OM	Order Management
OOB	Open Order Book
OPM	Object-Process Methodology
OR	Operations Research
OP	Order Promising
OWL	Web Ontology Language

pEWs	Positive Early Warnings
PGI	Planned Good Issue Date
PPOS	Plan Position
PSS	Power & Sensor Systems
ReCAST	Regional Customer Allocation Support Tool
RFP	Required for Plan
RMAD	Requested Material Availability Date
SC	Supply Chain
SCM	Supply Chain Management
SCN	Supply Chain Network
SCOR	Supply Chain Operations Reference
SCP	Supply Chain Planning
SE	System Engineering
SIN	Singapore
SKU	Stock Keeping Unit
SP	Sales Product
SysML	System Modeling Language
TTT	Target Transit Time
UCO	Unconfirmed Order
V&V	Verification and Validation
WIP	Work In Progress
ZLP	Zonal Logistic Planner
ZD-nEWs	Zero Day Negative Early Warnings

Abstract

Thesis title: Improvement of Advanced Planning System in Semiconductor Supply Chain System

Author: Behrouz Alizadeh Mousavi

A semiconductor manufacturing Supply Chain (SC) starts with intricate processes of chip manufacturing that extends from fabs to back-ends in a global network. These globalized value chains require months for raw silicon wafers to traverse to become a single integrated circuit. To manage these complex SCs, an Advanced Planning System (APS) is required. APS has an open architecture that allows bespoke integration of modules, algorithms, and performance measures to develop stable SC plans. The focus of this study is on APS within the semiconductor case study. Our objectives are to improve Master Planning, and Demand Fulfilment systems of APS through digitalization and reducing human interventions in planning. To address these objectives, the work in this thesis is separated into three parts.

The first part investigates this case-specific planning system and develops models through the use of Model-Based System Engineering (MBSE) methods. In this study, we compare three different standardized MBSE methods, namely, Web Ontology Language (OWL), Business Process Model and Notation (BPMN), and Systems Modeling Language (SysML). These MBSE methods support communication between different stakeholders involved in the management of these complex SCs. Within the thesis, we provide proof of the necessity of MBSE based methods for the management of these complex SCs.

The second part of this thesis develops a simulation framework to simulate the interaction between the planning system (APS) and material flow within the SC. This simulation framework was developed in Anylogic to reveal the root cause of internal nervousness, resulting in unstable Available-To-Promise (ATP), within the case study APS. This simulation work demonstrates the role of anticipation of lower-level behaviour in upper-level decision-making. The simulation modeling framework, as demonstrated here, could provide a risk-free environment to test, evaluate, and tune planning systems, such as APS in the management of complex SCs.

The third part proposes and develops a decision support tool for allocation planning within demand fulfilment. In a tight supply situation within the case study APS, planners maintain manually allocation plans. In this part, we proposed a mathematical model for allocating ATP to customers where demand exceeds supply. The bi-objective model maximizes customer service levels while keeping a maximum amount of stock, with this stock available for unforeseen planning situations. The proposed mathematical model was developed as a web-application decision support tool called the Regional Customer Allocation Support Tool (ReCAST). The prototype of ReCAST was applied to the case study to be used as a decision support tool by planners.

Chapter 1

Introduction

1.1 Motivation

In the new world of digital devices and technologies, semiconductors are among one of the fundamental elements of the digital world. Semiconductors are everywhere and their usage is growing with their introduction into new technologies like the Internet of Things (IoT), autonomous vehicles, artificial intelligence, and digital devices. Developments in the semiconductor industry have made Integrated Circuits (ICs) smaller, faster, and more reliable. Besides, increasing global demands drive semiconductor industries to introduce a wider range of products.

Semiconductor manufacturing comprises hundreds of steps that vary between products. The main steps in the manufacturing of semiconductor devices are front-end (wafer fabrication), and back-end (probing, assembly, and test). These products traverse to final products through the production of silicon wafers, involving the fabrication of integrated circuits on wafers, assembly of circuits, and testing finished products. Each of these steps is carried out in the global manufacturing network (Munirathinam and Ramadoss 2016). Recently, through digitization, this industry has become more competitive and customers have become more demanding. Long production lead times, dynamic demands, growing diversification of product types, shortening product life cycles, complex routing in global manufacturing supply chains, together with low confidence to commit to actual customer requests are examples of supply chain complexity in the semiconductor industry (Uzsoy, Fowler, and Mönch 2018; Batra et al. 2018).

Supply chains are a potential competitive advantage of companies that improve operational efficiency either by improving resource utilization or decreasing costs and inventory. In this competitive and growing environment, efficient supply chains are a key advantage. A supply chain's efficiency depends on every step in the value process, such as material procurement right through to the delivery of final products. Within these various factors, the information system and planning system are essential components that integrate decision-making processes (Prasad and Sounderpandian 2003). Within this rise of interaction and complexity in supply chains, planning systems are important elements.

Semiconductor industries are one of the most global and competitive types of supply chains. The characteristics of this industry make its supply chain planning complex and unique. Long production time, high capital cost, a wide range of products, short lifetime of products, global production processes, production yield, and demand uncertainties are the challenging characteristics of a semiconductor industry. To cope with these challenges, a mature and effective planning system needs to be deployed within these global value chains.

This planning system should be capable of developing decisions such as targets of global production, schedules for each production line, provision of predictions to customers on receipt of orders and fulfilment of demand, order shipments, consideration of limited resources, and to satisfy the overall financial strategy of the company. Besides, it should consider the interconnection and precedence of the decisions. The planning system in this complex environment needs to be hierarchically structured. For instance, production steps, planning horizon, and product types are examples of dimensions required in a planning system. To do these, semiconductor manufacturing uses an advanced planning system (APS) deployed on an Enterprise Resource Planning (ERP) system encapsulated within an information system (Wiers and Kok 2017; Stadtler, Kilger, and Meyr 2015).

Improving this complex planning system in semiconductor manufacturing is the main motivation of this thesis. In this thesis, we aim to investigate APS in the semiconductor industry with a focus on master planning and demand fulfilment. Within these processes of APS, we aim to investigate the root causes of instability in demand fulfilment and master planning. Besides, we are looking to improve the allocation planning of products to customers in demand fulfilment. This improvement is held through digitization of allocation planning by means of a decision support tool development based on mathematical optimization. In the next section, further details on the focused processes of a semiconductor supply chain and its APS are provided.

1.2 Semiconductor Supply Chain Planning

A supply chain system can be described using the Supply Chain Operation Reference (SCOR) model to structure its functions and processes. The SCOR model is a standard for managing, configuring, and analyzing supply chains (Stadtler, Kilger, and Meyr 2015). It provides a common terminology and structure within the whole global value chain. It comprises of six types of processes, namely, plan, source, make, deliver, return, and enable (Council 2008).

Plan configures all other processes. In the SCOR model, it has three-level. The aim of planning is to support decision making by evaluating alternatives for future activities and selecting the most appropriate or best options. As semiconductor manufacturing is complex, it is not possible to represent all elements inside a planning system. Thus, an abstract model of real material flow is used as a model of reality within a planning system. This model is the basis for the development of plans. The model in semiconductor manufacturing takes its main characteristics from an advanced planning system. The model has three main dimensions. The first is the planning level based on the planning horizon. It consists of three levels called long-term, mid-term, and short-term. The second dimension of the model is process types within the supply chain. It consists of procurement, production, distribution, and sales. The third dimension of the model structure is the hierarchy of planning tasks. Within the hierarchy, decisions move between the three-level to plan the supply chain. Based on these functionalities and di-

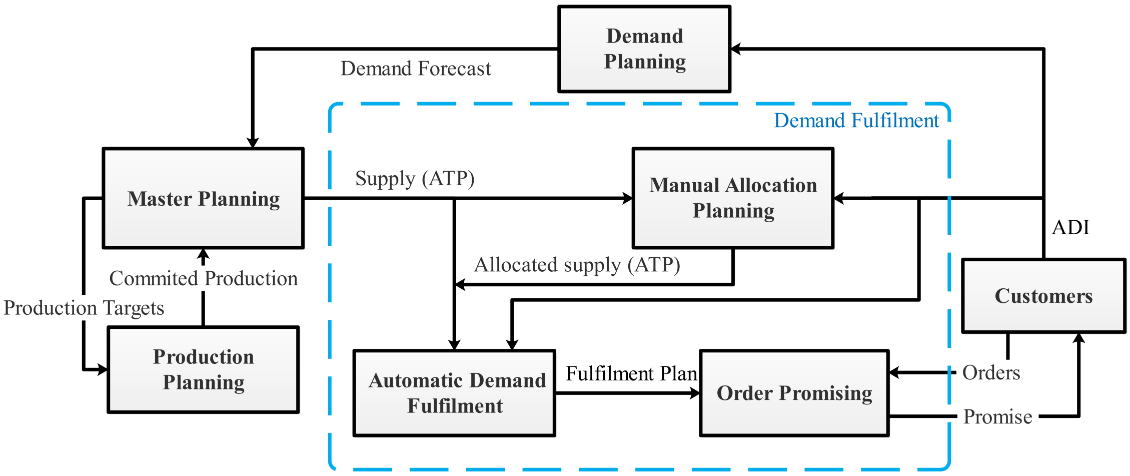


Figure 1.1: Overview of studied supply chain planning system in semiconductor manufacturing.

mensions where separate software modules are defined within supply chain planning. These modules of a semiconductor supply chain system are strategic network design, capacity planning, material requirement planning, master planning, production planning, inventory management, demand planning, and demand fulfilment (Mönch, Uzsoy, and Fowler 2018a; Mönch, Uzsoy, and Fowler 2018c; Uzsoy, Fowler, and Mönch 2018).

To fulfill demands in a volatile and dynamic environment, the studied semiconductor supply chain uses modules developed by various software vendors. This thesis will focus on master planning and demand fulfilment software and the planners involved in developing and maintaining valid plans. Demand fulfilment in this case study company comprises of manual allocation planning, automatic demand fulfilment and order promising. Other relevant processes are production planning and demand planning, which provide necessary inputs for demand fulfilment and master planning processes. Figure 1.1 provides an overview of the studied supply chain system.

In Figure 1.1 customer order details are input using two types of demand in the planning system. The first input is firm orders which we call orders and the second input is advanced demand information (ADI) that are demand forecasts of customers. ADI is not a firm order but an indication to notify the order promising process of possible future demands. On the basis of customer forecasts and orders as well as internal forecasts coming from marketing, sales, and operations functions, demand planning generates short- and mid-term forecasts for the total market demand. The goal of demand planning is to forecast market demand as accurately as possible. The results are provided to master planning which aims to match demand with supply. Master planning matches total demand with total resources to generate a global master production schedule. Resources are machine capacities, stocks, work in progress, and raw materials. The first output of master planning is production targets through interaction with the production planning module. Another output of master planning is current and possible future supplies also called available to promise (ATP). ATP comprises of current inventory as well as planned supply receipts for production. Demand fulfilment (dashed box in Figure 1.1 on the preceding page) has two main inputs which are ATP and order/ADI. Demand fulfilment has three main processes shown in this figure. ‘Order Promising’ is a tool for controlling orders, communicating with customers, checking Key Performance Indicators (KPIs). The other two aim to peg the ATP to orders or ADI handled by ‘Manual Allocation Planning’ and ‘Automatic Demand Fulfilment’.

Although APS solves complex planning decisions in a structured way, it suffers from weaknesses such as lack of transparency and complexity (Lin, Hwang, and Wang 2007). APS as a custom-specific planning system is deployed based on the requirements of applied industry (Wiers and Kok 2017). Thus, APS always requires maintenance and improvement which is challenging due to the complexity of procedures of the systems involved. Besides, these internal incompatibilities that need to be improved, involve various external factors that bound operational flexibility. Demand disorders, unknown customer behaviour, and production pullbacks are examples of stochastic external factors (Lee, Padmanabhan, and Whang 2004).

Planning is the process of taking into account future developments by identifying alternatives and providing directions for implementation. Due to the above-mentioned incompatibilities, which maybe internal or external, plans may need to be rectified automatically or manually. In order to evaluate plans, several KPIs were developed within supply chain management to check the quality and deviations of plans. These KPIs monitor plans, to indicate when plans need to be updated. An important goal of semiconductor supply chain planning systems is the design of effective KPIs, used to support industry to deliver the right products with the right quantity at the right time to customers (Batra et al. 2018). Having a properly defined KPIs is another challenge within the industry. If a KPI can not represent what happens within the supply chain, this inefficiency causes wrong decisions or lost of resources.

In this thesis, we investigate the supply chain planning system of a semiconductor case study. The software tools supporting demand fulfilment and master planning are typically of ERP and Advanced Planning System (APS). This thesis focuses on demand fulfilment, master planning, and related processes. First, we investigate this complex planning system using a system thinking approach and specifically Model-Based System Engineering (MBSE) methods. Using MBSE methods, we demonstrate the root cause of an inefficient KPI derived from demand fulfilment using simulation modeling. In addition, we discuss the neglected role of human planners in the manual allocation of planning in APS. We show how this manual intervention could be improved using mathematical optimization and develop a prototype decision support tool.

The output of this research will support the semiconductor's supply chain planning task to meet customer demands in a more efficient and on-time manner. Besides, it helps to improve the performance of human planners within demand fulfilment. In addition, the developed quantitative models within this research will support the industry to further digitize activities within APS by reducing human intervention in checking KPIs and improving the automation of customer allocation through the development of a decision support tool. In the next section, further details on the identified problems within APS are provided.

1.3 Problems Statement

Demand fulfilment and master planning have been studied as a research topic within the literature. However, the presented approaches ignore use within industrial practice, provide simplified assumptions, or tailor to the details of one set of processes without considering integrated modules. In reality industry can struggle with various challenges that have not been addressed so far. Some of these challenges observed (research questions that are addressed specifically in this thesis are detailed later in this chapter) during this thesis are:

- APS has a lack of transparency. There are evidences of nervousness and instability in plans through the generation of KPIs which require the control system or planners to be involved. However, the root causes of these nervousness that generate these KPIs are difficult to derive in the complex case study semiconductor supply chain studied here.

- Causalities are nonlinear and vague. The consequences of volatile inputs like demand or supply are not transparent. For instance, although the causes of ATP instability in a rolling horizon can be listed, revealing the exact reason is a challenging question.

- APS has several automatic algorithms that use mathematical optimization or rule-based algorithms, but at the same time, human planners are mandatory to modify plans in order to maintain them. The role of these planners and effects of their decisions on the efficiencies and instabilities of plans has rarely been studied.

- Planning in APS aims to be controlled centrally but at the same time, it needs decentralized modules. The balance between centralized and decentralized planning modules is another current challenge within industry.

- Modeling the complexity of a planning system using simulation and/or optimization requires a deep understanding of the system and the development of a profound conceptual model, which is another challenge within industry.

- Planning modules are developed by integrating different software vendors within a planning system. The process flow of steps and interconnections between software and databases may not be optimal.

- Planning algorithms within software vendors have different dimensions. Designing and improving the aggregation and disaggregation of information within hierarchical planning modules is a leading open challenge. For instance, master planning has short-term, mid-term and long-term planning horizons while demand fulfilment does not follow the same structure.

- Big data is available within APS, but sometimes it is hard to find relevant data specific to the problem being addressed, or the database may not have the relevant data.

The problems we aim to address in this research are related first to a repetitive instability in an existing KPI called Early Warning and secondly, to intractable human interventions in the development and maintenance of customer allocation plans. Early Warning is a KPI which aims to monitor the deviation in fulfilment plans in a rolling horizon. This occurs when the promised order could not be satisfied. In this case, this KPI notifies the planners of this deviation. This deviation is a sign of the incapability of the supply chain to satisfy planned deliveries. In this situation, human planners might need to be involved in the planning process to replan and to communicate with customers of the occurred deviation. Within this study, we aim to address this issue by finding the root causes of this nervousness using the research methodology described in the next section.

The second issue relates to digitization improvements in the APS case study where human planners have to manually allocate plans generated by the APS. Although customer allocation planning is performed automatically by the demand fulfilment module in APS, APS still requires human intervention to update plans in specific circumstances like supply shortage (see Figure 1.1 on page 5). Within Manual Allocation Planning, planners calculate allocation plans which are intractable, complicated, and non-optimal. We aim to address this issue by developing a web-based decision support tool developed using operations research methods. Based on these problem definitions, the following three types of research questions were addressed in this thesis.

Firstly, we carry out research into systems modeling approaches that can be used in the understanding, modeling, and knowledge sharing of advanced complex systems, in our case a complex supply chain system. In these systems, revealing an understanding requires a common language between stakeholders that facilitates clearly the benefits of applying analytical and scientific approaches. This research question resulted in us evaluating methods for modeling a complex supply chain system, with a specific focus on planning. Three MBSE methods were evaluated based on this task, which we feel demonstrated their benefits in knowledge sharing, finding the root causes of disruption in these systems, and facilitating the application of analytical methods. Chapters 3 and 4 address this research question.

The second research question focuses on the inefficiency of the Early Warning KPIs in the demand fulfilment module. This KPI is called Early Warning and its aim is to monitor the deviation of repromising within order management. To identify the root causes of this instability, we developed a hypothesis. Based on this hypothesis, we developed a simulation model empowered by data to imitate the APS that identifies the root cause of this instability. We further conducted data utilizing data from the real system to fully prove this hypothesis. The results of this research question are presented in Chapter 5.

The third research question relates to the implementation of improvement plans within demand fulfilment. Maintenance of allocation plans requires manual intervention by human planners. Human intervention in addressing allocation plans does not lead to optimal decisions and is prone to human bias. To address this research question, we aim to support planners in the task of product allocation to customers. We propose an advanced analytical method to solve this issue in APS. The aim is to improve manual allocation planning with a decision support system based on mathematical optimization. The tool was developed into a web-based application and was tested by planners within the case study company. Chapters 6 and 7 describe the results of this research question. Specifically, the research questions of this thesis are:

- Research Question 1: Evaluate how model-based system engineering languages could support disruption management within a complex supply chain system?

- Research Question 2: What are the root causes for Early Warning (KPI) nervousness which results from ATP instability in the semiconductor's supply chain planning?

- Research Question 3: How to move from low-level operational manual alterations by planners to more strategic decisions through the use of analytical decision tools?

1.4 Methodology

Research methodologies for use in supply chain research are: (1) surveys, (2) case study research, (3) action research, and (4) modeling (Kotzab et al. 2006) Identifying the correct methodologies to use in carrying out research into supply chain systems requires a balanced approach (Kotzab et al. 2006). Figure 1.2 on the following page details the research methodologies used in the research carried out in this thesis. As shown in Figure 1.2, first we performed our investigations by first carrying out a literature review into: first case study research into planning in complex supply chain systems and system thinking approaches for complex systems.

Using this preliminary knowledge, we then carried out interviews with supply chain experts familiar with the case study company. All these tasks in acquiring knowledge supported us in understanding the system and helped us to formulate initial research questions. Later, to better define the problems, we used MBSE to develop conceptual models used for communication with experts. In the next step, we developed quantitative models by using simulation, optimization, and data analysis. The developed quantitative tools were tested and deployed on the case study to complete the circle of action research within this thesis. Further details of each phase (blue dashed in Figure 1.2) are described as follow.

Our initial aim was to gain an initial understanding of supply chain planning within the industry partner. First, we investigated the available tutorials available within the industry partner. Second, we reviewed previous research carried out on this case study, which allowed us to perform interviews with supply chain experts using online/in-person meetings (see Appendix A.1 on page 233) and sending questionnaires to understand the challenges and suggestions in operating this complex system. This work consisted of open discussions which allowed us to gather feedback on our queries.

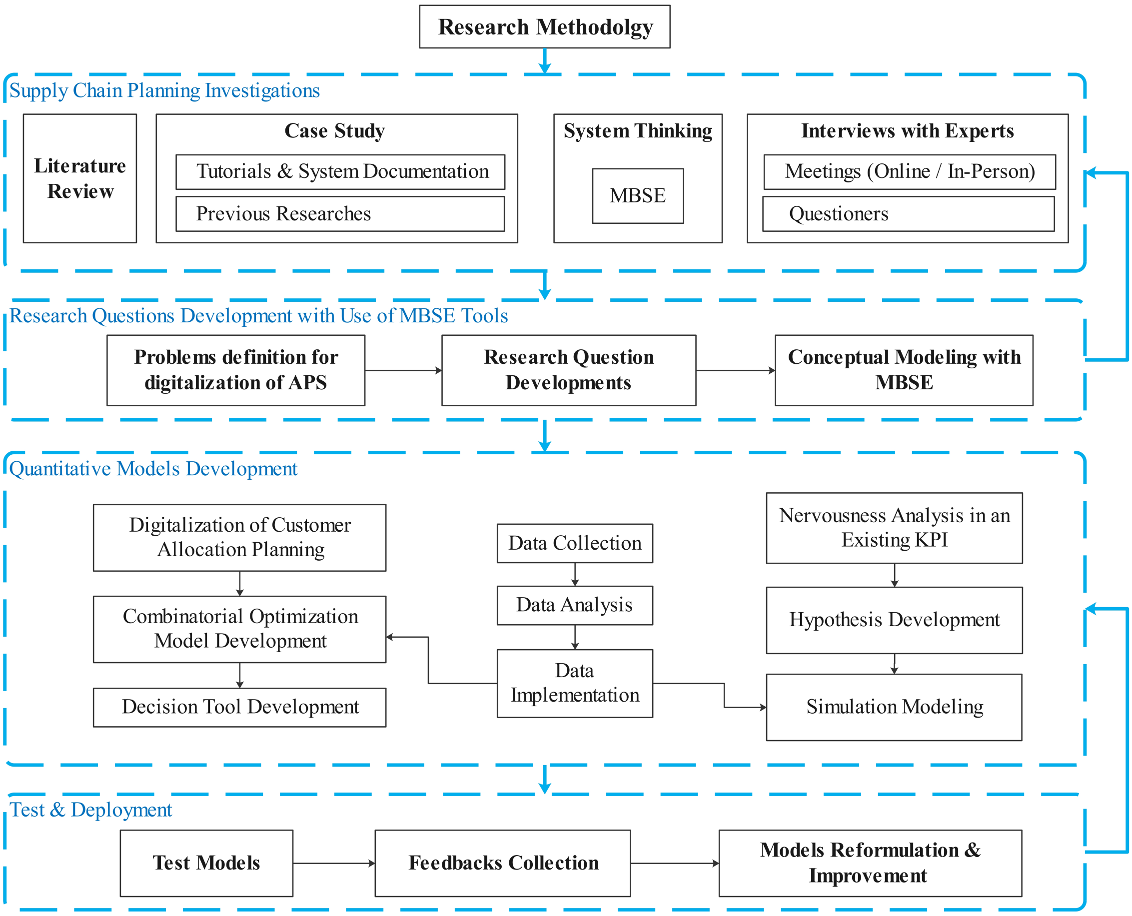


Figure 1.2: Applied research methodology.

In addition, to understand more in-depth the supply chain within this case study, we also used system thinking methods. Holistic system thinking perspectives allow us to maintain an understanding of all activities within the supply chain and it also clarified how the subsystems and processes interact and constitute the whole system (Johannessen 2005). System thinking methods supported us to develop an architectural view of supply chain systems and allowed us to develop conceptual models to support quantitative analysis. These models were used for communication between stakeholders and provided base lines for our defined research questions. The models included information flows, technical planning steps, and production flows within the supply chain case study.

The combination of these four steps in the first phase (Supply Chain Planning Investigations in Figure 1.2) were constantly updated during this research project. The result of the first phase was used in the Research Questions Development with Use of

MBSE Tools (see Figure 1.2). Within this Problem Definition phase, we developed two research questions that we addressed using quantitative modeling tools in the third phase (Quantitative Models Development in Figure 1.2 on the preceding page). In the first quantitative method, we identified instabilities identified by a KPI within demand fulfilment where a root causes of this KPI were unknown. To investigate the root causes, we simulated planning within this complex supply chain to prove our hypothesis. Finally, we carried out data analysis to further prove this hypothesis.

In the second quantitative method, we identified human intervention in allocation planning within order management. These manual modifications required a considerable amount of time and energy, while the output and its impact were not clear. We developed a quantitative decision support tool that digitalized this activity. The obtained method within the tool was based on an operations research method which was developed into a web-based application.

In the last phase, the developed quantitative models which were tested and deployed by industry. We deployed the prototype decision support tool on real case data. In addition, the provided results from both quantitative models are conveyed to industry managers and decision makers. The last phase complements the action research methodology as we used the delivered products and results in the case study.

1.5 Thesis Outline

This thesis investigates how demand fulfilment and master planning can be improved with the use of model-based system engineering (MBSE), and quantitative methods (simulation, data science, and optimization).

In Chapter 3 we introduce the planning system regarding master planning, demand fulfilment, and their relative modules. Then in Chapter 4, we discuss the use of MBSE for improving system understanding in complex supply chain planning. We compare three different languages, namely, BPMN, SysML, and OWL that were used to evaluate the root causes of nervousness in ATP.

In Chapter 5 to improve the transparency in demand fulfilment, we search for the root causes of nervousness in the planning system. We simulate APS based on a high level conceptual model which was presented in Chapter 4. Simulation modeling is a common problem-solving approach that improves system understanding by experimenting in a risk-free environment. Based on the simulation model, we investigate hypotheses for the root causes of nervousness in supply chain plans. The result from the simulation model was supplemented by a data analysis model that finally proves the hypothesis and reasons for instability in KPI. The outputs of this chapter proved the hypothesis and provide managerial insight regarding the maintenance of master planning and demand fulfilment algorithms.

Improving demand fulfilment would not be met without considering the effects of human planners. In Chapter 6 and 7 we aim to improve the decision-making of human planners in allocation planning by providing a decision support tool. This decision support tool is based on getting insight from planners, calculating allocation with the use of a multi-objective mixed-integer programming model, and providing visualization and scenario-based solutions. ReCAST is a web-application that aims to make available an advanced mathematical approach for use in practice.

Finally, Chapter 8 discusses the results, contributions, and limitations of the research presented in this thesis. Managerial insights are derived from the results and possible directions for future research are outlined.

Chapter 2

Literature Review

2.1 Introduction

Chapter one gives an introduction to the thesis, in which it is clear that there is a need for improvement and maintenance of advanced planning systems. The motivation of this chapter is to introduce the state of the related literature for complex planning systems for supply chains and other related topics. More specifically, we aim to review the following research streams:

- Supply chain planning systems and planning modules: this section identifies the complexity and challenges of SC planning systems. The relevant modules are briefly described to familiarize readers and we review nervousness within planning systems.

- Model Based System Engineering (MBSE): within this section, we aim to review the advancement in systematic modeling using computers. MBSE as a requirement for investigating planning systems has a myriad of benefits and challenges which we aim to review in this section.

- Simulation modeling in supply chain: this section aims to review the use of simulation modeling for the investigation and improvement of decisions in supply chains.

- Allocation planning in Advanced Planning System (APS): the last section relates to allocation planning in supply chain systems. This relates to the one research topic in this thesis. Here we review the current literature to present the developed decision support tools.

As the problems introduced in Chapter 1, in this thesis, we aim to reveal the root causes of nervousness within demand fulfilment and and improve the allocation planning in demand fulfilment by proposing a decision support tool. Thus, demand fulfilment within APS is the area of supply chain planning system which we aim to shed light on it in this chapter.

Although the focus of this thesis is on demand fulfilment and order management, we could not neglect the other planning modules and the complexity of their interrelationship. To systematically understand the SC planning system, we used MBSE language which develops a single source of truth from systems. This modeling approach is a novel method regarding studying supply chain systems. Thus, we provide the relevant literature here to clarify the objectives, trends, challenges, and benefits of using MBSE in previous researches.

2.2 Supply Chain Planning System and APS

Supply Chain (SC) as a complex system with stochastic and dynamic parts, has several elements where its total performance depends on individuals' performance, behaviours, and interactions within the designed network (Pundoor and Herrmann 2006). For understanding this complex, large-scale and interactive system, several standard modeling frameworks have been developed. The most famous model for Supply Chain Managemnt (SCM) is called Suppl Chain Operation Reference (SCOR) model. This standard was developed by the Supply Chain (SC) Council to model, describe, communicate, control, and evaluate SC configurations (Huan, Sheoran, and Wang 2004). The SC framework in semiconductor manufacturing in general and in our case study, the SCOR model and its elements of Plan, Make, Source (Supply), and Deliver which is used in Mönch, Uzsoy, and Fowler (2018b).

Within any modeling approach for a supply chain system, the planning system is the core which should be designed and implemented according to the case study (Wiers and Kok 2017). Due to the hierarchical planning of complex supply chain systems, organizations implement custom built APS systems, which is a key aspect of successful implementation (Zoryk-Schalla, Fransoo, and de Kok 2004). There is a lack of a clear definition of APS (whether it is considered as Advanced Planning System or Advanced Planning and Scheduling) (Vidoni and Vecchietti 2015), research such as Vidoni and Vecchietti (2015), Wiers and Kok (2017), Fleischmann and Meyr (2003), and Stadtler, Kilger, and Meyr (2015) try to propose common definitions and models for APS implementation. However, a common feature of APS is configuring interactions between levels of hierarchy. APS has a top model that sends instructions to base levels and creates anticipations regarding the base model using a model of the base-level model which resides inside the top model. The process model of these interactions structures the aggregation/disaggregation of data and models (Zoryk-Schalla, Fransoo, and de Kok 2004). Wiers and Kok (2017) states that APS is a custom-made implementation in a complex adaptive environment that requires updates and maintenance by supply chain experts. APS plans production using resources and capacities to meet demands in a planning horizon. A detailed discussion of the case-specific APS used in this study is described in the Chapter 3.

The literature for evaluating current deployed APS is scarce and mostly qualitative. Although researchers note the possible drawbacks of APS (Lin, Hwang, and Wang 2007; Zoryk-Schalla, Fransoo, and de Kok 2004; Halldórsson et al. 2007), the lack of quantitative analysis for implementing APS is a gap that needs to be considered by researchers for better design, definition, and application of APS. While there is a wide adoption of APS for many complex SCs (Lin, Hwang, and Wang 2007), there are weaknesses in their application. Using a APS in a semiconductor industrial setting, Lin, Hwang, and Wang (2007) researched an effective approach for APS implementation. Not only do they could not prove the correctness of APS systems, but also they reveal pitfalls. The most related pitfalls that our research tries to shed light on are (1) over-reliance on automated algorithms and technological solutions, (2) lack of transparency in APS-dominant system that causes difficulty in controlling APS, (3) Dependency of technological satisfactory performance on parameter fitting, data cleanup, system behaviour, and plan validation. In our case, we are looking to improve APS automated technology by tuning parameter settings through simulation experiments of ATP modeling.

Research exists regarding the improvement in the design and implementation of APS (Zoryk-Schalla, Fransoo, and de Kok 2004; Wiers and Kok 2017), research on proposing a new model for APS (Ponsignon and Mönch 2014), or evaluating APS qualitatively (Lin, Hwang, and Wang 2007; Halldórsson et al. 2007). However, there is little research on the systematic and quantitative evaluation of APS (Lin, Hwang, and Wang 2007). Thus, although APS and Available to Promise (ATP) were designed to stabilize production plans, which is successive in many cases (Moscoso, Fransoo, and Fischer 2010), in the case study researched here nervousness in ATP has been recorded that affects the order promising procedures (Lin, Hwang, and Wang 2007). Therefore, one of the novelties of this thesis is the quantitative evaluation of an implemented of APS using a multi-paradigm simulation to improve APS performance.

2.2.1 Instability in Planning Systems

The concept of planning instability or nervousness was first introduced by Steele (1975) to dampen deviations in MRP. With increasing complexity of planning systems and the move from Enterprise Resource Planning (ERP) to incorporate APS within a Hierarchical Planning System (HPS), the importance of the design and operation of these planning tools is important for the overall stability of these systems (Wiers and Kok 2017; Ponsignon and Mönch 2014).

The literature of a plan's stability using these tools could be investigated from many different aspects such is the vast amount of literature on this topic (Ponsignon and Mönch 2014; Panahifar et al. 2018; Sabri 2019; Demirel et al. 2019). The main streams of research on this topic are:

1. Nervousness in planning systems can be caused in production planning (De Sampaio, Wollmann, and Vieira 2017); or master planning (Herrera and Thomas 2009; Herrera, Belmokhtar-Berraf, et al. 2016); or in Supply Chain Planning (SCP) (Pujawan and Smart 2012; Demirel et al. 2019).

2. The sources of instabilities can be classed as intrinsic (internal operations and incompatibilities) (Pujawan and Smart 2012) or extrinsic (supply and demand uncertainties) (Hasachoo and Masuchun 2016).

3. Damping strategies (Ho 2005) that could be either reactive like inventory control and buffer stocks maintenance that will attempt to find a stationary state of the system while the nervousness still exist; or proactive which improves the planning system (Ponsignon and Mönch 2014) or research that finds the root cause of nervousness that is then used to dampen the complexity of planning as studied by (Sivadasan et al. 2013).

4. The structure of the planning system where the instability is evaluated. It could be for example in MRP system (Ho 2005); in a HPS (Moscoso, Fransoo, and Fischer 2010); a Rolling Horizon Planning (RHP) system (Meistering and Stadtler 2019); or in APS (Geier 2014), a topic which will be briefly discussed.

Regarding the first item, nervousness involves mostly two elements of a planning system which are production planning (Gelders and Van Wassenhove 1982) and master planning (Herrera and Thomas 2009; Herrera, Belmokhtar-Berraf, et al. 2016). Other research was carried out on Supply Chain Planning (SCP) (Demirel et al. 2019), and on nervousness in the whole planning system caused by system integration (Ho 2005). Demirel et al. (2019) modeled and evaluated the instability of supply networks by using generalized modeling, a methodology emanating from ecology, where they focused on the interaction between material flow networks with external instabilities in planning. To the best of our knowledge, APS's instabilities have been rarely researched (Moscoso, Fransoo, and Fischer 2010; Ponsignon and Mönch 2014). The analysis of nervousness in Master Schedule Planning (MSP) has been studied using different approaches. Herrera and Thomas (2009) minimized MSP nervousness by means of optimization and simulation. Other researchers studied nervousness in MSP by considering stochastic programming (Koca, Yaman, and Aktürk 2018); and Mixed Integer Programming (MIP) formulation (Herrera, Belmokhtar-Berraf, et al. 2016) to re-plan the master scheduling when instability emerges while considering the cost of rescheduling.

The sources of instability or nervousness could be internal or external as stated above. In previous works, researchers mostly focused on external ones. A managerial study by Pujawan and Smart (2012) showed that internal factors have lower impact on production planning nervousness. The external factors are generally related to the bullwhip effect (Li and Disney 2017), demand fluctuation by customers (Tiacci and Saetta 2012), and supplier side instabilities (Gruchmann and Gollmann 2017). In this paper, internal nervousness is researched arising from: (1) the information flow and timing of planning revision (Sivadasan et al. 2013), and (2) the structure of the SC network and software modules interactions and the timing of data entry (Demirel et al. 2019).

Damping strategies regarding planning instability are categorized into two types: reactive and proactive strategies (Pujawan and Smart 2012). The reactive strategies have been studied more than proactive ones and are mainly used to deal with external factors that are difficult to predict or forecast. Methods like stochastic optimization to address production planning instabilities are an example of reactive strategies (Koca, Yaman, and Aktürk 2018). In Herrera and Thomas (2009), an MIP model is used to reduce the nervousness of MPS. For example, a reactive decision making method was developed in Herrera, Belmokhtar-Berraf, et al. (2016) to deal with production planning instabilities. This method re-actively tries to find the optimal trade-off between nervousness and cost of rescheduling. On the other side, there are a number of researchers who investigated the use of proactive strategies for planning nervousness. Sivadasan et al. (2013) lessened the nervousness by omitting the complexity-adding information flow within the supply chain. The trade-off between planning revision time periods, quality of information, suppliers cooperation and forecasting accuracy are some instances of proactive strategies.

The type of the planning system can also cause instability as mentioned in the last item. Different planning systems were developed and progressed to deal with the complexity of production systems, to increase the efficiency of production and make the plans more stable. Another perspective towards instability analysis of planning systems is related to the architecture of planning systems, where Steele (1975) studied the nervousness from an MRP system. Further work on this topic was Carlson, Jucker, and Kropp (1979) who considered nervousness as the inconsistency of schedule set-ups and applied analysis using a dynamic lot-sizing algorithm to reduce nervousness within MRP systems. More recently, Li and Disney (2017) developed a measurement method to calculate nervousness to evaluate forecasts and orders-call-offs methods. They re-evaluated nervousness caused by forecasting errors in MRP systems. They mainly considered the problem jointly with the bullwhip effect.

HPS was developed to address the challenge of short and medium requirements of production planning (Fleischmann, Meyr, and Wagner 2015). The integration between the planning system levels and its consequences on a production plan's instabilities were studied in (Gelders and Van Wassenhove 1982). Moscoso, Fransoo, and Fischer (2010) addressed the problems triggered by HPS that lead to instability and re-planning. In this empirical study, they evaluated the causes related to lead time and bullwhip effects. Meistering and Stadtler (2019) also considered HPS and designed a stabilized-cycle strategy and compared it with period-based and order-based strategies. In the problem they studied, capacitated, multi-item and short-medium production planning.

In this section, we discussed the literature regarding APS and supply chain planning systems. The nervousness in a planning system is reviewed as a research area which this thesis aims to research. Besides, we identified that planning systems are complex and there is a lack of a systematic method for studying APS. In the next section, we discuss the literature of Model-Based System Engineering (MBSE) which we used as a systematic approach and pathway to investigating complex planning systems.

2.3 Model-Based System Engineering

MBSE can play a key role in the development of strategies for dealing with the growing complexity of Systems Engineering (SE), specifically in the areas of production and supply chain (Madni and Sievers 2018). In this literature review, we briefly discuss MBSE in different fields, but, because the applications are so varied, the main focus is on its use for modeling and decision making of complex manufacturing systems related to supply chain planning systems in the support of disruption management.

MBSE is one of the most important frameworks proposed in the past twenty years (Haskins 2011), where its comprehensive history and origin were discussed in (Haskins 2011). MBSE not only emanates from the modeling advancements created by the increase in computing capabilities and Information Technology (IT) improvements and automation, but also due to the lack of applicability of document-centering SE in large scale and complex production and design systems. The emergence of MBSE was mainly in aerospace and defence project systems, but with the adoption of Industry 4.0, and multiple stakeholders involved in industrial design and operation, it has gained prominence in the design and operation of production and associated processes (i.e., supply chains) (Estefan 2008).

According to Madni and Sievers (2018) MBSE is a framework in which a sole source of truth of the system or part of the system can be obtained. MBSE takes an integrative approach by analyzing how various perspectives and elements interact with an environment by defining which characteristics to consider or ignore based on the specific purpose in mind. In other words, a model can represent only a part of reality regarding the stakeholder requirements, but may contain several viewpoints from stakeholders regarding their questions and requirements (Madni and Sievers 2018). This is because SE is an interdisciplinary domain within engineering and management, that concentrates on the design, system architecture and maintenance of the whole system through the system's life cycle (Forsberg and Mooz 1991; Haskins 2011). A classification of SE knowledge areas was presented and reviewed by Yang, Cormican, and Yu (2019), where they note the increased complexity and multidisciplinary nature of SE that requires the involvement of many stakeholders.

While traditional SE only creates and manages system documents, MBSE is about developing, managing, and controlling them. In comparison to traditional approaches, MBSE has the advantage of maintainability throughout the life cycle. The Vee model is the most famous approach of the use of MBSE throughout the life cycle of a system. The Vee model was presented by Forsberg and Mooz (1991) to evolve the design overtime based on the stakeholders' requirements regarding system concepts and elements of design.

MBSE literature consists of many application domains and methodologies, here we focus on MBSE in design and operation of production processes and it's use for disruption analysis. Zdravković et al. (2011) describes examples of MBSE advantages in disruption management. In this work, the authors discuss the domain framework for Internet of Things (IoT) within implementation and maintenance projects. They categorized MBSE into three stages, system design (with Unified Modeling Language/System Modeling Language (UML/SysML)), application design (Domain Specific Language (DSL)), and interoperability (Ontology). This work truly delineates the importance of MBSE for efficient IoT system implementation. Besides, they emphasise the use of these approaches for the maintenance of a system, in which, disruption management is an integral part for efficient management. Also, Vallejo, Romero, and Molina (2012) propose an enterprise integration framework and model according to physical, application and business integration. The referential model used was Business Process Modeling Notation (BPMN) to monitor control and improve an extended enterprise for increasing interoperability and stability of a system. A comprehensive survey by Vernadat (2020) indicates that Enterprise Modeling (EM) has become a very active area in understanding, documentation, change management, re-engineering, architecture, performance evaluation, and system integration of organizations and enterprises. Although the benefits of EM and MBSE have become clear, they are many open research avenues, such as, ontological development, standardization, and language evolvement.

As MBSE becomes more widely used, its methodology also advances and improves.

The introduction of new MBSE approaches in the literature is clear evidence of this growth. For example, Beery (2016) introduced MBSE that expanded the use of MBSE with SysML (System Modeling Language) by providing greater detail on the method of using functional and physical architecture for external system performance. Mordecai and Dori (2013) proposed a new MBSE framework that increases interoperability, interconnectivity, interfacing, integration, and interaction in the context of system of systems in the domain of service systems. Object-Process Methodology (OPM) was used for this developed framework.

Great advances have been made in MBSE research in recent years (Madni and Sievers 2018) some of which has demonstrated the economic advantage of using MBSE environments in SE (Madni and Purohit 2019). In a comprehensive economic analysis of MBSE, Madni and Purohit (2019) analyzed the investment cost and possible gain of MBSE implementation and compared it with traditional SE approaches. The authors collected summary information from twelve industry sectors and defined a structured method for analysis. Of importance here are the results that MBSE implementation requires more upfront investment than traditional SE, but that in latter and longer stages of the system's life cycle (when the design is in operation) MBSE provides greater gains. Second, they showed which industry sectors, according to life span, environment complexity, and system complexity, gain more advantages from MBSE implementation. More recent surveys (Yang, Cormican, and Yu 2019; Cameron and Adsit 2018; Estefan et al. 2007; Huldt and Stenius 2019) have shown its increasing rate of application in other domains. INCOSE also defined the 2025 vision of MBSE towards the expansion of applications across industry domains (INCOSE 2014). Madni and Purohit (2019) main focus was on different types of systems that were continuous, discrete or combined (Law, Cheng, and Das 2015). Much research work has also been carried out on discrete event systems, the case study presented in this chapter.

These type of systems, also known as Discrete Event Logistics System (DELS), are defined as networks of resources through which goods and people flow (Mönch et al. 2011; Virtual Factory Lab 2020). These systems, discrete event systems, include factories, supply chains, warehouses, among others. McGinnis et al. (2011) presented an article on developing an ontology for simulation using SysML as a domain-specific language for a class of simulation applications. Other works from this group include Batarseh, Huang, and McGinnis (2015) who draws from methods within software engineering and computer science, specifically Model-Driven Architecture (MDA). They present a proof of concept study where they use a DSL (SysML) to develop a conceptual model of a manufacturing system and transform this to a simulation language Arena (Arena 2020). The proposed framework demonstrated how the problem domain, modeling, and knowledge analysis are integrated to create a DSL for the system of interest for domain experts by giving them engineering, rather than analysis tools, for specifying their problem. In the next subsection, we introduce MBSE methodologies used in the thesis.

2.3.1 Selected MBSE Methodologies

A primary goal of this thesis is to highlight the role MBSE has in disruption management. A secondary objective is to demonstrate three MBSE methodologies used on a complex supply chain case study outlining the advantages/disadvantages of each approach. As a digital representation of systems, MBSE requires a common language in a standardize format. MBSE methodologies, while seeking to be readable by both machine and human, define syntax and semantics to share the common knowledge of models between stakeholders. Comparing all MBSE approaches requires a detailed survey that is beyond the scope and contributions of this chapter. In this study, we aim to investigate the capabilities and advantages of MBSE approaches for analysing and understanding the sources of disruption in a complex supply chain system, which is a DELS.

SysML was selected as it provides a standard (SysML ISO Standard Document 2017) and comprehensive system specification paradigm. It has been used in analysing DELS (Feldmann, Kernschmidt, and Vogel-Heuser 2014; Batarseh, Huang, and McGinnis 2015). This provides a consistency in terms of model syntax and semantics, together with unambiguous graphical symbols, which can greatly improve communication. As well, since its adoption, SysML has enabled an increased adoption of MBSE practices across industry.

The number of industrial partners who have contributed to its development clearly indicates that practitioners recognize the need for SysML and show an eagerness to make standardized graphical modeling notation freely available.

In contrast to SysML, one of the advantages of BPMN, which is also a standard (BPMN ISO Standard Document 2013), is that it is an easy tool to understand, requiring a minimal amount of training for a basic level of understanding. As noted in Onggo et al. (2018), BPMN is easily understood by different stakeholders as it has been originally designed with business users in mind.

In addition to standard modeling approaches such as SysML and BPMN, OWL is World Wide Web Consortium (W3C) standard (W3C 2012) which aims to represent the exploitable web knowledge of computers. OWL's features include computer readability, knowledge representation and sharing, model discovery via search, or reusability.

SysML is an OMG (Object Management Group) standard that supports the specification, analysis, design, verification and validation of a broad range of systems and systems-of-systems. SysML helps create object-oriented models of systems that incorporate not only software, but also people, material, facilities, and other physical resources, while expressing both structure and behaviour for such systems (Nikolaidou et al. 2015). While SysML helps and facilitates requirements description, there are also work on requirements verification described using SysML (Feldmann, Kernschmidt, and Vogel-Heuser 2014). SysML is essentially a UML profile that represents a subset of UML 2 with extensions (Friedenthal Sanford, Moore Alan, and Steiner Rick 2014). Although UML has also been used to represent non-software systems, it is not ideally suited to this purpose and requires non-standardised use of model elements that can ultimately lead to confusion and incorrect interpretation of diagrams. To adapt UML for non-software systems, the developers of SysML attempted to remove the software bias and added semantics for model requirements and parametric constraints. SysML is more geared toward MBSE implementation in design and architecture of production systems to develop process integration and collaboration concepts within the system (Madni and Sievers 2018) while it is also examined and used for conceptual modeling and simulation modeling as an external analysis of DELS (Beery 2016; Batarseh, Huang, and McGinnis 2015).

BPMN provides a visual notation that is readily understandable, at the basic level, by all business users and Information Technology (IT) developers and it is mostly suitable for business entities and relations modeling. It allows the development of requirements for business processes with XML languages designed for the execution of business processes, such as WSBPEL (Web Services Business Process Execution Language). It has been argued that BPMN notation includes a small set of visual elements (Modelio 2018) that would allow a novice to understand a reasonably complex process. A formal description of the BPMN standard is given in Chinosi and Trombetta (2012), which one would be required to study to understand more complex processes.

BPSim (BPSIM 2018) is a Workflow Management Coalition (WfMC) standard to facilitate simulation of BPMN or XPD (KBSI 2018) models. More specifically, it allows the exchange of parameters of process analysis data facilitating structural and capacity analysis for pre-execution and post-execution optimization of a process. Critical analysis on BPMN and BPSim standard (Laue and Mueller 2016) states that these standards are powerful for the creation of business process models and help to promote process simulation. In addition, they facilitate the building of tools for modeling simulation independent from simulation tools. This notwithstanding, BPMN and BPSim may have limited technical capabilities in representing important features of complex flows in a business process (Onggo et al. 2018) while they represent enterprise systems with business perspectives that SysML is not truly capable of capturing. Other work by this group Proudlove, Bisogno, Onggo, Calabrese, and Levialdi Ghiron (2017) also demonstrate the application of BPMN for modeling and simulation of DELS in healthcare systems. Their aims toward fully facilitated modeling indicate the capabilities of BPMN for engaging stakeholders, facilitating communicating, and relieving coding gaps.

Ontology is a formal specification used to describe, define, and categorize the concepts and their relationships and can be applied to various domains in engineering. Studer, Benjamins, and Fensel (1998) defined ontology as "a formal, explicit specification of a shared conceptualisation of a domain of interest". These formal and unambiguous semantics can be processed by computers that can be used as communication tools between human agents and software. For the sake of this goal, machine-processable languages (XML, RDF, and OWL) are used to describe the entities and their logical relationships. Before we examine the selected OWL as an ontology language, the importance of ontology in modeling and analysis are reviewed.

Ontology has been examined through two main aspects, referential ontology defined as pre-image with strong epistemic nature which capture reality utilizing semantic relationships based on the representation of the real world; and on the other side; and methodological ontology with normative nature best used for further processing as formal semantics (Hofmann, Palii, and Mihelcic 2011). Zaletelj et al. (2018) propose an ontology of manufacturing systems which improve interoperability and composability of modeling and simulation. As discussed in Zaletelj et al. (2018) ontology can receive attention from different points of view such as modeling, information sharing and data exchange, knowledge sharing, life cycle management, planning overview, etc. A comprehensive review of the use of ontology in SE was done by Yang, Cormican, and Yu (2019). In this survey, the authors emphasized the lack of formal validated ontology and review it by structuring a categorized body of knowledge of SE. The evidence of this review reveals the growth of ontology-based SE as an MBSE methodology.

Although OWL has been approved as a standard web of data by W3C for web-accessible ontologies, it has been proposed as a standard in the domain of simulation (Silver et al. 2011). The authors develop a structure for DELS modeling through exploring classic and formalized DELS taxonomy (state-oriented model, event-oriented model, activity-oriented model, process-oriented model). This structure defines the hierarchy of properties of OWL and makes the modeling and simulation achievable through ontology. In the same vein, Kernan and Sheahan (2010) outline use of OWL for generic data modeling of an enterprise to be used in DELS to reduce the simulation modeling effort and the maintainability of models.

2.3.2 Supply chain disruption analysis with MBSE

In this subsection, we review previous research which used systematic approaches and MBSE for disruption analysis of a SC. Supply chains are interactive value chains of information and material networks that coordinate the delivery of products (Stadtler 2005). Analysing instability and disruptions within these complex flows has been researched (Ivanov, Dolgui, et al. 2017), but studying systematic methods for dealing with instability and disruptions in a SC has rarely been studied (Wu, Blackhurst, and O'Grady 2007; Bayar et al. 2016; Dekkers et al. 2012).

Ivanov, Dolgui, et al. (2017) reviewed the literature of quantitative approaches (such as, mixed integer programming, stochastic programming, and simulation modeling) for dealing with SC disruptions. This research identified model-based support as a prerequisite for quantitative analysis. The authors mentioned that: "The efficient application of model-based support for quantitative analysis implies a clear description of control processes in the case of different deviations and disturbances." Beyond applying quantitative approaches, there are some papers aimed at analysing SC disruptions with a systematic and methodological approach.

Beyond quantitative approaches, there are few systematic and modeling approaches which are aimed to facilitate the complexity of disruption analysis within SC. Wu, Blackhurst, and O'Grady (2007) presented a network-based modeling approach to understand how disruption will affect a supply chain system. This modeling aimed to evaluate the propagation of disruptions was based on the extension of Petri net. Although the proposed modeling can evaluate the disruption within the material flow of a SC, the interaction and inefficiency within the planning system could not be analysed. Dekkers et al. (2012) applied entropy assessment theory to the disruption of information within SC. By doing risk assessment as a function of an entropy score, the effects of information disruption on the supply chain structure are evaluated and coupled with discrete event simulation. This work truly shows the importance of information disruption but they are analyzed isolated from manufacturing. Bayar et al. (2016) also aimed to facilitate disruption analysis of a manufacturing system through monitoring disruptions and risks. They obtained knowledge-based ontology using OWL inspired by artificial immune systems. Obtaining the concept of biological immune system supported the authors to develop a framework and methodological guidelines for evaluating disruption.

In summary, comparing the selected MBSE methods and disruption analysis in a supply chain system identifies the lack of use of MBSE as a systematic approach for analysing SC disruption, specifically in a complex planning system. Thus, the motivation for this section are summarized as follows: (i) Use of MBSE as the sole source of truth to improve the understanding of the complex SC planning system by developing a common language between stakeholders; (ii) Present a MBSE pathway and framework for investigating complex problems within SC systems. The pathway demonstrates the systematic approach we used for dealing with disruptions and improving the SC system; (iii) Indication of the benefits and challenges regarding the use of MBSE in disruption management of SC.

2.4 Simulation evaluation of APS nervousness

To deal with nervousness, new integrated planning systems (e.g., Master Production Schedule (MPS), ERP, and APS) have emerged. APS is the most recent planning system which is designed to satisfy the needs of complex production planning and SCP systems like in semiconductor manufacturing (Hegde et al. 2004; Mönch, Uzsoy, and Fowler 2018a). Nervousness in ERP and APS has been rarely studied. APS consists of several integrated modules (Stadtler, Kilger, and Meyr 2015) developed based on heuristic algorithms to manage the planning processes. Planning sequences and events are patented by companies (Hegde et al. 2004; Chen, Srinivasan, and Syed 2005) but however have been studied by several researchers (Stadtler, Kilger, and Meyr 2015; Mönch, Uzsoy, and Fowler 2018c).

SC as a set of linked value chains (information, cash, materials, and decisions) that plays a central role in generating competitive advantages for production systems. In this complex integrated system, decision making is a crucial and challenging process. Modeling and simulation can assist decision makers to design, develop, analyze, and revise

SC systems (Oliveira, Lima, and Montevechi 2016; Kleijnen 2005). The three main topics where SC simulation has proven to be advantageous (more than 60 percent of publications) are: (1) understanding and diagnosing problems within SCs, (2) SC's performance improvement, and (3) experimenting new scenarios, models or projects (Oliveira, Lima, and Montevechi 2016).

The importance and complexity of SC issues in semiconductor manufacturing is increasingly important. Mönch, Uzsoy, and Fowler (2018a) discussed the importance of utilizing simulation modeling for addressing the challenges in understanding, improving, and evaluating the management of semiconductor SC systems. The use of simulation for evaluating SCP in semiconductor manufacturing has been used by several researchers (Mönch, Uzsoy, and Fowler 2018a) and the benefits of using Agent Based Simulation (ABS) is introduced and discussed in Van Der Zee and Van Der Vorst (2005) and Santa-Eulalia, D'Amours, and Frayret (2012).

While simulation is important in evaluation of SC networks and its management, the evolution of computation power and the ability of combining simulation modeling (i.e., multi-paradigm simulation of Discrete Event, System Dynamic, and Agent Based paradigms), have made it more feasible for researchers to analyse SCP systems using more complex simulation models (Mönch, Uzsoy, and Fowler 2018a).

2.4.1 Use of simulation for improvement within APS

Proposing new planning and scheduling methods, supported by mathematical optimization and simulation, is another research stream for the improvement of APS (Steger-Jensen et al. 2011). For instance, in the scope of APS in semiconductor manufacturing, Ponsignon and Mönch (2014) propose a heuristic optimization approach for master planning. In this work, the authors develop a simulation for master planning in the mid-term planning horizon to evaluate mathematical optimization performance regarding demand behaviour. In the proposed simulation model they evaluate the interaction of the planning system with the physical system during the planning horizon. Similar to our work (presented in Chapter 5), in the planning algorithm they consider the lead time of a product as an estimation of the cycle time that is the source of deviation between plan and executed plan. The focus of their work was on the front end of semiconductor manufacturing while the aim of our work is on the logistics of the back end, from distribution to end customer. Although, like the majority of the literature in this area (Ivanov, Dolgui, et al. 2017) they propose an optimization approach for the complex planning system, but the real system still works based on rule-based algorithms and the implementation of optimization in complex systems is still a matter of debate in the literature (Wiers and Kok 2017; Lin, Hwang, and Wang 2007). Their work uses a Discrete Event Simulation model combined with optimization, which could not mimic the whole planning system, specially decisions in the planning system and human interactions with this system. This is a gap which we aim to solve by proposing a multi-paradigm simulation modeling framework to model a SC planning system.

ATP as an output from APS, in the majority of studies, is modeled using mathematical programming to improve the efficiency and stability of the plans. Jung (2010) modeled ATP by considering customer priorities and penalty costs. Other work regarding the ATP calculation and updating time and structure was done by Plattner et al. (2013). They evaluated ATP creation based on the aggregation of data in an SC. Their work addressed the complexity of information sharing within the SC to improve the quality of ATP. They introduced the lack of an availability check as a drawback of several current ATP calculation systems, which leads to the loss of extremely important information for planning purposes.

As discussed in the previous subsection, simulation is an advanced quantitative approach for evaluating planning systems. Due to the complexity of SCs that require APS and the involvement of human planners (Wiers and Kok 2017), modeling and simulating of APS requires a proper understanding of the case-specific APS together with a set of research questions. This is required in order to formulate a conceptual model. In Santa-Eulalia, D'Amours, and Frayret (2012), the authors propose a top-down structure for a simulation model on APS. They propose agent-based simulation and they use SysML for semantic unification of stakeholders in order to deal with complexity of the system modeled. Although they properly delineate the role of agent-based simulation in performing the complex implemented in the simulation model of an APS, they did not clearly model the different aspects of APS that could be modeled, such as planners or the physical system as discussed in (Wiers and Kok 2017). Santa-Eulalia, D'Amours, and Frayret (2012) modeled APS, using an Individual Agent Organization Analysis (IAOA) method for assigning a role to agents and modeling functional elements of APS. Each agent performs its action and interacts with others to determine the behaviour of the system. This feature helped us to develop our model of APS (described in Chapter 5) where we simulated the system from time t to the next $t + \Delta t$. These agents could be as simple as inventory visibility control or complex as those used in production planning development.

As a conclusion to this section, we can state that most research in planning instabilities and nervousness has focused on external sources of instability caused in production planning. Moreover, the methods to dampen instabilities are mostly reactive. Although works like Demirel et al. (2019) proactively looked at the instability of a SC network caused by external activities, there are still several open avenues with respect to the instability of SCP systems due to internal instability related to the interaction of different planning subsystems and also due to the interaction of human planners.

In this thesis, we are trying to fill the gap in the literature with regards to investigating the nervousness in a complex APS embedded within a HPS caused by internal sources, with the goal of proactively dampening instability of this planning system. Our focus is specifically on APS and the nervousness within ATP. For this purpose, we use a multi-paradigm simulation modeling approach of DES and agents, a method that has not yet been exploited in this context. This study sheds light on the interactions between models of planning software modules and models of the physical flow of material within the production and logistical system and the interaction of these two subsystems with regard to the flow of information, planned revisions and update methods that may cause internal nervousness within the ATP. The aim of this research is to find the root causes of these errors within this complex planning system to assist SC experts in redesigning the operation of the APS.

2.5 Allocation Planning in APS

APS is an integrated platform that provides data centrally and decentrally for all planning and scheduling processes. The availability centrally of this data is of great advantage that makes APS an efficient planning approach. During the initial implementation of APS, several simplifications are considered to let the system deployment be completed. This availability of data provides opportunities for constant improvement of APS, although data is large, data transparency depends on organizational structure, system documentation, and integration. These capabilities of APS enable vast possibilities to improve APS functionality and performance. Improving planning decisions by deploying applied mathematics and combinatorial optimization is an open area of research in planning and scheduling. Although mathematical optimization is widely investigated in the literature, its application in real APS in industry has rarely been investigated (Stadtler, Kilger, and Meyr 2015). Modeling and developing a mathematical model for APS requires an understanding of APS processes, the strategic goals of planning, and availability of data. In this section, we discuss some optimization applications in demand fulfilment.

ATP in order management has been studied in the literature from different perspectives. Order promising, which is the allocation of products to customers, belongs to the literature stream ATP in APS (Stadtler, Kilger, and Meyr 2015). In this literature review, we will concentrate on ATP allocation and modeling methods for different production strategies, noting that our case study is based on a hybrid production strategy of Make-To-Stock (MTS), Make-To-Order (MTO), and Make-To-Forecast (MTF).

Within supply chain processes, Quante, Meyr, and Fleischmann (2009) implies that revenue management and demand fulfilment have similar aims. Several authors studied ATP allocation and consumption in APS with the objective of increasing revenue. In MTS production strategy, Meyr (2009) addressed the demand fulfilment problem in the lighting industry by clustering customers into different segments regarding their profitability. The proposed models, according to allocation with and without customer segmentation, were discussed and benefits of using customer segments instead of batch promising with single promising and deterministic known order strategies provided. They showed that prioritizing customers based on segments could possibly increase the profit within order promising.

In Babarogić et al. (2012), the authors considered the allocation of products to customers when a MTS manufacturing system is in short of supply. During the rolling horizon planning system, orders of segmented customers are satisfied without accumulation, which means the number of unsatisfied orders does not transfer to the next planning weeks. In addition, the lowest prior segments should be satisfied by a limited number and oversupply will be saved in stocks for the following planning weeks. The authors modeled the problem by maximization of customer service level which is defined as the fraction of customer orders delivered on time. The results of the model were compared with the results of a heuristic rule-based allocations. Lečić-Cvetković, Atanasov, and Babarogić (2010) dealt with order fulfilment in scarce supply by developing an algorithm to maximize the customer service level in different customer groups. The proposed algorithm prioritizes customers which are of a higher importance to the company with full allocations while only using partial allocations to lower prioritized customers. In addition, they considered back orders in their algorithm. Customer service level in this work was defined as the number of satisfied orders and the percentage of promised orders. While the customer service level was designed to consider long term business planning, the way that customers were classified into groups still followed the revenue management perspective.

Seitz and Grunow (2017) presented a new order promising method to promise orders when products and processes within the supply chain are flexible, customer's order lead times are heterogeneous, and demands are uncertain. Their problem was modeled based on a semiconductor manufacturer where orders should be promised online. They considered demand planning and order forecasts as prior steps before online promising. These steps support the model to cope with changes in production plans that are the result of newly arrived orders. Seitz, Grunow, and Akkerman (2016) modeled allocation planning in semiconductor manufacturing in which data availability and information sharing in a higher granularity level were considered (granularity level defines the level of aggregation/disaggregation within the hierarchical categorization of products or customers). He considered the order forecasts bias to qualify the data from individual customers. This data exploitation resulted in a better allocation plan especially for truthful customers. Cano-Belmán and Meyr (2019) dealt with allocation planning with short supply in multi-stage customer hierarchy. Central and decentral allocations were evaluated for heterogeneous customers which are different according to their behaviour, location, type of requested products, etc. Other authors also added more complexity to encounter with uncertainties in order promising like Grillo et al. (2018) who considered fuzziness and Gössinger and Kalkowski (2015) by considering robustness.

All the studied literature investigates the allocation of short supply to different customers based on customer segments, profit, customer services, or hierarchical system of customers. However, in our case, the role of human planner's decisions in allocation of ATP to customers is required. It enables keeping the flexibility of plans while using mathematical optimization. In fact, we aim to provide optimal allocation plans to be used by allocation planners as benchmarks in their decision making.

2.6 Conclusion

This chapter presents a comprehensive review of SC planning system, APS, instabilities in APS, MBSE for systematic investigation of complex planning systems, simulation approaches for improving APS, and the use of mathematical modeling and decision support tool for the improvement of allocation planning as quantitative approaches. The motivation of this chapter was to indicate the complexity of APS in semiconductor supply chain planning. Besides, we aim to present the gap of a systematic approach to the maintenance and improvement of planning systems, which we aim to improve through the use of MBSE in Chapter 4. Besides, we reviewed the gaps in the use of simulation modeling for studying nervousness in APS. Finally, we reviewed the previous work regarding the improvement of planning systems according to allocation planning and human involvement in planning.

In conclusion, using MBSE as a requirement of quantitative investigation in a complex SC planning system is the first area of the literature which we aim to improve. We discuss the acquired knowledge in Chapter 4. In addition, for revealing the source of nervousness in the planning system, we use simulation modeling and data analysis which we described in Chapter 5. Finally, to improve the allocation planning in APS through digitalization of allocation planning, we use OR methods and develop a quantitative decision support tool described in Chapter 6. These three chapters aim to fill the identified gaps within the literature and solve the identified problems.

Chapter 3

Semiconductor Supply Chain Case Study

3.1 Introduction

The purpose of this research is to evaluate and improve the performance of Advanced Planning Systems (APS) in semiconductor supply chains, specifically the demand fulfilment of APS. To carry out this research, an industrial case study is used as a basis for this research. The case study under investigation consists of an advanced planning system and demand fulfilment of a global semiconductor manufacturer.

Planning the value chain of a global high tech manufacturer like semiconductor requires a custom-specific system design. Thus, the the case's APS is specially designed for a particular Supply Chain (SC) planning system. To investigate and improve this uniquely implemented planning system, this chapter will describe the overall SC planning system of the case study company, with a focus on the relevant APS specifications and demand fulfilment processes. Consequently, this chapter is divided into three sections, with the first providing an overall overview of semiconductor manufacturing and SC, the second section describing APS and Enterprise Resource Planning (ERP) in the case study, and the third describing demand fulfilment and customer logistics management.

3.2 Overview of Semiconductor Case Study

Semiconductors are the backbone of technology development by linking the real world to the digital one. The semiconductor value chain links via fabs to back-ends to customers, which are globally dispersed with key operations worldwide. Semiconductor manufacturing is one of the most complex industries because of hundreds of different ranges of products, globalized supply chains, shrinking product lifecycle, volatile demands, increasing intricate production processes, and long production cycle times, which sometimes requires between four to six months to produce a product. According to a study by McKinsey & Company (Batra et al. 2018), semiconductor supply chain inefficiencies are a major cause of customer dissatisfaction concerning one of the important priorities of customers, which is on-time delivery. Delivering the right product at the right time and at the right location are the harsh obstacles that should be solved in supply chain planning of semiconductor manufacturing. Efficient delivery requires forecasting accuracy, execution quality, and inventory management. Due to varying supply chain issues, companies implement a customized planning system for improving forecasting, execution, and inventory. As a consequence, "the root causes are as complex as the supply chain itself" (Batra et al. 2018).

The case study is an Integrated Device Manufacturer (IDM), which is a type of semiconductor manufacturing company that designs, manufactures, and sells Integrated Circuits (IC). It is structured into four main divisions: Automotive (ATV), Power & Sensor Systems (PSS), Industrial Power Control (IPC) and Digital Security Solutions (DSS).

ATV with 45 percentage segment revenue supports the growing electrification of previously hydraulic and electromechanical systems plus a steady stream of safety, convenience, and lighting innovations. In addition, the ATV product and solution spectrum are powering the transition towards hybrid and pure electric vehicles, and towards higher levels of automation. Examples of ATV product applications are engine and transmission control, shock absorbers, and safety systems like ABS.

The PSS with 27 percentage of segment revenue offers a wide range of power and sensor technologies for efficient power management and high-frequency applications.

Examples of PSS product applications are power supplies for servers, PCs, and medical technology.

IPC with 14 percentage segment revenue provides semiconductors which are used for smart and efficient energy generation, storage, transmission, and consumption. Examples of IPC applications are photovoltaic installations, wind turbines, and high-voltage DC transmission systems.

DSS with 14 percentage segment revenue delivers security solutions and provides robust protection of devices, machines, identities, intellectual property, and data. Examples of DSS are SIM cards, payment cards, and security chips for passports.

Raw silicon wafers traverse through the semiconductor value chain and are eventually delivered to customers. The case study value chain delivers the above-mentioned categories of products using a global Supply Chain Network (SCN). The goal of SCN is to effectively combine the different production sites of the global network. Production sites in semiconductor manufacturing consist of Front-End (FE) and Back-End (BE). The case study global map of production sites showing presently all FEs and BEs is presented in Figure 3.1 . This global value chain is designed to manufacture products efficiently and flexibly.

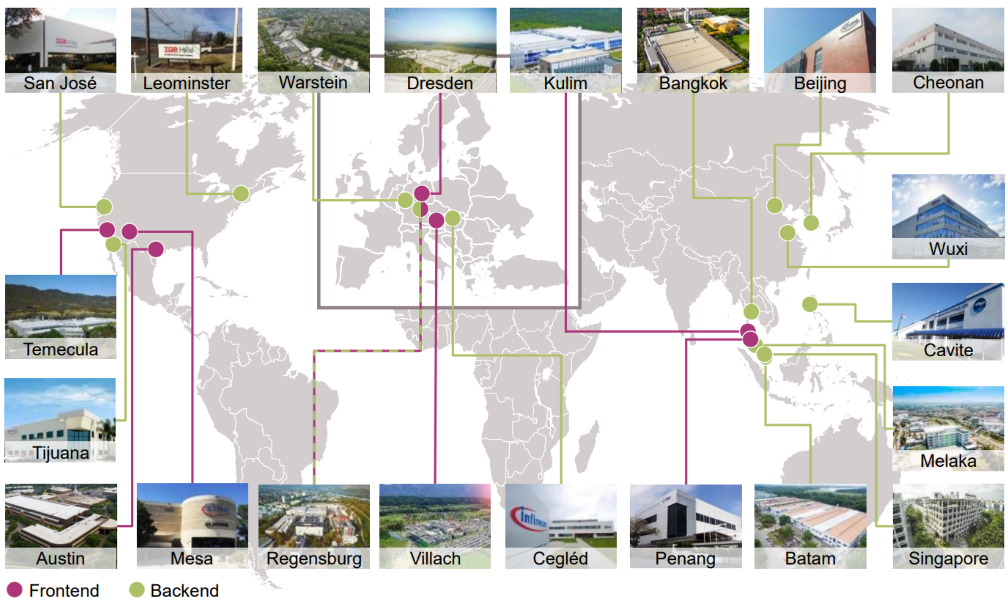


Figure 3.1: Global map of semiconductor's front-end and back-end productions sites (obtained from Dittmann 2018)

As depicted in Figure 3.1 on the preceding page, a SC planning system is required to support a global supply chain where flexibility is paramount to enable high utilization of capital equipment, which is costly. Figure 3.2 shows an example of the SC for a single product produced by the case study company. Typically, a single route in the SC is used at the start of production, but depending on the success of the product, a new SC may evolve over time, where each colour in the diagram represents an evolving SC, with the case study company outsourcing the majority of its production. For example, one route through the SC line consists of Fabrication (label: Fab Germany) occurring in Germany, then Bumping in Taiwan (Bumping B) Taiwan), Sort in Germany (SORT after Bump Germany), assembly in the Philippines (Assy Philippines), and final test in Singapore (Final Test SIN) when the final product is then distributed to the end customer.

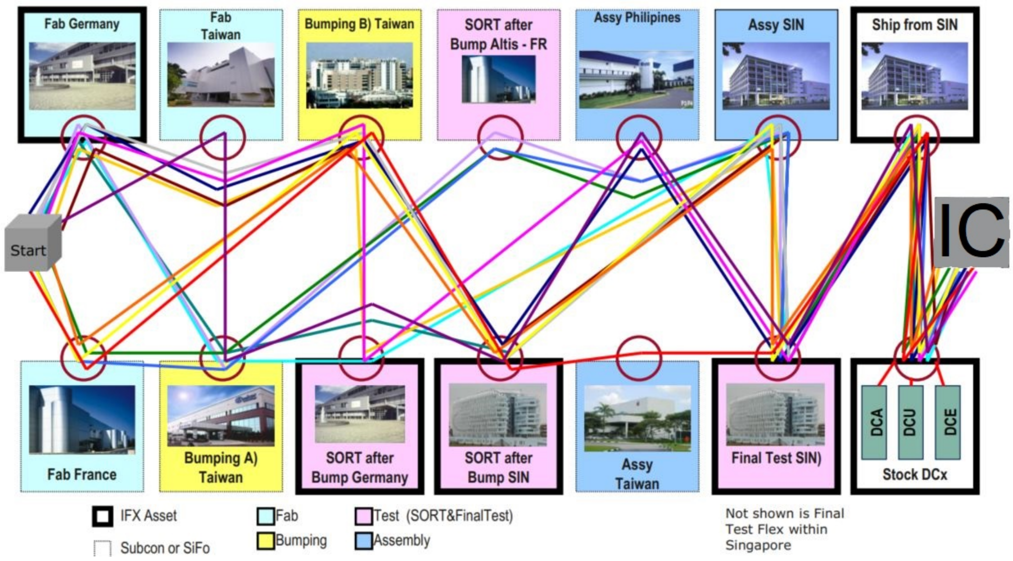


Figure 3.2: Global map of semiconductor's front-end and back-end productions sites (obtained from Ehm, Ponsignon, and Kaufmann 2011).

To manage this SC, a complex socio-technical system consisting of several subsystems is required that can be divided into:

- Production or physical system which are sets of machines and logistics.
- Planning system to maintain plans for the SC system, which differs from the Information Technology (IT) system.

- Expertise system or human planners who are required as “human glue” to manually intervene in the SC, using their cognitive insights.
- IT system which is a set of software modules and databases that interact with humans and integrate with different planning and optimization systems.

For understanding this complex, large-scale and interactive system, several standard modeling frameworks have been developed. The most famous model for Supply Chain Management (SCM) is called SC Operation Reference (SCOR) model. This standard was developed by the SC Council to model, describe, communicate, control, and evaluate SC configurations (Huan, Sheoran, and Wang 2004). The SC framework in semiconductor manufacturing in general and in our case study specifically follows the SCOR model. The focus of this thesis is on improving the planning system. Thus, in the next subsection, we describe the SCOR model for indicating planning system details.

3.2.1 The Supply Chain Operations Reference Model (SCOR Model)

The SCOR model is a process reference model that aims to engineer, benchmark, measure, and design business processes in a cross-functional framework. Supply Chain Council (SCC), an independent non-profit organization, developed and updated versions of the SCOR model. It standardizes the SCM processes to establish a framework of relationships and performance metrics among processes. It manages, practices, and aligns software features and functionality (Huan, Sheoran, and Wang 2004). As described in Zhou et al. (2011), the SCOR model has been widely applied by different companies like Intel, General Electric (GE), IBM, etc. The semiconductor case study also used the SCOR model for designing and managing its supply chain. The SCOR model divides into four levels as (i) top-level or process types, (ii) configuration level or process categorization, (iii) process element level or decomposed processes, and (iv) implementation level. Since we aim to use the SCOR model to better describe our case study supply chain planning, we focus on top-level or process types.

The top level of the SCOR model defines the scope and content of SCM. Five core processes of the top level are Plan, Source, Make, Deliver, and Return. As shown in

Figure 3.3, even a basic supply chain is chains of source, make, and deliver. On top of these interactive processes, Plan links them to manage the SCN. The SCOR model stretches from the source and deliver side of the organization or from the supplier to the customer. In the following, we define the five core processes while the make and plan of our case study is indicated in detail in the following sections.

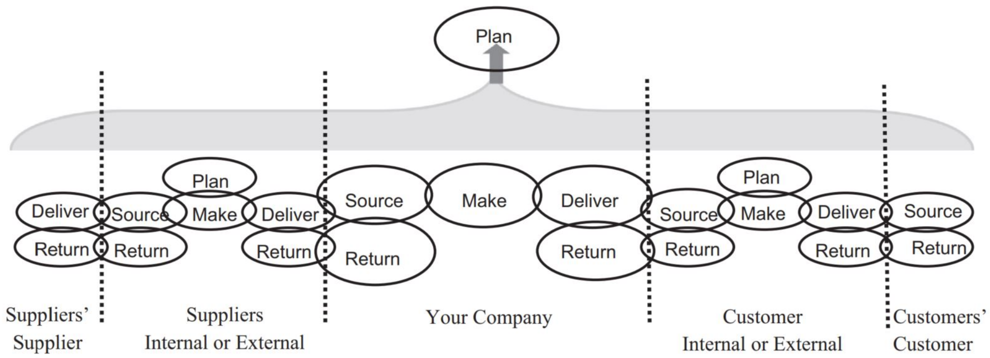


Figure 3.3: Top-level processes of SCOR model (obtained from Zhou et al. 2011).

Source: The Source process includes sourcing logistics for raw materials or finished purchased goods from internal and external suppliers as well as purchasing activities like market research and supplier evaluation and selection. The goals of Source are high security and quality of supply, low material, transportation and warehousing costs, and low inventory.

Deliver: The delivery process to internal and external customers is embedded in Deliver. It includes order management, shipping, invoicing, and order tracking.

Return: Return belongs to all processes that need to be done when procurement and final products have deficiencies and should return to suppliers or the organization, respectively. It consists of inspection, evaluation, transport, processing, and replacement.

Make: Make describes the production and production control processes trying to meet the target of on-time delivery to the warehouse. Furthermore, the Make operation deals with finding the right trade-off between high utilization and short cycle time, ensuring an efficient production.

Plan: Plan is the identification, assessment, and prioritization of supply chain resources and capacities as well as supply chain requirements and demand. Furthermore, both the resources and the requirements are matched with each other to schedule production and commit orders.

3.3 Semiconductor Planning System

Supply chains are very complex that make respecting every detail of reality within plans impossible. To coordinate activities and flows in the supply chain, plans abstract reality in a so-called model. A plan is only a successful one if it is executed at some point reaching the predicted goals defined within the plan. To have an efficient supply chain plan, especially when an order lead time is shorter than the full cycle time, semiconductor supply chains use advanced planning systems incorporated within their ERP. In this section, we aim to describe the fundamentals of planning in the case study from production to the delivery of end products.

3.3.1 Make in supply chain

As already briefly discussed in the previous subsection, the two main elements of semiconductor manufacturing are FE and BE. To traverse products between FE and BE, there are storage points called Die-Banks (DB) and Distribution Centers (DC). In DB, the semifinished products are stored, while in DC the final products are stored prior to delivery to customers.

FE divides into two subprocesses, Fab (fabrication) and Sort. Within the Fab different layers are built up on silicon wafers. The layers have different functions. For instance, in advanced ICs up to 40 layers are built. The wafer diameter ranges from 100mm up to 300mm defining the type and number of chips manufactured. A Fab process used on silicon wafers involves up to 500 chemical and/or physical processes steps are carried out. Lots of these steps need to be executed in clean rooms ensuring high chip quality. To obtain flexibility and to use capital intensive machines efficiently, Fabs work 24 hours and 7 days. Sort is the step after FAB where wafers are tested. Within SORT, broken IC are identified and will not be processed in the BE. After Sort, the diced chipsets are stored in DB. In the case study, there is an integrated global network of Fabs and Sorts to obtain flexibility (see Figure 3.4).

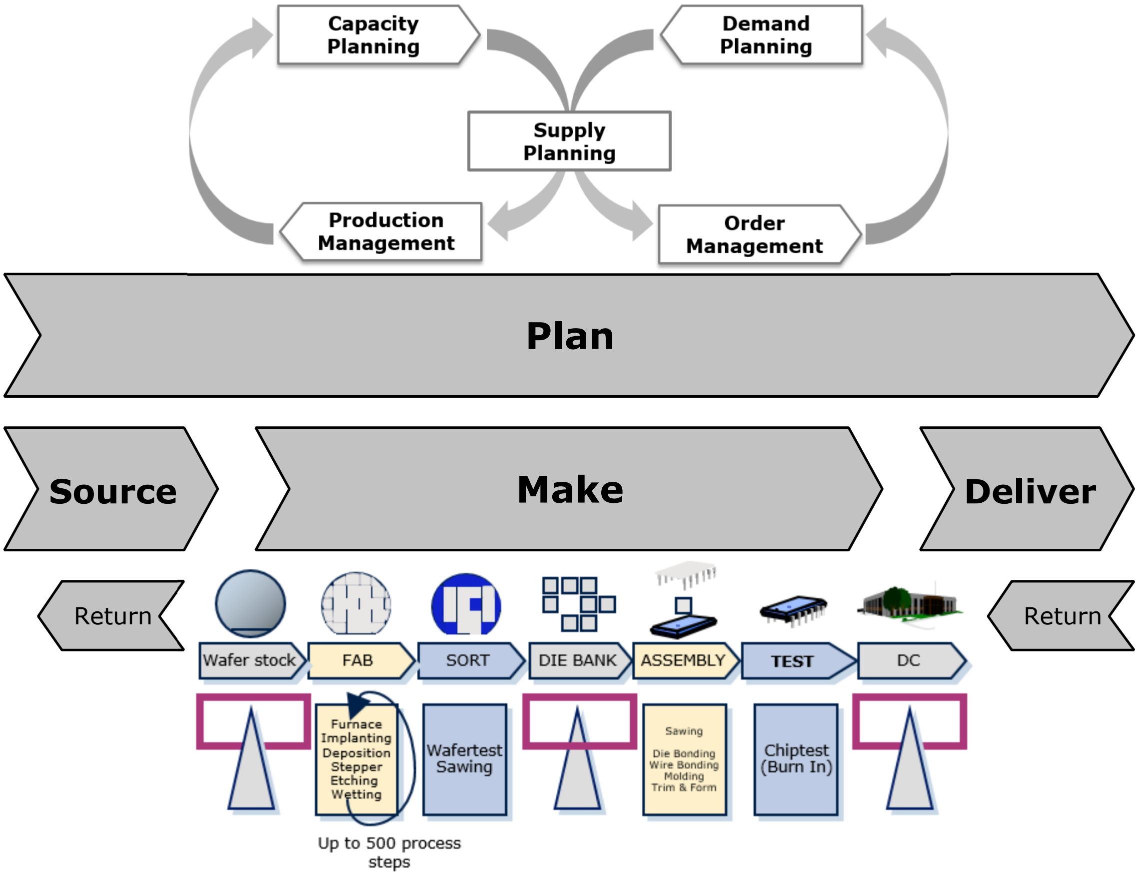


Figure 3.4: SCOR model in the case study's supply chain with make and plan details (derived from Ponsignon 2006, page 20)

The BE also divides into two subprocesses, Assembly and Test. In Assembly, first, the wafers are cut and split into single chips. Afterwards, chips are moulded into a final package to be protected and ready to be attached to a circuit. When the assembly finishes, the finial products are tested. In the Test step, the assembled products are checked to ensure good products are stored in DC. Finally, the tested chips move to finial DC in preparation for delivery to customers globally. This operation of this global and complex network requires a detailed plan, the topic discussed in the following subsection (see Figure 3.4).

3.3.2 Planning principles in the case study

According to Stadtler, Kilger, and Meyr (2015), Supply Chain Management (SCM) is "the task of integrating organizational units along a supply chain and coordinating material, information and financial flows to fulfil (ultimate) customer demands with the aim of improving the competitiveness of a supply chain as a whole". To handle global customers' orders in an increasing competitive market, the SCN needs to deal with divergent and convergent flows of complex networks in parallel (Wiers and Kok 2017). Planning system design, implementation and improvements aims to achieve the objectives of balancing networks of SCM. "Plan covers processes to balance resource capacities with demand requirements and the communication of plans across the supply chain. Also in its scope are measurements of the supply chain performance and management of inventory, assets and transportation, among others" (Stadtler, Kilger, and Meyr 2015). With plans, human experts aim to predict the future and decide the most appropriate operational events.

The semiconductor manufacturing case applied SCOR model to structure planning. Planning semiconductor manufacturing and its supply chain face challenges. In the operational level, long cycle times, large number of processing steps, re-entrant material flow of particular machinery, batch tools, and random failure are examples of the challenges within semiconductor SCM. At the tactical level and strategical level, short lifetimes of products, uncertainty in customers' orders, and globalization of production are examples of challenges. In addition, the semiconductor plan's solutions are derived from many interactive elements and there is no best solution, albeit many feasible solutions exist. Therefore, complex and challenging semiconductor manufacturing at the case study requires an advanced and complex planning system.

As shown in Figure 3.4 on the preceding page, the planning operation of the case is structured into five generic processes which are Demand Planning, Capacity Planning, Supply Planning, Production Planning, and Order Management. In the following, we briefly describe the processes.

Demand Planning assesses and aggregates supply chain requirements. For this purpose, the necessary activities are twofold: First, Sales and Marketing Planners forecast the internal sales demand based on customer data and take strategic products and market updates into account. Second, Supply Chain Planners utilize this forecast information and determine the operational demand for the next 26 weeks. The target levels for stock at each considered stocking point are specified, for example, with a higher weighting at the beginning or at the end of the supply chain plan. The output of the Demand Planning is a demand forecast without including any capacity constraints (Dittmann 2018). We discuss demand planning in further detail in Section 3.5 on page 62.

Capacity Planning involves providing information about the available capacities including bottlenecks. The output of Capacity Planning is the available capacities and bottlenecks in the production sites, which are then provided to Supply Planning. With a long time horizon, investment decisions are supported by this process (Dittmann 2018).

Supply Planning matches the available capacities from Capacity Planning to the requested demand from Demand Planning and prioritizes orders in case of short supply. If capacity and demand are unbalanced in the long run, the planning process recommends to adjust production capacity and investments. Supply Planning is the core of an Advanced Planning Systems (APS) in the case study. It is also called Master Planning. The outputs of Supply Planning are production requests to Production Management and supply to Order Management. Supply contains information on Available To Promise (ATP) quantities for each product in the respective time bucket for the next 26 weeks and information on planned production and availability for shipment at a certain time (Dittmann 2018).

Production Management connects planning activities with production sites. Production Management receives production requests from Supply Planning and makes sure that the desired supply is generated. Production Management defines the production start date and respective quantities over the planning horizon to deliver the products to the customer on time. Before the loading plan is released, optimization runs occur considering, for example, manufacturing and stock levels. Visibility to global production and inventory is provided by Production Management (Dittmann 2018).

Order Management represents the contact point with customers, which is performed by Customer Logistics Managers. The customer orders are promised in Order Management based on the supply provided by Supply Planning. Order Management establishes and communicates supply chain plans to the customers. Within that process, customer orders are received and confirmed according to the information from Supply Planning. After confirmation, the order's status is tracked continuously to discover mismatches early and enhance high delivery reliability. In case of short supply caused by uncertainties, capacity might be reallocated to the most urgent orders to ensure demand fulfilment. Some functionalities of APS are located in Order Management (Dittmann 2018).

3.3.3 Hierarchical Planning and Rolling Horizon

In semiconductor manufacturing, the planning system makes decisions of varying importance from very detailed operational decisions to high-level company strategies. For example, plans define the sequence of jobs in a photolithography station of a production site, the shipment quantity from BEs to DCs, or a decision on continuing production of a product or not. Besides, the plans should satisfy conflicting goals. For example, plans aim to reduce supply chain costs and cycle time, to increase supply chain flexibility and efficiency, to raise customer service levels, and to improve the robustness and accuracy of supply chain planning (Stadtler, Kilger, and Meyr 2015).

On the other hand, plans are not made for eternity and a predefined planning horizon indicates the validity of plans. Thus, within the case study, the importance of decisions to be made and the planning horizon classify the planning task into four linked categories (strategic, tactical, operational, and execution) as shown in Figure 3.5 on the facing page . The strategic planning or long-term planning deals with prerequisites for the development of the supply chain in future. For instance, in the strategic planning of the case study, a new product is introduced and a network for supply is defined. Tactical planning or mid-term planning defines rough quantities and time estimations of material flow and resource availability for the next six to twenty-four months. Short-term planning (operational planning) determines detailed instructions for immediate execution and control of all activities. Short-term planning is restricted to decisions from those made at the tactical level. The business scenario and production program are the most important elements of the case plans. The bottom level is the executive level with its Frontend Steering (FEST) and Backend Steering (BEST) processes describing the production requests in the highest details. All planning horizons are linked. The information contained in the lower level plan is reused in the next planning cycle for higher-level planning.

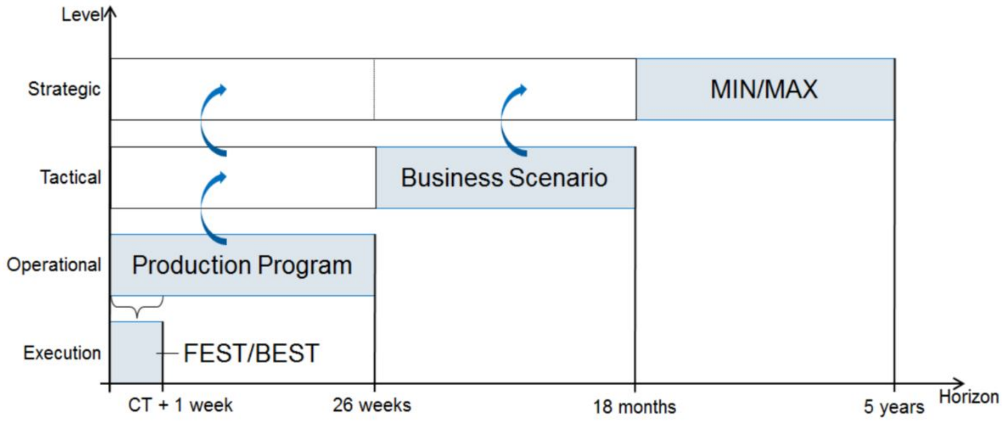


Figure 3.5: Planning horizon and levels in case study (obtained from Dittmann 2018, page 50).

All planning levels coordinate through the flow of materials and information. The decisions on the different levels of the hierarchy are coordinated by top-down instructions and bottom-up feedback. Therefore, the supply chain planning adopts the concept of hierarchical planning. Building on the seven principles decomposition, aggregation, coordination, model building, anticipation, disaggregation, and model solving, the hierarchical planning concept divides a decision problem into structured levels of hierarchically planning and links these levels with each other. Thus, a feasible solution of good quality for the original problem results (Seitz 2017). For better clarification of the planning level as a requirement of hierarchical planning, Stadtler, Kilger, and Meyr (2015) demonstrates these concepts in a supply chain matrix shown in Figure 3.6 on the next page.

This Supply Chain Planning Matrix indicates the planning task according to the planning horizon and supply chain processes (procurement, production, distribution, and sales), the flow of material, and information in the planning horizon. The matrix depicts the interdependencies of supply chain planning between time and activity. Vertical and horizontal information flow are the interconnections between planning levels through

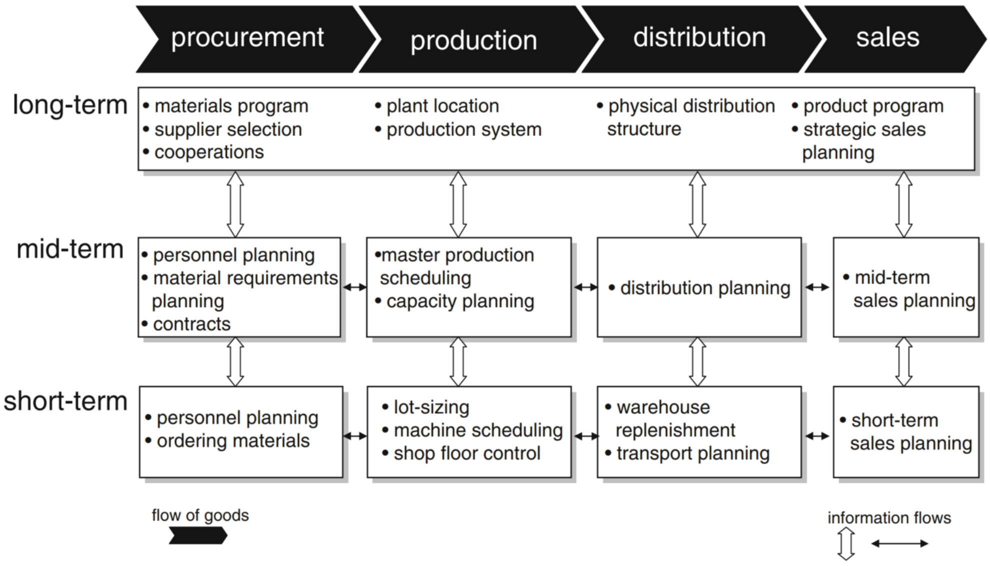


Figure 3.6: Supply Chain Planning Matrix (Stadtler, Kilger, and Meyr 2015).

the planning horizon which require aggregation and disaggregation. As discussed by Seitz (2017) "Aggregation along with the dimensions time and entity (e.g. product, customer, site, etc.) is used to reduce the complexity of the addressed planning problems. Rolling horizon schedules, i.e. the typical iterative re-planning scheme of industrial supply chain planning, result from aggregation along the time dimension."

Planning horizon divides to a defined period (in our case weeks and days). As reality deviates from the plan and as the uncertainties exist, when one defined period passes, the planned updates to the upcoming horizon, which is sometimes referred to as a rolling horizon. In addition to the time-driven update of plans, an event-driven update also causes replanning. The concept of the rolling horizon in hierarchical planning can lead to replanning. An example of a rolling horizon is shown in Figure 3.7 on the facing page, where the planning period is one week and CW refers to Current Week. For this example, the planning horizon is five weeks and at the beginning of each time (t) a plan is made that covers the next five upcoming weeks.

Another principle of planning systems is anticipation and reaction within the abstract model of reality. For each level, models of the decision problem are built. The model anticipates the capabilities and potential reactions of the model to subordinate levels and disaggregates the instructions from higher planning levels. By solving the

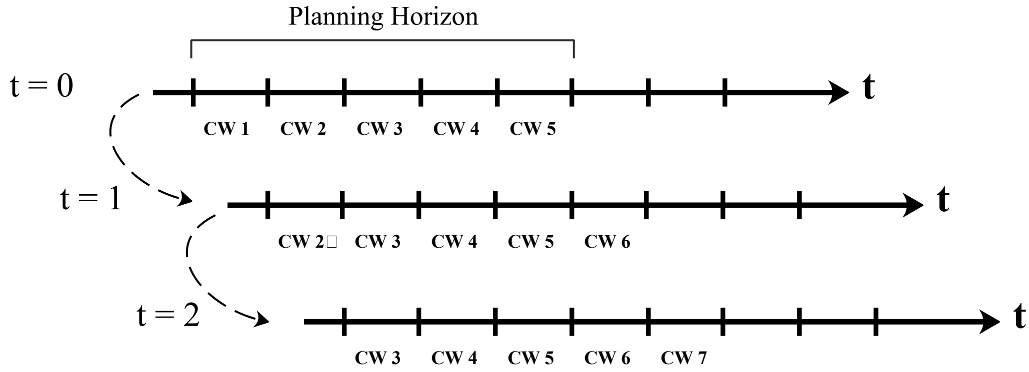


Figure 3.7: Planning Rolling Horizon.

models of the hierarchy properly, the original planning problem is solved. As shown in Figure 3.8 on the next page a planning system has a model of reality. Within each cycle of hierarchical planning, the Top Model creates a top-down influence on the Base Level affecting all components at the Base Level. The decisions in the Top Model are the instructions for the Base Level. On the other side, the Reaction of Base Level results from the Top Level Instruction. Besides, before giving instructions, the Top Model anticipates the reaction by modeling the behaviour of the Base Level, either implicitly or explicitly. This is an anticipated base model constructed based on aggregating information (Zoryk-Schalla, Fransoo, and de Kok 2004; Schneeweiss and Zimmer 2004). Understanding this premise is important for the basics of this thesis. We discuss the concept of an implicit or explicit model of reality in a planning system and the deviation between anticipation and reactions in Chapter 5 on page 117.

3.3.4 Decoupling Point and Product Dimension

In addition to the above-mentioned principles, in the semiconductor case, the concept of hierarchical planning system is constructed based on decoupling points, products dimension, and aggregation/disaggregation of information within the supply chain. Hierarchical planning allows the decomposition of overall planning tasks that considers interdependencies to find a solution in practice (Fleischmann and Meyr 2003). The decoupling point indicates the position in the material flow where the order-driven and forecast-driven activities meet which mostly coincide with stock points. The decoupling

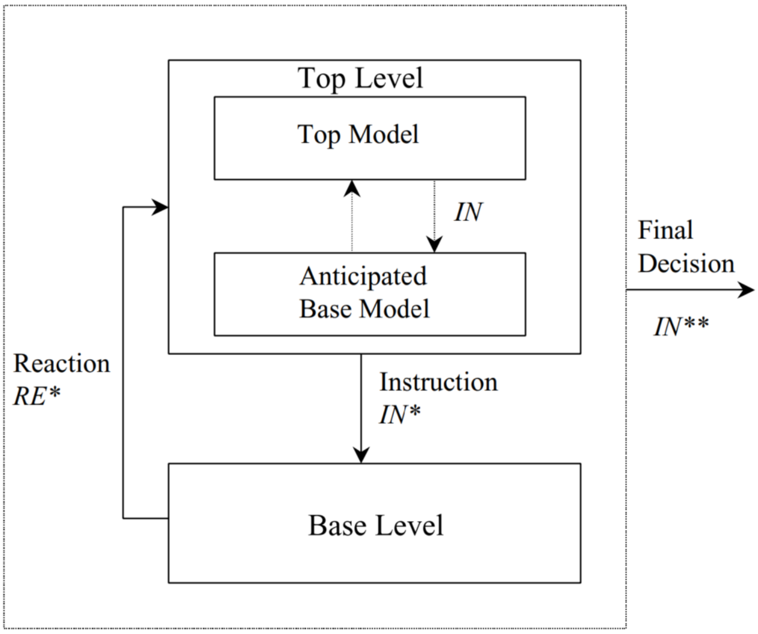


Figure 3.8: Anticipation, reaction, and abstract model of reality in hierarchical planning (Zoryk-Schalla, Fransoo, and de Kok 2004).

points provide the opportunity to reassess the material, semi-product, and product allocation within the supply chain flow. On the other hand, decoupling points define the decision points for planning (Wiers and Kok 2017).

Defining decoupling points depends on the demand penetration points in a planning system that structure the Make-to-Stock (MTS), assemble/configure-to-order (ATO/CTO), and Make-to-Order (MTO) business environments. As presented by Stadtler, Kilger, and Meyr (2015) and Fleischmann and Meyr (2003) which is shown in Figure 3.9 on the next page, in MTS, purchasing and production processes are triggered by forecasts where finished products are stored in stock. In an ATO environment, all components of the finished products are manufactured on the basis of forecasts. Assembly of components to finished products occurs when an actual customer order is received. In a typical MTO environment, only the SC's purchasing process is forecast driven. In the case study researched in this thesis, the business environment consists of all three MTS, ATO, and MTO allowing to locate decoupling points before FE starts (in a form of a forecast of demand), after FE in DB (in a form of semifinished products) and after BE in DC (in a form of finished product). Thus, the planning system needs to define a product (also called product granularity or product hierarchy) to distinguish, prioritize, forecast, and plan the hierarchical planning tasks.

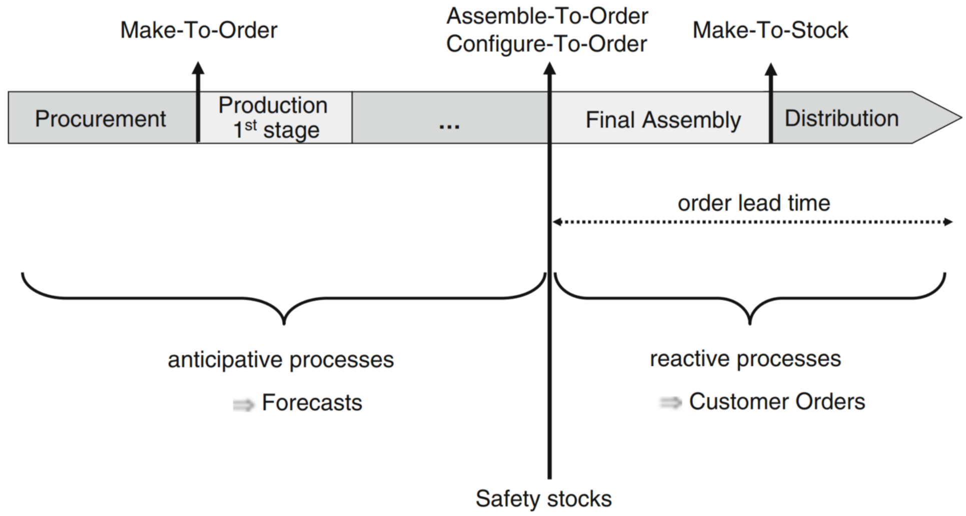


Figure 3.9: Decoupling points in MTS, ATO, and MTO business environment (obtained from Stadtler, Kilger, and Meyr 2015).

A product dimension is another aspect in structuring a planning hierarchy based on decoupling points. Within the case study, defining a product dimension allows the case study to clarify products according to their flow within production steps and according to the enterprise point of view such as finance view, planning view, production view, order management view, and marketing view. In the following, we define the most relevant and important product dimension regarding in this thesis.

Figure 3.10 on the following page depicts product dimensions according to the production view (Bottom left) and Sales and Marketing View (top-right) in the case study. Different hierarchical levels are explained as follow. These hierarchies in Master Data which indicates how the product dimension is stored in a standard framework for operational processes. Note that the product dimension also defines the aggregation and disaggregation of information within information systems in the case study (Ehm, Ponsignon, and Kaufmann 2011). In the rest of this section, the most relevant product dimensions regarding the objectives of this thesis are described (Seitz 2012).

Plan Position (PPOS): The highest aggregation level and linking element between the Sales and Marketing View and the Production View is shown on the upper end of the Product Master Data Hierarchy, which is the PPOS. It is used in Capacity Planning

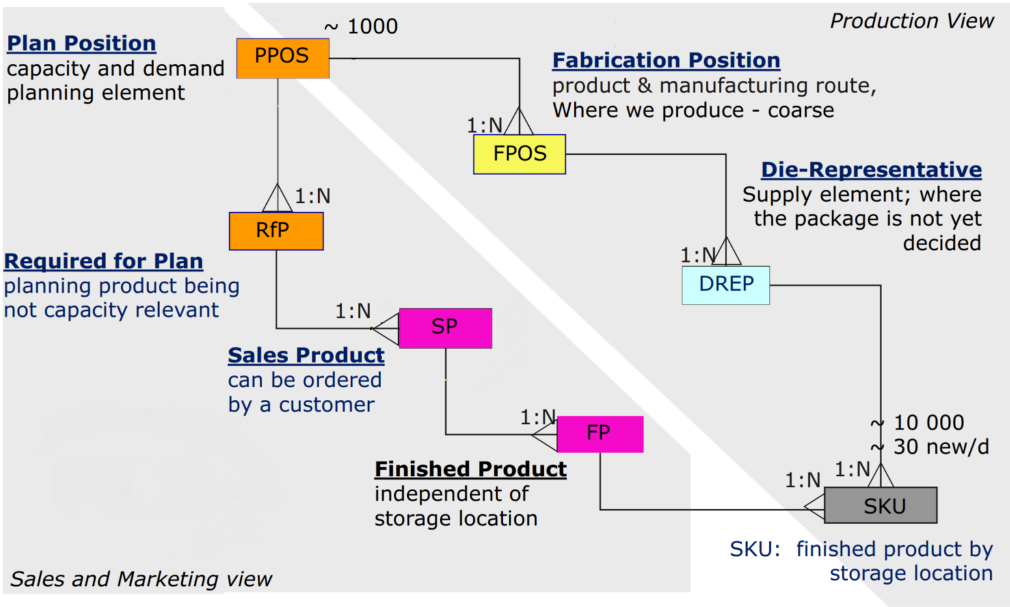


Figure 3.10: Product dimension in master data based on the sale and production view (obtained from Ehm, Ponsignon, and Kaufmann 2011).

and Demand Planning and describes the form, fit, and function of a product.

Required for Plan (RFP): The RFP granularity describes the aggregation level used for the Demand Planning Process. It is linked to the PPOS granularity in an N:1 (RFP:PPOS) relation.

Sales Product (SP): The product granularity on which customer orders are placed is called SP. It represents the external view of sellable products. SP has an N:1 (SP:RFP) relation on the RFP level. Equal to SP there is a product dimension called MA that represents the Manufacturing Product Type when the final product type is defined but it is in production (this is not shown in Figure 3.10).

Finished Product (FP): The SP is disaggregated to the FP level that describes the manufacturing route a product takes from the start of production until it leaves the site where it is tested. There is a 1:N (SP:FP) relation between SP and FP.

Stock Keeping Unit (SKU): The finest product granularity level and linking element between the Sales and Marketing View and the Production View is the so-called SKU. In addition to the information that is available on the higher aggregation levels, it also holds information about the storage location. The SKU is linked to the FP in an N:1 (SKU:FP) relation. Distribution Center Number (DC-N) is a product dimension equivalent to SKU that represents the finished product with respect to the location of the distribution center (this is not shown in Figure 3.10 on the facing page).

Fabrication Position (FPOS): In the Production View, the second highest product granularity after PPOS is the FPOS. Besides information about the form, fit and function of a product, it also describes the manufacturing route with a high degree of abstraction. The FPOS granularity is used for Capacity Planning and in the Supply Picture. PPOS and FPOS have a 1:N (PPOS:FPOS) relation.

Die-Bank Representative (DREP): When the production site of a product is fixed, but the sites where the process steps of assembly and test occur are not yet settled, the product is available in the Supply Picture on the DREP level. The DREP level is linked to the FPOS granularity in an N:1 (DREP:FPOS) relation and has a 1:N (DREP:SKU) relation towards the SKU aggregation level.

The concept of decoupling points and the product dimension defines the aggregation and disaggregation of information within the planning system. To better understand this, we describe the relation of product dimension with Make (see Figure 3.4 on page 45). The planned capacity before production starts in FE is FPOS and when the product is located in the first production process, its dimension changes to DREP. The DREP is disaggregated to MAs (equal to SP in product level), as soon as the product enters the last production process. When BE is finished, the MA changes to FP to represent the finalized products. As soon as an order or forecast gets promised to supply on the FPOS level, its granularity is broken down to the SKU level, as its production route and storage location are being fixed simultaneously. SP also is a higher level of FP that represents the sale products. Note that regarding the sale product, several FP can be interchangeably allocated for the same purpose. As a result, SP has 1:N relation with FP.

To sum up this section, we indicate the basic principles of supply chain and planning system in general and specifically in the case . In this subsection, we covered hierarchical planning, rolling horizon, anticipation/reaction, aggregation/disaggregation, and product master data in the semiconductor case study. This subsection is a prerequisite for discussing the concept of an Advanced Planning System (APS) and the implemented case-specific APS in the semiconductor case study.

3.4 Advanced Planning System

Finding the best plan will have three main challenges. First, plans aim to satisfy conflicting objectives. For instance, although customer service levels should be as good as possible, the inventory level should be minimized. Second, the large number of alternatives are infinite or combinatorially large, making one by one comparison impossible. Finally, dealing with uncertainty as plans are anticipations of future activities (Stadtler, Kilger, and Meyr 2015). Uncertainty implies the presence of different possible future realizations of demand, supply, and yields. Thus, planning intervals and hierarchical planning evolved to produce acceptable solutions and define the frequency of replanning, referred to in the literature as Uncertainty-Induced Complexity (UI-C). Another complexity of a supply-demand balancing problem is so-called Structural Complexity (S-C) that concerns the number of items, resources, item-item relations (relation of items and bill of material), and item-resource relations (routing of items along with different resources) (Wiers and Kok 2017).

The strength of Enterprise Resource Planning (ERP) is well-known in the integrated management and transactional backbone of companies (like human resources, finance, customer service, etc.), but there are planning challenges caused by the complexity and inefficient implementation of ERP, while acknowledging that ERP has useful planning functionality through the use of Material Requirement Planning I (MRP I) and MRP II (Stadtler 2005). This lack of planning support in ERP belongs to deficiencies of MRP I and MRP II, however, ERP should be seen as an informational backbone of companies. Despite the weaknesses of planning in ERP, such systems have been an important enabler for the development of APS, as ERP systems provide the essential data needed by APS. APS are deployed on ERP to fill the gap in planning found in ERP (Wiers and Kok 2017).

APS is a type of information system with functionality to support planning. APS consists of several software modules, each covering certain tasks in planning and scheduling.

APS very often needs to be custom developed specifically based on a company's requirements. APS develops plans based on hierarchical planning by using solution approaches like mathematical optimization, meta-heuristic, and rule-based algorithms. As shown in Figure 3.11, Stadtler, Kilger, and Meyr (2015) and Stadtler (2005) defines and categorizes planning modules and functionalities to plan and manage SC within the SCP-Matrix. Software vendors customize and deploy some of the APS modules within the ERP system. APS is a set of interactive modules to realize real execution in the supply chain by sending plans or targets (anticipations) and receiving executed actions (feed-backs). There is no common definition of APS, therefore we review the definition of the APS from a design and implementation point of view.

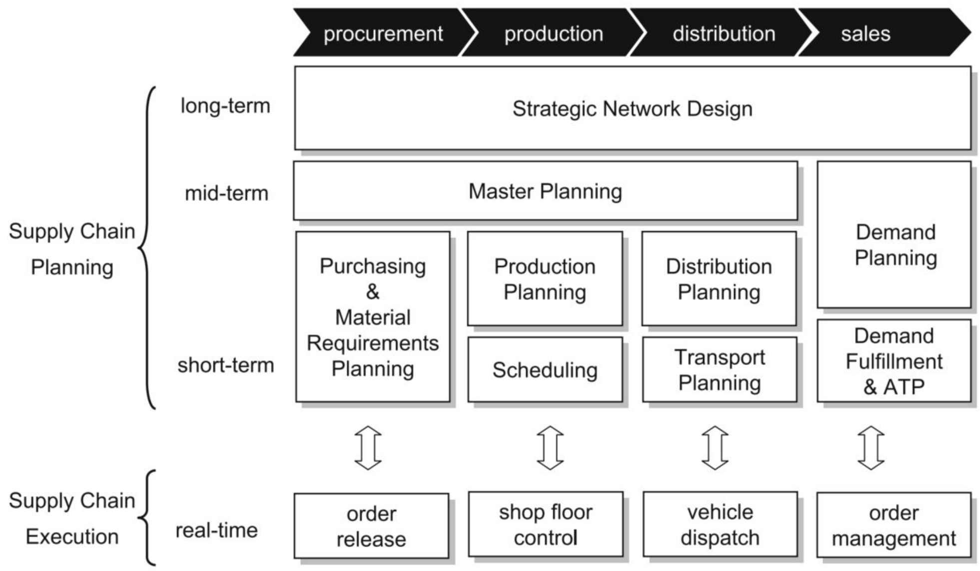


Figure 3.11: APS modules functionalities covering SCP-Matrix (obtained from Stadtler, Kilger, and Meyr 2015).

As defined by Wiers and Kok (2017), an APS is an interactive planning tool that has a model of the real physical system (implicitly or explicitly) and a set of engines for recalculating the consequences of planning actions immediately. The engines run custom-specific algorithms based on process flows of planning actions to generate plans and schedules. Required information for APS is provided by ERP and stored in random access memory (RAM). Usually, the information reads from different databases according to the designed process flow of engines. Besides, human planners manipulate plans or plan's input parameters of APS by the support of a graphical user interface (GUI).

The focus of this thesis is on the Master Planning and Demand Fulfilment & Available to Promise (ATP) functionality within the APS system in the semiconductor case study (see Figure 3.11 on the previous page). Master Planning aims to match operational demands with capacity bottlenecks to define targets of production planning and calculate possible future supplies called ATP. Demand Fulfilment (DF) in the semiconductor case study aims to provide promises and repromises to customer orders based on ATP and open orders. In the next subsection, we describe the modules in the case's planning system.

3.4.1 APS modules in the case study

To generate flexible plans, supply chain planning of semiconductor manufacturing uses hierarchical planning, APS, decoupling points, and rolling horizons. The case Supply Chain Planning (SCP) comprises of multiple software vendors deployed in the ERP system to meet the multidimensional planning requirements. The modules depicted in Figure 3.12 on the facing page are divided into five main categories of planning systems, Capacity Planning (red coloured), Demand Planning (orange coloured), Supply Planning (green coloured), Production Management (blue coloured), and Order Management (purple coloured) (discussed in subsection 3.3.2 on page 45). Each planning module contains a set of algorithms and engines which run either automatically or with the support of humans. As indicated by the legend given at the bottom of the figure, each module contains three separate types of information about the module, that is, the name of the system, its software vendor, and the human planners' used. The dashed boxes belong to systems owned by the semiconductor case study which are more dependant on the planners' inputs. On the top of the diagram, the four above systems (Aggregate Capacity, Scenario Planning, Marketing Demand, and Sales Demand) define the planning modules in the tactical level, also called a business scenario. The planning horizon of the Business Scenario is greater than six months, which aggregates information from the lower level Production Plan which has a planning horizon of 0-6 months.

The production plan or operational level, which defines the plans for the upcoming

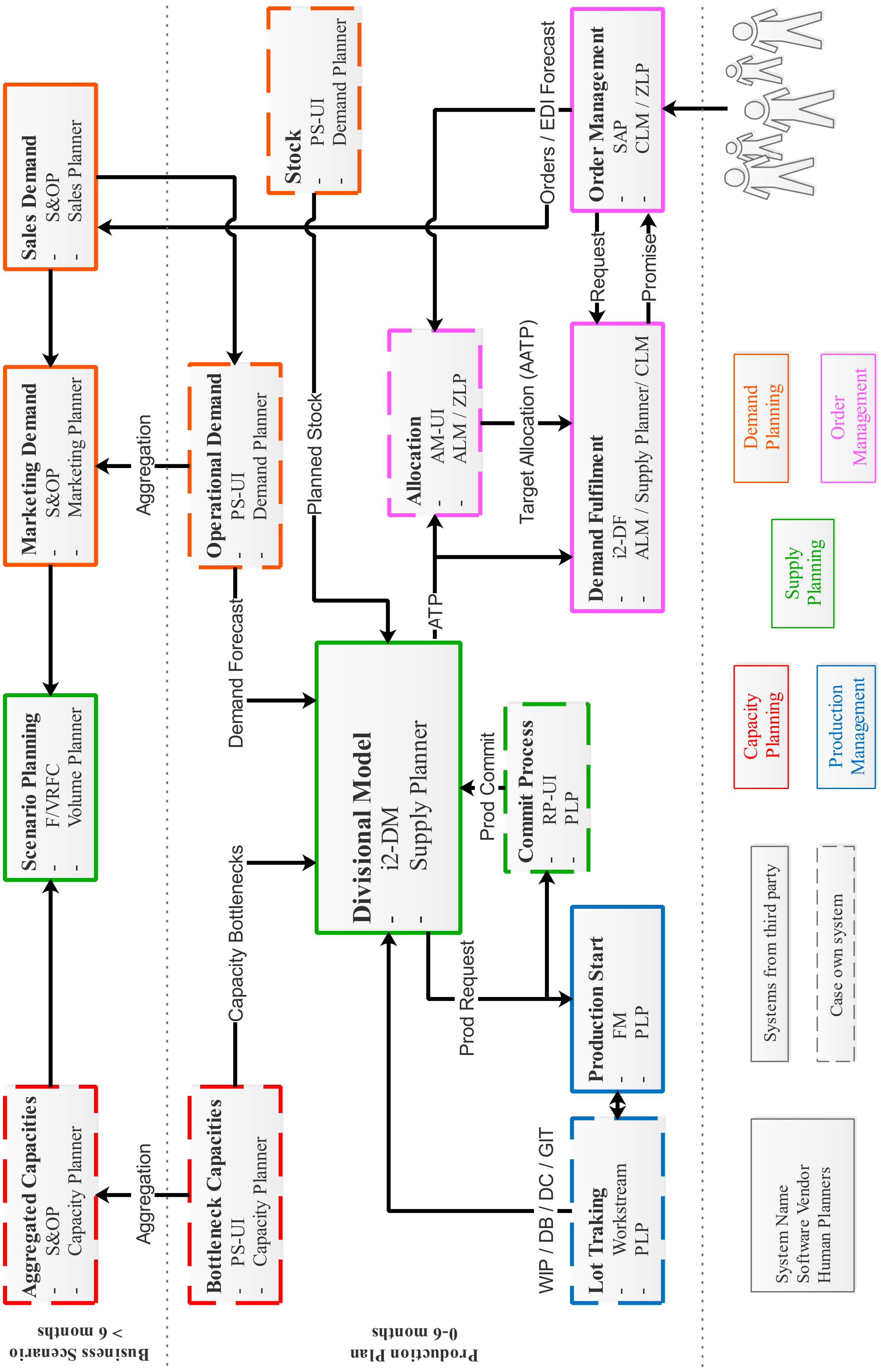


Figure 3.12: Case Study Supply Chain Planning Modules and Functionalities (derived from Achter et al. 2017, page 3512)

6 months (or 26 weeks), is developed in the semiconductor case study's APS. The overall functionality of the production plan starts by entering the order into Order Management. There are two types of orders, customer orders (these represent the obligation to deliver a product) and customer forecast or EDI Forecast (these represent an indicator to customers regarding what they plan to order and there is no obligation of delivery).

Orders divide into further categorizations that we neglect to consider in this thesis. For example, customer orders divide into standard orders, call off, consignment, replenishment, and material pull. These orders maybe handled by the SAP software depending on Customer Logistic Managers (CLM). They are negotiating with customers to balance the input orders according to the the semiconductor case study's strategy and availability of products. When a customer inputs an order, the planning system provides the delivery date and quantity based on the production plan, availability plan of products, and distribution plans. SAP is a gate for communicating with customers, providing promises, and balancing input orders. SAP software also provides a myriad of reports. One of the main functionalities of SAP is providing categorized and sorted orders (based on product type, dimension, requested time, and regions) for other modules like Allocation, Divisional Model (indirect), Operational Demand, and Sales Demand. In the case study, orders that are still open are stored in a database called Open Order Book (OOB). Besides, Order Management can promise orders using two approaches. First, as soon as an order arrives, it provides an online promise based on ATP. Second, it sends batched orders to Demand Fulfilment (DF) as a request. DF provides promises according to sets of rule-based algorithms (Batch-Rescheduling Cycle) and ATP. Another important module is Allocation with a software module called AM-UI (Allocation Maintenance- User Interface) in this thesis. This module plays a key role in managing the allocation of product to customers with the support of human planners, especially when the automatic algorithms of DF could not peg the products (ATP) to orders. For example, when the supply picture or ATP is less than demand. The output of Allocation is called Allocated Available To Promise (AATP) that indicate the amount consumed from Gross ATP. AATP subtracts from ATP in the automatic allocation steps of DF.

In the production plan level, Demand Planning (DP) calculates, prioritizes, and aggregates the demand forecast and planned stock. PS-UI is the software that supports a demand planner to develop demand forecast and stock plans. Demand Planning aims to generate prioritized demand every day. The prioritization covers planned stocks, forecasts, and orders. To plan the demand, three main elements are required, demand planner's input, target stock, and operational demand algorithms. It consists of nine different steps that are done automatically based on the automated rule-based algorithms and inputs used by the demand planners. The output of Operational Demand and Stock are calculated in the software (PS-UI) developed by the case study, but we separate them in Figure 3.12 on page 59 to better understand the steps. Thus, Operational Demand and Stock send one input to Divisional Model, which is the prioritized demands (forecast, orders, and stocks). Details of Demand Planning steps are described in Section 3.5 on the next page.

PS-UI also supports Capacity Planning by calculating capacity bottlenecks. The capacities of production sites based on product dimensions are gathered and calculated by the support of capacity planners. The output of Capacity Planning is used as constraints in calculating ATP in the Divisional Model.

The other SCP's subsystem is Production Management, which aims to schedule and track the status of production. Besides, it defines production targets to schedule the start of production. The Lot Tracking module is a developed software that integrates with the Divisional Model by monitoring the quantity of product dimensions in different stages such as: Work In Progress (WIP), Die Bank (DB), Distribution Center (DC), and Goods in Transit (GIT). In the opposite direction, Divisional Model creates targets of production according to location, product dimensions, delivery targets, and quantity targets. This production request (Prod Request) is sent to Production Start daily. These activities are monitored and controlled by a developed system called Commit Process. It identifies the production targets for production. Commit Process belongs to Supply Planning. The related software is called RP-UI, which is similar to other case developed software with the aim of supporting human planners.

Supply Planning consists of four software subsystems. The core module of APS belongs to the Supply Planning subsystem, which is denoted as the Divisional Model (DM). Capacity bottlenecks, demand forecasts, planned stocks, and visibility of products are all inputs of the DM. Every day, the DM is provided with these values that are used to run automated algorithms to match supply to demand and to provide visibility in the planning horizon. One of the main outputs of DM is Available to Promise (ATP). ATP enables efficient handling of volatile demands and supplies that provides promises to existing and upcoming customer requests. ATP as an inevitable part of APS is composed of different product dimensions (discussed in Subsection 3.3.4 on page 51) within the supply chain. Thus, it is like a current and future picture of available products that enables dynamic allocation of volatile products to dynamic orders. DM uses i2 Technology (i2 DM) for creating ATP which integrates with DF also developed by i2 technology (i2 DF). Further discussions regarding the DM algorithms and ATP generation are presented in Section 3.6 on page 66.

In this section, the overall functionality and definitions of the deployed APS were discussed. Although all modules in each system are considered as a part of APS (shown in Figure 3.12 on page 59), the Divisional Model and Demand Fulfilment are the main parts of APS, with both provided by i2-Technology. These two are the most automated parts of APS which are the focus of this thesis. Within the next sections, we aim to describe in detail the algorithms and processes of APS in the case study which are related to the scope of this thesis.

3.5 Operational Demand and Stock

With the focus on customer satisfaction, interlinked and coordinated planning systems are the key concepts behind APS in today's global and competitive semiconductor industry. APS equipped with demand forecasts are required to develop a coordination plan. In the case study demand is split into three major types: Customer Demand (represent obligations to deliver), Customer Forecast (represent what customers plan to order), and Demand Plan (represent the case study estimations of its future business). These three are the building blocks of inputs to carry out Operational Demand (see Figure 3.12 on page 59).

From the system point of view, demand planning at the production program level can be divided into seven subprocess steps: Demand Plan entry, Order Categorization, Demand Netting, Demand Disaggregation, Demand Alternate, Target Stock Entry and Demand Prioritization. All of these tasks are done in a tool named SPLUI. Most of the planning tasks are automated and work according to the rules defined by the demand planners. The seven subprocess steps shown in Figure 3.13 are an element of Supply Planning and Order Management (see Figure 3.12 on page 59). Next, we describe each of these seven subprocesses.

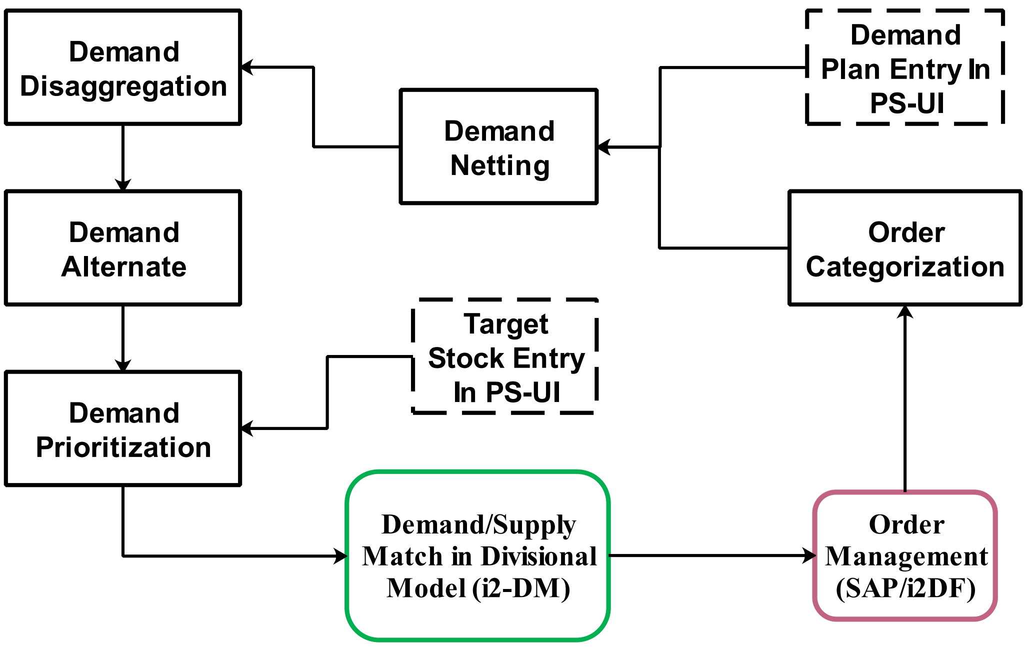


Figure 3.13: Demand Planning steps in production level in the case study.

Order Categorization (see Figure 3.13) in the case study aims to automatically categorize orders into different demand types, with the aim of keeping the commitment of already confirmed orders and to provide sorted orders for the netting step. The three prioritized categorizes are:

- Confirmed on time (CO) that means the customer wish dates for when the order is delivered and this date is confirmed by the company.

- Late confirmed order (LCO) that means the delivery date confirmed by the company for this order is later than the customer wish date for the order.
- And unconfirmed order (UCO) that there is no confirmed delivery date available yet for this order.

The orders are categorized in the finished product level. After categorization, orders are sent to netting in the sequence of wished date, demand type, and order ID called ECT (Earliest Creation Time) rule. First orders are arranged by their wish date. The oldest wish date comes first. If two or more orders have the same wish date, they are arranged by their demand type (listed above). First comes CO, then LCO and finally UCO. If some orders get the same wish date and have the same demand type, they are sorted by their Order ID with the lowest ID first, the highest ID last (Wagensonner 2018).

Demand Plan Entry In PS-UI (see Figure 3.13 on the previous page) is a parallel step carried out with Order Categorization. Demand planners calculate and enter the estimation or forecast of demand into PS-UI as a demand forecast. The Demand Forecast is an estimation of what customers want in the future according to marketing and business. These estimations consist of the number of product pieces for the upcoming 6 months by considering ramp-ups/down or seasonal effects. The estimations are entered into PS-UI every week. The quantities are not constrained by the actual supply, but developed according to the amount that the case study could sell to customers. The demand forecast is generated at different product levels (like PPOS, MA, or DC-N). They also have three prioritizes that are forecast with allocation priority (AP-FCST), forecast with order priority (OP-FCST), and forecast with normal priority (NP-FCST), respectively (Wagensonner 2018).

Based on these prioritizes, the demand forecasts and categorized orders are netted in Demand Netting (see Figure 3.13 on the preceding page). In Demand Netting, the categorized orders on hand are automatically mapped to demand forecasts using a greedy algorithm that divides between backward and forward searches based on the requested delivery date of each order. During netting three rules are followed. First, in a given period, an order is netted to the highest priority. Second, if all demand forecasts for one type (e.g. AP-FCST) is consumed in the requested week, an excess forecast of the specific type in the previous weeks is used; and if it is consumed in the previous week then the excess forecast of the specific type in the following weeks is used. When the forecasts in one priority finishes, it goes to the next lower priority of demand forecasts. As forecasts contain different product dimensions, orders are all in the finished product dimension (DC-N). Thus, within the algorithm's rules of demand netting, first look for a forecast target on the DC-N level, if it could not net, it tries to net with other dimensions such as MA, SP, and PPOS, respectively.

The next step is the Demand Disaggregation. As discussed, demand forecasts are in different product dimensions, while orders are in the finished product dimension. From this perspective, the objective of Demand Planning in the case study is to provide reliable forecasts to be used in DM and DM only plans on the DC-N level of a product dimension. Besides, when forecasts could not net to orders the excess forecasts remain in the higher granularity dimension (Aggregated level). Thus, the excess forecast needs to be disaggregated to the finished product level.

After developing the disaggregated demand plan, the entered Target Stock is created by demand planners and added to the disaggregation output. The case study to protect against demand fluctuations and to improve the reliability of delivery consider buffer stocks. As shown in Figure 3.12 on page 59, Target Stock or planned stock is represented in a separate module to Operational Demand. It shows that operational demand and planned stock are different while both are calculated in PS-UI. In the Stock module, planners define the minimum and maximum level of stocks individually. This is not related to monitoring or representing the current level of stock. It aims to manipulate the stock level strategies of the company within supply chain planning. A stock target could be seen as a demand that the case study keep for upcoming orders or fluctuations within real orders.

In the final step of demand planning, disaggregated demand and planned stock are prioritized within Demand Prioritization. This step sorts all demand and stock in one repository and priorities them for use by DM. The standard priorities in the case study are Retain Stock (stocks use in allocation time), Safety Stock (stocks use to hedge against supply fluctuations), AP-FCST, OP-FCST, Ramp-Up Stock (use for ramp-up products and can not be consumed by demand), Excess Order, NP-FCST, and Min Stock.

Consequently, in the Demand Planning system, the outputs are prioritized demand and stock plans for the 6 months planning horizon on the finished product dimension (DC-N). This output is one of the inputs of DM for calculating ATP.

3.6 Divisional Model

Planning in an ERP system depends on MRP. MRP assumes that beyond the standard production lead time, no material constraint exists. MRP assumes any preferred quantity of supply is available for fulfilling incoming orders (Stadtler, Kilger, and Meyr 2015). MRP has deficiencies not only in planning resources, material availability, and capacity planning but also in allocation and synchronization (Wiers and Kok 2017). ERP could not take account of production's constraints and demand uncertainties, which propelled companies to deploy Master Planning within ERP. Master Planning creates a comprehensive mid-term supply plan that takes into account all sourcing, manufacturing and logistics processes by considering possible production capabilities and resource bottlenecks.

In the semiconductor case study, Master Planning is the main part of APS and SC-Matrix which is handled by a module called Divisional Model (DM) (see Figure 3.6 on page 50). DM aims to consider all capabilities and bottlenecks within production. DM balances demand and supply within the planning horizon. To do this, it requires to have a model of reality and a defined set of processes to handle the complexity of matching demand with supply.

In this part, we define the aim and scope of DM in the case study from three different perspectives. First, we discuss the functionalities and rule-based algorithms within DM. Second, we clarify the Process Flow within DM to indicate the steps in transferring inputs to outputs. Third, we explain the DM's input and output that consist of visibility, interactions, and ATP. Since defining ATP is one of the main purposes of this chapter, we consider ATP in a separate subsection. Thus, in this section, we will take a closer look at the current DM of the case study according to the scope of this thesis.

3.6.1 DM functionalities

The goal of Supply Planning (see Figure 3.12 on page 59) is balancing demand with supply. The purpose of Supply Planning is to align the results of the Capacity Planning and Demand Planning to determine what can be sold into the market. Every week, alignments occur to improve the output of planning. Supply Planning provides inputs for Production Management and Order Management. The core of of supply planning at the operational level are handled by DM, which consists of several different functions.

The DM aligns capacity and demand planning for production. Therefore, the inputs of DM are the results from PS-UI on the demand and capacity planning side (see Figure 3.12 on page 59). The output of the so-called Demand-Supply Match (see Figure 3.13 on page 63) is then transferred to the order promising interface and the production site interface. DM provides the supply capability of the case study as the available supply which is used in Order Management.

For generating plans, the Demand-Supply Match algorithm uses backward and forward calculations. For these heuristic calculations, there are sets of prioritizes that are discussed in Section 3.5 on page 62. Demand-Supply Match search supplies backward based on prioritized demands. In this search, it looks for products in stock, transit, BE, and FE, respectively.

Demand, as discussed in Section 3.5 on page 62, consists of netted demand with orders, minimum stock, and safety stock. Capacity bottlenecks are calculated in each production step which is inputted to DM. Supply provides stock levels & WIP (Work In Process), GIT (Goods in Transit), and the requests that are in Frozen Fence (FF) (Req in FF) and Allocation Fence (AF) (Req in AF). This information is provided for the whole planning horizon. The DM is based on priorities that start the backward search which matches demand with supply. The output of this Demand-Supply Match sends requests or production targets to AM-UI (see Figure 3.12 on page 59).

The DM's algorithm is as follow. Demand-Supply Match search for products in DC stock. If the demand matches, it is ok, otherwise it searches for WIP and GIT backwards. When the demand and supply match, the ATP is generated. Any demand without sufficient supply needs a new production request. When the capacity is not sufficient, DM will try to produce it in earlier weeks. For example, consider that the demand is six pieces and there are two pieces in stock. Therefore, four more pieces are required. In the backward search, three more pieces were found in a BE. Since there are no more pieces in a BE, the remaining piece should be set as a production target of a BE.

3.6.2 Process Flow

The DM and APS run on centralized engines (Machines). Most of the considered algorithms within this thesis are automatic algorithms. They consist of rigid steps from gathering data, to providing solutions. The designed steps for generating solutions within algorithms are called process flows. Here we aim to describe the process flow of algorithms and relevant data gathering in DM. The process flow of DM is integrated with the process of order management. The important point here is that the designed process could be the source of inefficiency and instabilities within planning.

The process flow of DM and Batch Rescheduling are shown in Figure 3.14. In this figure, the period is one day for the engine planning process (0:00 to 24:00 see Figure 3.14).

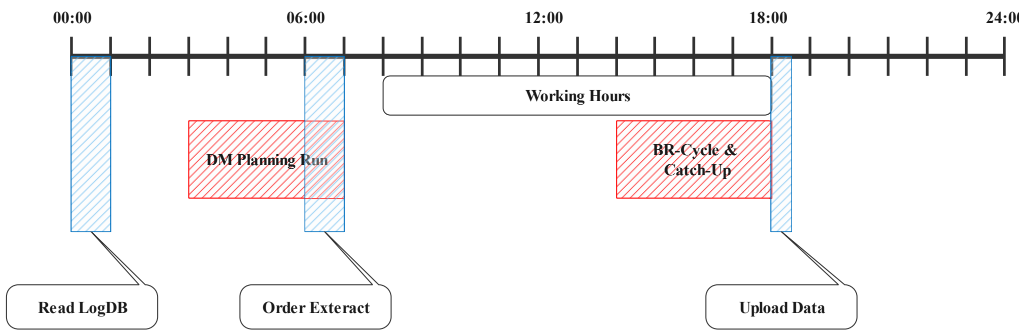


Figure 3.14: Process Flow of Divisional Model and Demand Fulfilment.

Every day (except one day) at 0:00 am, data for DM planning is extracted into a database called LogDB. Order data is also extracted into the database called logDB-

Orders. From 3:00 am until 7:00 am, DM planning occurs and calculates the newest ATP values. At 6:00 am another order data extraction occurs that updates the order changes that happened. A Batch Rescheduling Run occurs from 2:00 pm until 6:00 pm. Afterwards, there is a catch-up process that synchronizes changes of existing orders and new arriving orders. Finally, new data is uploaded to SAP and, based on this data a new SAP Early Warnings (EW) Report is generated. Further details of OM are discussed in Section 3.7 on page 72. All data read from different databases is denoted with Die-Banks in the data base, for example, LogDB Orders in Figure 3.14 on the facing page.Note that this figure is a sample of processes flow, in the real case there are many other modules and databases.

As discussed in Subsection 3.3.3 on page 48 APS contains a Top Model (see Figure 3.8 on page 52) of how its actions are executed to facilitate feedback. This model sends instructions to the Base Level and calculates plans based on anticipating changes to the Base Level. The model of reality in the case study is implicit, which means there is no exact structured model as a model of reality in DM (Wiers and Kok 2017). While there is no *explicit* structured and documented model inside DM, the process flow is an aspect of an existing model. For such a model, when and what data should be extracted and what algorithms should run inside DM.

3.6.3 Input and Output Data

As indicated, DM receives data from different systems and modules such as forecast data which come from demand planning, capacity bottlenecks that incorporate for each potential resource the details of the production steps, and the minimum and maximum level of stock for each inventory node. DM needs not only current stock (stock level in DC), Work in Progress (WIP), and products in logistics (Goods in Transit), but also input parameters for modifying internal models. For instance, the production yield of each specific production plant. This value uses anticipations calculated within DM's model of reality (see Figure 3.8 on page 52).

Figure 3.15 on the following page shows the inputs and outputs of DM considered in this thesis. In the semiconductor case, five main databases export data to DM. Capacity Bottlenecks and Planned Demand which were discussed in the previous sections. The other three items are: WIP which represents the products that are in production, divided according to decoupling points (WIP in the back end, front end, and in planning (FPOS)); Stock which is the availability of products in DCs; and finally, the Goods In Transit (GIT) which represent the number of products being shipped between decoupling points.

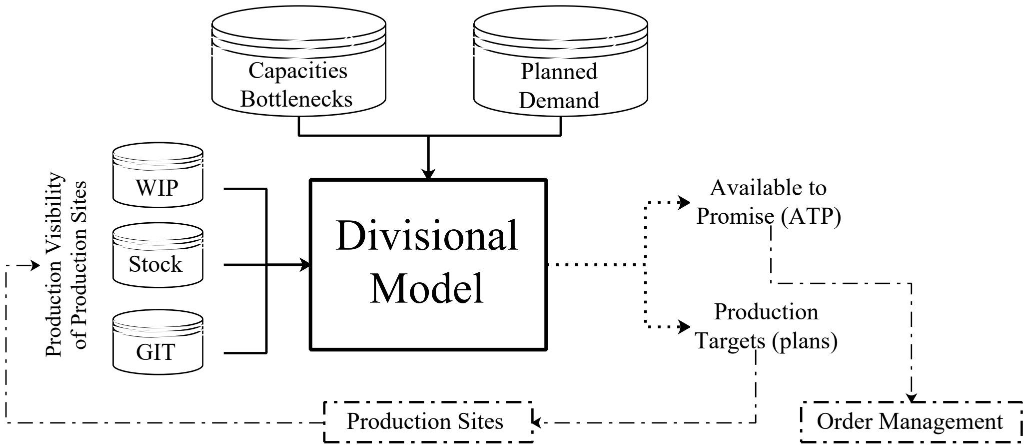


Figure 3.15: DM's Input and Output Data.

The critical thing about these databases is that they are extracted once a day or once a week, while a new value could be added to the database at anytime. The act of gathering or reading the data is an interaction between the real execution and the planning system. This interaction (data extraction) is equal to the reaction from the Base Model to the Top Level model, discussed in Subsection 3.3.3 on page 48. The sequence of interactions is the window of visibility (Stadtler, Kilger, and Meyr 2015). Planning decisions are made based on the visibility (reaction) and anticipation and will update with new visibilities. Thus, the time window of visibility has a direct effect on the instability of plans. As mentioned in Stadtler, Kilger, and Meyr (2015) "This 'window of visibility' is often derived from the average production lead-time of this industry and thus depends on the WIP level. If actual production lead-time is below the average, forecast accuracy will be high. If production lead-time is above the average, forecast accuracy will be low as sales do not get an accurate demand signal outside the window of visibility. This increases the risk that purchasing will procure the wrong materials, production will start the wrong production orders, WIP levels increase even further".

The outputs of DM are: (i) Production Targets that are used to develop plans for production sites and (ii) Available To Promise (ATP) that used for demand fulfilment. The Production Targets aim to balance the demand with what is in production. It consists of product categories, production sites, due dates, and quantities. The next subsection describes ATP.

3.6.4 Available To Promise

In most of today's APS, available and not yet reserved stock, as well as the future supply of goods provided by the Master Planning Processes, are called Available to Promise (ATP) (Stadtler, Kilger, and Meyr 2015). In the semiconductor case, ATP are generated weekly and is a core aspect of the planning function. ATP represents all current and future supplies that are available to fulfill demands. Thus, ATP is the main supply-side input of order management. ATP is consumed by different modules within order management by pegging the ATP to open orders or demands. Pegging does not mean that a segment of ATP is permanently reserved for that demand or order, ATP is rather a method which allows the promising tool within APS to calculate supposedly feasible delivery dates for an order (Stadtler, Kilger, and Meyr 2015).

ATP is typically organized in the dimensions of products, time, and location of requested demands. Product dimensions in ATP provide aggregation levels according to the defined decoupling points. This different aspect of ATP is also called ATP granularity. For instance, in the case study, the ATP granularity consists of finished products (MA), products in the first production step of FE (DREP), and planned capacity (FPOS).

ATP is maintained in discrete-time buckets, which mostly correspond to the time bucket of master planning. ATP generates weekly for the whole planning horizon. In the planning horizon, ATP is represented in the long term as weeks, but nearer a shipment (a week before shipment), ATP is represented in the time bucket of days. Figure 3.16 on the following page represents how ATP is generated according to the planning horizon and time buckets. When supply is on DC-N or MA level (Finished product dimensions) and it is close to the current day the granularity of ATP shifts from weeks to days as shown in Figure 3.16.

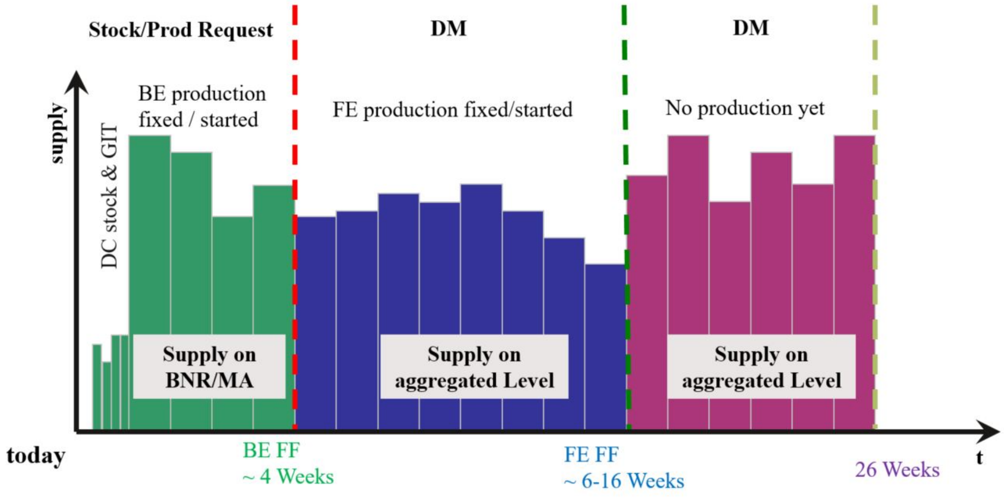


Figure 3.16: The Schematic of ATP in the planning horizon.

The finished product dimension (DC-N or MA) will be located in DC stock (SKU), logistics system (GIT), or the production start date is fixed, will all be denoted as MA since they are categorized as finished product types. Figure 3.17 on the facing page represents a schematic of ATP in two consecutive planning horizons. Current Week (CW) represents the week number in a year calendar with the planning horizon divided into 26 weeks. The figure consists of a portion of the planning horizon (four weeks) for two consecutive weeks. The DM uses this data to calculate the ATP for the whole 26 weeks. Thus, Figure 3.17 on the next page shows the output of the DM run at $CW = i$ and $CW = i + 1$. The Green columns are the ATP in the finished product granularity which divides into three types. The yellow shows the aggregated granularity according to the FE production step (DREP). It infers that the calculated ATP at week i for week $i+n$ ($Q_{i,i+n}$) should be equal to the calculated ATP at week $i+1$ for week $i+n$ ($Q_{i+1,i+n}$), otherwise the ATP becomes unstable.

3.7 Order Management System

Although there is no universal definition for Order Management (OM), however, its main purpose can be stated as providing reliable delivery to customers. In today's competitive markets, reliable and fast promises increase customer satisfaction and market share.

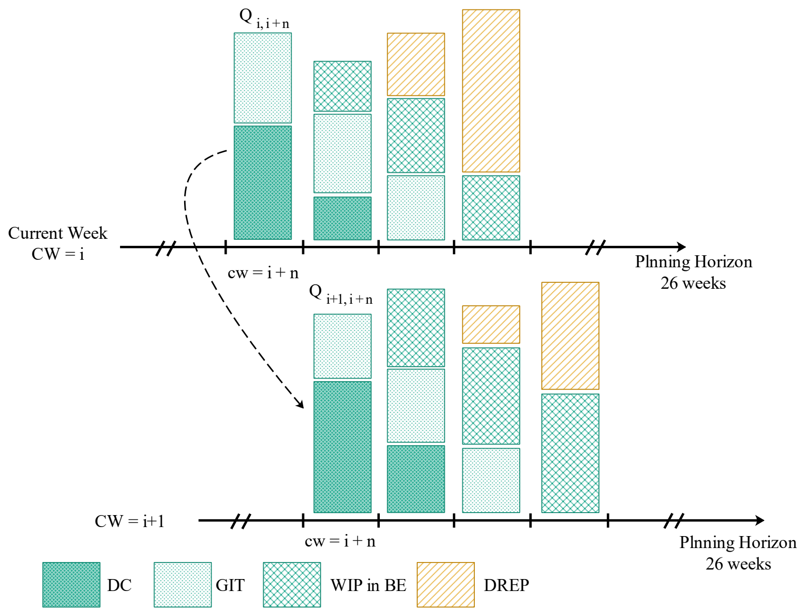


Figure 3.17: Example of ATP Granularity in Planning Horizon.

Stadtler, Kilger, and Meyr (2015) provides reasons for the complexity of OM, such as (i) increased number of products, (ii) product configuration during ordering, (iii) shortening product life cycle, (iv) flexible pricing policies, and (v) demand fluctuation. All these reasons exist in the semiconductor industry and in the case study.

OM in the case study aims to set the delivery date as close as possible to the costumer's requested date, with the aim of maximizing profit and supporting production efficiency. APS planning capabilities deal with complex OM to improve delivery dates and to provide quotations (Stadtler, Kilger, and Meyr 2015).

The case study OM comprises of three planning modules (see Figure 3.18 on the following page): Allocation, Order Management (OM), and Demand Fulfilment (DF)). The corresponding software tools for these modules are AM-UI, SAP Sales and Distribution (SAP SD), and i2DF, respectively. In Allocation (also called Allocation Maintenance), part of ATP is reserved for groups of customers. This module handles manual maintenance of allocation to dampen fluctuations. The output of Allocation is Allocated ATP (AATP). OM provides a communication interface between customers and different internal planning modules. Customer orders and EDI forecasts are received by SAP SD. This software sends confirmation of orders to customers and provides a platform for CLMs to communicate with customers. The received orders by SAP SD are sent to Demand Planning and DF. Besides, SAP SD keeps the open orders in a database called Open Order Book (OOB). In addition, it creates performance reports. Finally, the DF module calculates the confirmation date according to the Online Order Promising requests of SAP SD. Every day, all orders in the OOB are repromised to check if the previously confirmed dates are still feasible. This daily task is called Batch Rescheduling Cycle (BRCycle).

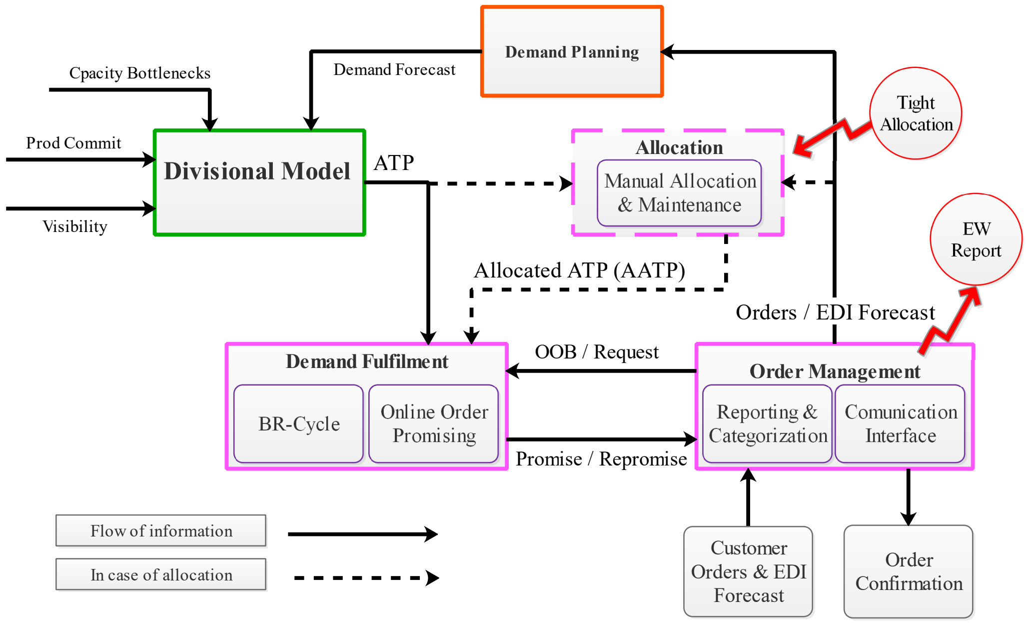


Figure 3.18: Order Management Modules and main responsibilities.

3.7.1 Online Order Promising

An order request consists of a wish date and a quantity that is sent to i2DF. i2DF calculates a promise date by following a rule-based heuristic algorithm. To do this, the engine searches the ATP schedule backwards in time, starting at the Requested Material Availability Date (RMAD). RMAD is the calculated date by considering the selected date of the customer minus the transport time and the pick and pack time. The Pick and Pack Time is the time needed for packing and preparing goods for shipment at the DC. For a backward search, there is a limitation, which defines how far the algorithm is allowed to search backwards. This limitation is the so-called Consume Earlier Fence (CEF) (see Figure 3.19 on the next page). If the search cannot find enough supply in the backward search, a forward search will be carried out in time. The algorithmic search for available products has three different promising policies. As discussed in Seitz (2012) and Alexander Seitz (2013), the policies are as follows (see Figure 3.19):

- **ALL_ON_TIME**: When the Promising Policy ALL_ON_TIME is active, the i2DF will only promise an order, if the whole requested quantity can be shipped on the RMAD. If on-time delivery is not possible, no promise will be generated. This promising policy is only used in BR-Cycle.

- **ALL**: With the Promising Policy ALL, splitting of orders into several items is not allowed. The i2DF will promise the full quantity in only one shipment, even if the promised date is later than the RMAD. This promising policy is only used in BR-Cycle.

- **ASAP (As Soon As Possible)**: The ASAP Promising Policy is used in both the Online Order Promising and the BR-Cycle. ASAP allows splitting orders into several items to ship as much supply as possible and as soon as possible to the customer. The first (partial) promise/shipment will be from the RMAD.

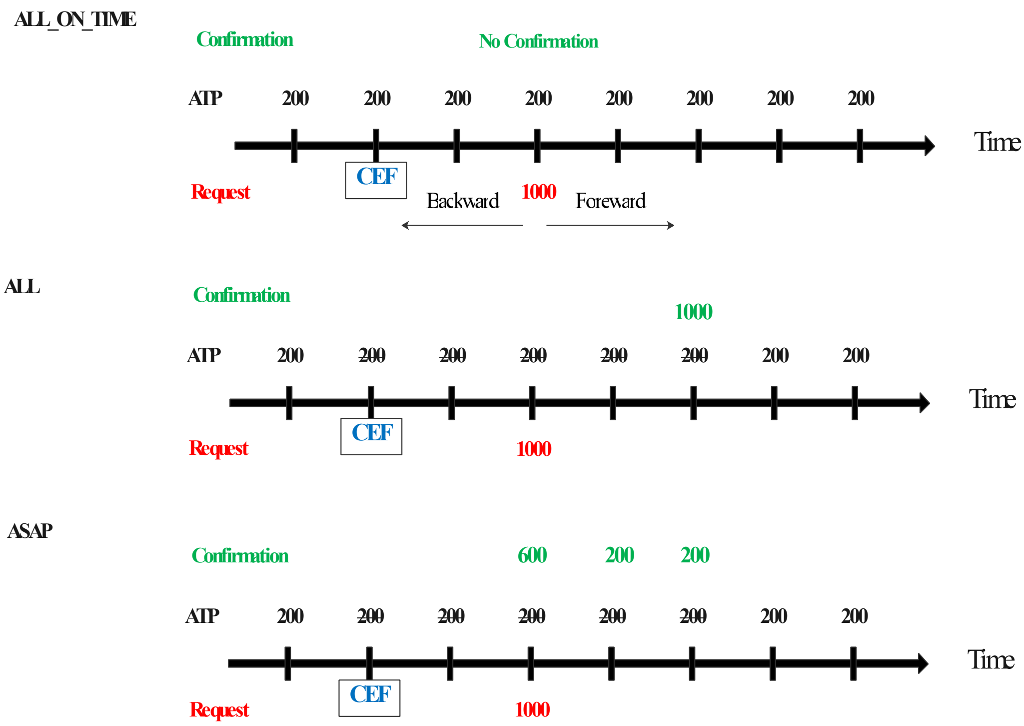


Figure 3.19: Order Promising Policies and Search Method (obtained from Seitz 2012).

Note that in all the policies a backward search is conducted first. If the order promising do not meet completely or partly, a forward search is executed. The ASAP policy indicates that each order (so-called Order Line) can split into several Schedule Lines. Thus, the notation of $CQ_o^s(t)$ represents the confirmed quantity of Schedule Line s in Order Line o in time t. The Confirmed Material Availability Date (CMAD) of each order line are also denoted by $CMAD_o^s(t)$.

3.7.2 Batch Rescheduling

Since the ATP is a dynamic online order promising system it provides a promise as soon as orders are entered. This causes the APS to reschedule all the open orders every day. To improve these online promises, every day except one day the BR-Cycle algorithm runs to reassign ATP to orders. BR-Cycle considers all the orders in a batch mode (Polwin 2018; Seitz 2012).

First, it unpegs all the ATP from OOB. Then it books all orders into ATP again. To do so, all orders are sorted with ascending CMAD and planned with the policy ALL_ON_TIME on their CMAD. If an order fails to be planned in this step, it will be set aside until all orders have their chance to be repromised using this policy. All orders left in this step are then repromised using the ALL policy. If still, orders remain to be promised after this step, they will be promised with the ASAP policy (Seitz 2012).

Second, the Cross-Confirmation Run is used to find possible CMAD improvements. It first freezes the remaining ATP (so-called Net-ATP) then it books all orders using BR-Cycle logic but it sorts the orders by their RMAD. If the CMAD improves, the promised date of Cross-Confirmation Run is used.

The Last phase in BR-Cycle only runs on a given day. It is like a Cross-Confirmation Run but considers the complete ATP. The new CMAD is denoted with $CMAD_o^s(t+\Delta t)$.

3.7.3 Early Warning

If the CMAD of a Schedule Line shifts during the BR-Cycle, a warning report will be created by SAP SD that is called a Early Warnings (EW). The aim of EW is to indicate

CMAD updates for the possible required intervention of human planners. Note that this warning system cannot capture the shift in the promised quantity and it only tracks the changes in promised date (Polwin 2018; Seitz 2012; Alexander Seitz 2013). If the new CMAD is later than the old one, it is called a Negative Early Warnings (nEWs).

$$CMAD_o^s(t) < CMAD_o^s(t + \Delta t) \tag{3.1}$$

On the other hand, if the new CMAD is sooner than the old CMAD, it is called a Positive Early Warning (pEWs).

$$CMAD_o^s(t) > CMAD_o^s(t + \Delta t) \tag{3.2}$$

Positive Early Warning (pEWs) and nEWs are just a representation of a complex interactive SC system. nEWs is more critical than pEWs since it pushes back customer orders. On average, five percentage of order lines per day have nEWs. If the schedule line (s) is split into multiple schedule lines s', s'', $\cdots$, the latest new CMAD is compared with old CMAD.

$$CMAD_o^s(t) > \max(CMAD_o^{s'}(t + \Delta)\ , CMAD_o^{s''}(t + \Delta t)\ , \cdots) \tag{3.3}$$

SAP SD compares the old and new CMAD of Schedule Lines that is created in i2DF. The result is a list of all EWs. This report contains information on the affected order lines and the old and new CMAD. The root causes cannot be deduced from Early Warnings since they only indicate that the CMAD has changed but contain no information of why the old CMAD could not be repromised.

The root causes can not be obtained from Early Warnings as they simply indicate changes in CMAD. The report does not contain any details for the reasons of the nEWs or pEWs (Polwin 2018). As Polwin (2018) indicated "The CMAD of a Schedule Line can not only change during the algorithmic repromising. Each customer has a Customer Logistics Manager who is responsible for the customers Order and can manually reassign supply to change the CMAD." As an example, Figure 3.20 on the next page depicts the frequency of EWs. Frequency here means the percentage of Order Lines that change and create EWs based on the total number of Order Lines. Note that the following three figures are used to illustrate different effects and are based on real data from the case study internal reports.

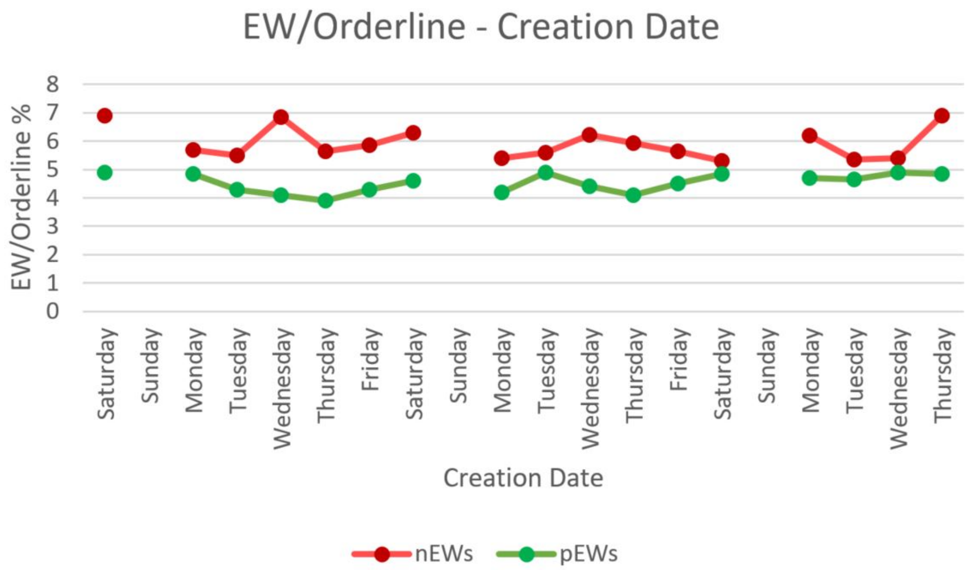


Figure 3.20: Frequency of Early Warnings per Order Lines.

Two comparison metrics are also extracted from CMAD. First, the difference between the new and old CMAD, which represents how many days the new CMAD is sooner or later than the old CMAD in time t. It is so-called CMAD Shift and denoted by $\Delta CMAD_o^s(t)$.

$$\Delta CMAD_o^s(t) = CMAD_o^s(t + \Delta t) - CMAD_o^s(t) \tag{3.4}$$

Figure 3.21 on the facing page (again presented as an example) represents the frequency of CMAD Shift per order line that is calculated in a past week.

Second, there is a difference between the current calendar date τ and the old CMAD. It shows how many days before the old CMAD an Early Warning was created. We call it Date Relative Difference $DRD_o^s(t, \tau)$ and it is denoted as follows:

$$DRD_o^s(t, \tau) = CMAD_o^s(t) - \tau \tag{3.5}$$

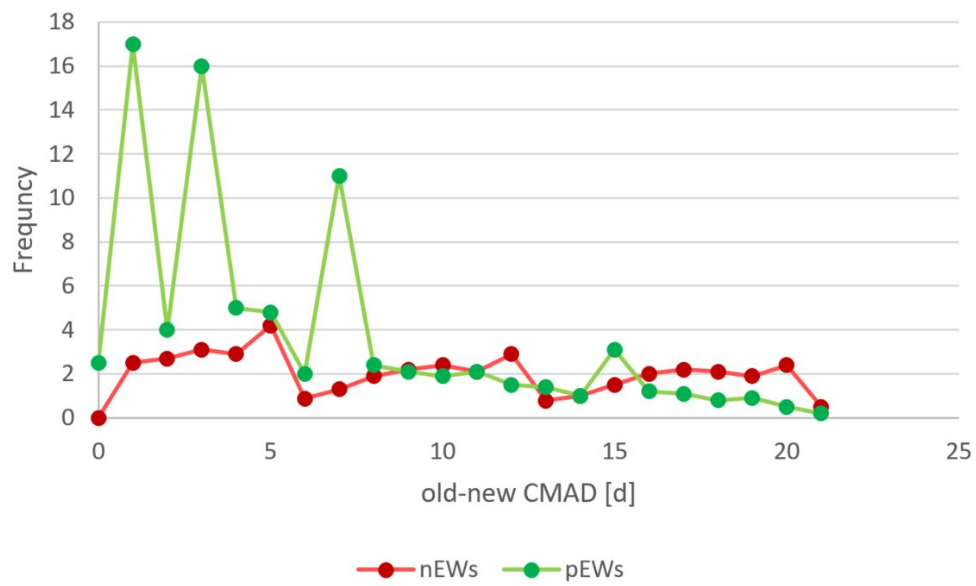


Figure 3.21: CMAD Shift for nEWs and pEWs per order Lines.

Figure 3.22, as an example, represents the frequency of Relative Date per order line that is calculated in a past week.

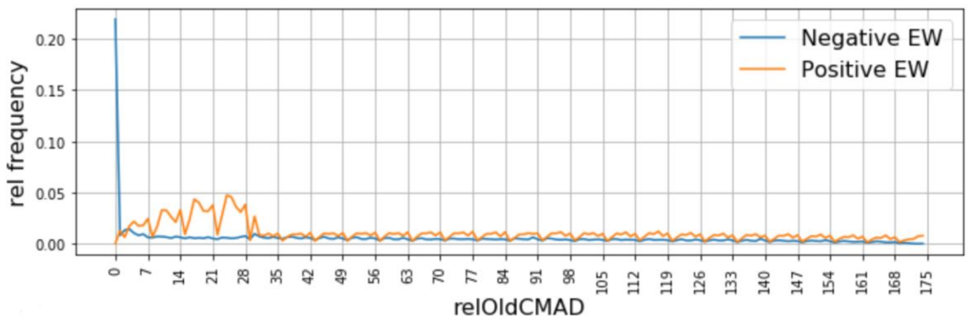


Figure 3.22: Date Relative Difference between new and old CMAD for nEWs and pEWs per order Lines (obtained from (Polwin 2018)).

It is foreseeable that nEWs behaviour should not remain the same in a rolling horizon, when the demand pattern changes. However, there are similar patterns in nEWs during the planning horizon that are related to weekends and the change of process flow within weekdays. In addition to this, it is observed that the phenomenon is very robust to perturbation and volatile planning conditions. The phenomenon stays constant in time, so it might be related to the APS system itself.

Any changes in previous confirmations are indicated by EWs and one important factor in this data is how many days before the old CMAD the EWs are created. For instance, if it creates a nEW for a confirmed order in the next 10 weeks and pushes it back to week 11, the human planner has 10 weeks to adjust this change or it may alter within the rolling rescheduling runs. The difference between a batch rescheduling run date (creation date) and old CMAD indicates this Relative Difference (RD) of each EWs.

According to the nEWs diagram of DRD (see Figure 3.22 on the preceding page), around 30% of all nEWs are created on the day of the previously confirmed CMAD ($DRD_o^s(t,\tau) = 0$). This phenomenon is called Zero Day Negative Early Warning (ZD-nEWs). There is not the same issue with pEWs.

To reveal the root causes of ZD-nEWs more than four research projects were conducted (Seitz 2012; Polwin 2018; Alexander Seitz 2013; RAMÍREZ 2015). With the support of SC experts, seven hypotheses were developed and some of them investigated. Previous research projects mostly focused on order management and DF processes and algorithms for the source of this disruption.

Background of ZD-nEWs Investigations

The early warning algorithm calculates the difference between new and old CMADs to reveal shifts in plans. It is just a comparison of new and old CMAD within the IT system but does not reveal the reasons why these EWs occur. The early warning indicates changes of previously given confirmations. It shows shifts of plans within the planning horizon for the same order lines. If EWs are created many days before the calendar date, the case company has time to react and adapt to a new situation. Contrary, if it happens a week or a day before final delivery, it means they should invest time, money and energy to solve this issue either by contacting customers or by finding products to allocate. In the worst case, if the problem were unsolved, the cost would be customer satisfaction, a cost that is hard to measure.

Depicting EWs relative to current calendar dates for different BR-Cycle runs, Polwin (2018) showed that approximately 25% of all negative early warnings are created on one day relative to a current calendar day. It means that for the orders that should arrive tomorrow or in less than a week, the case company feels that this is high and could cause customer dissatisfaction. Consecutive runs of EW reports certify that this unknown phenomenon happens regularly. The previous studies show that these ZD-nEWs are not true as the supply exists (discussed below) but with rescheduling it appears that capacity is not available. These ZD-nEWs, which are generated although supplies are available, are the so-called fake nEWs.

There are some possibilities for root causes of these fake ZD-nEWs (Unreliable nEWs) like lack of visibility and data update issues and/or system design, and/or system configuration that all should be related to the internal incompatibility of the designed systems. To deal with these challenges many researchers conducted several types of research in this semiconductor case study. Seitz (2012), in his master's thesis carried out data analysis on this issue and defined and tested a hypothesis for this phenomenon. He developed seven hypotheses relative to unreliable nEWs and evaluated these using mainly data analysis. Studying these hypotheses and test examples helped develop a good understanding of how data extraction could be carried out. The most important and relevant hypotheses are:

Hypothesis 1, CMAD vs GID: A high percentage of nEWs is caused by this hypothesis. This hypothesis means that SAP generates EWs only based on the change of CMADs and does not take the Planned Goods Issue Date (PGID, also called PGI) nor the quantity of the confirmed order line item into account. While the orders are still open until the date that goods actually leave the DC.

To examine this hypothesis, Seitz (2012) took a sample of orders in SAP to verify or falsify this hypothesis. Since they did not find useful data for this analysis. The authors selected orders with an average pick and pack time for more than one day. They assume that order line items are deleted from OOB (Open Order Book) once PGI is issued. To certify this hypothesis, they need to know that the PGI trigger caused the deletion from OOB. However, there is no document in this regard, but this issue is understood within the case study.

Within the extracted example, they show that, nEWs are generated for each day when the PGI is within the CMAD date and it happens before a Catch-Up process within i2DF is executed. Although the hypothesis seems rational and the example shows it properly, the root of delay might not belong to pick and pack time in DC. It could belong to steps before DC and when the product is in the GIT.

Another considerable hypothesis in Seitz (2012) is hypothesis 6 that was examined in Alexander Seitz (2013). The hypothesis is related to a mismatch in deletion of supply from DC stock and deletion of orders from OOB. The base of this idea is that the supply (ATP) is deleted as it is listed in stock delivery while the orders will be deleted by the difference in time. Investigations of this hypothesis falsify the hypothesis and provide a good insight in that both OOB and ATP deletion depends on PGI and Delivery Note which has no effect on ATP errors.

A comprehensive data analysis of the EW report is conducted by Polwin (2018). In his master's thesis, he interpreted the following points from data analysis that could provide insights for this research. He showed that the likelihood for EW is larger for a Schedule Line with CMAD equal to the current date. This indicates that the reasons for ZD-nEWs is more than the CMAD irregular distribution. He also indicated that most of ZD-nEWs belong to specific SPs that could be used in data interpretation and analysis of this study.

Previous studies have some limitations and challenges as follows:

- The early warnings from i2 technology do not contain additional schedule line item splits generated by the batch run – the full size of the original schedule line item will always be shown on the latest schedule line item generated during the batch run.

- In case of potential for positive early warning, this means that if there is a quantity remaining on the currently confirmed date, no EW will appear.

- In the case of positive and negative EWs for the same line, only one nEW will show.

- The realized dates after acceptance of the early warnings in SAP could not be captured in the data.

- To evaluate the outcome of a batch run, it might be more meaningful to look at the realized order confirmations before and after in SAP.
- The EWs control system compares CMAD of schedule lines and it could not capture the quantity shifts of orders.

All and all, hypothesis 1 of Alexander Seitz (2013) shows the importance of the difference between what is planned and what is realized. They showed that when the actual executed plan is later than the plan's target, which will not be reflected in the BR-Cycle, this may create fake nEWs. For dealing with issues based on hypothesis 1, they propose data analysis that filters OOB or fake nEWs to lower the work of CLMs and provide transparent nEWs for them. However, our hypothesis aims to adjust APS parameters to avoid the creation of ZD-nEWs.

3.8 Allocation Planning and Human Intervention

Although APS consists of several automatic algorithms, it can not replace human activities. In the case study, algorithms cover a vast amount of operations based on formal and simplified planning rules. To perform an end-to-end supply chain planning, human intervention is mandatory. Examples of interventions are the modifying of automatic algorithms, handling exceptions, maintenance of plans, management of escalations, modifying APS plans and other activities. In fact, APS is a decision support tool for human planners to efficiently handle the planning and scheduling of the whole supply chain. Wiers and Kok (2017) indicates that human interventions as implicit systems have a large impact on supply chain performance. In this section, we aim to introduce human interventions in order management and allocation within the semiconductor case.

Allocation of limited resources to demands is the main goal of allocation planning. Allocation of ATP to orders is usually carried out by automatic algorithm within APS. However, these normal business cycles can not meet all business situations. The semiconductor case faces situations where demand exceeds the available supply. Planners in APS can categorize products for allocation. For instance, the allocation situation arises when the delivery performance decreases.

There are two types of allocation, Product Allocation and Customer Allocation. Product Allocation is a manual distribution of available bottleneck capacities between affected products and supply plants over a defined time frame. Product Allocation results are used as targets in supply generation. During Product Allocation, the demand (orders on hand) is higher than the available capacity. Thus, it needs to be decided which product gets how much capacity. Therefore, a run rate per product has to be defined and entered in SPLUI (AP-FCST) for the generation of supply (GROSS ATP) for order confirmation.

Customer Allocation distributes the available supply between end customers (called leaf-sellers). By Customer Allocation, The semiconductor case aims to achieve given commitments to meet customer requirements. Customer Allocation is performed by the software Target Allocation Setting User Interface (AM-UI) module. Customer Allocation is the responsibility of Zonal Logistic Planner (ZLP).Note that this name is created for this thesis due to confidentiality of using industry terminology. The ZLP uses the AM-UI tool for Customer Allocation and leaf-sellers to allocate available supply to customers or a group of customers.

The process of allocation start with the determination of the general Allocation Situation. Then, according to the type of allocation, the relevant Allocation Logistic Manager (ALM) or ZLP are assigned to identify an Allocation Situation. After that, allocation maintenance is performed by the responsible planner according to the defined business logic. These steps require the gathering of data, communication, and decision making by the responsible planner. The result of allocation maintenance is the modification of quantity or lead times of orders or change in targets. The modification needs to be announced to customers. Another important terminology is Tight Supply. When the expected duration of the Allocation Situation is less than two months and only a few types of products are affected, it is named Tight Supply.

To manage an Allocation Situation different roles and responsibilities are defined for planners. Figure 3.12 on page 59 shows the abbreviations of roles according to the planning module and type of allocation. For instance, ZLP handles Customer Allocation by using AM-UI. In allocation situations, all roles need to be involved and be aware of the allocation process. The interfaces allow them to communicate through tools and reports.

An Allocation Situation could be regional or global, which is driven by events or strategic priorities. These events and strategies come from sales and marketing. Here we discuss two main roles of planners, ALM and ZLP. ALM handles Product Allocation. They decide on whether to put a product in allocation or not. They are responsible to communicate allocation announcements and modifications. They define volume splits for production plants and products within global allocation. ZLP handles Customer Allocation. They evaluate plant supply and set the supply allocation of products on and off for the regional customers where necessary. They assign the planned supply to leaf-sellers levels for specific products using AM-UI.

The responsibilities and roles of the human planner for managing an Allocation Situation are more than what listed here. However, it is clear that the planners are crucial for managing the supply chain. They are an implicit element within the supply chain system, where their performance affects the performance of the supply chain. For instance, ZLPs change the quantity and time of allocation for specific products by consuming the Gross ATP. Then the new AATP is used in DF, while the effects of this shift in AATP on the demand fulfilment have not been studied. To improve their performance, systematic and structured supports are mandatory. In this regard, providing decision support tools to ease the process of decision making in their allocation decisions is demanding.

In customer allocation, ZLP should allocate the available short supply to the current customer's orders. This allocation regularly takes time and effort since they should allocate quantities for several weeks between customers. Human allocation usually has some drawbacks such as bias decision between customers, selection of simplified answers, not selecting the optimal solution according to a criteria, and the possibility of mistakes. Besides, it is a time-consuming task for ZLPs which are human capital for the company. We aim to provide a decision support tool for allocation planning to not only ease the allocation planning for ZLPs but also to improve the performance of allocations. The decision support tool is based on mathematical optimization and is called the Regional Customer Allocation Support Tool (ReCAST).

3.9 Conclusion

In this chapter, the supply chain planning system and the implemented APS of the case study are reviewed. The motivation of this chapter was to introduce the case-specific APS, discuss challenges with details, and indicate the complexity of the planning system. The objectives of this thesis are to reveal the root causes of identified nervousness in APS and improve APS through digitalization of allocation planning. To obtain these goals, the following three Chapters 4, 5, and 6 discuss the used approaches and obtained results.

APS has different dimensions, hierarchy, and planning principles according to production requirements, which runs through the interactions of IT, software, physical, and human systems. To understand these complex systems, communication with stakeholders, the proposal of questions and solutions, we require a solid modeling approach and common language. To solve these obstacles and provide a shared understanding, we used Model-Based System Engineering (MBSE). We used three MBSE languages within this study which are discussed in Chapter 4.

Based on the results of Chapters 3 and 4 we proposed research questions regarding Early Warning and revealing the root causes of demand fulfilment's nervousness. Besides, the outputs of MBSE are used as a conceptual model for quantitative analysis with simulation and data analysis. Moreover, the results of these two chapters reveal the importance of human intervention in APS. With the focus on demand fulfilment, we identified the customer allocation planning which is handled manually. To solve these manual interventions of human planners, we propose a decision support tool based on mathematical optimization discussed in Chapter 6 and 7.

Chapter 4

Model-Based System Engineering

4.1 Introduction

Over the past two decades, systems thinking has come to economic systems, socio-technical systems (man-made systems), and engineering systems. Within all these domains, concept modeling has become the foundation of system definition, design, analysis, and synthesis. As global systems become wider and more complex, Model-Based Systems Engineering (MBSE) has emerged as an essential tool for learning from real systems (Foster 2005; Vespignani 2012).

According to the International Council of Systems Engineering (INCOSE) (International Council on Systems Engineering Website," n.d.), MBSE is "The formalized application of modeling to support system requirements, design, analysis, verification and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phase" (Walden et al. 2015). From this definition of MBSE it can be seen that, while MBSE has been shown to be beneficial to support system design and analysis (Huldt and Stenius 2019), here the objective is to introduce MBSE as a proper proactive approach towards managing disruptions (internally and externally), revealing the causalities of disruptions, and analysing and synthesising the system under study. To reach this goal, we examined three different MBSE methodologies (Systems Modeling Language (SysML), Business Process Model and Notation (BPMN) and Web Ontology Language (OWL)) for the evaluation of sources of disruption in the domain of a supply chain planning system. We believe MBSE can shed light on the sources of disruption in the short-term while helping to further develop more resilient production and supply chain systems in the long term.

MBSE elucidates system paradigms through which a formalized, centric, single source of truth as a core form of description for configuration management can be posited (Madni and Sievers 2018). MBSE is not just a set of tools and languages, rather a systematic exposition of system description. Although system modeling with MBSE is still in its initial stage of maturation, it has been used by many real applications in large-scale and complex systems. MBSE has been applied in many domains of system science such as system architecture, validation and verification (V&V), quantitative and simulation analysis, and improvement through the whole life cycle of systems (Estefan et al. 2007; Huldt and Stenius 2019).

More stable systems are considered as resilient systems. However, any man-made system like production, supply chains, finance, etc. is prone to disruption. Numerous approaches for preventing disruptions have been proposed throughout the literature pertaining to different scopes of operation management and enterprise/production systems. Although these vast domains of expertise assist in guiding towards more stable and efficient artificial systems; as Foster (2011) mentioned, in economic systems, the "decision-maker is presumed to be faced by a measurable set of opportunities and constraints that are amenable to the discovery of a stable equilibrium solution" which within the complex context of dynamic systems is hard to achieve. Encountering with disruption through the life time of a system is inevitable, but dealing efficiently with disruption and risks can be better dealt with in the long term using a complex systems perspective (Foster 2011).

Disruptions are events and occurrences of a system moving from an equilibrium stable state to an unexpected state (Wu, Blackhurst, and O'Grady 2007; Pavlov et al. 2017; Ivanov, Sokolov, and Dolgui 2014). The use of MBSE for facilitating disruption management could be: (i) reactive which supports system engineers to mitigate the consequences of occurred disruption in the best possible way (Yildiz et al. 2016) or, (ii) proactive by delineating efficient system and adopting proper systems thinking to develop a more resilient system against disruptions (MacDonald et al. 2018; Ivanov, Sokolov, and Dolgui 2014; Pavlov et al. 2017).

To the best of our knowledge, the role of MBSE and its tools for investigating and managing disruption to find the root causes of disruptions has rarely been studied. As a result, the contributions of this chapter of the thesis are to:

- Introduce and extend the scope of MBSE for enterprise system analysis and synthesis.
- Present MBSE use in the context of disruption management.
- Evaluate and compare MBSE tools for system resiliency, integration, and quantitative analysis.
- Examine MBSE tools for disruption analysis in a case study of semiconductor manufacturing supply chain planning.

Comparing the selected MBSE methods and disruption analysis in a supply chain system identifies the lack of use of MBSE as a systematic approach for analyzing SC disruption, specifically in a complex planning system. Thus, the motivation for this chapter are summarized as follows: (i) Use of MBSE as the sole source of truth to improve the understanding of the complex SC planning system by developing a common language between stakeholders; (ii) Present a MBSE pathway and framework for investigating complex problems within SC systems. The pathway demonstrates the systematic approach we used for dealing with disruptions and improving the SC system; (iii) Indication of the benefits and challenges regarding the use of MBSE in disruption management of SC.

The remaining part of this chapter is organized as follows. To provide an example of the use of MBSE for disruption management, in Section 4.2 on the following page, we detail part of the planning system that we investigate systematically by MBSE. In the following section, which discusses the general structure of a supply chain planning system in addition to highlighting the demand fulfilment part. Section 4.3 on page 94 illustrates the benefits of the use of MBSE as referential and methodological system modeling for disruption management. The used pathway of MBSE languages regarding how we applied MBSE models within the course of this study is discussed in Subsection 4.3.1 on page 95. After discussing the pathway, the three applied languages are discussed in further details by introducing the developed models. Based on these discussions on the developed models, in Section 4.4 on page 110 we discuss the use of three MBSE modeling methodologies and compare their advantages/disadvantages and provide managerial perspectives towards the use of MBSE for system modeling and disruption management. Finally, in Section 4.5 on page 115 we conclude the study according to use of MBSE modeling and analysis.

4.2 The case study challenges

As discussed, the objectives are to improve the order management by finding the root causes of unexpected errors within the control system. As depicted in Figure 3.1 on page 40, an SC planning system is required to support a global supply chain where is paramount for enabling the most efficient use of capital equipment, which is extremely expensive. In order to manage this SC, a complex socio-technical system consisting of several subsystems is required that can be divided into:

- Production or physical system which are sets of machines and logistics.
- Planning system to maintain plans for the SC system, which differs from the IT system.
- Expertise system or human planners whose expertise are required for tuning the SC, using their cognitive insights.
- IT system which is a set of software modules and databases that interact with humans and integrate with different planning and optimization systems.

For understanding this complex, large-scale and interactive system, several standard modeling frameworks have been developed. The most famous model for SCM is called

SC Operation Reference (SCOR) model. Using the SCOR model, the SCP system is shown in Figure 3.4 on page 45, where the capacity is balanced with demand over a six month period within strategical planning level. In short-term planning (tactical), decisions are taken driven mainly by the Advanced Planning System (APS). APS match the Bottleneck Capacities, prioritized Operational Demand, Stock level, and availability of different granularity of products in the global supply chain (Visibility Lot Track). The main outputs of APS are production/logistics targets and Available To Promise (ATP). ATP is the current or future supply of products based on visibility of the production system or developed plans. In production management processes, productions are scheduled in different parts where visibility on the flow of materials in the global production processes is ensured. Finally, order management contains the processes of receiving orders, ensuring contact with customers, and providing promises based on the ATP.

Acting at the core, APS merges different software systems, expert's decisions, strategic regulations, production feedback, and other relevant modules, each coming from one of the four SC subsystems. Within APS, the planning processes that satisfy the customer demands are called order management (which consists of allocation, demand fulfilment, and order management). The main functions of order management include: matching the available supply with dynamic demands, promising reliable delivery dates to customers, dealing with changes of customer orders, providing data regarding future demands, negotiation with customers, allocation of shortage supplies to customers, etc. (Seitz and Grunow 2017; Stadtler, Kilger, and Meyr 2015).

Uncertainties in the supply chain can come about for a variety of reasons. Different global suppliers, complex manufacturing processes, variant products, mass customization, production and logistics strategies, and variant order channels can each be a factor (Seitz and Grunow 2017). Satisfying uncertain demands with constantly changing supply requires repetitive interactions of the four subsystems based on defined processes and upcoming events. Each paradigm emergence out of these interactions has chains of causalities that are not linear or even clear. For example, in consecutive runs of the ATP (which happens each day), changes occur which may be due to the modification of an order, maintenance of production for other products, lack of compatibility of IT processes, late database updates, inaccurate planner intervention or many other reasons. Thus, the causalities are the result of several nonlinear interactions. This is why discovering the source of the disruption is so complex. MBSE models help us understand how all working parts of a system and its subsystems interact, how ATP operates and how to identify the source of EW.

4.2.1 Demand fulfilment and ATP instabilities

As discussed in the case study Chapter 3, the Demand Fulfilment (DF) process provides the framework for fulfilling promises on requested orders. To maintain flexibility, deal with uncertainty, and develop a stable plan, APS utilizes a central computation engine. Orders or predictions of orders are calculated during the initial stage according to a strategic plan, rendering the demand for upcoming weeks and even months as output.

Based on this planned demand and by monitoring the constraints and conditions of production sites in both the back and front end of production, the divisional model creates a picture of future supply including Work In Progress (WIP) where the initial steps of its production have not yet started. These stored WIP, and future supplies are called ATP. ATP feeds DF software modules, allowing planners to generate fast and reliable promises. The first date and quantity promised to the customer by consumption of ATP creates a baseline for orders on delivery time. Therefore, if there are any instabilities in the ATP, a ripple effect ensues which moves throughout the rolling horizontal plan. This inevitably leads to unconfirmed orders and customer dissatisfaction.

These instabilities, including inconsistencies in promised delivery dates and quantities, may also trigger events such as changes to production plans. All of these variables are taken into account during a process called order promising. Order promising runs every day to peg the orders with ATP again. If the result of the reschedule is not the same as promised, a control system is triggered and generates an alert, called Early Warning (EW), to notify software modules and planners regarding the shifts in a promised schedule. An EW alert can be positive if the new promised date is earlier than the previously confirmed date; or negative if not. This control system is a part of order management and it reads data from different subsystems within the planning suite of IT tools.

4.2.2 Challenges and MBSE objectives

Because planning occurs each day, consecutive observations of EW show that a repetitive, daily pattern of changes in promises before confirmed delivery. This causes disruption, requiring planners to manually mitigate in order to reduce the ripple effects of the disruption case by case. It also creates disruption in the planning system, from which no root cause has been identified in this complex global SC planning system. To discover the origin of this error, we used MBSE to investigate and model the system as a basis for developing hypotheses and quantitative tools. This allowed us to define research questions regarding stabilizing ATP (Mousavi, Azzouz, and Heavey 2019), which is defined in more details in the next section. It is worthy of note that this manual intervention of planners mitigates against applying Industry 4.0 concepts which is a core theme of the Productive4.0 project (Productive4.0 2017).

MBSE allowed us to model, explore, understand, share, and debug this centralized, complex, and global SC planning system. In order to pinpoint the epicenter of the disruptions and identify the ensuing errors, a model as a single source of truth needs to be developed. With this aim in mind, we examined different modeling methodologies with their advantages and disadvantages through our study which we will be discussing in detail in the next sections.

The results of this MBSE study will shed light on our project objectives of finding the source of planning errors (disruption) observed in the operation of the control system. Here, the objective is to introduce MBSE as a proper proactive approach towards managing disruptions (internally and externally), analysing and synthesising the system under study and revealing the root cause of disruption. In the next section, we discuss our pathway towards implementing these tools in the evaluation of different MBSE methodologies within the context of disruption management and instability analysis. This has enabled us to pinpoint the source of the planning errors (EWs) generated by the control system within the order management system of the case study in question. SysML, OWL, and BPMN are three approaches that we reviewed for the purposes of this study.

4.3 MBSE of system modeling and disruption analysis

How MBSE could be efficiently standardized is still a matter of debate. There are questions as to how it can be applied and which tools and methodologies are more compatible with each domain. To the best of our knowledge, MBSE can support disruption management from four perspectives.

1. Firstly, MBSE increases the resiliency of a system against risks by: (1) system structuring and knowledge sharing, (2) making rigorous decisions, and (3) propelling “as is” state and proactive exploration against risks (Joannou et al. 2019; D’Ambrosio et al. 2019; Madni and Sievers 2018).

2. Secondly, MBSE with a collection of integrated models in the form of a “sole source of truth” stored in a single repository, empowers the development of quantitative and analytical investigations of a system. For instance, we can cite the benefits of using MBSE for simulation modeling (Onggo 2009; Liston et al. 2010).

3. Thirdly, MBSE reinforces system management during a disruption by creating a language for communication which facilitates system understanding and that ensures the interconnection among the various perspectives held by stakeholders. The usefulness of MBSE integration feature for disruption management depends on how it is applied in systems where there is no defined measure to evaluate the quality of MBSE implementation projects (Madni and Sievers 2018; Albers and Zingel 2013; Albers and Lohmeyer 2012).

4. The final advantage of MBSE for dealing with system disruptions is in the realm of dynamicity and technological advancement of systems. MBSE with V&V features and visual representation could ease the process of change in systems. When a

new technology or change is about to be adopted by the system, the use of MBSE as a road map can facilitate the design of more resilient systems against any foreseen disruptions caused by stakeholders. V&V could verify any negative consequences caused by changes and alleviate them through a shared understanding of the system. (Ramos, Ferreira, and Barceló 2011; Madni and Purohit 2019).

The use of MBSE for disruption management has opened new avenues of research. In this section, we focus on the role of MBSE in defining elements and causalities of disruptions within the studied system by comparing the benefits of using different MBSE tools. With this aim in mind, we first outline the advantages of MBSE in disruption management, then go over the pathway of using MBSE tools within the project. Finally, we discuss the results of obtained tools individually.

4.3.1 MBSE case pathway

To formulate hypotheses and eventually propose a proper line of action, it is important to understand the SC system in question. This requires a certain level of abstraction. This abstraction as a conceptual model of a system defines the method and approach towards finding the root of the error within the complex planning system. During our investigation within the identified problem, MBSE tools were shown to provide a shared understanding with three main advantages: methodological features, and gaining an understanding of the system with stakeholders.

1. First, MBSE has two types of applications: (1) for system design and architecture and (2) for external analysis of a system, which is mostly discussed here in relation to simulation modeling and mathematical programming. There is a need within manufacturing and SC systems to share and crystallize knowledge. Unfortunately, digital and referential models are lacking. It is possible, however, to apply methodological modeling on a case by case basis as we move toward a more comprehensive referential model for development projects. An example of this can be seen in our case study. While there are different resources available as

system knowledge documentation, the lack of a model in a single repository in an interconnected and maintainable model is obvious. Confusing updates, different perspectives of the same topic from different stakeholders, and loss of knowledge are the main issues experienced during investigation of material within the case.

2. Second, during the course of this system investigation, several challenges emerged for the gathering and sharing of knowledge. Firstly, there was a lack of a standard language to describe interactions. Secondly, different terminologies were used in the description of the system. A case in point is that while all the learning materials in our study were available, they were developed and designed by different stakeholders. Thus, a model is required to be not only a visual representation, but an efficient method of transferring knowledge to users as well (Hettinger et al. 2015). Through the use of MBSE tools, we will be able to develop a common language and visualize the shared understanding for better communication.

3. Third, the MBSE models we developed reveal the causalities and interactions of a system's elements. Causality in a complex system is pervasive in analysis (Wagner 1999). The key to finding the internal or external root causes of a disruption is being able to define reciprocal causalities in the form of linear causes and then combining them to form a system. This allows us to analyze nonlinear causality in order to find the internal or external root causes of the disruption. Understanding the causalities of complex systems without visualization and a structured standard is very difficult to achieve. The use of MBSE allows visualization of processes, requirements, interconnections, and integration, and reveals the causes between elements.

In the project Productive 4.0, our aim is to improve the order management of a complex semiconductor SC. The case was further complicated by the fact that the stakeholders studying the system did not have the same understanding of the supply chain in question. The varying MBSE methodologies play the role of liaison between stakeholders for questions of research, definition, and communication. The pathway presented

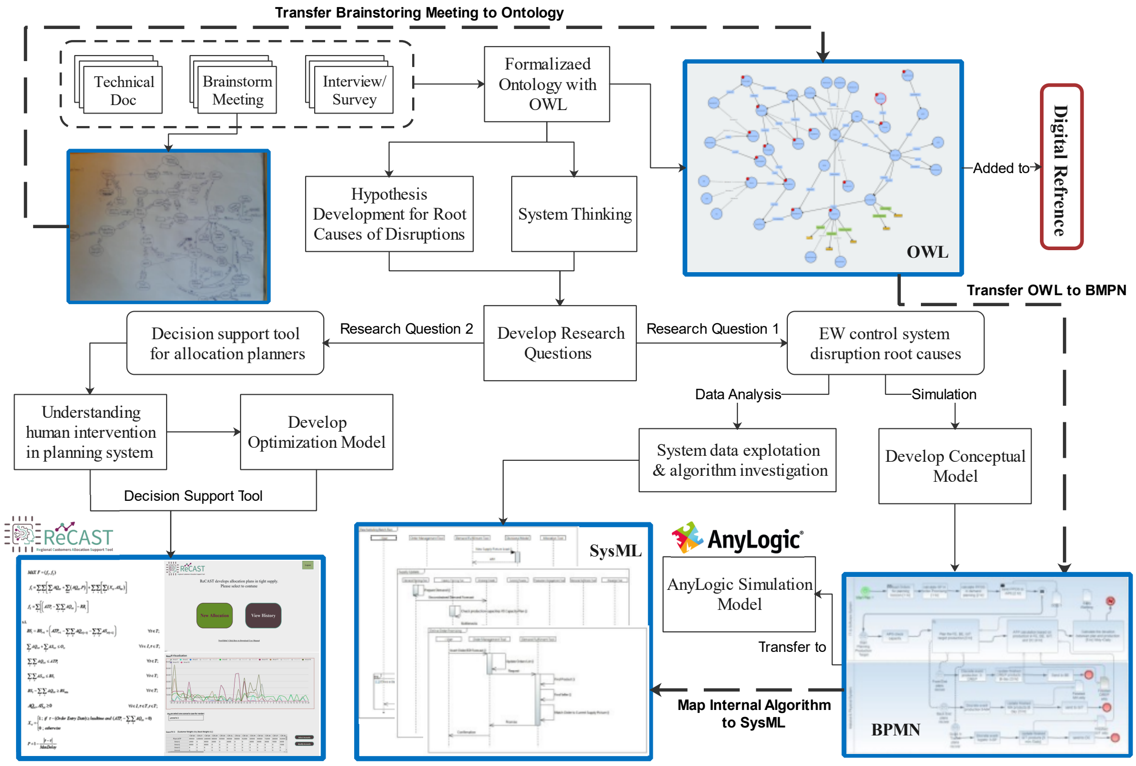


Figure 4.1: Pathway for implementing and using MBSE approaches for system understanding, communication with stakeholder, define research question, conceptual modeling, decision tool development and depth modules investigation.

in Figure 4.1 illustrates the role of the three different MBSE methodologies within this research.

To shed light on the selection of tools for the understanding of systems disruption and analysis, we investigated three different MBSE methodologies with their relevant methods (OWL, BPMN and SysML), with their application to the case study described in the following subsections. However, first we provide an overview of how we used these methods. Figure 4.1 shows that we initially participated in the Digital Reference's group meeting of Productive 4.0 (Work Package 7 (WP7)), which used OWL as its MBSE methodology for analysing the planning system and relevant disruptions. Technical documents, brainstorming meetings (see Appendix A.1 on page 233), and interviews with experts supported us to understand the planning system, define system disruption, and discuss our research questions. With the aid of two supply chain experts and five other researchers in other domains of the system, we participated in modeling a digital reference of the planning system. The hand-drawn diagram derived from the initial brainstorm meetings was then transferred to an ontology of OWL and using Protégé (Protégé 2020). This indicates separable classes (subjects) of the planning system linked by the property of another class or object.

The first draft of the model was used as a baseline in subsequent meetings within system thinking and hypothesis development. These models formed a digital reference (Ehm, Ramzy, et al. 2019) allowing us to communicate our research questions and provide a common language between the stakeholders of the Corporate Supply Chain Innovation Department (Germany) and the Customer Logistic Management (Ireland) whom we were jointly working with. The outputs of OWL lead us to define two research questions regarding order management within the planning system. First, the EW issue (defined above) for which we wanted to find the root causes and for which we used simulation (see Figure 4.1 on the preceding page) and second, an allocation issue that human planners deal with manually for which we proposed a decision support tool based on operations research.

In the workshops, OWL was used to map out high-level classes and relations, allowing us to gather knowledge from SC experts and to identify EW using simulation, enabling us to identify root causes. To achieve conceptual models of real-world systems, a preliminary abstraction step was used.

In the first research question, once we arrived at the system definition and defined the hypothesis using OWL, a conceptual model to support simulation was built using BPMN, which allowed us to communicate with non-simulation experts, such as SC and Information Technology (IT) experts. This provided a common language for the meetings to ensure we used the correct conceptual model to build an Anylogic (Anylogic 2020) model (Mousavi, Azzouz, Heavey, and Ehm 2019b).

Using the BPMN model, we were able to verify and develop an initial set of hypotheses to experiment with to find the root cause of disruption within the SC planning system. However, we used SysML as a system of communication. The simulation model could mimic the system behaviour and prove the hypothesis, but for extracting the correct data sources we needed to know further the details of the promising algorithms and rule-based APS. To achieve this goal, SysML with different diagrams played a crucial role in allowing us to map the internal algorithms to the supporting Sequence Diagrams. These diagrams were used in online meetings (see Appendix A.1 on page 233) to plan data extraction with the lead principal supply chain manager, IT experts, allocation managers, supply chain engineers, and project managers.

In the second research question, using OWL and BPMN, we identified and investigated parts of the planning system that human interventation are required to update to maintain plans. Based on these MBSE languages we held meetings with planners, allowing us to select an allocation planning situation which requires human calculations which are intractable and possibly a source of instability. MBSE supported us to clarify the goals and limitations of this allocation planning problem. MBSE languages allowed us to understand business processes within the SC planning system, allowing us to develop this decision support tool, which we called a Regional Customer Allocation Support Tool (ReCAST).

4.3.2 Web Ontology Language

Within work package seven (WP7) of the Productive 4.0 project, the authors participated in developing the Semantic Web-based digital twin as a digital reference of a semiconductor manufacturing supply chain (Ehm, Ramzy, et al. 2019). The Digital Reference ontology designed to benefit: the automation of the sales network, assist in building standard simulation models, setting-up a planning structure, use for data preparation of deep learning, and knowledge sharing. With the aim of this ontology development as stated above, we communicated with stakeholders, improved our understanding, and explicitly shared our knowledge. It initially provided us with an easy understanding of disruption within the SC system and other challenges that we dealt with quantitatively by the means of simulation analysis and mathematical optimization.

Figure 4.2 on page 101 presents in greater detail a part of the OWL file for the ontology of a semiconductor order management system and the whole ontology of the semiconductor SC seen through an overlapped diagram. It contains classes used to create different objects based on different property values, relations and object properties. A relation can be described as a concept which connects various classes while object properties are the definition of relations equal to human language. With this small expression of the model, it could be implied that OWL is very good for system understanding and knowledge sharing. It helps us to understand the essence of the planning system's entities and their relations to better create our abstraction of the studied system that we modeled later with BPMN.

OWL as a system modeling language has a high abstraction level. Protégé is a tool, which provides web-based editing functionality, visualization, and model reusability. As presented in the screenshot of the order management part of our case study depicted in Figure 4.2 on the next page, the OWL consists of the definition of system elements and their relationships while features of each element could also be stated. In our case, the main objective is revealing the sources of disruptions in the control system generating EWs. The OWL allows visualization of the control system when the ATP-based promised order could not be satisfied in the new scheduling run, which can be resolved automatically by the software or may need to be manually resolved by a planner. The data that it uses for the calculation and instabilities analysis depends on which IT system is used, mainly the divisional model part which has its own computational algorithms, input data, and software packages. To represent this knowledge, a part of the developed OWL is presented in Figure 4.2 on the facing page.

Nervous Early Warning (nEWs) are the result of the control system that we are focusing on, which shows the orders where the promise could not be met. *nEWS*, a **[class]**, is the result of, *OrdersRescheduling* (not shown fully in Figure 4.2 on the next page), another **[class]** and *isResultOf* is an **[object property]**. *OrdersRescheduling* results in *OpenOrderBook* and its data contains *ConfirmedQuantity* and *ConfirmedDate*. *OpenOrderBook* consists of *OrderLineItems* which consume *ATP*. Allocated Available to Promise *AATP* provide the *Target Allocation*. Customer Logistic Coordinator (*CLM*) use this nEWs report to provide *Promise*. Based on this example of the OWL, one can see how it can clearly depict classes and properties for system knowledge understanding. It

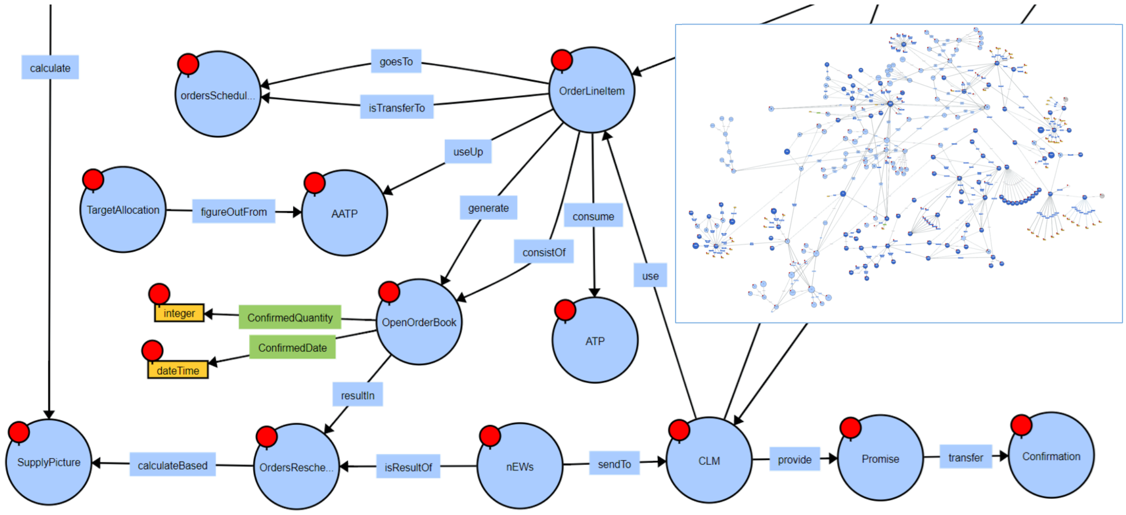


Figure 4.2: OWL of Order Management and Early Warning Control System developed by Protégé and visualized by WebVOWL. The whole digital references of the semiconductor manufacturing and supply chain presented in the box on top-right.

is case dependent and the definition of classes depends on the perspective of modelers and their objectives. In addition to facilitating communication and knowledge sharing with experts, the system ontology developed by OWL gave us insight into the search for the root causes of internal sources of disruption within the system.

To improve our system understanding, communicating with stakeholders, and participate in the semantic project, we developed a complete OWL model of planning system in the case study companyIFX with the focus on order management and ATP creation. Figure 4.3 on the following page shows the OWL model of the case planning system with its planning modules separated with colored lines. Note that these lines are not part of OWL and are drawn here to facilitate understanding of the figure.

As discussed in the above example of Figure 4.2, the models developed by the Protégé. Within this tool, sets of classes should be defined within their hierarchy. These classes relate to each other with sets of Objects properties and Data properties. The Classes and relevant properties of the developed model in Protégé are shown in Figure 4.4 on the following page.

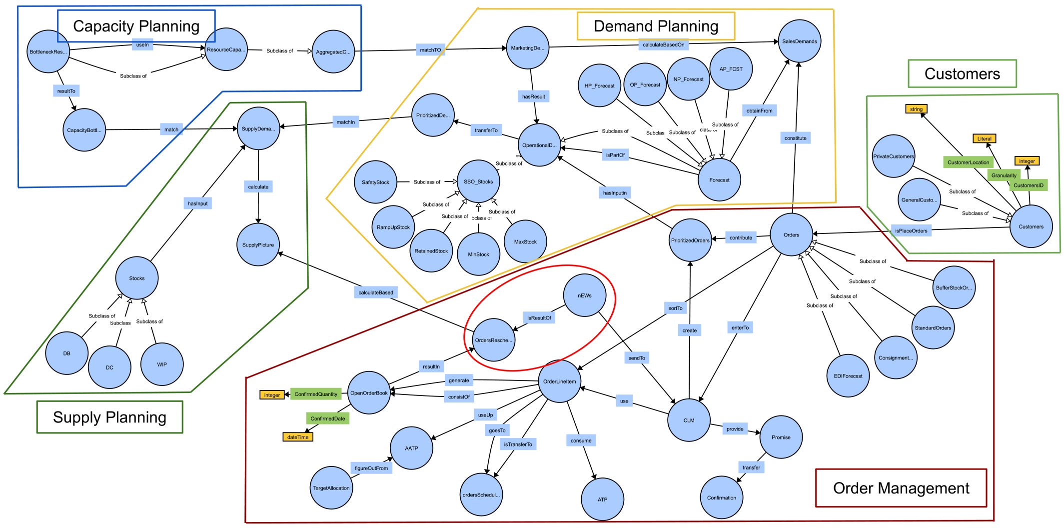


Figure 4.3: Structured and highlighted OWL model for domain of study.

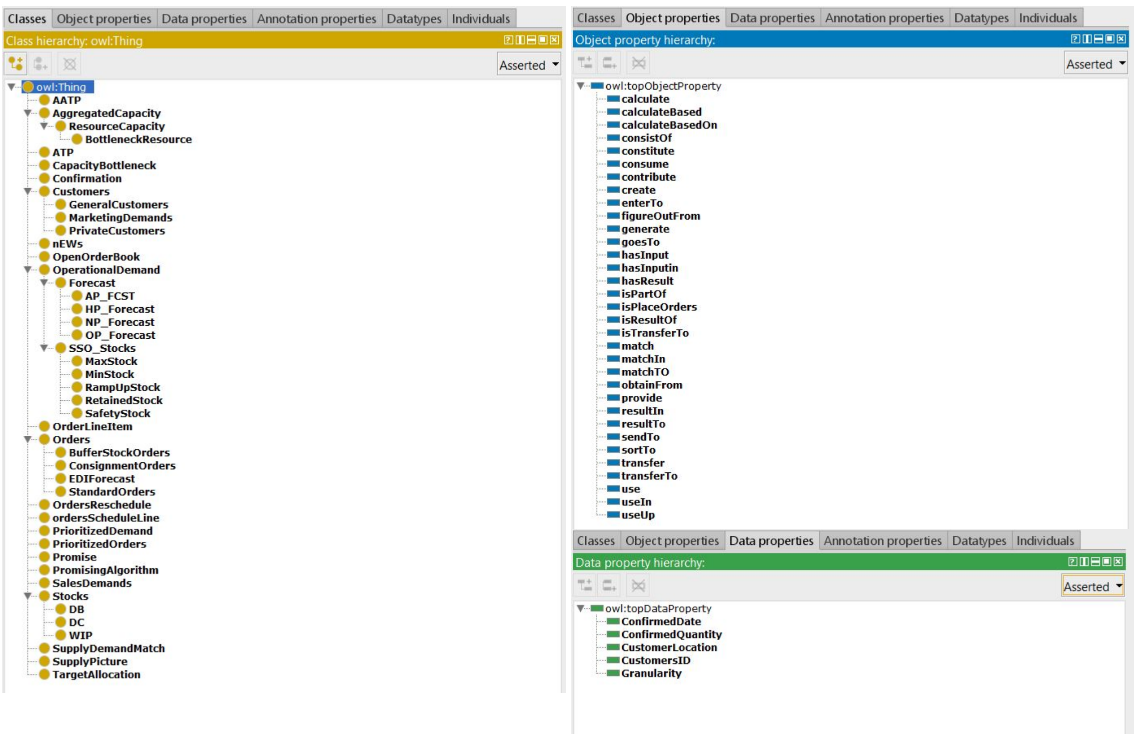


Figure 4.4: Developed properties for OWL model in Protégé.

4.3.3 Business process modeling notation

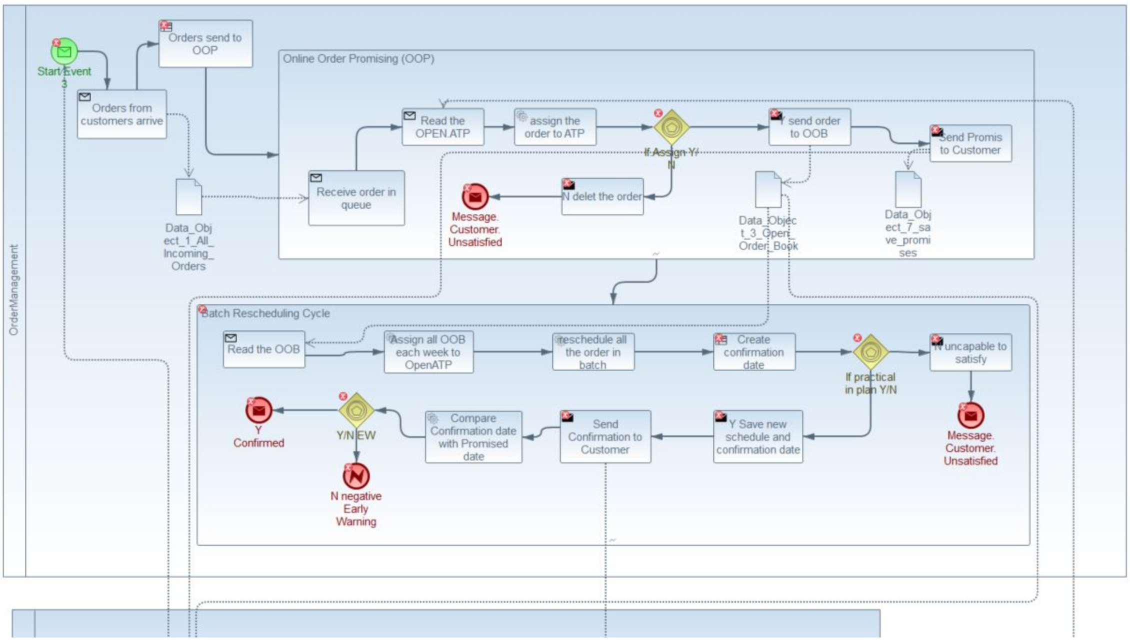
BPMN has become a de facto graphical process modeling language because of its comprehensive semantic, simplicity, and expressive knowledge sharing. Different commercial and open source tools support BPMN and it has capabilities to be extended (Zarour et al. 2019). BPMN with expressive and sufficient notations can map processes in different business levels which is in alignment with supply chain system modeling and simulation analysis. As presented in Figures 4.6 on page 106 and 4.5 on page 105, we used BPMN via an Eclipse plugin for capturing the steps of our simulation model. The notations like start (time-based or message-based) and end processes with varying types of tasks (human-based, service-based, etc.), hierarchical structure for depicting different levels or subsystems, in addition to model validation feature, make BPMN the closest MBSE approaches to discrete event simulation modeling.

To simulate the reality of the system and discover the reasons why nervous Early Warning were issued, this BPMN conceptual model was built for Order Management and related processes in planning and manufacturing system of the case study SC. These models are developed based on the knowledge gained from OWL model presented in Section 4.3.2 on page 99. These models are used within meetings with stakeholders and experts to improve our knowledge of the planning system in addition to discussing the proposed contributions and abstraction of the simulation model. Besides, it show the detailed discrete steps considered in the planning level of the simulation model.

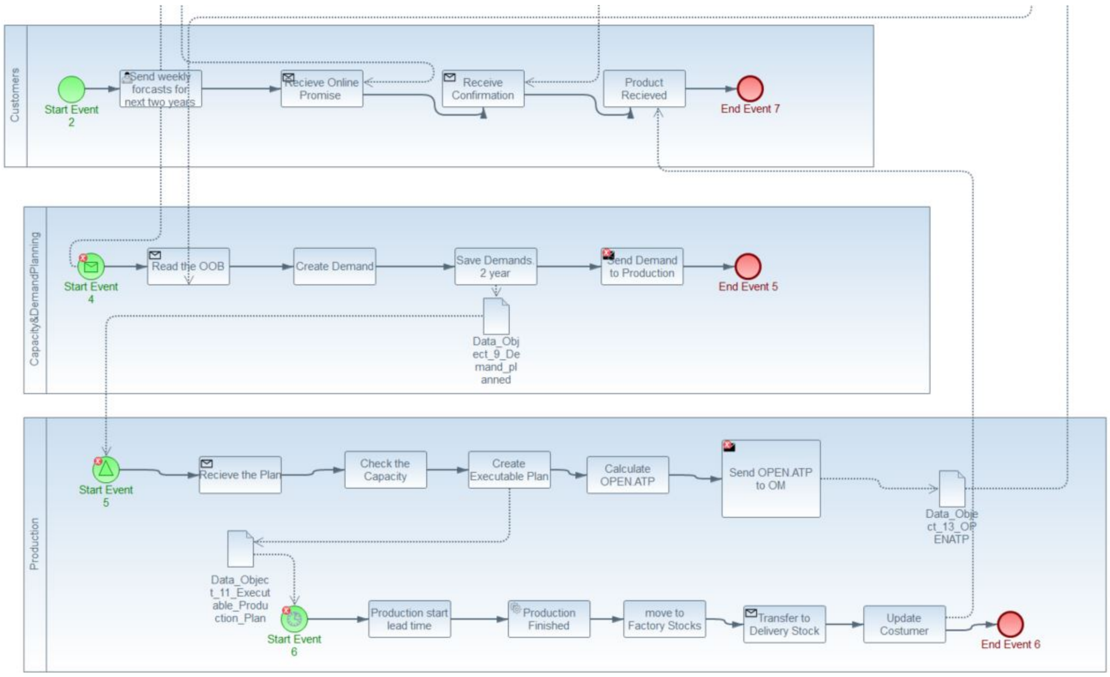
To model the supply chain and discrete event based on the research questions, we developed two BMPN models. In Figure 4.5 on page 105, a model of detailed processes within planning system comprised of related software modules is presented. This figure aims to imitate the detailed discrete steps considered in the simulation model. The considered levels are Order Management, Capacity&Demand Planning, Production (production planning), and Customers interaction with planning system. In Figure 4.6 on page 106, a higher level of the the case study planning system and manufacturing is modeled. This figure is divided into two levels of IT & Software System which represent Information flow; and Material & Physical System which represent material flow in manufacturing high level steps. Further descriptions of the figures are presented in the rest of this subsection.

In order to understand the processes within the planning system we developed the BPMN model presented in Figure 4.5 on the facing page. This model shows the main four modules which we aimed to understand deeply. First, the Order Management, which was the main focus by considering its subsystem (In top box). Order Management comprises of two sub-processes named Online Order Promising (OOP) and Batch Rescheduling Cycle (BR-Cycle). These two related to two function in SAP and i2DF software modules. OOP aims to receive orders online and provide promise for customers and BR-Cycle aim to reschedule what promised in a predefined period (See Section3.7 on page 72). Second, the Customers which we aimed to understand how customers put orders, negotiate, and receive their orders. Third, The Capacity&Demand Planning which we aimed to understand the relevant planning module in connection with order management. And finally, Production module which aims to control and plan production. In Production we require to understand as we wanted to model the interaction between planning flow and material flow.

Figure 4.6 on page 106 represents the developed conceptual model for the simulation of the EWs control system in a higher level but presents the material flow. The EWs system is defined by two separate business layers: (1) *Material & Physical System* and (2) *IT & Software System*. In the bottom layer, production lines in the *Front End*, *Back End*, and *Good in Transit* (distribution) get their plans from the *IT & Software System* (top layer) based on time-events which are mentioned inside the processes. The import and export of data from and to databases are also presented as *OOB1* (Open Order Book) which store the unsatisfied orders in a data store. This model clearly shows how demand and production plans are updated inside the system and how this information is shared between different modules related to the EWs control system. The result of ontology modeling by OWL in addition to the BPMN model provided insights into the possible sources of internal disruption within the EW system, that could benefit some quantitative analytical approaches, that in our case is, a simulation modeling approach.



(a)



(b)

Figure 4.5: BPMN model of planning modules in the case study with focus on BR-Cycle and Online order promising of Order Management.

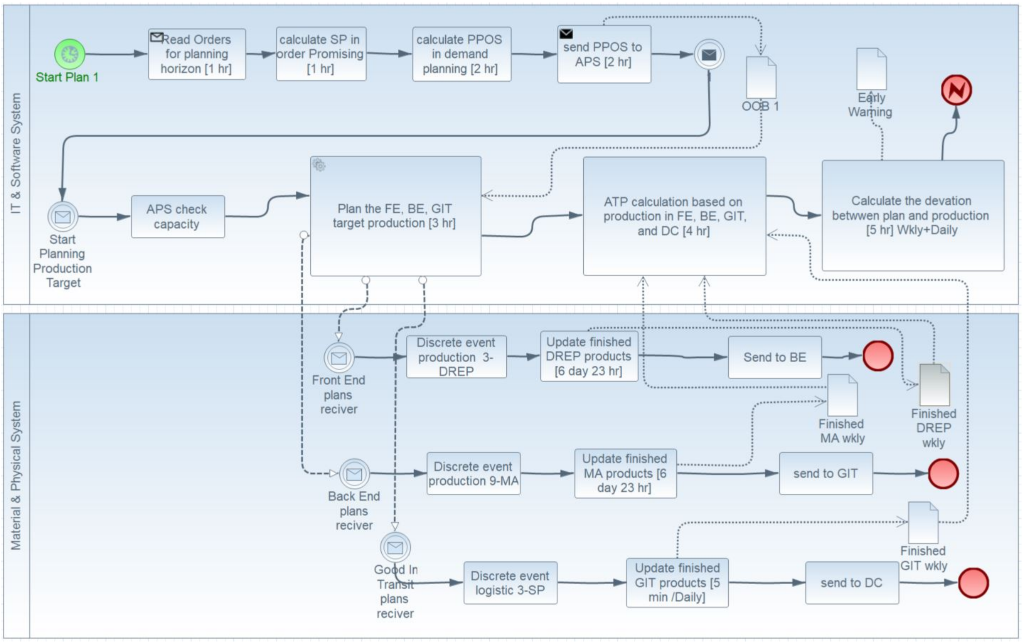


Figure 4.6: Business Process Modeling Notation with Eclipse. It consist of two layer of IT system and Physical system and each module (box) could be expanded in deeper layer. SP, PPOS, DREP, MA, and GIT are product granularity according to real system.

4.3.4 System modeling language

Compared to OWL and BPMN, SysML is able to express deeper levels of detail about a system. This could represent the subsystems of a system from different perspectives such as the sequences of processes and collaboration between individual entities, the system behaviour from start to finish of a specific activity, etc. SysML diagrams are categorized into three categories: (1) behavioural diagrams, (2) structural diagrams, and (3) requirements diagram (Liston et al. 2010). Even though the evaluation of SysML diagrams is beyond the scope of this research, we use it for requirements gathering to assist distinguishing the sources of internal causes of instability and allow better disruption management within the system. In fact, EWs are the result of mismatch between first promises and re-matching of the orders with supply. Thus, it is important to understand how both of them are generated.

Figure 4.7 on the next page presents the sequence diagram of the *Online Order Promising* which is the process responsible for providing first promises to the customer. In fact, entered orders by the customer should pass through two steps before the order is booked into the ATP to generate the promised date. The first step is product finding (*Find Product*

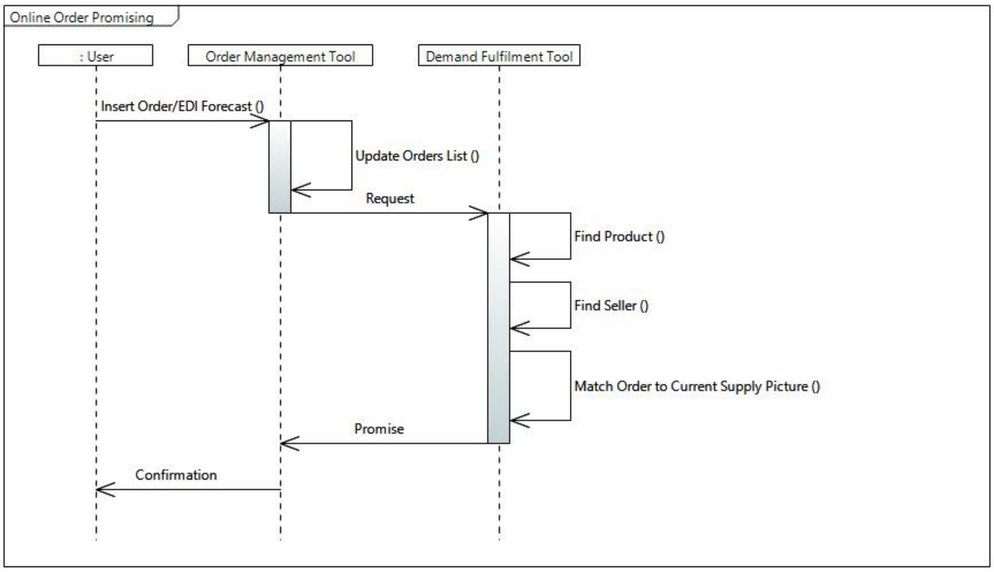


Figure 4.7: Online Order Promising Sequence Diagram.

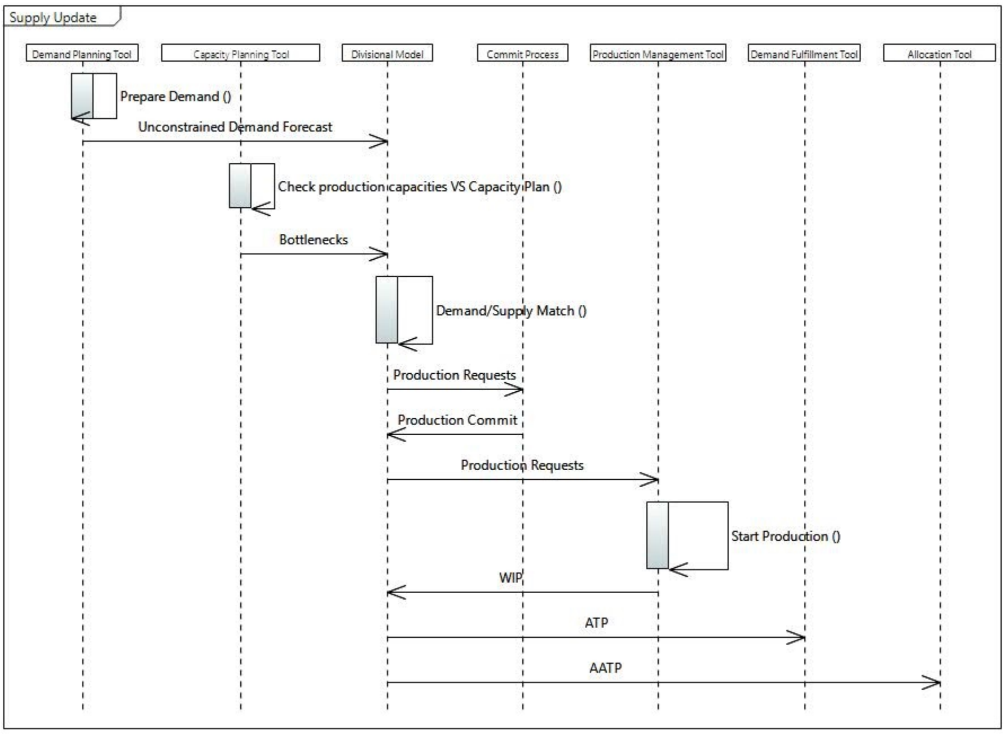


Figure 4.8: Supply Update Sequence Diagram.

()) where a list of products is generated. This list includes all finished products that are associated to the ordered sales product, as well as alternative products. In a second step, the seller of the order is determined (*Find Seller ()*). Using this information, the demand fulfilment tool promises the order (*Match Order to Current Supply Picture ()*) and sends a *Promise* to the *Order Management Tool* that confirms quantity and date to the customer.

Figure 4.8 on the preceding page illustrates how the supply picture is created, which we denote *Supply Update*. As we can see, the *Demand Planning Tool* prepares the demand plan which is aggregated and categorized based on specific rules and subject to defined priorities. Then, the calculated *Unconstrained Demand Forecast* is sent to the *Divisional Model*. At the same time, the *Capacity Planning Tool* checks production capacities against capacity plans and then *Bottlenecks* are defined and sent to the *Divisional Model* as well which matches demand and supply. The result of this process will be *Production Requests* that are sent to the *Commit Process* which needs to commit these requests. After commit, *Production Requests* are sent to the *Production Management Tool* where real production will be started (*Start Production()*). Information about Work In Progress (*WIP*) will be submitted to the *Divisional Model*, which creates the supply picture (*ATP* and *AATP*) (Allocated Available to promise (AATP) that is sent to the *Demand fulfilment Tool* and the *Allocation Tool*.

The sequence diagram presented in Figure 4.9 on the next page illustrates the *Rescheduling Batch Run* which is designed to repromise all the orders against the last ATP/AATP every day. This module certifies the feasibility of confirmed delivery dates. As presented, the *Demand fulfilment Tool* loads the new supply picture from the *Divisional Model* (Figure 4.8 on the preceding page shows more details about how supply is calculated). Then, if a product is on short supply (*Product on Allocation*), target allocations are loaded from the *Allocation Tool* and the *Allocation Planning* is performed. (Note as SysMl allows hierarchical modeling this shows a lower level model.) Orders are repromised based on the new supply picture in *Orders Repromising* and *Cross Confirmation Run* processes. Once a week, another process is performed which is the *Improvement Run*. As a result of *Batch Rescheduling*, if the orders cannot be confirmed on this date, the new promised dates and

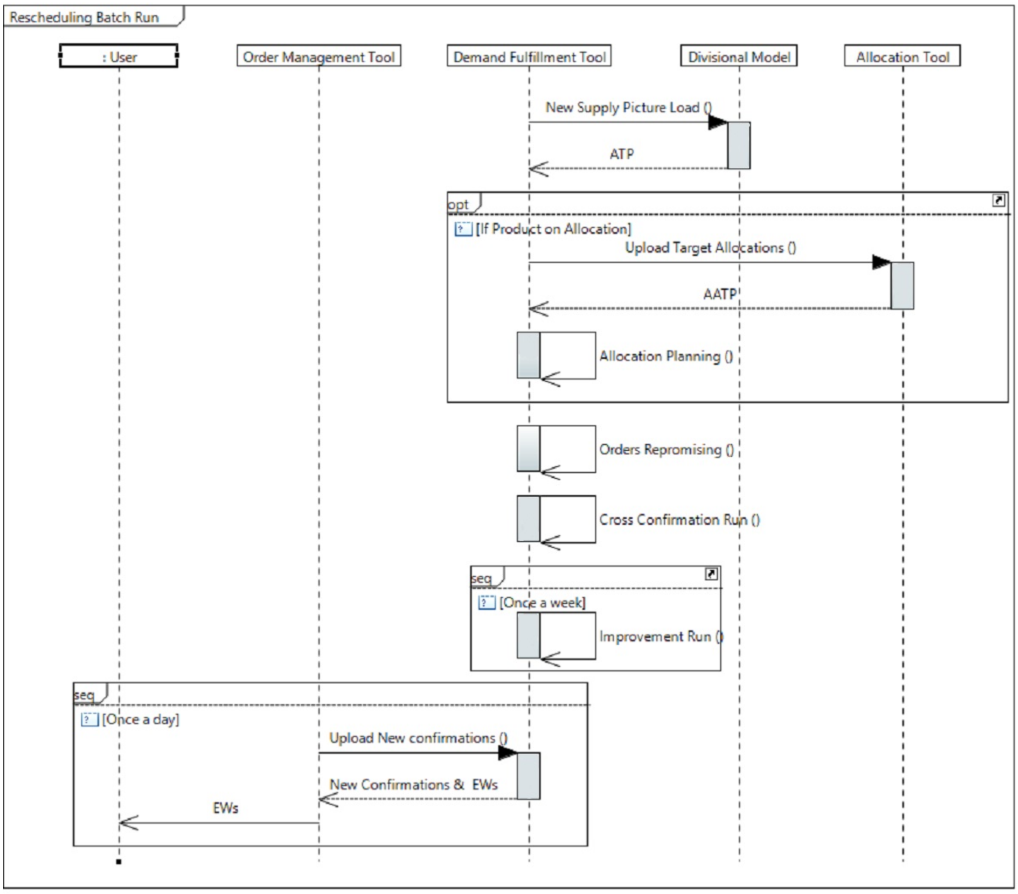


Figure 4.9: Rescheduling Batch Run Sequence Diagram.

quantities will be reflected and *EWs* will be issued by the system. In fact, the *Demand Fulfilment Tool* provides information regarding supply change to the *Order Management Tool* through *EWs* which will forward them to the customers.

Based on these three sequence diagrams that provide clear visual representation of system behaviour, understanding the modules and system entities involved in the disruption propagation in the form of EWs are easier to understand. Visually showing how the demand is inserted, how the supply is updated, and how the re-promising process is done makes the analysis of the system more practical. Theses diagrams used for defining the right databases and data extraction schedule to be used for data analysis to find root causes for EWs. The output of SysML supported us to discover the sources of disruption that causes these EWs.

4.4 Discussion

In this section, we focus on the lessons learned from the use of MBSE languages for advancing disruption analysis and improvement of supply chain planning system with the use of MBSE as an initial step. We delineate how MBSE improves disruption management and supply chain improvement by discussing the impacts of MBSE methods used for disruption analysis of the the stated problem in addition to managerial perspectives, limitations, and future work.

4.4.1 To what extent did MBSE support disruption management?

Disruption management in stabilizing a supply chain or manufacturing system involves not just a single approach or solution. In fact, it requires concrete methodological steps to deal with instabilities. The ultimate objective of disruption management is that each stakeholder in a SC or manufacturing system, who, as shown here are from different domains (IT, planning, manufacturing, logistics, and management), should all recognize and understand the system, essentially as a "single source of truth" (Madni and Sievers 2018). This will become more important in the future as systems, with the wider adaption of Industry 4.0, will become more complex and a wider set of domain experts will be involved in their design and operation.

A missing chain in disruption management that is investigated by few researchers (Wu, Blackhurst, and O'Grady 2007) is the lack of structured requirements gathering methodology for linking the phenomena of disruption with relevant quantitative methods for improved management of systems. In our case, the use of MBSE provided a common language between researchers and system experts. A back and forth communication was the means for developing MBSE models to help us to efficiently understand the system and build quantitative models.

Having a referential or methodological MBSE of planning and supply chain systems will support personnel. First, referential aspects provide a common language to better collaborate for recognizing vulnerabilities and viable criteria, and second, it supports decision makers to develop the conceptual modeling for quantitative analysis (in our example simulation and optimization methods) for analysing the current situation and proposing solutions in an efficient way.

4.4.2 To what extent did OWL affect SC disruption analysis?

In the Productive 4.0 project OWL was used as a language for developing digital references within the project. In our case study, we used OWL to develop an ontology of order management of the planning system in order to increase our business understanding, define research questions and to gather requirements from different stakeholders, i.e., SC experts, IT, manufacturing, etc. This initial knowledge gathering exercise allowed us to gather information of planning elements in the SC and propose research questions with SC experts.

Our lesson learned from ontology development using OWL as a part of semantic web technology is that it is easy to learn and convenient language with several tools designed in alignment with machine readability technologies. It has practical applications in automation and digitization of machine processes especially Cyber Physical Systems (CPS) (Yang, Cormican, and Yu 2019). It is very flexible and easy to maintain during the whole system's life cycle compared to SysML and BPMN. As a referential model for SE, OWL could convey system knowledge better than other approaches by defining the relations between defined classes. However, it could not visualize all the processes in the system. Moreover, it has no standard and defined structure of syntax and semantics but there is inconsistency in how it models the same system. The other weakness of ontology with OWL, especially in comparison to SysML, is that it could not keep the consistency and connection between different perspectives of system users and stakeholders within the life time of SE.

4.4.3 To what extent did BPMN affect SC disruption analysis?

The OWL model enabled us in revealing the main elements of disruption in planning, but it was not enough to map the conceptual model for quantitative analysis. The second MBSE approach we used in our case study is BPMN. We developed BPMN for simulation analysis of our case study. It is an easy language with visual notation to support communication with experts. Using this methodology in the conceptual model developed we could distinguish between planning and physical systems and show their interaction in addition to hierarchical structure which is demonstrated in Figure 4.6 on page 106. The developed simulation model depicts the disruption in the planning system and allowed validation due to the ability to communicate with the users of the SC system (Mousavi, Azzouz, Heavey, and Ehm 2019b).

One of the advantages of BPMN is that it is an easy tool to understand that requires a low training level for a basic stage of understanding. As noted in Onggo (2014), BPMN is easily understandable by different stakeholders as it has been originally designed with business users in mind. As well, a recent survey proposed by Harmon (2016) stated that BPMN is becoming a widely used standard for modeling business processes since the percentage of those interested in BPMN increased from 22% in 2005 to 64% in 2015. It could easily convey vast amount of information to stakeholders and experts without in-depth knowledge in MBSE. BPMN as methodological method of MBSE is very useful for understanding and developing hypothesis by quantitative methods like simulation modeling. However, there are some issues with the use of BPMN in term of simulation support. In fact, as stated in Rosing et al. (2015) operational simulation is out of the scope of BPMN and this standard was not specifically designed for simulation modeling. This may explain the limited technical capability of BPMN and BPSim to represent some important features of complex flows in a business process which limits the usefulness of these standards for complex simulation models. Onggo et al. (2018) identified resources, queues and Key Performance Indicators (KPIs) as modeling elements that are relevant to closing the gap between a BPMN diagrams and simulation. As well, it should be noted that BPMN ambiguities may also raise the risk that different tools adopting different interpretations of BPMN which may influence the automatic generation of executable simulation code (Onggo et al. 2018).

4.4.4 To what extent did SysML affect SC disruption analysis?

Although simulation analysis provided insights for the root causes of disruption, it is an abstraction of reality and it could not completely prove the hypothesis of the root source of EW within the planning system. Further investigation of the source of disruption in the IT system required a deeper investigation of software algorithms, database communication, and process flow of functionalities within the planning system. These all lead us to better see the benefits of SysML as a detailed modeling approach. The presented sequence diagram in Section 4.3.4 on page 106 is the result of a process of investigation into the planning software algorithms.

The output of SysML helped us to plan our data science investigation to validate the hypothesis of disruption sources through further extension of the initially developed simulation model. SysML has more capabilities in referential modeling of a system and the gained insight in this regard is described in the rest of this section.

SysML provides a standard and comprehensive system specification paradigm (Willard 2007). This provides a consistency in terms of model syntax and semantics, together with unambiguous graphical symbols, which can greatly improve communication. It could detail individual elements of planning and the relations hierarchically. As well, since its adoption, SysML has enabled a large recognition and an increased adoption of MBSE practices across industry.

SysML is nevertheless criticised for providing too much freedom to the modeler, allowing important information to be represented in an obscure manner in a SysML diagram (Herzog, Pandikow, and Syntell 2005). A further weakness of SysML is the associated learning effort. It has been reported that it took around 1.5 man months to train project teams to an acceptable competency in SysML and a SysML tool (Alexander et al. 2007).

4.4.5 Managerial perspective toward the use and selection of MBSE approaches for disruption management

Above, we mentioned the limitations of the three selected MBSE methods that we used. Importantly, from a managerial perspective, the different MBSE methods provided a platform for us, as researchers, to gain a very clear insight into the complex SC planning of a global supply chain. While the MBSE methods where not panacea, we felt that they were a necessary approach for the methodology we applied, which was to gain a good understanding of the system and the problems of the system before the application of quantitative tools. However, MBSE methods have several challenges that we highlight here, at a high level, to be improved to better support system modeling. These are:

- Lack of concrete quantitative performance indicators to evaluate the quality of the implemented MBSE projects.
- MBSE project implementation highly depends on the domain of application.
- The selection of standards and tools depends on complete understanding of objectives, time, broadness, and budget of the project.
- The participation of stakeholders and modelers' expertise define the achievements of MBSE implementation projects.

- The MBSE implementation is tightly connected to change culture within the organization.

As a result, to efficiently deal with these challenges, developing a proper implementation methodology and choosing the right MBSE approach require consideration of stakeholders' perspectives, definition of the domain of application, making clear the requirements, planning the change management approach, and the strategic plan for project implementation to achieve lasting success.

4.4.6 Limitations

First, the gained knowledge and insight on each modeling approach had an impact on output of the next MBSE applied method. As previously discussed, we developed OWL, BMPN, and SysML, respectively. Second, MBSE is a wide and vast area of knowledge, here by necessity we where restricted to the domain of one case study. The aim of these tools for us is to assist in revealing the source of disruption and modeling the system based on the developed hypothesis. Third, each of these approaches has several features which we did not need to cover in our case study. Fourth, measuring the efficiency of MBSE in SC disruption management is difficult. In this regard, we only evaluate the MBSE models by comparing the real data and simulation results. Finally, here we aim to evaluate MBSE in the initial stages to help understand disruptions in SC systems and to assist in the support of quantitative methods, where the MBSE methodologies were used to communicate with the different stakeholders within the system.

4.5 Conclusion

Advanced MBSE methodologies and tools have been introduced as referential and methodological SE approaches. MBSE has several benefits over document-centric SE and wider domains of application such as system architecture and improvement, change management, and quantitative analysis. Several tools and approaches have been developed and evaluated according to the requirements of SE. It has been shown that keeping the MBSE up-to-date throughout a system life cycle is harder than its initial application. Although MBSE is still maturing, the cost of SE analysis with MBSE tools is decreasing in overall costs (Madni and Purohit 2019). In addition, MBSE has undeniable benefits on improving the systems resiliency and validating its performance in possible occurrences of risks (Madni and Sievers 2018). As demonstrated in this article, the advantages of MBSE are unambiguous and they have been proved in many research projects and in real applications (Huldt and Stenius 2019).

In this work, we introduced MBSE approaches, its domains of application, and a brief investigation on current approaches and relevant tools. We selected three different MBSE methodologies and their tools for evaluating MBSE in the context of analysing disruption of a complex supply chain planning system. We examined these tools according to a real case application where the scope is analysis of disruption within this system. This case study is related to our work on debugging and improving a semiconductor SC system. OWL, BPMN, and SysML where the examined methodologies in this work. According to this case study each of these methodologies impacted on our work in different ways: with OWL which is easy to understand supporting our initial understanding of the case study, BPMN, which is again easy to understand, provided good support for simulation model development, and SysML which we found we did require to provide further understanding of the planning systems within the case study. However, selecting the best one depends mainly on the objectives and domain of application (Rashid, Anwar, and Khan 2015) which we try to shed light on in this regard.

In this chapter, we introduced the use of MBSE for managing disruptions within the SC system and revealing the root causes of nervousness and improving APS. However MBSE application for disruption analysis of DELS require more investigation by researchers. For filling gaps, we suggest further case application, tools evaluation, and categorization of application domains. Furthermore, developing ontology for transferring MBSE to quantitative analysis could add value to disruption management studies. Investigating the reasoning capabilities of MBSE tools in analysing developed model is another open avenue that need to be considered by researchers.

Chapter 5

Data-Driven Simulation Modeling of Advanced Planning System

5.1 Introduction

With the growing globalization of production systems, the complexity of supply chains as socio-technical systems is escalating, which, consequently, increases the importance of strong planning systems. Plans are developed to structure production in end-to-end supply chains that can experience nervousness due to uncertainties which may result in unsatisfied customers.

The Supply Chain Planning (SCP) system is responsible for developing production targets taking into account stochastic and dynamic demands. To obtain a stable planning system, different approaches were developed and customized. These included an Enterprise Resource Planning (ERP) system that, to manage complex SCs, developed an additional software called an Advanced Planning System (APS) that uses hierarchical planning. These SCP systems support structured production planning in complex Supply Chains (SCs) and in our case study a semiconductor SC (Stadtler, Kilger, and Meyr 2015).

In production systems and their related supply chain, plans and schedules are controlled using socio-technical systems to gain expected output (Wiers and Kok 2017).

Plans may not be completely realized because of the uncertainty of events but the aim of a planning system is to develop a smooth and stable plan. APS evolved from ERP, as they provide the ability to optimize supply chain scheduling by determining the best schedule given the multiple constraints of equipment and material labour within a rolling horizon.

As discussed in Section 3.7 on page 72, one of the major reasons for adopting APS is to provide reliable promises to customer orders. Within the case study company, a Demand fulfilment (DF) module within APS is used to provide promises to customers in a yearly planning horizon, whether the order is committed or forecast. To provide these promises, APS needs to know the requirements of products and at what stage these products are in production, transport, or in stock. These visibilities and interactions by means of Available To Promise (ATP) and production plans (anticipations, instructions, and reactions discussed in Subsection 3.3.3 on page 48) identify the performance of the planning system according to nervousness. Any shift in ATP (ATP instability and nervousness) forces the planning system to check for possible required adjustments on promised orders. With the presence of nervousness and disruption within ATP, companies are exposed to DF risks that lead to unexpected costs and unsatisfied customers (Levy 1995). Viewed as such, SCP is a complex adaptive system operating within a complex environment.

Although the external causes of nervousness and instabilities in supply chain planning systems were previously considered in the literature, internal nervousness is only partially investigated. Internal (intrinsic) nervousness results from how the subcomponents of the planning system interact, rather than external (extrinsic) nervousness which results from how external activities (i.e., suppliers or customers) interact with the production system. In this chapter, we study internal nervousness of a supply chain planning system in the semiconductor case study using simulation analysis. We propose a novel multi-paradigm conceptual model for the SC system as a basis for developing the simulation model. Verification, data analysis, validation, and experimentation are conducted to evaluate a hypothesis related to the root causes of system disruption and its ripple effects on the studied network. This work demonstrates the use of simulation-based decision support towards better decision making and decision analysis for a planning system through increasing stability.

As discussed in Chapter 3, the case study supply chain planning consists of a complex APS. Although we indicate the observed challenges with Early Warning (EW) (Zero Day Negative Early Warnings (ZD-nEWs) in Section 3.7.3 on page 76) it is not easy to understand the root causes of this disruption. To support decision-makers to find the root causes of instability, many scenarios in the dynamic SC system have to be considered. The challenges are worse when experimenting on a running planning system, due to the effects and risks of experimenting on the real system. Methods are available for evaluating a planning system from computer sciences, statistics, and operations research, which includes simulation modeling, that can be used to imitate a real-world system and its constraints. Simulation enables the consideration of the interdependency, dynamicity, and stochasticity of a real system. Besides, with simulation analysis, experimentation could interactively be conducted (Santa-Eulalia, D'Amours, and Frayret 2012; Achter et al. 2017).

Thus, the focus of this chapter is to investigate the internal nervousness within master planning and order management. Herewith, using the simulation model, we aim to evaluate APS to test the hypotheses of the cause and effects of ATP nervousness and specifically Negative Early Warnings (nEWs) on the APS. Therefore, we introduce simulation modeling, discuss the fundamentals of multi-paradigm simulation modeling, review the problem and hypotheses, introduce the conceptual model and simulation model, verify the simulation by applying case data, conduct validation and experimentation, and finally consider APS data analysis as a further validation for the root causes of internal nervousness within APS.

5.2 Simulation Modeling

Simulation is about imitating the operations of real-world facilities or processes (systems) by using computer techniques. To simulate systems, it is required to understand sets of assumptions about how the real system works. These assumptions allow the creation of a model, which is an abstraction of the real system. If the relations within the system or questions are simple enough, it is possible to answer questions with analytical solutions (such as algebra, calculus, or probability theory). However, usually, the real system is too complex that requires the use of simulation (Law 2013).

Simulation modeling facilitates the evaluation and understanding of a real system by conducting experiments. These experiments provide a risk-free environment to find and test solutions (Borshchev 2013). Figure 5.1 indicates the process of developing risk-free solutions for a real-world problem. It shows the phase of mapping a real-world problem to a model. The process of abstracting a real-world problem to a model, a process that is less formalized and is thus viewed as art rather than science. It is obvious that any model could not represent every detail of the real-world. Thus, all models are wrong, but a model could answer specific questions or what-if scenarios. As mentioned in Borshchev (2013): "The whole modelling thing is actually about finding the way from the problem to its solution through a risk-free world where we are allowed to make mistakes, undo things, go back in time, and start all over again."

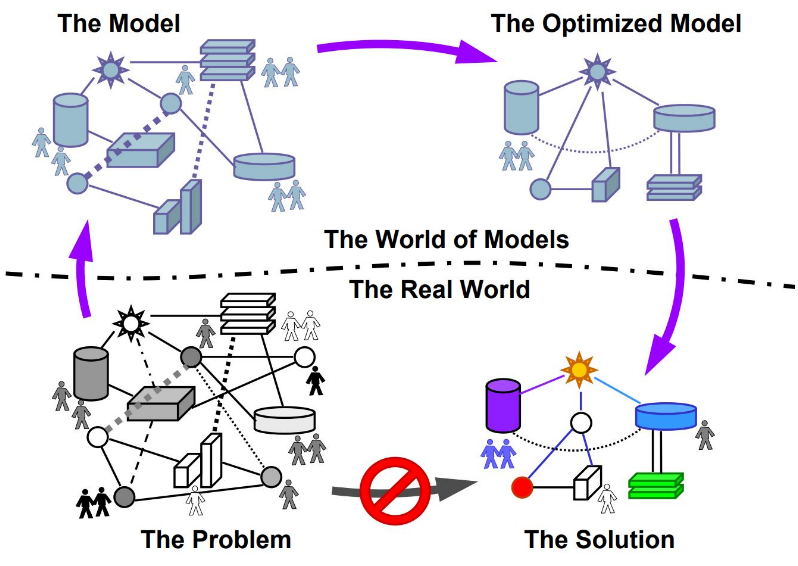


Figure 5.1: Simulation modeling, real-world, and solution implementation (obtained from Borshchev 2013).

As discussed in Law (2013) and Borshchev (2013), in modern simulation modeling, there are three main methods that each suit a particular range of abstractions. First, Discrete Event (DE) modeling which operates at a low and medium level of abstraction. A

DE modeler considers models as a process where entities pass through logical sequences of steps with limited resources. DE models are stochastic as they depend on stochastic inputs like service time. Second, Agent-Based (AB) modeling is typically used in detailed to high-levels of abstraction. AB allows a modeler to model a system where they are not sure of how the system behaves as a whole. AB technology comes from object-oriented programming, UML, and statecharts. In AB the modeler imitates individual objects that may interact with other objects and environments. Finally, System Dynamics (SD), which is primarily used for strategic modeling, captures feedback in causally closed structures. SD simulates a system in an aggregate level. SD models are mostly deterministic, but can include stochastic elements.

Choosing the right method depends on a simulation project goal, data, and nature of the system. However, it has been shown that one method could not conform to the whole problem (Borshchev 2013). Thus, all three methods could be used in one single model. In this regard, AnyLogic (by using the object-oriented platform) lets a modeler imitate a different level of abstraction with three methods in one model. Figure 5.2 on the next page shows some of the multi-method approaches. For instance, 'Processes inside agents' is when DE models are developed inside agents. Then the agents interact with each other or in one environment. In another word, the behavior of agents comes from internal DE. A good application of this multi-method approach is in the modeling of complex business processes (Borshchev 2013).

As discussed in Chapter 3, the case study is a complex supply chain planning system. We aim to identify the root causes of ZD-nEWs observed in the Order Management system. As discussed in Section 3.7.3 on page 80 there are various hypotheses that could lead to ZD-nEWs. We are particularly looking to the source of instability within ATP. Thus, our simulation model is required to capture how ATPs are generated. To model this complex system, we will use multi-method simulation of AB and DE. We discuss the multi-method in the following subsection.

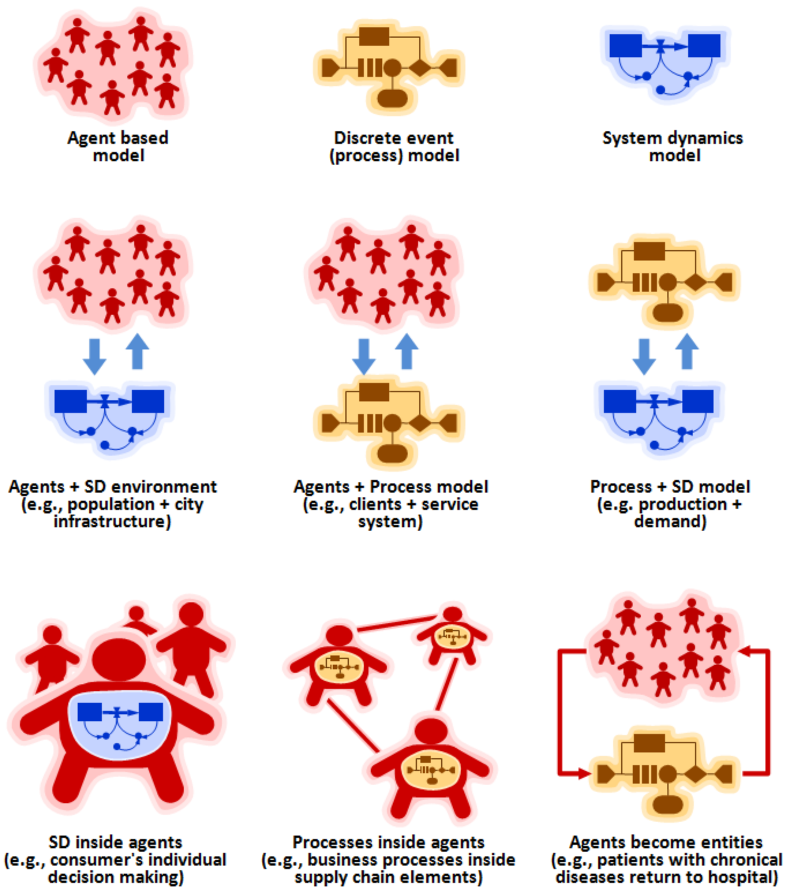


Figure 5.2: Examples of multi-methods simulation (obtained from Borshchev 2013).

5.2.1 Multi-Method DE & AB

The case study SCP system is a complex socio-technical system. The total performance of this SCP system depends on the individual performance of production systems, information systems, human interventions, procurement, and several other interactive modules and elements (Pundoor and Herrmann 2006).

Any SC system could be divided into two main flows: (1) Material Flow (MF) which is the flow of products from raw material to final products, and (2) Information Flow (IF) which models the flow of data and decisions developed by algorithms and planners that go through the system to shape the flow of material in an efficient way.

In the semiconductor manufacturing case study, MF divides into Front End (FE), Back End (BE), transit, and distribution with decoupling points. Each production site is a discrete system with a process-centric nature that could be modeled as a DE. Each separate production site like a front end in Germany can be considered as an agent. These separate agents receive production targets and deliver products. They interact with each other in an environment controlled by the IF system.

IF constitutes of the five subprocesses discussed in Chapter 3. The APS consist of separate software modules with separable functionalities. We could define the individual steps of each planning module, thus each of these elements could represent as an agent in the environment of supply chain planning.

Information systems and production systems interact with each other through visibility (from material flow to information flow) and production targets (from information flow to material flow). Timing of information updates, frequency of updates, events that trigger actions, and sequences of actions are examples of challenges regarding modeling these complex socio-technical systems.

In the developed simulation model, each production site or planning functionality is a DE. However, since they interact in a complex environment, each of them should locate inside an agent. This muti-method approach is what we used in the developed simulation model. The advantages of this approach are:

- Enables the simulation of complex planning systems.
- Allows capturing a wide range of abstraction levels within the supply chain.
- Facilitates developing and experimenting with new planning functions within the planning system.
- Enable simulating the effect of human planners (agent) interaction with the planning system.

Further discussions on the conceptual model of the case study planning system are discussed in the following subsections.

5.3 Problem Review and Hypothesis

In Chapter 3, Positive Early Warnings (pEWs) and nEWs were described. In their description, it was stated that nEW is more critical than pEW since it pushes back customer orders. On average, 5 percentage of order lines per day have nEWs.

It is foreseeable that nEWs behaviour will not remain the same during each planning time echelon within a rolling horizon. For revealing the root causes of ZD-nEWs five research projects, were described in Section 3.7.3 on page 80. In these research projects seven hypotheses were created and some of them investigated (Alexander Seitz 2013). Some of these research projects focused on order management and DF processes and the algorithms used to try and find the root cause of disruption. The hypothesis of this research is on ATP generation and master planning.

The source of nEWs maybe a difference between APS lead time (anticipations from the top model see Figure 3.8 on page 52) and real execution time (reactions from the base model). Stated differently, the APS parameter for a cycle time of the logistic duration between BE and the distribution centre may deviate in reality but not be reflected in the planning system. The case study company would require proof of the root cause of nEWs, rather than change the planning system, which ships billions of euros of products per year, a change that could adversely affect end customers and, also, planning staff within the company. Note that since there are thousands of products, lack of data transparency, and the occurrence of an error in DF (not ATP generation), it is not possible to simply compare ATPs in the rolling horizon. Moreover, there are hundreds of logistics lines.

Therefore, we aim to simulate the master planning, order management, and production sites of the case study. For this level of abstraction, we discuss in the following sections details of the simulation model that allows experimentation of this hypothesis. This hypothesis will clearly show the effects of the deviation between anticipation and reaction. In the next subsection, we elaborate the hypothesis.

5.3.1 Hypothesis of Target vs Actual Transit Time

In the case study, product in different granularity flow through FE, BE, and DC. In each production site a product's granularity transfers to the another granularity. For this global flow of material, there are internal logistics systems that are outsourced to third parties. APS as a centralized engine of the planning system has clear and important functions. APS defines decisions to improve the efficiency of material flow.

First, it matches supply and demand by considering capacity bottlenecks and aggregated demands. The result of this netting occurs in the operational level involving priorities and hierarchies. The APS develops targets and plans for material flow systems (instructions). Concurrently, it fulfills demand and provides information about the availability of products (ATP). ATP contains granularity of products and reads data from production targets, Goods in Transit (GIT), and Distribution Center (DC) stock (see Figure 3.15 on page 70). It loads this data every day at 9:30 and 12:30. Then, the BR-Cycle pegs the orders (Open Order Book (OOB)) to ATP. The BR-Cycle also runs every day and it considers priorities of orders. Requested Material Availability Date (RMAD) and Confirmed Material Availability Date (CMAD) also consider prioritization. EW is a report generated from SAP, which is a software module that runs after i2DF (BR-Cycle).

Any APS has a model of the real physical system inside. The process flow is designed based on this model and the data that is gathered from executing production and logistics. The internal model of APS is designed based on the evaluation and parameters obtained from reality (anticipations). To model the real system inside APS, we must use parameters obtained from the system. The hypothesis that is investigated here is that the real system is dynamic, but the planning system does not incorporate this uncertainty. This means that the model inside APS would not stay validated for long runs. Therefore, the APS system requires both maintenance and human planners to be involved. For example, the model of the system in the case may lead to the OOB and ATP being deleted caused by a PGI (planned goods issue date) message, or it may depend on the lead time of each logistics route. As discussed in Chapter 3, previous research about ZD-nEWs considers BR-cycle, change in order updates, or the data gathering flow within APS. None of the previous work investigated the effect of ATP instabilities on ZD-nEWs.

If we assume that the error comes from ATP instabilities and since the ZD-nEWs happens one or two days before the current date, the issue should be related to the final granularity of ATP which will relate to GIT from the BE to the DC or product in stock level (SKU). Thus, we focus on the final step of delivery which is the internal logistics between BE and DC. More specifically, in this case study we examine the anticipation from a logistics system called Target Transit Time (TTT) and the time of real transfer called Actual Transit Time (ATT). Therefore, this hypothesis examines the difference between Target vs Actual Transit Time (TTT vs ATT).

For better clarification, see Figure 5.3. In this figure, the schema of two consecutive planning horizon ($t = i$ and $t = i + 1$) is depicted. It relates to ATP calculation and the values of products considered as Stock, GIT and BE. The available ATP quantity calculated at $t = i$ for week number $i + n$ is the summation of three different values ($Qi, i + n$ = Stock + GIT + BE). Based on the available data and the internal model, APS anticipate the available quantity for week number $i + n$ that would be calculated at $t = i + 1$. This called Target. What is planned by APS is depicted in Red and titled as Target. However, as time passes, at $t = i + 1$ what is executed deviates from what is calculated as Target.

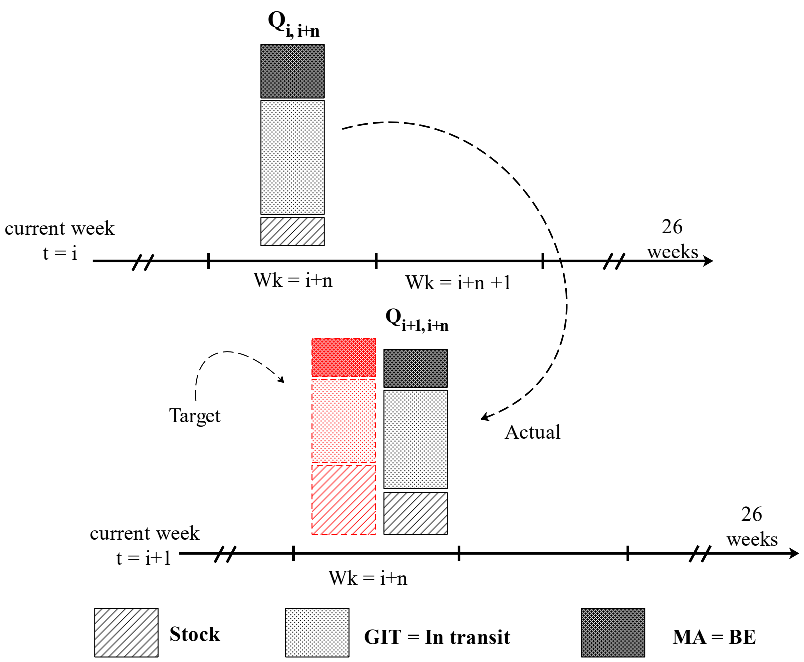


Figure 5.3: Examples of anticipation and reaction in ATP calculation with different product values.

For example, in Figure 5.3 the Target has a bigger value of Stock as the planning system anticipated as a bigger portion of GIT will arrive. Stated differently, all GIT that is expected to arrive in the next time period is still in GIT and the parameters used within

APS is a weak estimation of reality. What is more, the whole quantity of Target is bigger than the Whole quantity of Actual.

As a result, the hypothesis of TTT vs ATT aims to connect the ZD-nEWs to the difference between TTT and ATT in GIT for certain routes, i.e., short routes. The reason for choosing a short route is that the ZD-nEWs happens in days that are near to the current day. For instance, the internal logistics route from Singapore BE to DC in Europe usually takes a couple of weeks, which would be anticipated by APS. However, transport from the Back End in Singapore to DC in Singapore is not a long route like moving from Asia to Europe. Thus, based on a SC experts' ideas we focus on the route between SIN (BE in Singapore) and DCA (Distribution Center Asia).

To understand the relations within the hypothesis, there are different databases within the case study. To the best of our knowledge, revealing the source of ATP shifts in data, connecting the ATP data to Transit data, and finally, its relation to SIN-DCA is not a straight forward task. This is why we must simulate the supply chain system. It helps us to understand the relations, relevant data and providing a risk-free environment for experimentation.

To experiment with the root cause of ZD-nEWs in ATP instabilities, we should model the planning system of the case study. When the simulation model is verified and validated, we can test this hypothesis. In the following sections of this Chapter, we present a conceptual model and provide details of the developed simulation model.

5.4 Simulation Conceptual Modeling

A conceptual model is a crucial step in building a simulation model (Santa-Eulalia, D'Amours, and Frayret 2012; Proudlove, Bisogno, Onggo, Calabrese, and Ghiron 2017) and an important step in interacting with persons knowledgeable about the system being modeled. Building a conceptual model provides a shared understanding between stakeholders and the model builders. Several conceptual modeling methods are presented in Chapter 4 and in Mousavi, Azzouz, Heavey, and Ehm (2019a). Using BPMN we developed a conceptual model shown in Figure 5.5 on page 130. The conceptual model allowed us to share an understanding with system experts and to effectively abstract and develop the simulation model.

In our conceptual model, the supply chain divides into two flows. First, the material flow that models the production steps for transferring raw material to the finished product. Within the simulation model, each production site is considered as an agent that has a DE process inside. As there is a network of production sites in the case study (see Figure 3.2 on page 41), we developed an abstract network of FE, BE, DC, and logistics.

Second, Information Flow to manage the material flow is modeled. Each software module in the the case study planning system is modeled as an agent. These agents, modeled in an abstract way, capture the functionality of software modules. Modeling SC planning as agents provides a means to abstract the relevant interactions and functions within APS.

The simulation model used was developed by AnyLogic. The conceptual model was developed using BPMN (see Figure 5.5 on page 130) and a schematic demonstrating the abstraction used is shown in Figure 5.4 on the next page. With this modeling approach, we simulate the interactions between material flow and information flow.

In Figure 5.4 on the facing page, the selected level of abstraction of MF and IF is shown. Since the focus is on ATP generation and deviation between Target and Actual transit time (anticipation and reaction), many details of the activities within Demand Planning (DP) and Order Management (OM) are neglected. Instead, in IF, each planning module (boxes) is considered as a separate agent. The main functions of the IF agents modeled are listed in this figure. In the Flow of Information, two databases are modeled, Customer Orders and ATP Instabilities which are part of the planning system. These two databases read orders and store ATP instabilities from the APS. In the Flow of Material part of the diagram, an abstracted model of MF is extracted from the real production network, and this models the interaction within MF and the interaction between MF and IF. The MF comprises of three FE, two BE, and three DC. In addition, the logistics is modeled between BE and DC. FEs, BEs, Logistics, and DCs are defined as separate agents. Each agent in MF has a DE process inside. The flow of products within the supply chain

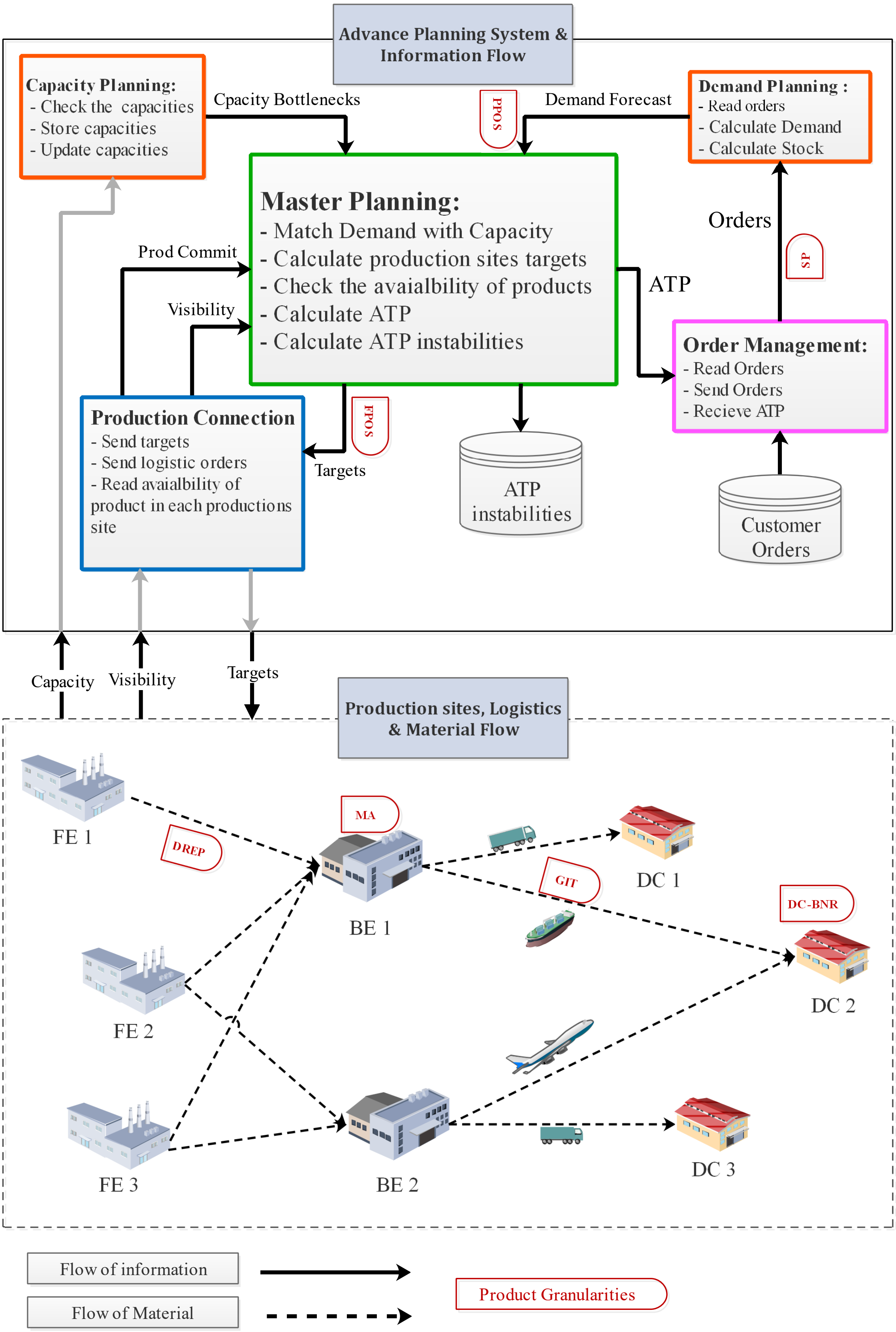


Figure 5.4: Abstraction model of the case study supply chain for evaluation of ATP instabilities.

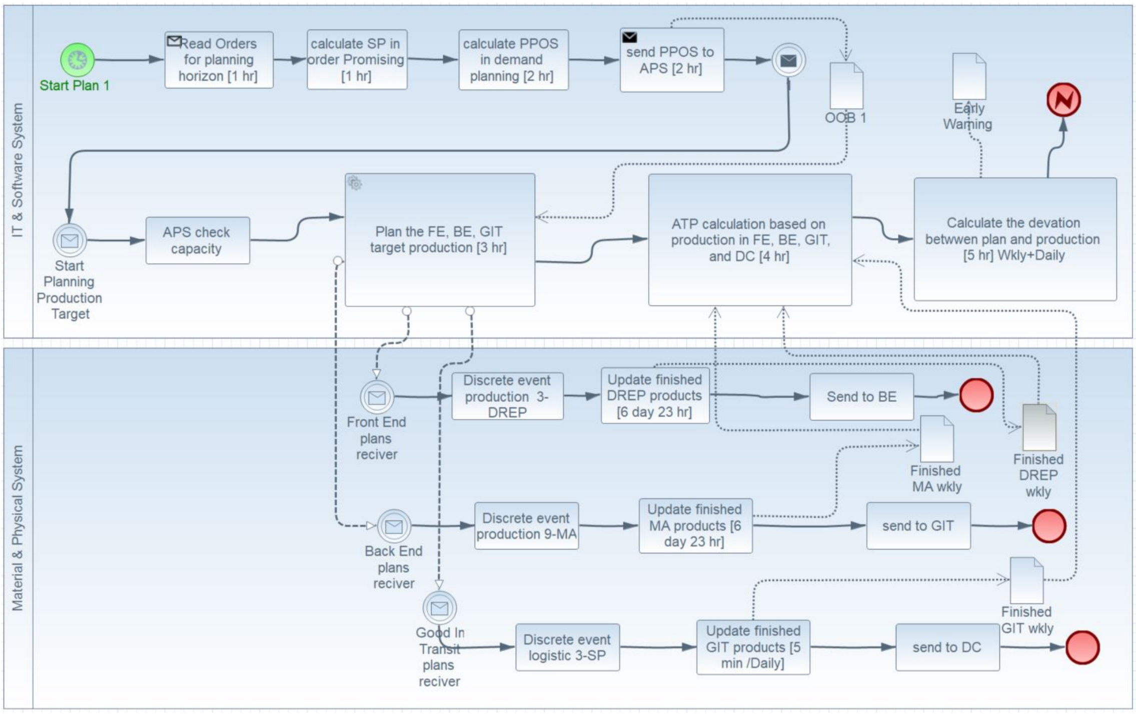


Figure 5.5: Business Process Modeling Notation of conceptual model.

are also highlighted in Figure 5.4 on the previous page.

Within the model (see Figure 5.4 on the preceding page), the interaction between the IF and MF in the system is simulated by reading the quantities of Capacity (free capcities of the production sites) and Visibility (availability of products) from the MF agents. Moreover, the IF agents develop Targets (production plans). These are the considered types of interactions between IF and MF.

Beyond the visualization of the simulation model in Figure 5.4 on the previous page, a BPMN is developed (Figure 5.5) as a conceptual model. The BPMN also shows the same interactions discussed above. The BPMN is like discrete steps or sequences of processes. In Figure 5.5 BPMN has its own standard notation which supports modelers to better understand the sequences within the simulation model. For instance, the start of a process could be a message base or time base (like the 'Start Plan 1'). When it is message base, its trigger depends on messages, for instance, the 'Start Planning Production Target' depends on the receive of a message that PPOS in APS is received. Then it triggers the 'APS check capacity' function.

Besides, the BPMN comprise of two layers of MF and IF. It divides to two levels of

MF and IF like Figure 5.4 on page 129. The IF shows the steps from reading order data to calculating ATP instabilities. For instance, the required data regarding unsatisfied orders store in 'OOB 1'. Beyond data storage the interactions also show in this figure. For example, when a Plan (target) sends from IF to a relevant production site, a message triggers to start production or put the product in a queue. When the production is finished, the granularity changes to the next level. Note that the timestamp of triggering of the events is shown in the name of each process. For instance, when it is written [3 hr], it means that this process runs after three hours pass from the start of each week or the relevant time bucket.

Since the planning and production system of the semiconductor SC is large and complex, using the above modeling approach, the modeler can focus on the specific aspects of his/her investigation. The above modeling framework is very useful to evaluate the performance of a planning system in addition to analyzing external factors like demand fluctuation, and developing digital twins of SCP systems (Santa-Eulalia, D'Amours, and Frayret 2012). We used this conceptual model to simulate the ATP nervousness caused by the lack of incompatibility between software parameters and uncertainty in the production yeilds. These components are described later in this chapter, but next we describe the assumptions in the model.

5.5 Simulation Modeling Assumptions

Based on the research questions, the main goal of the simulation model is to evaluate the performance of ATP generation within APS. To evaluate APS, the simulation model should imitate the planning system and the algorithms of the case study. Thus, we simulate the customized APS in the case study including its algorithm and it interaction with the discrete flow of material. The simulation model first goal is to show the effect of anticipation/reaction within the internal model of APS. This model indicates the importance of APS input parameters and the tuning of parameters used by APS.

The proposed conceptual model allows abstraction to address the research hypothesis allowing the simplification of the real system. In this section, we discuss the assumptions according to research questions and limitations.

As shown in the case study chapter, ATP is generated for the complete planning system. However, capturing all the elements of the planning system is not possible in one project or be computational feasible or necessary in developing a simulation model. For example, there are various demand priorities within APS. Simulating all types of demand and considering separate inputs will make the model too large. However, the results will not differ greatly with regard to answering the research question. As a result, we abstract or neglect some parts of reality. In the following, we detail the assumptions used in the model.

5.5.1 Products and Customers Assumptions

Three customers are considered in the simulation model. This restriction does not affect ATP generation. The ATP granularity consists of Nine different MA (see Subsection 3.3.4 on page 51):

$$(MA = \{MA1, MA2, MA3, \ldots, MA9\})$$

three DREP:

$$(DREP = \{DREP1, DREP2, DREP3\})$$

and three FPOS:

$$(FPOS = \{FPOS1, FPOS2, FPOS3\})$$

The MA could be in BE, in GIT or in stock. So each MA could be in four stages as planned, either in BE, WIP, GIT, or DC.

5.5.2 Simulation Time-frame Assumptions

The simulation model has three indices of time. First is the Week number (indicated by W or CW). In this simulation, we provide 32 weeks of orders. Second index is Planning Horizon (PH). PH shows the length of weeks for planning. Within the simulation model, PH is equal to 10 weeks. Third index is Rolling horizon (RH). Since the planning system

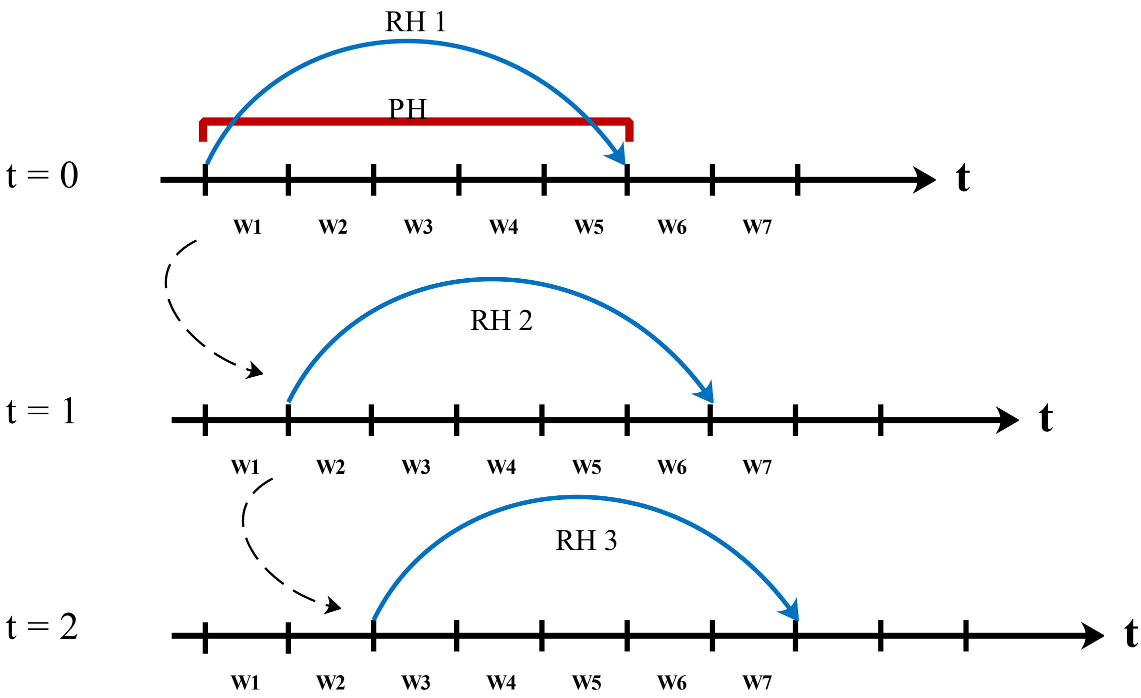


Figure 5.6: Difference between PH, RH, and week numbers.

every week calculates a plan on a rolling base, this identifies each of this rolling horizon with a number. Thus, the length of RH is equal to PH. The simulation model runs for 23 RH, which literally means that it calculates plans for the next 10 weeks for 23 times. To better clarify the difference, here we provide an example and a diagram (see Figure 5.6). In this figure, 7 weeks of data are shown. PH is equal to 5 weeks and there are 3 RH. For instance, RH2 is the plan which calculates at the beginning of week 2 for the next five weeks (PH).

5.5.3 Demand Type Assumptions

We consider "commit orders" and "min stock" as the demand types within DM. The other demand types are neglected. The rationale for this assumption is that considering other demand types like forecasts and stocks does not capture the effect of the difference between what is planned and executed. The simulation model aims to reflect the deviation between what is planned and what is executed in ATP generation (see Section 3.6.1 on page 67).

5.5.4 Constant Demand Assumptions

The demand for the upcoming 10 weeks are all received once per week. In contrast to reality, the demand is considered as fixed values and customers cannot change orders. To develop the ATP, the input orders are not considered equal. However, a repetitive pattern is used where constant values of orders are repeated. For example, if we consider the planning horizon is six, we need to have six-week demands which are constant like $[200, 200, 200, 200, 200, 200]$. However, considering demands like this make the recording of the instabilities challenging. Thus, in our model we consider a different and repetitive pattern of demand like $[200, 100, 300, 200, 100, 300]$ which in summation has equal value. This difference makes recording the instabilities easier. A screenshot and description of customer demands is shown in Figure A.2 on page 236 in Appendix A.2 on page 234.

5.5.5 Production Sites Assumptions

Each production site is modeled as a server. Thus, we do not simulate the detailed production steps of manufacturing. In contrast, we use DE to simulate the stochasticity of manufacturing. Production sites receive targets and put them in queues for a single server which models production. Production yields are modeled and are stochastic based on statistical inputs. Thus, by this means, we can simulate the overall performance of each production site in a connected network.

5.5.6 Capacity Planning Assumptions:

Each production site has a capacity according to input granularities. When a production site produces three different granularities (for example, in BE1), the capacity divides equally between them. The capacity assigned to production targets using the first come first serve rule.

The above assumptions facilitated the modeling required to address the objectives of this research. Although more details could be included which would not cause computation issues, we tried to keep the model as simple and efficient as possible. The above-

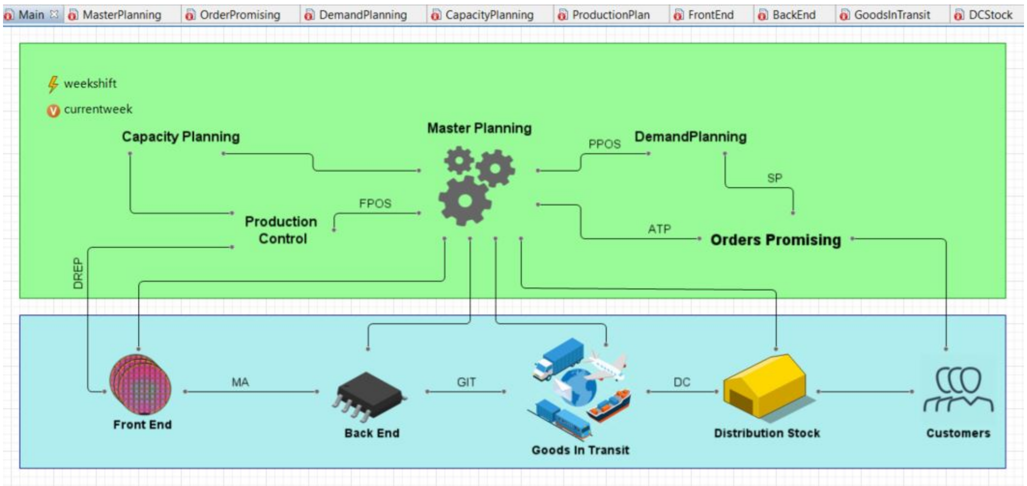


Figure 5.7: Main elements of the case study simulation model in AnyLogic software.

mentioned assumptions model all aspects of Master Planning required to calculate the ATP based on the logic obtained from the real algorithm while the inputs such as the type of products were simplified or reduced. Based on these assumptions, the simulation model components are discussed in the following section.

5.6 Simulation Components

The simulation model was developed using AnyLogic 8.3 University Version. AnyLogic is a Java based simulation software. This simulation software that supports multi-paradigm simulation, discrete event simulation (DE), agent-based simulation (AB), and system dynamics. In this research, AB and DE are used in one model to imitate ATP generation of APS in the case study. We aim to test the source of instability that comes from the difference between anticipation and reaction within APS. In this section, we describe the simulated model components that divide into DEs, ABs, and database elements. Figure 5.7 shows the simulated agents, connections, interactions, and granularity of products. Further details can be gained in Appendix A.2 on page 234 which describes the different elements of the model.

5.6.1 Production Sites and Network

The whole network of the production system is extracted from the real network, where the product flows from BE, FE, and DC. Figure 3.5 on page 49 shows the global production sites and Figure 3.1 on page 40 shows the network samples between these sites. By keeping the ratio (of the products) in these examples, we developed three FE and two BE and three DC. All production steps are not connected to each other similar to the real network. The quantity that ships between production sites is calculated using first-come-first-first-serve (FCFS) in Master Planning. In addition, each production target according to the capacity are defined in Master Planning.

Each production site is located in one agent. The agent receives the targets and sends the available products to buffers in production. The agents work like a communicator to the planning system. Inside each agent, there is a DE of the production line. For example, Figure 5.8 on the facing page shows the DE simulation of the FE1 which is responsible for the $DREP1$ granularity. The 'drep1' receives targets, and puts them in the queue ('que1DREP1'). Time Measurements (clock symbol) calculate the production time that is spent on each product. 'Production DREP1' is the same in all production steps for FEs. The delay time is evaluated dynamically, which maybe stochastic and may depend on the agent as well as on any other conditions. In the case study, The Front End production takes around 4 weeks. Here in this example, we use a triangular distribution with mode equal to 27 days (min and max are 24 and 30 days). Note that this is an example that we evaluated using different combinations of these statistic inputs, which will be presented later. In the 'WklyStock1' queue, the final product is gathered for shipment each week. In the final step in Figure 5.8 on the next page, the 'exitDREP1' is defined to deliver the finished product to the logistics of the next production site in the SC according to the targets of APS. The values of these queues are stored in a variable and collections ('varWklyProduction1' and 'colWklyDREP1') to be used by an agent for interaction with APS. The agent in this example updates the targets, capacity, and buffer stocks using the event called 'eveUpdateDrep'. This event follows the process flow (discussed in Figure 3.14 on page 68) of the case study and runs at the end of each week in the FE production

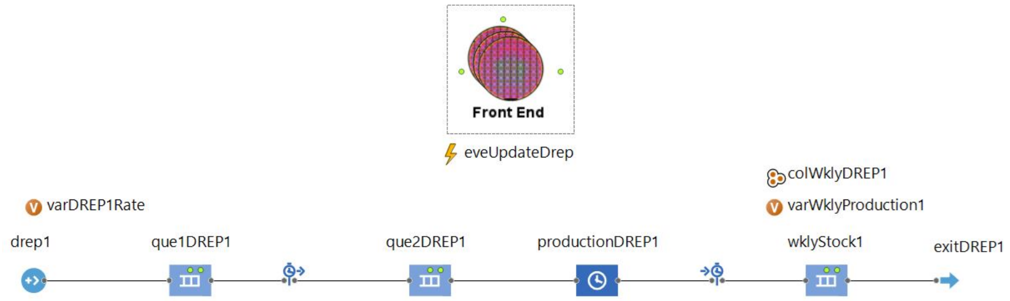


Figure 5.8: Discrete Event Simulation of Front End production line inside an agent.

site. This event could be modified to evaluate the interactions between the production site and APS.

The discussed approach for connecting AB and DE is carried out inside an agent in this example and is replicated for all production and logistic elements of the simulation model. All production and logistics agents and related DE are presented in more detail in Appendix A.2 on page 234.

Before describing the model in more detail below, note that a collection in Java is a set of variables that are stored in the form of an array. A collection in Java is a framework that provides an architecture to store and manipulate groups of objects. A Java collection can achieve all operations that you perform on a data such as searching, sorting, insertion, manipulation, and deletion. This helped us to better transfer the data between agents and simplified the coding of the planning module as a collocation could easily be used in loops.

A set of DE processes according to the logic of the production site or logistics route where developed using the DE paradigm. DEs receive targets from agents using a set of variables. A DE has some input parameters like delays, capacity limits, and queues. These DEs push values for variables and collections related to agents. Finally, based on time-based events, agents are sent these variables or collections which are transferred on to the planning system. This communication is the interaction of production sites with the APS. These communications between production lines support's APS to update plans and to send new targets, which are sent using time-based events in the planning agents.

5.6.2 Simulated APS

The APS in the simulation model consist of Order Promising, Demand Planning, Capacity Planning, Production Control and Master Planning. Each of these agents imitates the discussed abstraction in Section5.4 on page 127 and assumptions given in Section 5.5 on page 131. In this section, we discuss the main agent which is Master Planning describing some of the important functions. The rest of the agents with their functions are presented in Appendix A.2 on page 234.

Master Planning:

Master Planning is the core of APS in the case study and the focus of this research. Figure 5.9 on the next page shows the simulated master planning agent in AnyLogic. To make the model transparent and maintainable, this agent's elements are divided according to functionality within the planning horizon. The planning horizon consists of ten weeks and the relevant granularity is shown in the presented arrow on the bottom of Figure 5.9 on the facing page. The functionalities are also divide into three levels for calculating 'Planning Targets', 'ATP calculation', and 'ATP Nervousness & nEWs'. For example, between weeks one to three products are in BE. In this period, 'ATP Calculation' using the 'maATP' function is programmed to calculate the available products at the MA level. 'beProdTarg' is programmed to calculate the production targets for BEs according to demand and capacity. All functions are triggered by time-based events within the agents. Collections are used to store the results within agents. They are similar to random access or volatile memory (RAM) in APS (Wiers and Kok 2017).

As shown on the planning horizon arrow in Figure 5.9 on the next page, the planning period is a week until the products are in the BE. When the BE's productions are finished, the products are shipped to DC. The visibility and monitoring of ATP in the last week within DC are calculated daily as the delivery due date time line are measured in days. Besides, the ZD-nEWs hypothesis will focus on this last delivery. Thus, we calculate the ATP, instability, and targets based on days similar to real APS in the case study. For this purpose, the events related to GIT and DC granularity are triggered daily. For instance, 'eveGIT' will monitor and calculate the available product in GIT at 04:00 AM of each

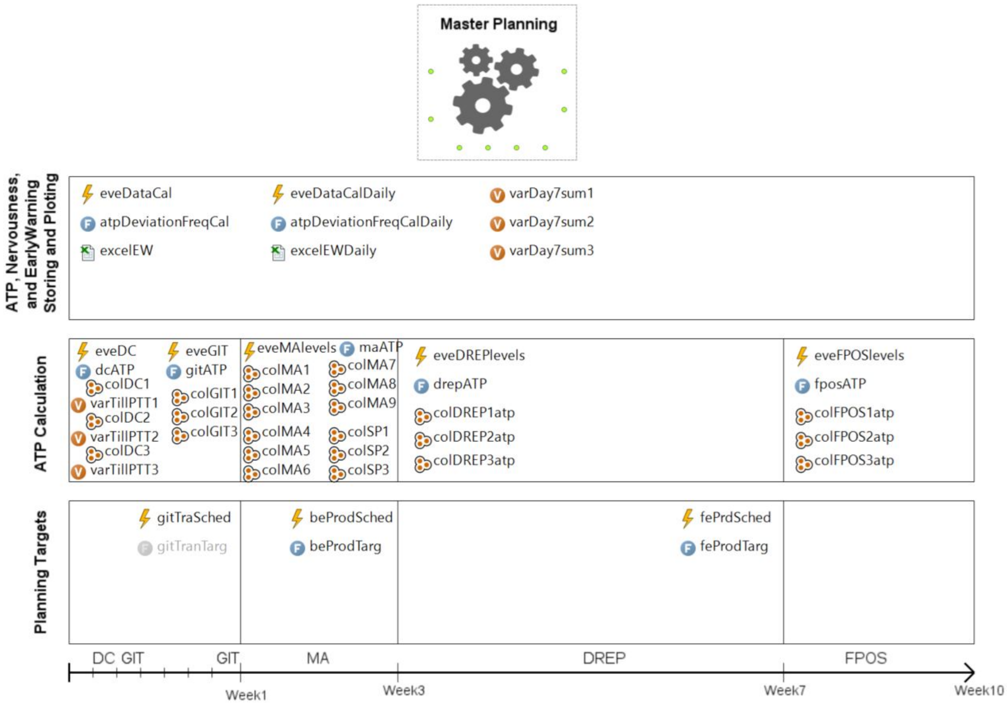


Figure 5.9: Simulation model description of master planning.

day.

To capture the instability within ATP, we need to define more functions than is found in reality in Master Planning. The 'atpDeviationFreqCal' is defined to calculate the weekly instabilities for products in MA, DREP, and FPOS. This function is triggered by a weekly time-based event called 'eveDataCal'. In addition, 'atpDeviationFreqCalDaily' is coded to calculate the daily instabilities in GIT and DC granularity. The reason for moving from weekly to daily is that in reality ATP will measure delivery in days rather than weeks, with weeks used for the medium term planning horizon. The daily calculation of ATP also triggers a time-based event called 'eveDataCAlDaily' (see Appendix A.2 on page 234).

Both of these functions calculate the nervousness percentage (N_t) of what is planned (targets) at time $t-1$ ($ATP_{pl,t-1}$) and what is available at next rolling time t ($ATP_{ava,t}$).

$$N_t = \frac{ATP_{ava,t} - ATP_{pl,t-1}}{ATP_{pl,t-1}} \times 100 \tag{5.1}$$

We focus the calculation of the ATP instability at different decoupling points. The reason is that the product granularity changes at these points and targets are calculated based on anticipations. The validation diagrams (see Figure 5.11 on page 143) represent this percentage of nervousness. All other agents in the model of the planning system follow the same simulation structure as Master Planning.

5.6.3 Databases

One of the main features of AnyLogic that support us in modelling APS and relevant production systems of the case study is the easy integration with Databases. This feature helps us to import real data from transit times to the simulation model. Besides, it helps us to store the results. Since the simulation model outputs contain 23 different runs of the consecutive rolling horizon for all granularities of ATP, storing the outputs in collections and variables is impossible. Thus, with the database feature of AnyLogic, we import the order tables, import transit times, and store ATP nervousness.

5.6.4 Limitations and Possible Improvement

The developed simulation model in this work aims to answer the questions according to the defined hypothesis. However, based on the performance of the simulated model, the approach could be extended to a more general simulation model. In a scale of a larger project than this PhD, the model can be extended to simulate the whole automatic parts of APS and to further detail the production sites. This would allow the simulation model to more closely move towards a digital twin of APS. It could be used for further examination and testing of APS in a risk-free environment. In the case study, this focus must be on Master Planning as the core of APS. Thus, in this subsection, we discuss the limitations of the current simulation model and possible improvements.

The first is Demand Planning which was simplified according to the research question addressed in this thesis. We did not consider all priorities and algorithms in demand planning. Second, the orders are considered as constant input, while in reality customers can change orders based on contract terms. In future extensions, customers could be modeled by agents having internal state-charts. These agents could interact with the model to put or modify orders. Yet in our model, the orders are kept constant and uploaded once in a rolling horizon. Third, capacity planning and relevant algorithms were not defined completely. We just model the visibility within Capacity Planning. Fourth, the production sites and their DEs simulated models are captured at a very high level where only the overall statistical behaviour of each production site is modeled. These models could be further elaborated where production priorities and complexity could be captured, including, for example, different products at each production site and more details modeled within each production site. Finally, order management was not included in the current model, this could be further elaborated on within a future model.

Another approach that could be examined is defining product granularity as agents instead of entities. If used, product agents could interact with planning and production agents. In addition, they could change to other agent types within granularity agents. The agent-based library of AnyLogic is capable of doing this. Moreover, the available examples in AnyLogic could help future research in this regard. Another interesting and rarely studied area is considering human planners. Human planners are agents that have complicated states and interact with the planning system to add or modify values within APS. With this approach, the missed planning system in our work like allocation planning (see Section 3.8 on page 83) could be modeled.

5.7 Simulation Model Verification

To make reliable decisions, one crucial step toward developing a simulation analysis is model Verification and Validation (V&V). Verification is defined as a means to ensure that one built the system right, while validation is a means to ensure that one built the right system (Walden et al. 2015). There are different techniques to verify a simulation model, including writing and debugging the computer program in modules, a structured walk-through of the program, the Interactive Run Controller (IRC) or debugger, running the simulation with different parameter inputs to carry out sensitivity analysis by checking to see that the output is reasonable using some measures of performance, observation of animation of the simulation output, etc. (Law 2013).

In our case, we used the last technique where we set up different scenarios with

Figure 5.10: DE model of a shipment line from BE to DC inside Good In Transit agent.

different ATT and TTT values. Then we tracked the simulation output and the values of the internal variables. The focus of nervousness evaluation is on the logistic system from BE to DC. Thus the Good In Transit agent is where we define Target Transit Time (TTT) and read Actual Transit Time (ATT). For better clarification, In Figure 5.10, the variable 'varPlannedTransitTime1' stores the TTT in the format of days. The 'shipment1' also simulates the real transit time modeled by its statistical inputs.

To examine the hypothesis of deviation between ATT and TTT as a cause of nervousness in ATP, we design three scenarios to experiment with, which are presented in Table 5.1 on page 144. Each scenario is replicated 10 times and the average deviation for RH 14 (see Subsection 5.5.2 on page 132) are plotted in Figure 5.11 on the facing page. Note that RH 14 are summed across the replications. The reason for selecting one specific week is that each replication of the simulation model consist of 23 RH and if we averaged along the rolling horizons it would not mimic the deviation of the planning system (it would be the average of one rolling horizon). Thus we select RH 14 and calculate the average of this RH deviation in 10 replication. The average deviations of RH 10 and RH 20 for 10 replications are provided in Appendix A.3 on page 240.

The simulation model time unit is hours and it runs for a period of 23 RH. The first 7 weeks are designed to allow the model to pass the transient phase, which means that the first order pass all the production steps and arrive in distribution center. Figure 5.11 on the facing page shows the percentage of nervousness in ATP for the final 10 weeks before the unit is delivered, with weeks depicted in the x-axis from week 10 to 2 and from week 2 to the delivery date, the unit changes to days.

To facilitate the interpretation of the diagrams, we explain the verification diagrams. In Figure 5.11 on the next page if there is no difference between the MF (or ATT) and IF (or TTT) each week, the output for each product will reside at "0" in the Y-axis ("FRE-

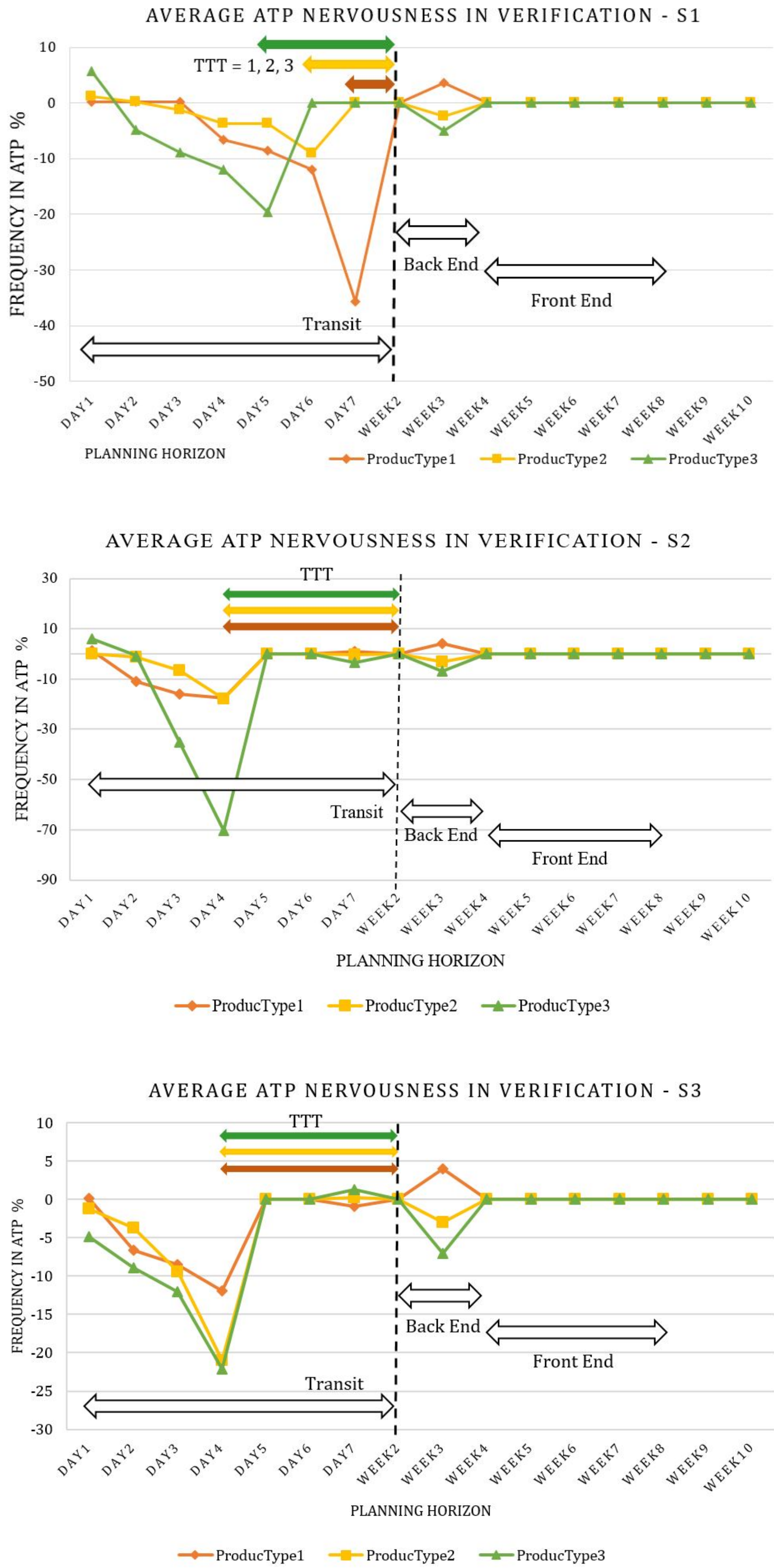


Figure 5.11: Results of verification according to the values in Table 5.1.

Table 5.1: Experimented scenarios.

	Scenario 1		**Scenario 2**		**Scenario 3**	
Product Type	TTT	ATT	TTT	ATT	TTT	ATT
1	1 days	Triangular (1, 4, 3)	4 days	2 days	4 days	Triangular (2,3.5,3)
2	2 days	Triangular (1, 4, 3)	4 days	4 days	4 days	Triangular (3, 5, 4)
3	3 days	Triangular (1, 4, 3)	4 days	6 days	4 days	Triangular (2, 6, 3)

QUENCY in ATP %"). In scenario one (S1), product type 1 in BE (week 3) has almost + 4% deviation which means the delivered quantity from FE to BE is 4% more than what was planned. For product type 2, which uses a transit time of 2 days in planning (yellow), the product arrives at day 6 with a deviation of around 10 % from what were planned (-10%).

When the ATT between BE and DC can not meet the TTT, ATP nervousness in ATP is expected, mainly when it is very close to shipping the promised orders. As presented in Figure 5.11 on the preceding page, the simulation model verifies this hypothesis. It shows ATT and TTT mismatch as a source of ATP nervousness. The average of deviations generated by the simulation model is validated by comparing the values of EW frequency extracted from real data, which varies from 18 to 34 percent.

In scenario 1, we considered different TTT for each product type while the statistical distributions of ATT were the same (i.e., a triangular distribution with the maximum value of 4 days, a minimum value of 1 day and a mode of 3 days) and the same seed was used in the replications. In Figure 5.11 on the previous page the results of scenario 1 show that, as far as the TTT increases (from 1 to 3 days), the nervousness in ATP decreases. This could be explained by the fact that a larger TTT makes the system less sensitive to possible delays in transit. However, this setting may result in a higher level of stock that should be kept in inventory until the shipping date arrives.

In scenario 2, we considered the same TTT for all product types while we used different settings for ATT values. The ATT are deterministic to evaluate the effects of other production steps. The results show that, when ATT is bigger than TTT, there is a big shortage in the expected product arrival date. For instance, in product type three where ATT is 6 days and TTT is 4 days, there is 70 percent shortage in day 7. At the same time, even when the ATT is less than TTT, there is still a shortage that could be explained by nervousness in the previous production steps and mostly in the back-end. For instance, in product type 2 there is 15 percentage deviation. This scenario shows that the deviation in the simulation model not only comes from the late delivery in the logistic system, but it also relates to previous steps in production. It shows that the effects of the whole production system considered within the simulation model.

In the last scenario, TTT was also considered the same for all product types and we set ATT to follow different triangular distributions with a mode less or equal to TTT. In verification we simply use triangular distributions based on our understanding of the manufacturing processes. Later in validation of the simulation model we provided more educated distributions. The deviations in production types 1, 2 , and 3 are 12, 21, and 23 percentage, respectively. In comparison to the other scenarios, the recorded ATP nervousness and EW frequency have the lowest deviations.

Based on this verification, the APS planning and production system interact. The simulation model imitates the developed functionalities within APS. It shows that the difference between TTT and ATT could be a source of instability within APS. The results of verification were presented to the case study experts. Consequently, specific logistic routes were suggested for further examination with the simulation model. Three routes are between BE in Singapore and a DC in Asia. These routes are between the same destinations but various product types are shipped by different third parties. Thus, we aim to define and extract the relevant data for testing the hypothesis of this logistic route in the simulation model. The next section discusses the data preparation to be used in the verified model.

5.8 Hypothesis Data Preparation

The results of the verification show that the incompatibility between TTT and ATT could cause ATP nervousness. Based on the verified results and with the support of experts, we focused on finding industrial data related to ATT values between BE and DC. We extracted 3 months of logistic data from BEs to DCs. Finding the proper data in the right time interval was the first step. Then we cleaned the data using the R Studio with

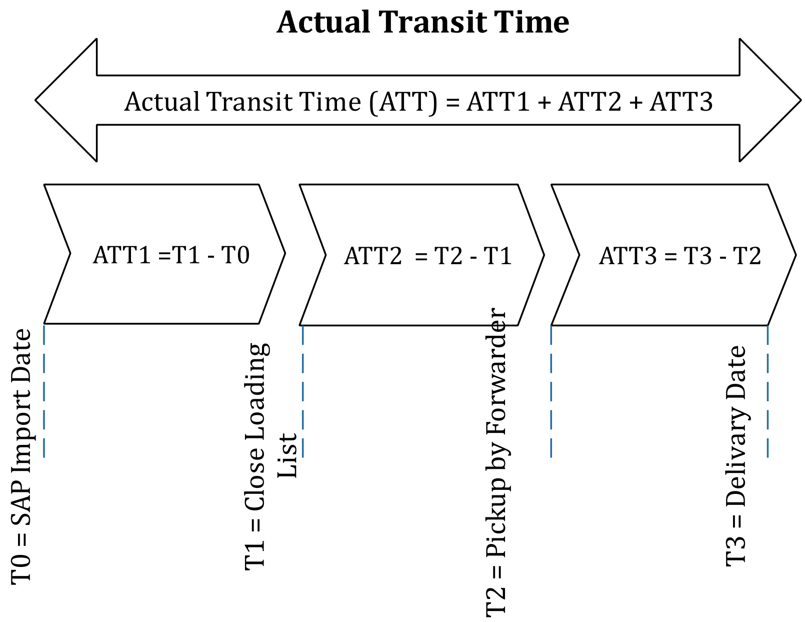


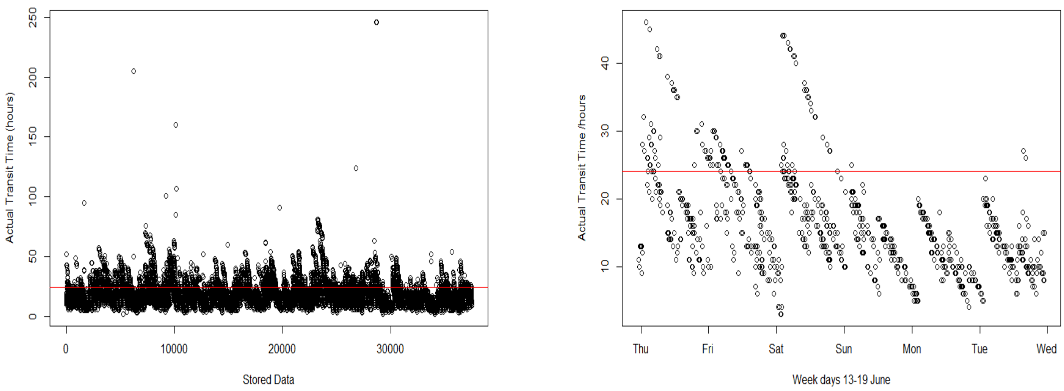
Figure 5.12: Actual Transit Time components.

Python. After cleaning the data, explanatory models were developed to extract statistical features. The outputs of this step were prepared using a compatible format for use in the simulation model. Finally, we added the data into the simulation model. These steps are discussed in further detail in the following.

Transit Monitoring and Analysis (TMA) is the datasheet that we used for this data analysis. TMA stores the transit times between different BEs and DCs. It contains ATT and TTT which are categorized based on the product types (ATV, PSS, IPC, and DSS see Chapter 3). The extracted data contains three months of transit data containing around 70000 rows. Thus, we need to do an explanatory analysis to better understand the transit time.

Both ATT and TTT in the TMA data divide into three different transit time intervals. Figure 5.12 shows the three components of ATT. For instance, ATT2 is a period of time from closing a shipment list in the BE to its pickup time by a third party. The summation of ATT1, ATT2, and ATT3 is the value of ATT which is used as a value in APS. Similarly, the TTT is divided into these three periods, but its values are fixed and are defined by APS. In this example, the total TTT is 24 hours, which means that APS anticipation of transit time is 24 hours. For analyzing the collected data we used R. Appendix A.5 on page 243 contains some examples of R code used to conduct this data analysis.

Data cleaning is the first step. Rows with negative or zero ATT values were deleted from the data. Later, an explanatory data analysis was conducted by visualizing the data for different products within the different time frames. Figure 5.13 shows two examples of explanatory visualization. Figure 5.13a visualizes all cleaned data according to the value of ATT. The red line in this figure represents the value of TTT which is 24 hours. Furthermore, data visualization was performed to search for any pattern in different product types or periods of time. Figure 5.13b is an example of an ATV product type for one sample week (from 13 to 19 for July 2019). Similarly, the horizontal red line shows the TTT.



(a) ATT visualization for all the TMA data.

(b) A week sample of ATT in TMA.

Figure 5.13: Examples of explanatory data analysis.

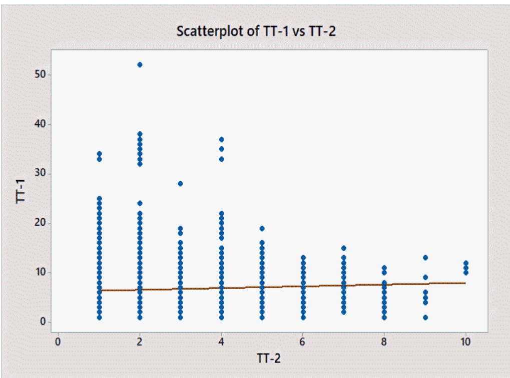
The results of explanatory data analysis show that there are considerable numbers of ATT with a higher value than TTT. However, there is no evidence of patterns within the total ATT. Table 5.2 on the next page shows the average ATT based on days for all product types. This table was extracted based on the whole period of TMA data. Since the BRCycle runs daily, here we change the time scale from hours to days. Thus, the TTT is equal to one day (24 hours) and all ATT greater than one means there was a deviation between what was planned and what was executed.

Since there is no descriptive statistic for ATT, we move to further detail and evaluate ATT1, ATT2, and ATT3 and their relations. For this purpose, linear regression between ATT components was conducted (see Figure 5.14 on the following page). Figure 5.14a on the next page shows the linear regression between ATT1 and ATT2. It indicates that

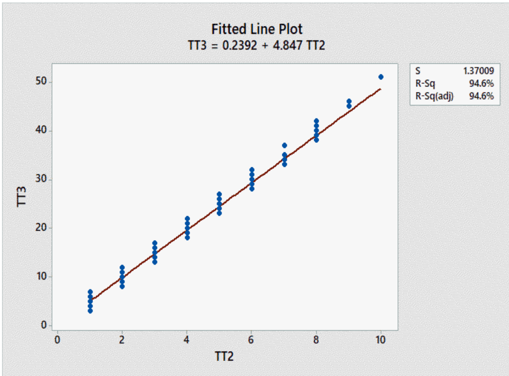
Table 5.2: Average ATT for all products types.

	ATV	PSS	DSS	IPC
ATT = 1	0.86	0.84	0.546	0.875
ATT = 2	0.13	0.151	0.403	0.115
ATT = 3	0.007	0.008	0.038	0.008
ATT $\geq$ 4	0.003	0.001	0.013	0.002

ATT1 which is mostly related to the automatic closing of the list, has no relation with ATT2 which is the waiting time until the third party picks the package. On the other hand, Figure 5.14b shows that there is a strong linear relation between ATT2 and ATT3. If the third party collects the package late (TT2), then shipment will be delayed (TT3). This provides insight for better monitoring and controlling the handled of the logistics by the third party.



(a) Regression analysis between ATT1 and ATT2.



(b) Regression analysis between ATT2 and ATT3.

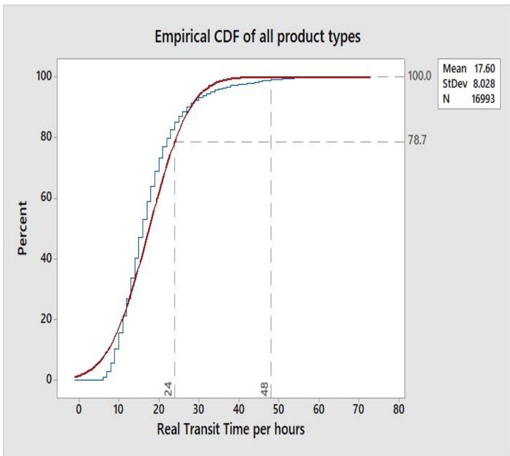
Figure 5.14: Linear Regression test on ATT components.

The main objective of this data analysis is to use this data in our simulation model to validate the hypothesis. Thus, we need to obtain a statistical distribution for the ATT data. The goal is to use a distribution as a delivery time of shipment within the simulation model. Besides, TTT uses as a simulated APS parameter (TTT in the simulation model). While we tried to fit various formats of data to well-known distributions, the results of Goodness of Fit revealed that none of the distributions would fit (see Figure 5.15 on the next page). Figure 5.15a on the facing page presents the output of Minitab and the p-values and Anderson-Darling goodness-of-fit statistics (AD) that indicate the incompatibility of the data with well-known distributions. Figure 5.15b on the next page shows that the ATT does not fit a normal distribution. Thus, we used the empirical distribution of the data. In addition, cumulative analysis for all product types revealed that almost 22% of the products arrive after the planned time and most arrive before day 2 (48 hours).

Goodness of Fit Test

Distribution	AD	P	LRT P
Normal	141.992	<0.005	
Box-Cox Transformation	187.554	<0.005	
Lognormal	193.762	<0.005	
3-Parameter Lognormal	398.612	*	0.000
Exponential	267.074	<0.003	
2-Parameter Exponential	1933.810	<0.010	0.000
Weibull	151.112	<0.010	
3-Parameter Weibull	428.145	<0.005	0.000
Smallest Extreme Value	131.775	<0.010	
Largest Extreme Value	177.911	<0.010	
Gamma	172.430	<0.005	
3-Parameter Gamma	657.852	*	0.000
Logistic	141.937	<0.005	
Loglogistic	179.839	<0.005	
3-Parameter Loglogistic	359.160	*	0.000

(a) Goodness of Fit results - Minitab.



(b) Cumulative Distribution Fit.

Figure 5.15: Fit distribution to ATT data.

Consequently, to apply the real data in the simulation model, we transferred the data to an empirical distribution in AnyLogic. The next section discusses the validation of the simulation model with ATT empirical data and TTT equal to 24 hours.

5.9 Simulation Model Validation and Experimentation

The validity of the model is determined regarding the specific purpose it is built for and the research questions to be addressed. A model may be valid for one set of conditions and not for another. A model is considered valid for a set of experimental conditions if the model's accuracy is within its acceptable range.

There are various techniques for validating simulation models such as animation, event validity, and face validity (Sargent 2013). The method that we use in this work aims to imitate the interaction between APS and manufacturing. This interaction of anticipations and reactions depends on the statistical distribution of manufacturing. Besides, the focus of the hypothesis is on EW and ZD-nEWs. Thus, the validation in our case depends on comparing the real EW data with results of the simulation model. However, there are differences that need to be highlighted.

First, the simulation model can create ATP instabilities that might cause EW in demand fulfilment. Thus, the model does not imitate EW exactly, but it shows that the ATP instabilities as the main input of DF calculations that generates EWs. Second, the ATP data is not accessible to be compared exactly with the simulation results. Even if it exists, the real ATP contains thousands of products with wider complexity dimension than the simulation model. Thus, it would not be possible to compare the real or frequency of ATP with the simulation output. Thus, comparing real EW or ATP with simulation outputs are not meaningful, but the simulation models fed with real data will imitate the trend of ATP instability. This trend shows the model can mimic the behaviour of a real supply chain system, resulting in validation of the simulation model.

Another approach of validation is the evaluation of anticipation/reaction within the modeled APS. This is done by changing the TTT parameter in the simulation model. The simulation model has three logistic routes connected to TMA data. Relevant to each of these routes, there are parameters in APS. APS uses these parameters to calculate anticipation from reality. Within the performed validation and experimentation, we modify these TTT parameters to record and analyze their effects on ATP instability. The rest of this section relates to the extracted results used for validation and experimentation.

To validate the simulation model, the empirical distributions of logistic routes were applied to the simulation model (see Figure A.8 on page 239 in Appendix A.2). Besides, three scenarios based on different service time distributions of manufacturing processes were applied. Table 5.3 on the next page shows the scenarios used for this validation of the trend presented in the following graphs. The presented distributions in this table are used for the manufacturing time of FEs and BEs. The distributions of logistic routes, which are the focus of this thesis are directly obtained from real data. The parameters of the distributions (FEs and BEs) were gained from an educated guess based on available information from the case study and after discussions with the case study experts. We used different distributions related to time periods with various variances and parameters to show that the distributions do not have a major effect on the logistic routes. Further investigations in this regard could improve the simulation model. Note that the simulation model was also tested with other scenarios, but since the results did not deviate considerably, we neglect to present all of these here.

Table 5.3: Simulation validation scenarios.

	Scenario 1	Scenario 2	Scenario 3
Front End 1	triangular(24, 27, 30)	erlang(0.15, 5, 27)	normal(0.5,27)
Front End 2	triangular(25, 29, 26)	pareto(20, 27)	normal(0.5,27)
Front End 3	triangular(24, 29, 27)	gamma(1, 0.5, 27)	normal(0.5,27)
Back End 1	triangular(12, 15, 13)	erlang(0.2, 2, 13)	erlang(0.3, 2, 13)
Back End 2	triangular(12.2, 14.2, 13)	pareto(20, 13)	erlang(0.5, 1, 13)
Back End 3	triangular(12.2, 14.2, 13)	gamma(2, 0.8, 11)	erlang(0.4, 2, 13)
Back End 4	triangular(12, 14.7, 13.5)	erlang(0.2, 2, 13)	pareto(20, 13)
Back End 5	triangular(12, 14.7, 13.5)	pareto(20, 13)	pareto(20, 13)
Back End 6	triangular(13, 13.8, 13.2)	gamma(2, 0.8, 11)	pareto(20, 13)
Back End 7	normal(0.5, 13)	erlang(0.2, 2, 13)	gamma(3, 0.2, 13)
Back End 8	normal(0.5, 13)	pareto(20, 13)	gamma(4, 0.15, 13)
Back End 9	normal(0.5, 13)	gamma(2, 0.8, 11)	gamma(4, 0.15, 13)
Logistic 1	Empirical ATT of ATV	Empirical ATT of ATV	Empirical ATT of ATV
Logistic 2	Empirical ATT of DSS	Empirical ATT of DSS	Empirical ATT of DSS
Logistic 3	Empirical ATT of PSS	Empirical ATT of PSS	Empirical ATT of PSS

The scenarios are replicated ten times. Each run contains 23 rolling horizons (see Subsection 5.5.2 on page 132). The planning horizon for each experiment is 10 weeks. Since we focus on movement of product from BE to DC, that week is divided into the time unit of days to better capture the instability frequency. Tables 5.4 on the next page, 5.5 on page 153, and 5.6 on page 154 are the average of 10 replications for the selected RH 14. Each table belongs to one logistic (L) route. All values are the percentage of nervousness and zero means that there was no deviation from what was planned. A negative value means that the available ATP is less than planned. For instance, -8.3918 percent for day seven of scenario 1 (S) in Table 5.4 on the following page, which shows the percentage of shortage in ATP for 7 days before the end of the modeled planning horizon. The instabilities in Week three are the deviation between FEs and BEs.

As mentioned, the model runs for 23 rolling horizons. The first seven RH were set for warm-up. From RH eight, the model passes its transient phase to get products through the planning weeks to be delivered. For instance, Table 5.7 on page 155 shows the out-

Table 5.4: Average frequency of ATP instability for the first logistic route for ten replications.

Logistic 1	Scenario 1	Scenario 2	Scenario 3
Day 1	0.61446	0.926488	0.981385
Day 2	0.61446	0.842072	0.957249
Day 3	-1.89192	0.842072	0.846249
Day 4	-1.89192	0.823506	0.812455
Day 5	-1.90347	0.590187	0.781422
Day 6	-2.24674	-0.7913	-1.18073
Day 7	-8.39819	-10.1897	-15.1417
Week 2	0	0	0
Week 3	-0.86957	-0.91304	-0.84569
Week 4	0	0	0
Week 5	0	0	0
Week 6	0	0	0
Week 7	0	0	0
Week 8	0	0	0
Week 9	0	0	0
Week 10	0	0	0

puts of one run for all 23 RH. Note that weeks 5 to 10 of the planning horizons are not shown in the table as all values are zero. Figure 5.16 on page 156 shows that the moving ATP frequency in the last last 7 days of the modeled planning horizon, week 2, and week 3 (ATP frequencies in weeks 4 to 10 are zero, similar to week 2, thus we do not present them in this figure). The figure is the result of one replication of RH 14, where Day 7 has the largest deviation from what was planned. Day 1 to 5 have the same deviation quantities which shows the importance of the ATT which follows an empirical distribution in comparison to TTT which is equal to one day (day 7, which is the most deviated day is another sign of this). Note that we only sample one replication here to show the frequency in the whole planning horizon. Thus, further replication shows the same behaviour. Further runs of validations and experimentation regarding other logistic routes are provided in Appendix A.4 on page 240.

Testing the model with empirical distribution of ATT and real TTT extracted from industry shows that the percentage instability caused by logistic routes and incoptatibility between ATT and TTT. The next step is to propose a solution to experiment with using the simulation model. Thus, we increase the value of TTT. For Scenario 3 (S3) and logistic route 1 (L1), we increase the length of TTT from 1 day to 2 days, which is

Table 5.5: Average frequency of ATP instability for the second logistic route for ten replications.

Logistic 2	Scenario 1	Scenario 2	Scenario 3
Day 1	0.506833	4.33755	4.0245
Day 2	0.506833	6.245986	2.00154
Day 3	-0.30767	4.041564	3.872444
Day 4	-0.31001	3.819472	3.547114
Day 5	-0.72079	2.853944	1.979996
Day 6	-4.13099	-2.24707	-4.70905
Day 7	-24.2415	-26.3908	-30.4429
Week 2	0	0	0
Week 3	-0.43478	-1.95652	-2.13043
Week 4	0	0	0
Week 5	0	0	0
Week 6	0	0	0
Week 7	0	0	0
Week 8	0	0	0
Week 9	0	0	0
Week 10	0	0	0

equal to the increase of TTT in TMA from 24 hours to 48 hours. Figure 5.17 on page 156 shows new ATP instabilities for each planning day and week. This figure depicts the full planning horizon for one replication. The experimentation result shows that the average instability decreases by around 10 % for the planned delivery day (day 6 in Figure 5.17 on page 156). Note that since there is no difference between ATP frequencies in days 1 to 5 all plots are merged on top of each other. Thus, all lines cannot be clearly seen in this figure.

To clarify the consequences of modifying TTT in APS, two replications (R1, R2) for each scenario (from Table 5.3 on page 151) are depicted in Figure 5.18 on page 157 for logistic route one. This figure depicts the frequency values of planned delivery days for the whole RH (rolling horizon) where all runs belong to the first logistic route. In this figure, S3-Exp-R1 and S3-Exp-R2 show experimentation results where TTT is 2 days. The instabilities in these experimentation cases are mostly positive, which shows a reduction in nEWs, when compared to the other results (indicated by solid lines) which all have a TTT of 1 day. Results of other logistics routes are provided in Appendix A.4 on page 240.

In conclusion, the simulation model indicates that the role of anticipation in APS and its effects on planning stability and performance indicators. With this modeling

Table 5.6: Average frequency of ATP instability for the third logistic route for ten replications.

Logistic 3	Scenario 1	Scenario 2	Scenario 3
Day 1	0.796545	0.872857	0.002405
Day 2	0.796545	0.872857	0.002405
Day 3	-8.05712	1.021086	0.438937
Day 4	-8.19577	0.85526	0.375805
Day 5	-8.22117	0.345194	-0.04212
Day 6	-8.38345	-1.04596	-1.85886
Day 7	-12.0006	-12.5488	-14.2296
Week 2	0	0	0
Week 3	-0.08696	-0.30435	-0.13043
Week 4	0	0	0
Week 5	0	0	0
Week 6	0	0	0
Week 7	0	0	0
Week 8	0	0	0
Week 9	0	0	0
Week 10	0	0	0

approach, not only do we model the complex supply chain system and its interactions, but also we conducted experimentation on our hypothesis. Although the simulation model provides insight into our case study, it does not completely prove the hypothesis. Thus, based on the results of the simulation, we search for the root causes of ZD-nEWs by carrying out an additional data analysis project within SC. The next section discusses the results of the data analysis to examine the ATT vs TTT hypothesis.

5.10 Root Causes Data Analysis

The outputs of the simulation model proved that the delay in logistic routes could be one of the reasons for ZD-nEWs. Although the hypothesis was proved by the simulation model, it does not mean that ATT vs TTT is the only reason for ZD-nEWs. Thus, after the hypothesis was proved, the question arises of what percentage of ZD-nEWs comes from each specific route or which of the logistic routes has the highest amount of nEWs and ZD-nEWs. To answer these questions, we carried out a business data analysis as part of the work reported here. In this section, the pathway and outputs are presented. Appendix A.6 on page 250 contains the data science model developed using

Table 5.7: Simulation validation output for whole rolling horizon of scenario one and logistic route one.

S 1 /L 1	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	W 2	W 3	W 4
RH 7	0	0	0	0	0	0	0	0	0	0
RH 8	16.88	16.88	16.88	16.88	16.88	16.02	4.98	0	3	0
RH 9	-3.23	-3.23	-3.23	-3.23	-3.23	-4.17	-12.14	0	-3	0
RH 10	-1.57	-1.57	-1.57	-1.57	-1.75	-2.44	-10.65	0	-7	0
RH 11	4.01	4.01	4.01	4.01	3.63	2.1	-8.21	0	10	0
RH 12	0.37	0.37	0.37	0	-0.74	-1.85	-9.24	0	-3	0
RH 13	4.9	4.9	4.9	4.71	4.51	3.14	-7.25	0	7	0
RH 14	-6.12	-6.12	-6.12	-6.29	-6.65	-7.73	-16.91	0	-3	0
RH 15	8.24	8.24	8.04	8.04	7.84	6.86	-2.55	0	-3	0
RH 16	-6.13	-6.13	-6.13	-6.3	-6.48	-8.06	-16.29	0	0	0
RH 17	2.23	2.23	2.23	2.23	1.86	0.74	-11.17	0	0	0
RH 18	-5.9	-5.9	-5.9	-5.9	-6.08	-6.8	-14.31	0	3	0
RH 19	7.69	7.69	7.69	7.69	7.69	5.96	-3.65	0	0	0
RH 20	-2.7	-2.7	-2.7	-2.88	-3.06	-4.32	-12.97	0	-7	0
RH 21	-3.07	-3.07	-3.07	-3.25	-3.25	-4.15	-11.73	0	3	0
RH 22	1.14	1.14	1.14	1.14	0.76	-1.52	-11.01	0	7	0
RH 23	0	0	-1.08	-1.26	-1.62	-3.24	-12.25	0	0	0
Average	1.05	1.05	0.97	0.88	0.64	-0.59	-9.71	0	0.44	0

Jupiter NoteBook (*Jupyter Notebook* 2021).

Transit data of logistic routes, master planning data (ATP), and demand fulfilment outputs (nEWs) are the three main ingredients that need to be evaluated. However, this data is stored in various formats on separate databases that have dissimilar keys and ids. Thus, finding proper data and connecting them is not a straight forward analysis. In addition, due to confidentiality of data, we could not access this data within the company's databases. To solve these issues and find the effects of logistic routes on ZD-nEWs we conducted the following steps:

- **Define data requirements**: In this step, we search and extract relevant data. Several meetings were held with experts to find the relevant data (see Appendix A.1 on page 233). Based on this feedback, the outputs of this step were to extract transit data from 'TMA405' and nEWs from 'BO-Universe'.
- **Collect data samples**: After defining the relevant data samples, logistic route data for nEWs for the same planning weeks were extracted with the support of the case study experts. Samples were evaluated to identify whether the specific

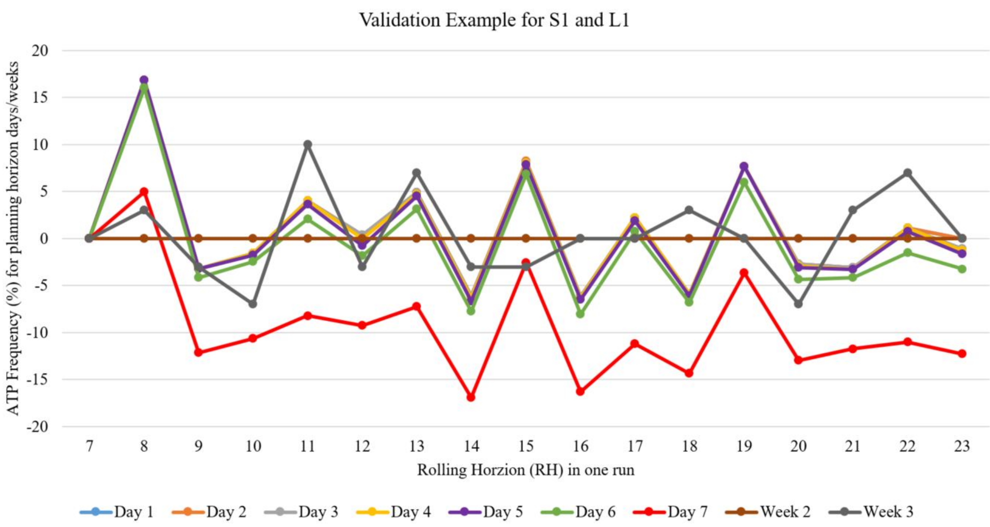


Figure 5.16: Validation outputs for scenario 1 and first logistic route for one replication.

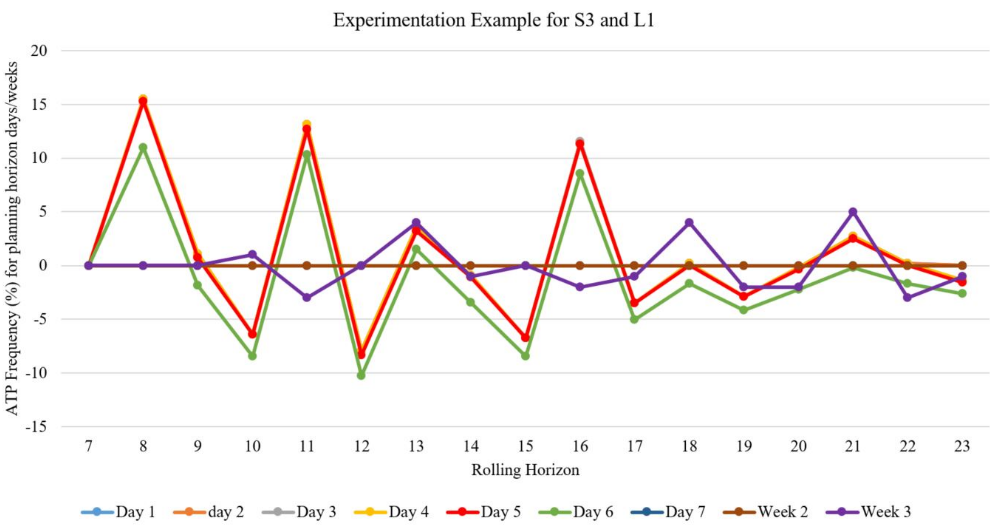


Figure 5.17: Experimentation outputs with TTT=2 day for scenario 3 and first logistic route for one replication.

shipment was related to nEWs.

- **Improve data understanding**: By evaluation of samples, we learned that the two data sets do not have a similar identifier to allow integration. Thus, we searched for other relevant data sets in the case study to connect these two data-sheets. The result of this search was finding the 'TMA 101' which contains both identifiers of 'TMA 405' and 'BO-Universe'.

- **Prepare and collect data**: Based on the three identified data sources, data was collected. As the data is confidential, only some samples were provided by the

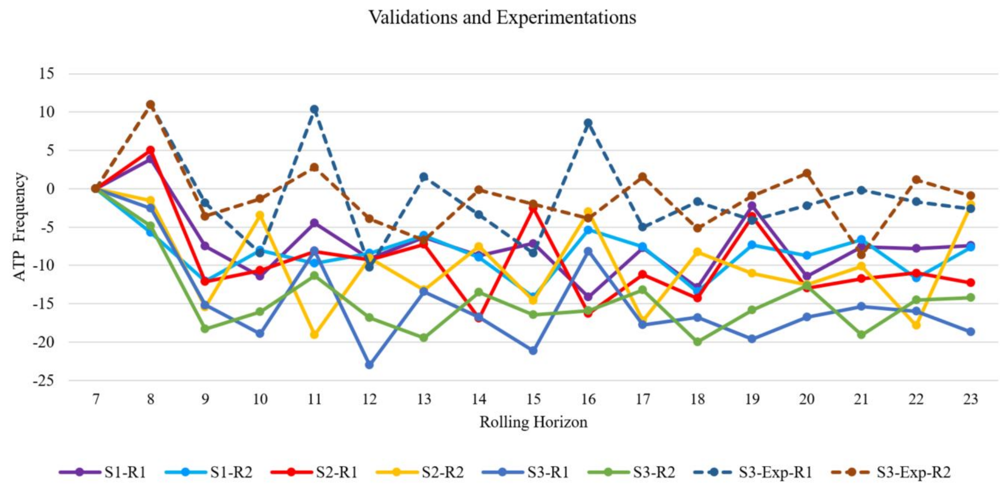


Figure 5.18: Experimentation and validation outputs comparison for Logistic Route 1.

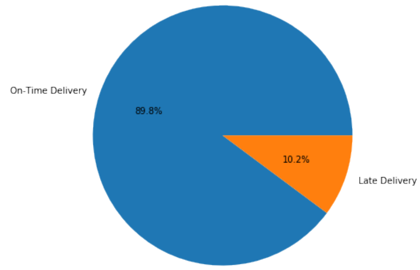
case study to let us develop a data analysis model. Within the extracted samples, we choose CW 33 from 2020 for a specific logistic route.

- **Develop model for analysis**: Based on the final data-sheets we developed a generic model using Jupyter NoteBook. The objective was to find the nEWs and ZD-nEWs that caused $ATT > TTT$ in a logistic route. In addition, the model should be applicable on different case examples within the case study.

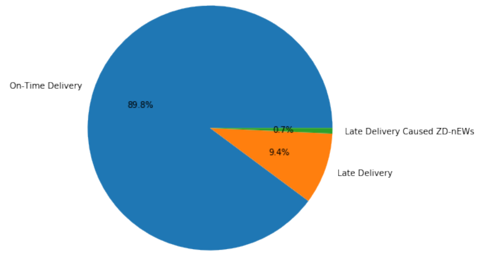
- **Evaluate the results and receive feedback**: The results of the data analysis showed that there is a connection between ZD-nEWs and logistic routes.

Three data sheets are required for analysing the ZD-nEWs that come from late delivery in the logistic route. 'TMA 405' is the transit data that contains ATT and TTT. This data should be filtered for the specific route in the selected period. 'BO-Universe' which shows all nEWs for the global supply chain. Finally 'TMA 101' which contains the SP values of each shipment. As mentioned, the developed model evaluation on sample data and further analysis in other logistic routes will need to be conducted by the case study industry experts due to the confidentiality of data. The results presented in Figures 5.19a on the next page, 5.19b on the following page, 5.19c on the next page, 5.19d on the following page are based on the data analysis where 10.2% of the ATT is bigger than TTT (see Figure 5.19a on the next page). Around one percent of all finished transits in the data caused ZD-nEWs (see Figure 5.19b on the following page) which is equal to 7.3%

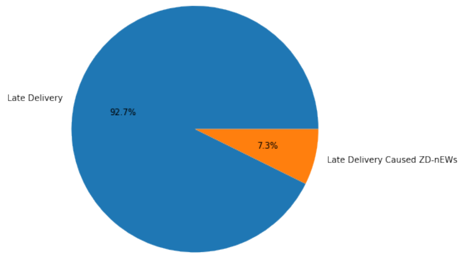
of all the late deliveries by this specific logistic route (see Figure 5.19c). In this sample analysis, all ZD-nEWs caused by the selected logistic route was equal to one percent of all ZD-nEWs (see Figure 5.19d).



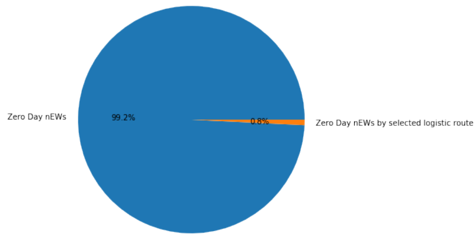
(a) Comparison of deliveries in a selected logistic route On-Time or Late delivery.



(b) Percentage that arrive on-time, late, or late delivery that cause ZD-nEWs.



(c) Percentage of late delivery in TMA that caused Zero Day nEWs.



(d) Percentage of Zero Day nEWs caused by the selected logistic route in TMA data.

Figure 5.19: Results from the selected delivery logistic route.

There are still challenges in data analysis to find the root cause of ZD-nEWs. One of the identifiers that was used to connect the three data-sheets was the SP number of each delivery package. The SP value is unique for sale products, but it could repeat if the selected period of the logistic route was more than 2 weeks. It means that one logistic route might be connected to several ZD-nEWs because there is no unique key in the data to compare. To solve this, we considered checking the quantities of orders. While it filters the result in a better way, but errors are still possible.

5.11 Conclusions

In this chapter, we present the application of a multi-paradigm simulation for the analysis of APS in the case study complex supply chain planning. We analyzed the internal nervousness of a semiconductor SC system using simulation, where we simulated the

the case study SC with a reasonable level of simplification. The simulated model aimed to imitate ATP generation and to test the hypothesis regarding the root causes of nEWs. The outputs of the simulation model supported the decision maker to improve the demand fulfilment and the tuning of the APS parameters.

The proposed conceptual model was a novel paradigm used to model the interactions of the SCP system (information flow) and the production system (material flow). Verification, validation, data analysis, and experimentation of the model were also presented. Moreover, we introduced and discussed the effects of the defining planning parameters on the performance of the whole SC system. Finally, a data science study was discussed and conducted following results from the simulation model.

To evaluate internal nervousness within APS, we examined the effect of ATP instability on the demand fulfilment and its relative control module (Early Warning). We demonstrated the effects of deviation between ATT and TTT that resulted in ATP nervousness. The results state the importance of visibility in the internal interactions of SC networks to reduce instability. The proposed proactive approach for dealing with ATP nervousness and SCP instability opens avenues for further studies that could be conducted on how to optimally tune planning parameters to proactively reduce internal nervousness. This study shows the importance of tuning within the Top model and Base model (see Figure 3.8 on page 52) in a complex implementation of APS.

The simulated APS was confirmed by real data analysis which demonstrated the same behaviour shown in early warning reports. This research explicitly proves the importance of parameter tuning and maintenance of APS internal model. Updating APS's internal model certifies the quality of APS algorithms according to the real and dynamic model of the physical system. In complex semiconductor manufacturing, the life cycle of a product is short. To deal with rapid changes in the physical system, parameter tuning and model maintenance of APS are crucial, and in best practices, many planners adjust the APS regularly (Wiers and Kok 2017). Human planners should be supported by decision support tools like simulation, where planners could be alerted by simulation for the need for APS tuning.

The result of the simulation model does prove the researched hypothesis, but did not provide the percentage of internal nervousness. However, it provides a better understanding of APS interactions to define an efficient data extraction strategy. It supports the decision-makers in dealing with a lack of transparency in APS. Besides, the proposed model provides a baseline for future simulation modeling of complex APS. For instance, further extensions of the model could be used for analyzing the impact of other volatile factors like orders, test new algorithms for APS, or generate data for data that requires advanced analytical methods like artificial intelligence and machine learning.

Finally, the level of abstraction in this work has shortcomings. For example, the model of the physical system in APS is a simplification of reality, where, for example, all priorities of orders were not considered, aspects of APS algorithms were not captured, and it generalizes the shop floor of production sites.

Over and above, the proposed conceptual model captures enough detail and complexity of the planning system to answer the hypothesis. However, it could be reinforced by modeling human planners as new agents. These human agents should interact with the manual part of a planning system like AM-UI in the case study. The effect of human planners on APS is rarely discussed by researchers, but they are essential elements for successful APS. Human planners construct an informal and invisible system that affects a plan stability. We believe this conceptual model could be extended to develop a digital twin of parts or the whole of the planning system.

Chapter 6

Manual Allocation to Mathematical Optimization

6.1 Introduction

Advanced Planning System (APS) improves three factors of competitiveness, namely, cost, quality, and time. In addition, it makes processes more transparent and provides a basis for advanced optimization techniques to solve complex decision problems (Stadtler, Kilger, and Meyr 2015). Product allocation of APS is the topic of this chapter.

Linear Programming (LP) and Mixed-integer Programming (MIP) are powerful optimization techniques and can be applied to APS, such as Master Planning and demand fulfilment. For instance, the match between capacity and supply is discussed in Chapter 3 on Section 3.6.1 on page 67. This process is handled by heuristic rules and manual intervention of planners, however, many of these decisions could be replaced by optimization tools. This possibility within APS is widely neglected within industry and academic research, while the current optimization technologies and solvers could solve the problems optimally. When these mathematical optimization techniques are incorporated into a computer application, they can support decisions within APS (Williams 2013).

Although research identifies mathematical optimization as one of the key parts of

APS, it has been shown in Chapter 3 that planning in the case study primarily uses rule-based heuristic processes. In addition, the the case study planning system also uses auxiliary software and human planners. Examining demand fulfilment and order management, the case study has two elements within order management. First, the case study plans demand fulfilment within i2DF in an automatic manner. Second, following from the output from i2DF human planners implement these plans where there are many other decisions required (see Section 3.8 on page 83). In specific but common situations, these planners may have to allocate products to customers completely manually. In conclusion, the allocation of products to customers in the case study is handled by a mix of rule-based algorithms and human decisions.

Allocation planning in the case study is not limited to product allocation to customers. There are two allocation types, product allocation and customer allocation, with each based on the decision hierarchy (see Section 3.8 on page 83). Demand Fulfilment (DF) as a part of APS was introduced which provides information on orders and order updates, which is connected with other software modules of APS that set due dates, provide promises, and monitor orders for delivery. The more reliable the dates and quantities planned by means of DF and Order Promising (OP) are, the more reliable and satisfied the customers will be. To handle these within manual allocations, various allocation types should be conducted within APS. For example, allocation of products to different sales offices and allocation of products to customers. As discussed in Section3.8 on page 83, for planning allocations, the roles are defined within the hierarchical planning system.

Where demand outstrips supply, this will result in shortages to end customers. Or when customers change the requested date or quantities, a modification will be required for plans. In such a case, decisions need to be made on how to allocate supply to customers. Thus, allocation maintenance by human planners is required. Customer satisfaction requires accurate order promising that leads to better cooperation, as well as trustable orders and forecasts from customers. As a result, customer satisfaction through a trustable promising system leads to more accurate planning for production.

In this regard, APS in the case study provides allocation planning to customers' orders based on "Available To Promise" (ATP). Lack of supply, escalation, and excess demand is propelled by competitive plant capacity, dynamic behaviour of ATP, orders, and demand forecasts in demanding industries like semiconductor manufacturing. When demand exceeds supply, APS needs the support of experts (human intervention) about the time and amount to be allocated to customers. In the case study planning terminology, it is called an Allocation Situation (AS). This feature of APS keeps the flexibility of planning to find feasible optimal decisions regarding allocations.

In this chapter, we propose a mathematical model for the optimization of ATP allocation to customers in AS, where demand exceeds supply. The mathematical model aims to support human planners to define allocation quantities in a faster and better way. Thus, the mathematical model is used as a core of a decision support tool to analyze allocation scenarios. The objective of the proposed mathematical model aims to maximize the customer service level, which is directly related to customer satisfaction while keeping a maximum of stock. We report that the obtained results from the mathematical model were validated using real data from the case study company. The developed decision support tool is called Regional Customer Allocation Support Tool (ReCAST). ReCAST was developed as a prototype and deployed in the case study and is reported in the next chapter.

6.2 Problem statement

Demand and supply volatility's cause instabilities within plans. As discussed in Chapters 4 and 5 these instabilities need to be dampened by improving and maintaining APS. To deal with instabilities within DF, a module in the Order Management system called Allocation is used (see Figure 3.18 on page 74). The aim of this module is to maintain allocation situations that could not be handled by i2DF, which requires manual intervention. The related software is called AM-UI which is the case study custom specific software. Therefore, when ATP generates data, it is sent to both AM-UI and i2DF. AM-UI has a priority over i2DF. Thus, what is consumed from ATP in AM-UI will be denoted as AATP (Allocated Available To Promise) and taken from ATP, with the remaining ATP used in i2DF.

There are two types of allocation within the Allocation module, namely, Customer Allocation and Product Allocation. Both of these are manual allocations. Customer Logistic Management (CLM) is part of the case study order management system. It aims to develop and manage customer allocations. When demand exceeds supply, a flag is raised by the Allocation Manager (ALM) which means an allocation task is assigned to a Zonal Logistic Planner (ZLP). ZLPs allocate the product in AS to customers. Customer Allocation planning is the area that we focus on in this chapter.

Dealing with tight allocations is a daily business of ZLPs. These processes consist of the allocation of one product to many customers for a planning horizon of approximately a year on a weekly basis. For instance, if a product needs to be allocated between three customers for forty-five weeks, it means that the ZLP should identify the allocation amount (or at least check), for example, $3 \cdot 45 = 135$ weeks. For this manual planning task, they need to check a customers' history of demand, follow the case study's planning strategies according to products on allocation, check the level of ATP for the whole planning horizon, and consider the level of stock and customer orders. In all of these cases, an ZLP should consider customers equally and modify the allocation strategies according to the business situation of the product under allocation. These decision makers require a high level of experience and knowledge which requires a highly skilled human resource.

Based on investigations by the case study project manager and held meetings with planners (see Appendix A.1 on page 233) within the CLM pools in the case study, a rough estimation (that was not scientifically rigorous) that showed each ZLP deals with customer allocation more than 80 times per month. This manual allocation takes around 30 minutes for an experienced ZLP to be finalized in AM-UI. In the global Supply Chain (SC) of the case study there are 29 ZLPs which handle customer allocation based on the region of the CLM pool. Consequently, more than 1160 hours of the case study human resources are spent on customer allocation per month. Note that in this estimation, we gather data from experts and underestimate the values to have a minimum. It could be implicitly concluded that for each of these allocations there could be other hidden costs within this manual allocation. For example, costs of teaching and experimenting manual allocation for the ZLPs; or cost of intractability or human errors within the processes.

Through meetings with ALMs, ZLPs, business process owners, IT engineers, and SC planning experts, we identified this challenging allocation planning problem carried out using ZLPs. This dependency of APS to planners is necessary, but the way they need to interact manually makes the allocation planning challenging and intractable problems which we aim to solve by automation of allocation planning through a mathematical model and development of a decision support tool. Within this investigation, we developed a mixed-integer programming model to allocate products to customers as discussed in this chapter. The developed mathematical model is used as a core of a decision support tool described in Chapter 7. This decision support tool still needs planners insights, but it diminishes their effort for allocation and improves the quality of the allocation plan. In the next section, we discuss the developed mathematical model and discuss its performance through validation using real case data and describe sensitivity analysis.

6.3 Allocation Situation Modeling

Allocation planning of the case study's APS is performed in a hierarchical system in which products or ATP are assigned to nodes and leaves of the hierarchical tree (see Section 3.3.3 on page 48). Leaves represent individual customers and upper nodes describe the aggregation of customers based on different criteria such as regions, as described by Vogel and Meyr (2015) and Cano-Belmán and Meyr (2019). Therefore, the allocation should cover nodes in addition to leaves (customers). In such an environment, the ATP allocated to higher levels (i.e., upper nodes) defines the amount of ATP for each region. The allocated ATP in each node is used to satisfy the customers in the region. The ReCAST mathematical model is designed for allocating leaf sellers in CLM pools based on regions.

The model formulation was developed to allocate limited supplies to customers in multi-stage allocation planning. The allocation plan will perform this within a time horizon. In fact, it should be noted that this model was developed in the context of improving customer service level within demand fulfilment. During interviews with experts, business owners, and allocation managers in the order management department, it was revealed that profit is not considered at this level, but products are allocated to improve customer satisfaction. Therefore, profit is considered at a higher level in the planning hierarchy in product allocation. Thus, in an allocation situation executed by ZLPs the aim is to satisfy customers and control stock level. Therefore, the ZLPs consider the following assumptions within their allocation plans:

- Promised date (CMAD) and quantities.
- Orders could split and satisfy through the allocation horizon.
- Contract clauses of each customer.
- Level of buffer stock that should be kept from ATP weekly.
- Use buffer stock based on business situation.

As discussed, one of the advantages of APS is the availability of data for decision making. AM-UI creates data from an allocation plan which contains ATP, and a list of orders for a selected planning horizon, usually a year in weekly time frames. Based on the available information and gained knowledge, we proposed a mathematical model. The model aims to develop an allocation plan in the allocation situation. The Mixed Integer Programming (MIP) model aims to increase the customer service level of allocation plans under the previously presented constraints. The total supply (ATP) and customer demand are deterministic and known at the beginning of each planning horizon. Given the purpose of the model, these are the parameters and decision variables on which the model is based:

1. Indices
    - i: customers
    - τ: confirmed time

- t: time

2. Parameters

- $O_{i,\tau}$: Order is the quantity requested by customer $i = 1, ..., I$, confirmed at time $\tau = 1, ..., T$. $O_{i,\tau}$ which is known for the whole planning horizon.
- ATP_t: The Available To Promise (ATP) is known for the whole planning horizon and is equal to ATP_t at each $t = 1, ..., T$. The total demand at time t is usually more than ATP_t.
- BS_{min_t}: ZLPs specifies the minimum of the cumulative buffer stock, BS_{min_t} which is an input parameter for the model.
- IBS: Initial Buffer Stock is the level of buffer at the beginning of selected planning horizon.
- $X_{i,t}$: ZLPs could use buffer stock if customer i follows the lead time for the order at time τ. It is a binary input.
- RB_t: The ZLPs can add more to the buffer stock from ATP_t, which is called Reserved Buffer Stock RB_t. The value of of RB_t is also a parameter predefined for each time t and the ZLPs try to reach this level and keep as much as possible from ATP_t, that is not used.
- $MaxDelay$; The number of weeks that an order could be delayed.

3. Decision variables

- $AQ_{i,\tau,t}$: Allocated or promised quantity to customer i in time t referred to an order previously confirmed at time τ. This quantity is consumed from ATP_t.
- $AS_{i,\tau,t}$: Allocated or promised quantity to customer i in time t referred to an order previously confirmed at time τ. This quantity is consumed from buffer stock. ZLPs are allowed to consume from buffer stock to satisfy the order.
- AR_t: Allocated quantity from the reserve buffer based on the ZLP goal and the ATP data.

- BS_t: Buffer stock level for each week.
- Z_t: Allowance of using from stock for orders in time t. This is an binary variable.

Based on the indices, parameters, and decision variables, the mathematical formulation is as follow.

$$\text{Maximize} = (f_1, f_2) \tag{6.1}$$

$$f_1 = \sum_i \sum_\tau \left[\sum_{t=\tau} AQ_{i,\tau,t} + \sum_{t\neq\tau} (AQ_{i,\tau,t} \times P) \right] + \sum_i \sum_\tau \left[\sum_t AS_{i,\tau,t} \right] \tag{6.2}$$

$$f_2 = \sum_t AR_t \tag{6.3}$$

Subject to:

$$\sum_t AQ_{i,\tau,t} + \sum_t AS_{i,\tau,t} \leq O_{i,\tau} \qquad \forall i \in I, \tau \in T \tag{6.4}$$

$$BS_t = IBS \qquad for \;\; t = 1 \tag{6.5}$$

$$BS_t = BS_{t-1} + ATP_{t-1} - \sum_i \sum_\tau AQ_{i,\tau,t-1} - \sum_i \sum_\tau AS_{i,\tau,t-1} \qquad \forall t = 2, 3, ..., T \tag{6.6}$$

$$\sum_i \sum_\tau AQ_{i,\tau,t} + AR_t \leq ATP_t \qquad \forall t \in T \tag{6.7}$$

$$\sum_i \sum_\tau AS_{i,\tau,t} \leq BS_t \qquad \forall t \in T \tag{6.8}$$

$$AR_t \leq RB_t \qquad \forall t \in T \tag{6.9}$$

$$BS_t \geq BS_{min_t} \qquad \forall t \in T \tag{6.10}$$

$$\sum_{\tau} AS_{i,\tau,t} = \sum_{t} AS_{i,\tau,t} \qquad \forall t \in T, \;\; i \in I, \;\; \tau \in T \tag{6.11}$$

$$Z_t \leq \left(\frac{\sum_i \sum_\tau AQ_{i,\tau,t}}{ATP_t} \right) + eps \qquad \forall t \in T \tag{6.12}$$

$$AS_{i,\tau,t} \leq bigM \times X_{i,t} \times Z_t \qquad \forall t \in T, \; i \in I, \; \tau \in T \tag{6.13}$$

$$P = 1 - \frac{|t - \tau|}{MaxDelay} \tag{6.14}$$

The objective function in Equation 6.1 represents the maximization of two different objectives f_1 and f_2. These two functions are in contradiction. Within the developed decision support tool, ZLPs could apply various scenarios according to their business goals. These scenarios are applied through weights used to balance these two functions. These weights support ZLPs to manage trade-offs between two competing goals. The first objective function in Equation 6.2 represents the maximization of customer service level as it is the sum of the allocated quantity where the allocated quantity when $t \neq \tau$ is penalized. The penalty is calculated in Equation 6.14. As far as the promising date moves forward in time, the model decreases the value of the promised quantity based on this distance. In fact, the model tries to satisfy the demand closer to τ (previously confirmed time to the customer). The second part will be active only when the customer follows the lead time condition and ATP_t could not cover the requested quantity. Therefore, ZLPs could use the buffer stock. The second objective function Equation 6.3 models the reserved quantities that the allocation manager tries to keep by defining this goal. In the following, the model's constraints listed above are discussed.

The first constraint Equation 6.4 refers to the total allocation from ATP or stock that must be less than the requested orders. Note that an order could split into several parts based on the maximum delay given in Equation 6.14. Maximum delay or delivery window is a term in the customer's contract within the case study. Equations 6.5 and 6.6 aim to control the level of buffer stock. They refer to the calculation of buffer stock at time t based on the level of buffer stock at $t-1$ and the amount that is consumed by ATP and buffer stock. Besides, the total allocated quantity and reserve from ATP and buffer stock at time t should be less than the available to promise and buffer stock which is modeled in Equations 6.7 and 6.8. Equation 6.9 is added to control the level of reserve buffer stock based on the ZLPs' inputs. The buffer stock variable also should be bigger than the minimum values which are defined by the ZLPs in Equation 6.10. Note that this number is different from Reserved Buffer stock that the allocation manager needs to add to the buffer stock at time $t+1$. Equation 6.11 describes the allocation from stock that should happen when $t=\tau$. Equation 6.12 refers to a binary variable Z_t. This variable becomes one when all ATP is consumed by $AQ_{i,\tau,t}$. This binary variable controls the next Equation 6.13 to force the use from stock when the ATP is finished and there still are available orders. The input binary parameter input by the ZLPs, $X_{i,t}$, is also multiplied in this equation to consider the case study strategies and a planner's goal.

The above-mentioned model gradually evolved through meetings and testing. First, we developed the initial idea in sets of meetings with SC experts and ZLPs. In this step, we were required to understand what are the decision variables and constraints that ZLP considers implicitly within their cognitive decisions. In addition, what are the case study strategic benchmarks and guidelines required by ZLP to consider. Thus, we developed meetings which are available in Appendix A.1 on page 233.

Second, we investigated further the available data. The output of this investigation of the APS available data led to a better understanding of requirements. During a series of meetings, we clarified how ZLPs uses this data for their calculation. Note that decision making between planners may vary based on their understanding of the allocation situation, which makes the understanding of processes challenging.

Third, based on the meeting's outputs, available data, and questionnaire answers, we developed the initial model. In the initial development phase, the MIP model was programmed using the YALMIP toolbox in MATLAB (see Appendix A.8.1 on page 293) with real data from the case study applied to this model. The used solver was GUROBI 9.0 -academic version. The model's output was discussed with the allocation business owners and ZLPs which lead to improvements in the mathematical model and a quality of improvement in the allocation plans developed.

During debugging and testing of the initial model in the fourth iteration, we found a missing variable in the mathematical model. Adding this new variable was considered implicitly by ZLPs. This variable is AR_t which changes the shape of the model. This decision variable was examined by ZLPs. After they certify the usefulness of the addition of this new decision variable, we modified the model. This new variable allows us to divide the total allocation to each customer to two decision variables $AQ_{i,\tau,t}$ and $AS_{i,\tau,t}$ which are the allocated quantities from ATP and stock respectively. Previously, an ZLP considered these two calculations in one output from the model.

Fifth, we tested the new data in the new model. When the quality of the solutions was certified by ZLPs, the capabilities of the model were proved to SC experts. Thus, we moved to develop a decision support tool in a web framework. Thus, for further improvements of the model, we moved the model to Python (see Appendix A.8.2 on page 299). In the next chapter, we discuss the developed tool (ReCAST). In the rest of this chapter, we discuss the outputs and analysis of the developed MIP mathematical model, which is the basis of the ReCAST tool.

6.4 Results and experimentation

Within the modeling phase of the project, we extracted five real cases of allocations. This data set were used within the development of the model and further testing on new data was performed by ZLPs data which will be discussed in the next chapter. Here we aim to examine the mathematical model using real data from the case study company which is presented in Table 6.1 on page 173.

In this table, the planning horizon is from $CW1$ to $CW49$ (CWi Current Week i) and there are two customers ($C1$ and $C2$). Reserve buffer stock which is an important goal of ZLPs to obtain from stock, is denoted by RB_t . We consider this value constant in this example, but ZLPs can modify it based on their understanding of business situations. The Initial Buffer Stock (IBS) in this example is 535500. Another parameter defined by ZLP is the minimum buffer stock, denoted by BS_{min_t}. These values are considered constant in this example based on ZLPs ideas, but in real examples ZLPs can modify them based on their requirements. X C1 and X C2 are representative of the $X_{i,t}$ parameter which define whether the customers are allowed to use from stock ($X_{i,t} = 1$) or not ($X_{i,t} = 0$). In this example, all customers are allowed to use stock. Similarly to the previous parameters, ZLPs can modify these values.

As mentioned, the proposed mathematical model is multi-objective. To solve it, we used the weighted sum approach where the objective functions are aggregated by multiplying them with weights and summing over them. The sum of weights is equal to one. The weights are the inputs from planners. In this example, we used 11 combinations of weights:

[[1,0],[0.9,0.1],[0.8,0.2],[0.7,0.3],[0.6,0.4],[0.5,0.5],[0.4,0.6],[0.3,0.7],[0.2,0.8],[0.1,0.9],[0,1]]

Another parameter that needs to be defined by ZLP is the packing unit which relates to the product type on allocation. In this example, the packing unit is 500. The last input parameter is $MaxDelay$. This parameter identifies how many weeks the model is allowed to postpone and split on an order. Max delay is used in the calculation of the penalty function in the mathematical model.

The different combination of weights between f_1 and f_2 are presented in Table 6.2 on page 174 and Figure 6.1 on page 175. Applied examples contain two customers with 49 weeks of a planning horizon. Note that both functions are similar in units thus the summation of weighted objective functions would not cause any issue.

Table 6.2 on page 174 shows 11 scenarios that start from high to low weights for allocation to customers. When the weight value is one, it shows that the model prefers to allocate to customers first rather than keeping reserve goals. In this table, the total

Table 6.1: Sample data for optimization model.

	Week	ATP	Order C1	Order C2	RB_t	BS_{min_t}	X C1	X C2
1	CW1	101500	60000	68000	100000	400000	1	1
2	CW2	0	50000	68000	100000	400000	1	1
3	CW3	115500	0	68000	100000	400000	1	1
4	CW4	163500	0	0	100000	400000	1	1
5	CW5	191000	0	83000	100000	400000	1	1
6	CW6	138000	195000	183000	100000	400000	1	1
7	CW7	150000	50000	83000	100000	400000	1	1
8	CW8	154000	50000	0	100000	400000	1	1
9	CW9	128000	45000	80000	100000	400000	1	1
10	CW10	150000	0	180000	100000	400000	1	1
11	CW11	93500	0	0	100000	400000	1	1
12	CW12	146500	155000	0	100000	400000	1	1
13	CW13	140000	50000	70000	100000	400000	1	1
14	CW14	150000	15000	70000	100000	400000	1	1
15	CW15	150000	0	0	100000	400000	1	1
16	CW16	150000	135000	0	100000	400000	1	1
17	CW17	150000	50000	70000	100000	400000	1	1
18	CW18	150000	0	80000	100000	400000	1	1
19	CW19	150000	0	80000	100000	400000	1	1
20	CW20	150000	85000	80000	100000	400000	1	1
21	CW21	150000	35000	80000	100000	400000	1	1
22	CW22	150000	0	80000	100000	400000	1	1
23	CW23	150000	0	80000	100000	400000	1	1
24	CW24	150000	0	0	100000	400000	1	1
25	CW25	150000	0	80000	100000	400000	1	1
26	CW26	150000	170000	80000	100000	400000	1	1
27	CW27	163500	0	60000	100000	400000	1	1
28	CW28	150000	0	130000	100000	400000	1	1
29	CW29	136500	149000	40000	100000	400000	1	1
30	CW30	150000	0	30000	100000	400000	1	1
31	CW31	152500	0	30000	100000	400000	1	1
32	CW32	150000	0	70000	100000	400000	1	1
33	CW33	147500	149000	70000	100000	400000	1	1
34	CW34	150000	0	104000	100000	400000	1	1
35	CW35	150000	0	100000	100000	400000	1	1
36	CW36	150000	0	70000	100000	400000	1	1
37	CW37	150000	149000	60000	100000	400000	1	1
38	CW38	150000	0	80000	100000	400000	1	1
39	CW39	150000	0	80000	100000	400000	1	1
40	CW40	150000	0	76000	100000	400000	1	1
41	CW41	163500	0	76000	100000	400000	1	1
42	CW42	136500	149000	76000	100000	400000	1	1
43	CW43	150000	0	76000	100000	400000	1	1
44	CW44	150000	0	76000	100000	400000	1	1
45	CW45	162500	0	0	100000	400000	1	1
46	CW46	137500	149000	0	100000	400000	1	1
47	CW47	159500	0	0	100000	400000	1	1
48	CW48	150000	0	0	100000	400000	1	1
49	CW49	150000	0	0	100000	400000	1	1

Table 6.2: Comparison of results using different objectives' weight values.

No.	Senarios (f_1 Weights)	1	0.9	0.8	0.7	0.6	0.5	0.4	0.3	0.2	0.1	0
1	Summation of ATP	7081000	7081000	7081000	7081000	7081000	7081000	7081000	7081000	7081000	7081000	7081000
2	Summation of Orders	4925000	4925000	4925000	4925000	4925000	4925000	4925000	4925000	4925000	4925000	4925000
3	Allocated from ATP (AQ)	4176000	3878500	3554000	3315000	3210500	3171500	3029000	2829000	2287500	2287500	0
4	Allocated from Stock (AS)	749000	1046500	1371000	1610000	1714500	1753500	1749000	1442500	118000	118000	0
5	Customer Allocation Objective	9848	8788.68	7678.8	6594.49	5568.48	4613.1	3496.52	2383.02	926	463	0
6	Reserve Stock Objective	0	582	1354.2	2178	3003.6	3793.5	4792.2	5870.9	7669.6	8628.3	9587
7	Summation of Objectives	9848	9370.68	9033	8772.49	8572.08	8406.6	8288.72	8253.92	8595.6	9091.3	9587

available ATP and total requested order are also identified. Summation of allocation from ATP and from Stock define the allocation quantities which will be shipped to customers each week. Finally, the optimal values of each objective are the sum of rows 5 and 6 that gives row 7. These values are shown in Figure 6.1. This figure shows the relation between the objective function values in different scenarios based on several weight combinations. When the weight of f_1 is equal to zero, the model prefers to keep all ATP as a reserve buffer stock and nothing is allocated to customers. When the weight of f_1 is less than 50%, the two objective functions become equal. This example was tested based on input parameters that show the effects of the multi-objective on the final solutions.

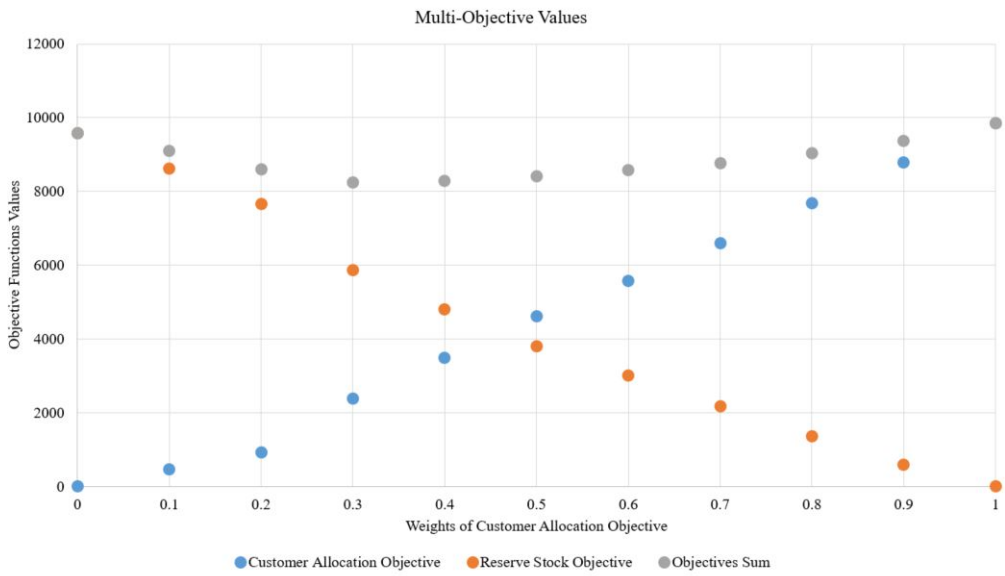
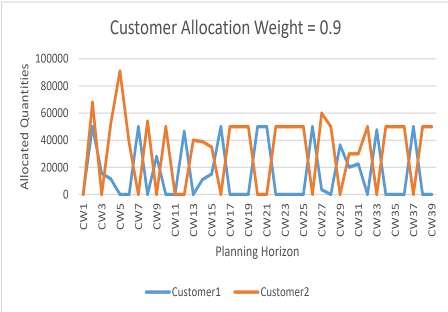


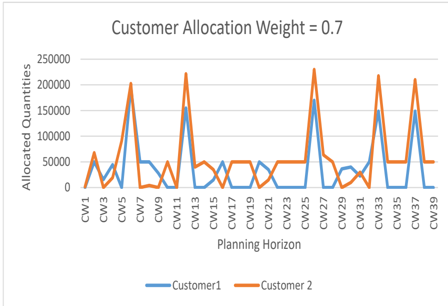
Figure 6.1: Comparison of multi-objective values between scenarios.

The model was tested on various scenarios that demonstrated the competency of the model and how it depends on the inputs of ZLPs. In other words, the model can develop a solution based on a business situation rather than a ZLP's preference. Figure 6.2 on the next page visualizes the allocation plan for four selected scenarios with a selected planning horizon of 39 weeks shown by CW (Current Week). Figure 6.2a show the quantities allocated to each customer every week. It shows the allocation plan when the customer weight is 0.9 and the stock weight is 0.1. The allocated quantities to customers one and two at week 17 are equal to 0 and 50000. The comparison between these figures shows the effect of weighted objectives on the allocated quantities to each customer.

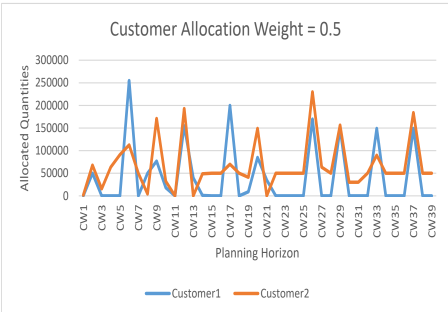
Comparing Figure 6.2a with Figure 6.2c, shows the model with 0.9 customer allocation weight allocates more quantities to customer two between weeks 1 and 5. However, in Figure 6.2c a major part of the allocation to customer one occurs in week 6..



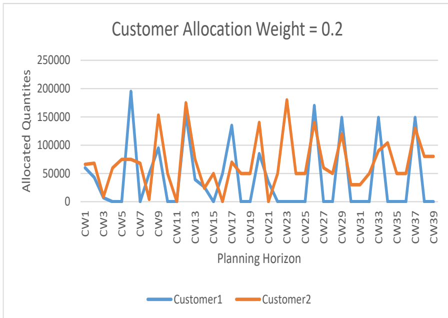
(a) Customer Allocation weight = 0.9.



(b) Customer Allocation weight = 0.7.



(c) Customer Allocation weight = 0.5.



(d) Customer Allocation weight = 0.2.

Figure 6.2: Comparison of allocation plans based on different scenarios.

6.5 Sensitivity Analysis

The presented mathematical model solves the allocation problem and provides exact optimal solutions. However, it is possible in real cases that specific business conditions need to be considered. For instance, a customer requests for a specific quantity in a critical week for its business. To solve this exception, ZLPs might modify one or two specific weeks. In these situations, ZLPs require a mathematical model to calculate the optimal plans but might need to modify the optimal solution for a small number of weeks (possibly one or two). For these modifications, ZLPs usually need to use product from stock. To keep the optimal solution and modification flexible, we proposed a heuristic approach for analyzing the effects of using from stock or not for ZLPs.

To provide this analytical tool for ZLPs, we performed a sensitivity analysis on the model. Since the model is MIP, duality theory could not be applied. Thus, we could not calculate the dual values using solvers. To address this issue, we fixed the binary variable of the optimal solutions to change the model to LP. Then we change the value of the binary variables one by one and calculate the gradient of the objective function. By comparing the value of the binary fixed objective with the global optimal (normal objective), we could see the effects of the change in a plan. The concept of the heuristic algorithm for this sensitivity analysis is similar to duality since it aims to calculate the cost of change in one of the variables. The pseudocode of the sensitivity analysis is presented in Algorithm 1. This is similar to calculating the dual variable of binary constraints. The plot of this analysis is presented in Figure 6.3 on the next page.

Algorithm 1 Sensitivity Analysis.

Z_t^* is the optimal solutions of binary variable.
$f_1^* + f_2^*$ is the optimal value of objective functions.
Constraints 6.12 on page 169 and 6.13 on page 169 are the binary-constraints related to binary variable Z_t
for All scenario weights i **do**
    for All the planning horizon j **do**
        if Z_j^* in secenario i is equal to 0 **then**
            change the value to 1
            set the changed variable as a parameter
        else if Z_j^* in secenario i is equal to 1 **then**
            change the value to 0
            set the changed variable as a parameter
        end if
        Change the binary-constraints by replacing the variable Z_j with identified
        parameter in If statement above.
        Optimize the model with the new binary-constraints.
        Calculate the differences between the new optimal value with new binary
        -constraint and the global optimal $((f_{1,bin-fixed}^* + f_{2,bin-fixed}^*) - (f_1^* + f_2^*))$.
    end for
    Calculate these differences for all scenarios
    Plot the normalized differences for scenarios, in whole planning horizon
end for

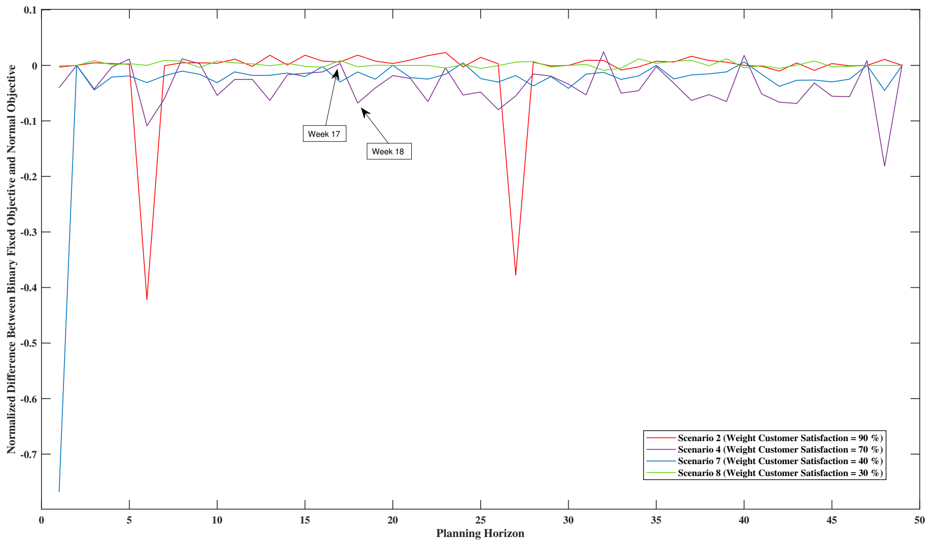


Figure 6.3: Sensitivity analysis of using from stock or not.

The result of this sensitivity analysis presents a plot for ZLPs to be used as a benchmark for decision making regarding the modification of using from stock or not. As ZLPs are only allowed to use product from stock when ATP is consumed, this analysis can support them in understanding if they need to modify the optimal solution of ReCAST. This plot supports them to compare the impact of using or not using from stock in a specific week to select the most suitable weeks for changing the plan of using from stock. As an example, output from the use of Algorithm 1 on the preceding page which is presented in Figure 6.3 shows the sensitivity analysis of using from stock for 49 weeks of a planning horizon (49 weeks) in four scenarios, where different customer weights are used. Each point in this figure represents one run by changing Z_t^* in that week from the optimal binary variable to its opposite value (see Algorithm 1 on the preceding page). The y axis of this plot shows the normalization of this difference and the negative values show how much the new model loses in comparison to the optimal model. For instance, the optimal value of the binary variable in week 17 and 18 for Scenario 4 are both zero which means the model does not use from stock in these weeks. However, if the ZLPs wants to use from stock in this week and now wants to decide which week is the most suitable, then week 17 is recommended as it has a lower consequence on the optimal value. This means that using from stock in week 17 causes a lower lost in the optimal value of the function. Note that this feature was not applied in the final prototype of ReCAST, which is presented in the next chapter.

6.6 Conclusion

APS can be improved by applying decentralized mathematical operation research methods. It is felt that planning systems will always need human planners and applied mathematics can improve their decision making. In advanced planning systems, the main functionalities are handled centrally. The central system cannot be replaced fully by mathematical optimization as some researchers suggest. However, the availability of data from APS can support decentralized mathematical optimization as is shown here. In this chapter, we discuss the proposed mathematical model for customer allocation planning in APS. The model is a multi-objective MIP model which was tested on various real data from industry which showed its competency and capabilities. The chapter also presented further investigation regarding possible sensitivity analysis conducted on the model to better support decision makers. This model is used as the core of the decision support tool described in the next chapter.

Chapter 7

Regional Customer Allocation Support Tool (ReCAST)

7.1 Introduction

This chapter will present the details of a prototype web-based decision support tool developed using the mathematical model presented in the previous chapter, demonstrating its applicability as a decentralized decision tool. Planners may not be familiar with mathematical optimization, therefore we aimed to develop ReCAST as an easy access and user-friendly tool for planners to be implemented using the case study Advanced Planning System (APS). The motivation of this project is to improve allocation planning by replacing manual calculations with an optimized support tool. Note that this tool will not replace the role of a planner, but change their role from the need to do manual allocation to optimally allocating based on the objective function and following the case study current business goals. This project and the developed tool is named the Regional Customer Allocation Support Tool (ReCAST).

In this chapter, we aim to discuss the steps and processes of designing and developing ReCAST. These include:

- **Requirement Analysis:** to define the basic structure of the tool.
- **Design:** this will describe the steps involved in the design of the web-based tool.

- **ReCAST:** the steps involved in the development of this tool.
- **Test and Debugging:** to define the errors and test applications using real case data obtained from Zonal Logistic Planner (ZLP).
- **Deployment and Delivery:** the handover of the ReCAST tool to the case study company with relevant documentation.

Finally, in this chapter we provide conclusions on the development of the ReCAST tool for digitization of allocation planning and how it integrates with APS in this complex planning of a supply chain.

7.2 Requirement Analysis and Design

We carried out a requirements analysis for the ReCAST tool. In the requirements specification (see Appendix A.7 on page 256), we focused on the analysis of a user type, user's expected tasks, user interfaces, user interactions, and system performance. The aim of requirement analysis is to define the structure and scope of the project. These specifications were used to build a simple prototype for meeting (see Appendix A.1 on page 233) with stakeholders for validation of their requirements. Project managers, ZLPs, developers, and Supply Chain (SC) experts provided feedback during these meetings, which supported us in modifying some of the user interaction operations and business processes based on the previous prototypes we presented (see Appendix A.7 on page 256). The three main objectives of the ReCAST prototype are:

- The aim is to obtain the optimal customer allocation to be used within AM-UI as benchmarks by the ZLPs during allocation planning.
- The generated optimal solutions by ReCAST are based on the ZLP input configurations. These configurations come from the insights that ZLPs have or gain from marketing, product types, business situations, etc. Thus, ReCAST should receive these configurations from the end user.

- ReCAST should have a convenient and easy-to-use design. ZLPs are required to manually import data from a local Excel file, add configuration data, run the mathematical optimization model, select the proper scenario from the generated mathematical solutions, and finally export from ReCAST the allocation plan to AM-UI as a structured Excel file.

The process flow of ReCAST is described in Figure 7.1 on the facing page. This figure shows that when a Target Allocation flag is raised, the ZLPs are notified that an allocation situation is required and they need to develop a new allocation plan. Allocation plans are manually added using the AM-UI module. When they feel that the allocation situation is complex enough that they need ReCAST, they export the allocation situation to an Excel file. The exported file is then imported to ReCAST. ReCAST reads the data and parses them to a compatible format to be used in the mathematical model. For this, it needs the inputs from ZLPs which are the selection of planning horizon, their strategies by identifying the weights of scenarios, delivery windows, packing unit of product on allocation, minimum buffer quantity, and their goal of keeping ATP as a reserve buffer stock. After entering inputs, ZLPs runs ReCAST. The model displays scenarios in a new page using visualization tools. These visualizations are aimed to facilitate the process of scenario selection for ZLPs. After selecting the most relevant scenario, the ZLPs exports the results. Now another parser transfers the results to an Excel file which is compatible with AM-UI. The user then needs to import the result from ReCAST into the AM-UI tool.

7.2.1 User groups

In ReCAST, the primary user is a ZLP. In addition, an Allocation Logistic Manager (ALM), a Customer Logistic Manager (CLM), or a Supply Planner may use ReCAST. These users would be defined as high-level users in ReCAST, and this may need to be taken into consideration in the design of the tool. Currently, we have successfully validated the business logic for ZLP. As the software is designed for internal use, there is not a requirement for a high level performance because the number of such users is low. Users of ReCAST have also stated that data stored beyond three months could be erased,

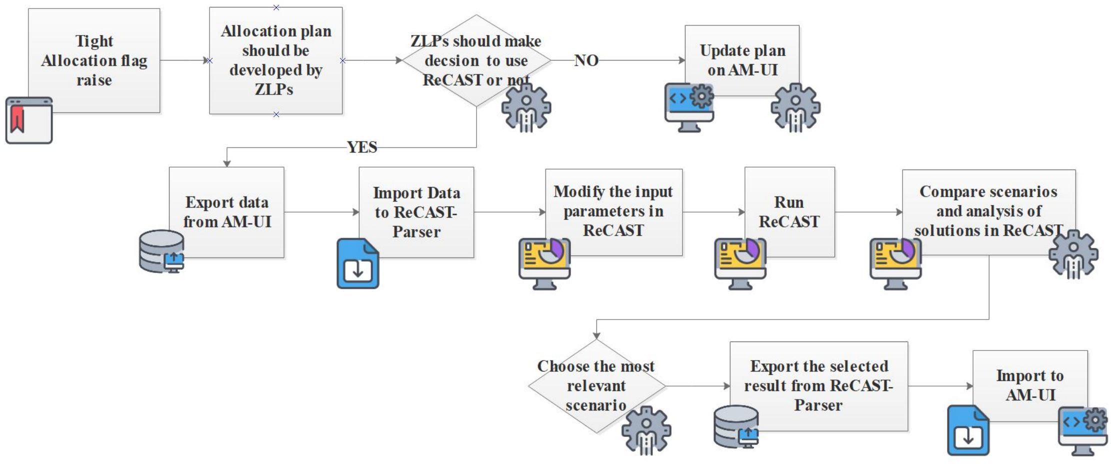


Figure 7.1: Process flow of using ReCAST in the case study APS.

as the planning horizon moves and historical data will not be required.

There are three states for an ZLP when they are using ReCASTS (see Figure 7.2):

- State A: not logged in state: At Login page, Register page, Forget Password.
- State B: logged in state: At Index page, Initial Scenario page (Import Data from Excel file), Task Information page. Including All pages at state C.
- State C: working state: At Config Optimal Model page, ReCAST Analysis Result page, Scenario Modification page.

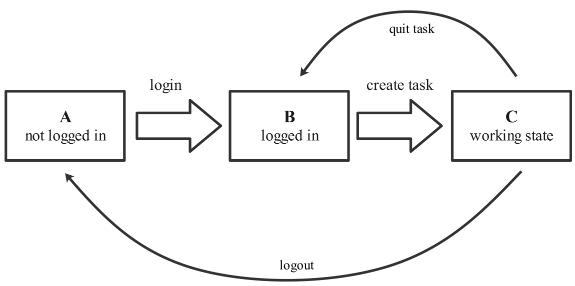


Figure 7.2: ReCAST user's states and transits.

The transits between states are:

- A to B: From not logged in state to logged in state, by triggering a login event.
- B to C: By creating a task for running ReCAST. A task can be considered as a business logic flow for generating allocation plan scenarios.

- C to B: By quitting the task or finishing the task, the ZLP is currently doing.
- C to A: By logout.

7.2.2 Functional Requirements

This subsection provides information on all known ReCAST functions and services. It is more relevant to the specific functional descriptions related to business logic and all user interactions between the system. Use case diagrams are essential to describe functional requirements, based on the system's actor, showing the interaction of users in the system, and it also represents the relationship between each function. Each circle in the Figure 7.3 represents a use case for a system boundary that describes a usage scenario context to capture the requirements for user interaction with the system. In this figure, there are two types of relationships, extended and include. Extend is used when a use case adds steps to another first-class use case. Include is used to extract use case fragments that are duplicated in multiple use cases. The included use case cannot stand alone and the original use case is not complete without the included one. The main use cases are described as follow:

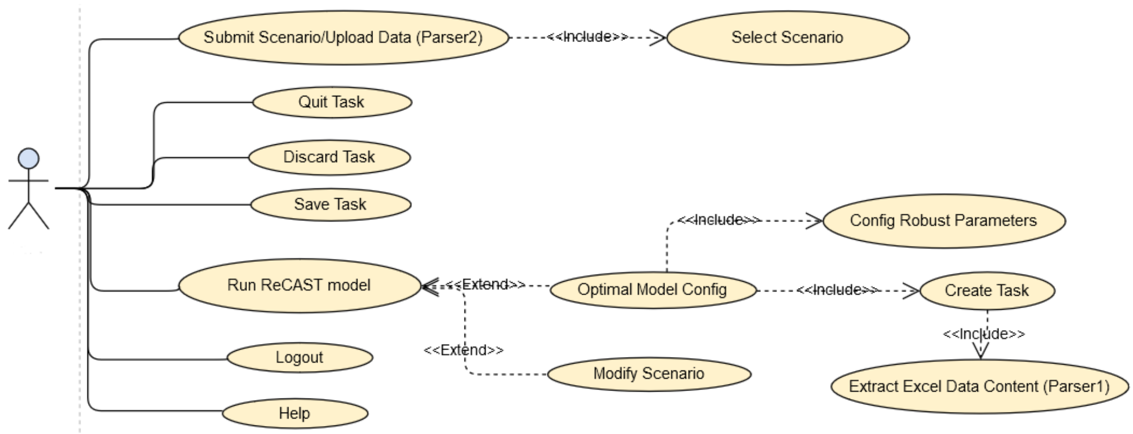


Figure 7.3: Functional requirement.

- **Extract Excel Data to Content (Parser1)**: The use case is used for importing data from an AM-UI Excel file. Firstly, ZLPs upload the file and input parameters manually; and then, the file with parameters will be posted to the server (AWS). Note that the word server here means the Amazon Web Service (AWS), which is the server we deployed the web-application on. The server then checks the file and parameter's validity and extracts all the data from that file using an Excel file parser.

- **Create Task**: When an excel file uploads to the server and to continue, users should define a task in ReCAST. The aim of task definition is for avoiding confusion between various allocation plans. Besides, the assigned task names and descriptions are useful for storing and retrieving the data within the database.

- **Submit Scenario/Upload Data (Parser2)**: The use case is used for exporting a selected scenario data to a template Excel file for uploading to AM-UI. After all scenario results are displayed, ZLP will select a scenario and then click 'Export'. The browser will send data to the the server, and the server will run Parser2 for generating an Excel file compatible with AM-UI. Then the server will send the generated Excel file to the browser to be downloaded by ZLPs. Moreover, the server will write the results and product information to the database for the convenient use for the next time.

- **Scenario Selection**: ZLPs select the most appropriate scenario based on the initial input data. In this use case, some visualization features are considered to facilitate scenario selection by ZLPs.

- **Run ReCAST model**: There are two ways for users to run the ReCAST optimization program, from the Configuration page and from the Modification page, but no matter which approach is used, they will always trigger a click event to run the optimization model. Note that the modification page of ReCAST was designed in Requirement Analysis documentation, but it was not developed in the presented prototype version handed over to the company.

- **Modify Scenario**: On the 'Modification' page, the table of scenario information can be modified by a ZLP. Allocated from ATP (A-ATP) and Allocated from Stock (A-Stock) are editable, but the sum is a fixed number. (There is a relationship between these three parameters: $A - ATP + A - Stock = Total - Allocation$.) Also, the user can click the 'Check' browser to post data to the optimizer server to run the ReCAST optimization model. This then will return results to be displayed. If the user does not click 'Check', there will be no data updated. Note that this feature did not developed in the delivered prototype to the company. The design of the modification page is available in requrement analysis in Appendix A.7 on page 256.

- **Optimal Model Config**: Each individual problem requires a modification of input parameters. In this use case, the ZLPs define the configuration for the mathematical model.

7.2.3 Design and website module segmentation

ReCAST aims to be accessible in a browser using the internet, which makes the application widely available throughout the case study. Thus, ReCAST is a web-based application. The website is divided into three modules: front-end, back-end, and server.

The front-end is implemented using HTML, CSS, and JavaScript; Website back-end was developed using the Django framework with the Python language; Server connectivity is provided by Amazon Web Service (AWS). The Django built-in server uWSGI was used for development. Most of the front-end user interaction is implemented using the Javascript framework Jquery.js and the style design is implemented using the CSS framework Bootstrap. Figures 7.4 on the facing page and 7.5 on the next page represent the website and technical modules used within ReCAST[1].

The data visualization module provides ZLP with visualization of scenarios so they can be compared. This was implemented using Echarts.js which is developed using JavaScript. The parser is the input and output module of the system, which provides functions for processing Excel. This uses two frameworks, HandOnTable.js to implement the display effect in the front end, and the Pandas framework is used to generate

[1]In the development of the ReCAST web application, I want to acknowledge the contribution of Mr Zhikang Tian.

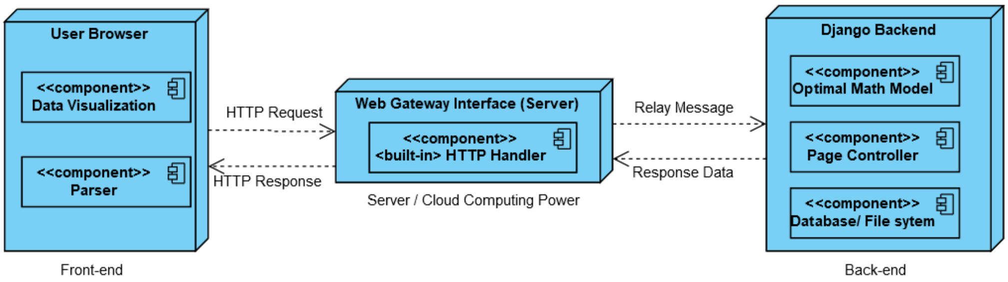


Figure 7.4: ReCAST system modules.

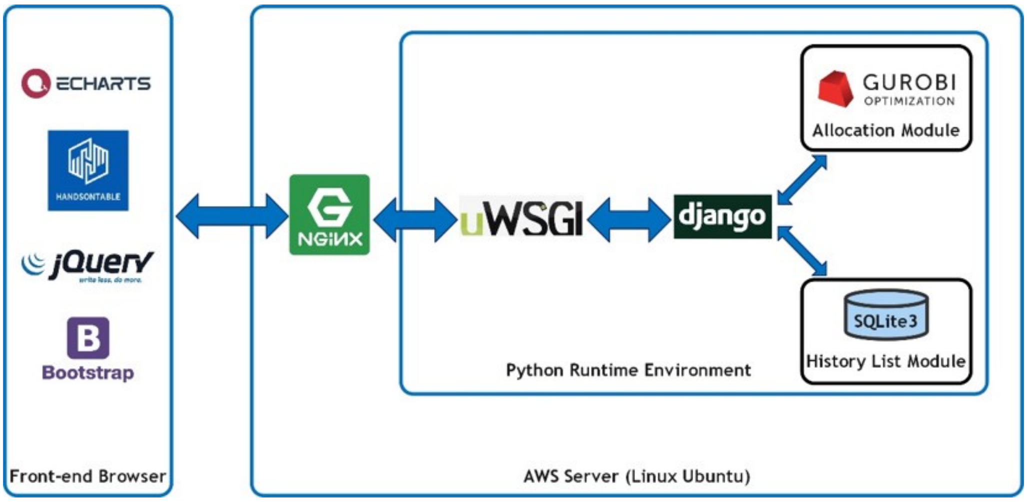


Figure 7.5: Technical modules.

the Excel file in the back end. Django's built-in uWSGI can be used as a server to quickly implement access functions by combining with cloud computing platforms, which uses the HTTP access request processing function. Optimal Math Model provides the core business algorithm to generate allocation plan scenarios by invoking the Gurobi mathematical API interface. Page Controller defines a webpage jump relationship (route logic) in the Django framework to manipulate which pages should be visible in ReCAST. The database module provides system persistence capability, the user's history task records all data, which is provided by Django with its built-in access to SQLite.

7.3 ReCAST Prototype Demonstration

ReCAST is a decision support tool that aims to integrate with the use of APS, which provides planning data that the mathematical model proposed (see Chapter 6) can use. Thus, ReCAST is not only an optimization model but also a web application made up of various technologies to allow easy input of data by ZLPs. ReCAST reads allocation data from the case study's APS, parses data, receives a planner's inputs, obtains insights and planning strategies from the case study, provides visualization, generates solution scenarios and finally exports the selected result in the format readable by the case study's APS, through the AM-UI software module. The optimization model and web application can handle any type of allocation situation and input data. In this section, we describe the functionalities and features of the ReCAST prototype. PyCharm is the IDE we used to develop ReCAST modules. Appendix A.8.3 on page 306 shows some screenshots of the developed modules in PyCharm. ReCAST was developed using Git version controller with the address of the git repository given in Appendix A.8.3 on page 306.

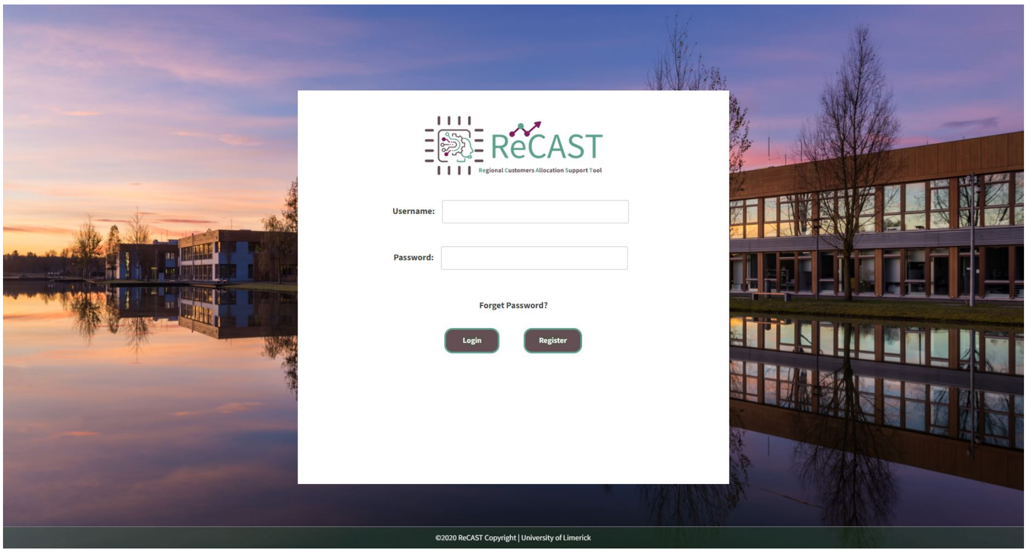


Figure 7.6: ReCAST login page.

In the rest of this section, the ReCAST prototype is presented. Each planner has a user id and password. They can login to ReCAST from anywhere using an internet connection (see Figure 7.6). In the first page, they can check the history of their allocation or create a new allocation. Developing a new allocation plan is as follows. When the allocation situation occurs, the ZLPs can download the task as an Excel file. They should upload the Excel file of the allocation situation from their local computer (see Figure 7.7 on the next page).

By uploading a file, they can review the content of the allocation situation and define a new task in ReCAST. They choose the planning horizon of the allocation required. For

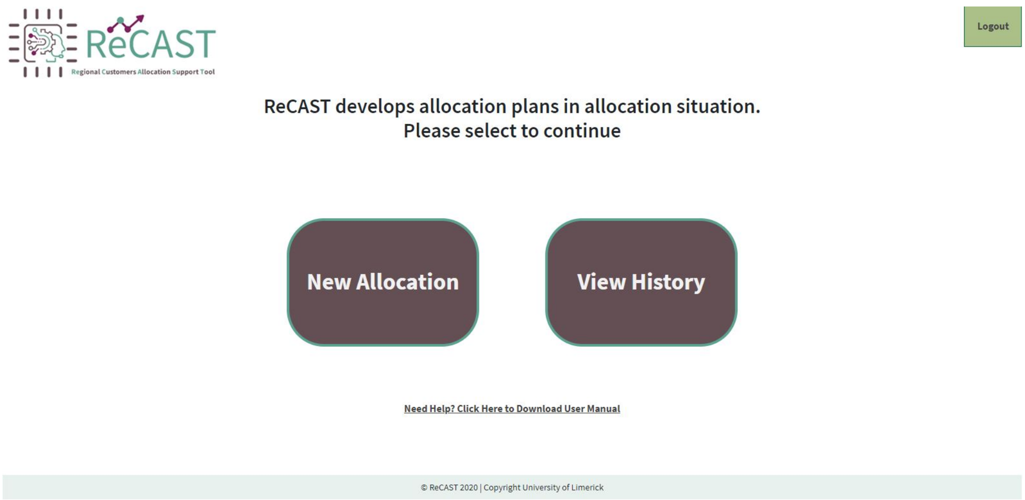


Figure 7.7: ReCAST index page.

example, in this case from week number 25 in 2020 to week number 20 in 2021 (see Figure 7.8 on the following page). As customer allocation are defined based on product type, planners also need to identify what is the packing unit of that product in the logistic system. Finally in this page, they create scenarios according to the case study's strategy for the product on allocation. They identify the weights between allocating more products to customers or keeping the level of stock. This feature lets planners define different scenarios to check various strategies and solutions (see Figure 7.8 on the next page).

After uploading the file to ReCAST and defining the new task, planners should identify the allocation limitations, goals, and allowances of using a product from stock, the screen shown in Figure 7.9 on page 191 is shown. This shows that each order could be allocated to different portions of the planning horizon. Max Delay is the maximum number of weeks that the last part of the order could be satisfied. Minimum Buffer stock identifies the lowest level of stock. The planner could simply apply their ideas to the whole of the planning period or modify week by week (see Buffer Stock Allocation in Figure 7.9 on page 191). Reserve buffer stock also shows the goal of keeping product from ATP and reserving it for stock. As can be seen, this value also could be applied globally for all periods in the planning horizon or be modified weekly. Finally, the planner could restrict the model to allocate some customers in some weeks. By default, all customers are considered similar and all are allowed to use product from stock. However, in some cases, the allowance of using a product from stock needs to be modified

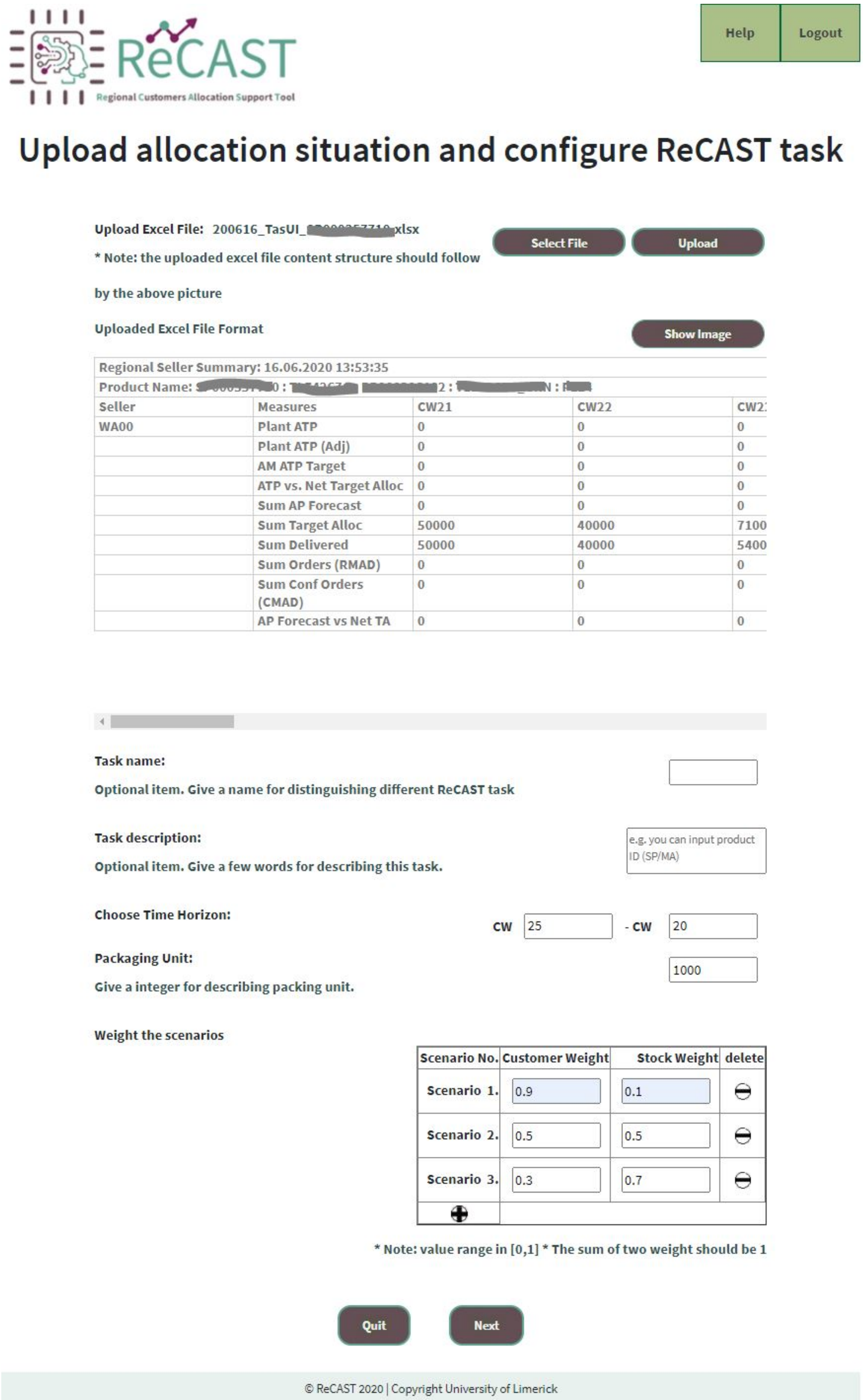


Figure 7.8: ReCAST new allocation task creation page.

by planners (see Figure7.9).

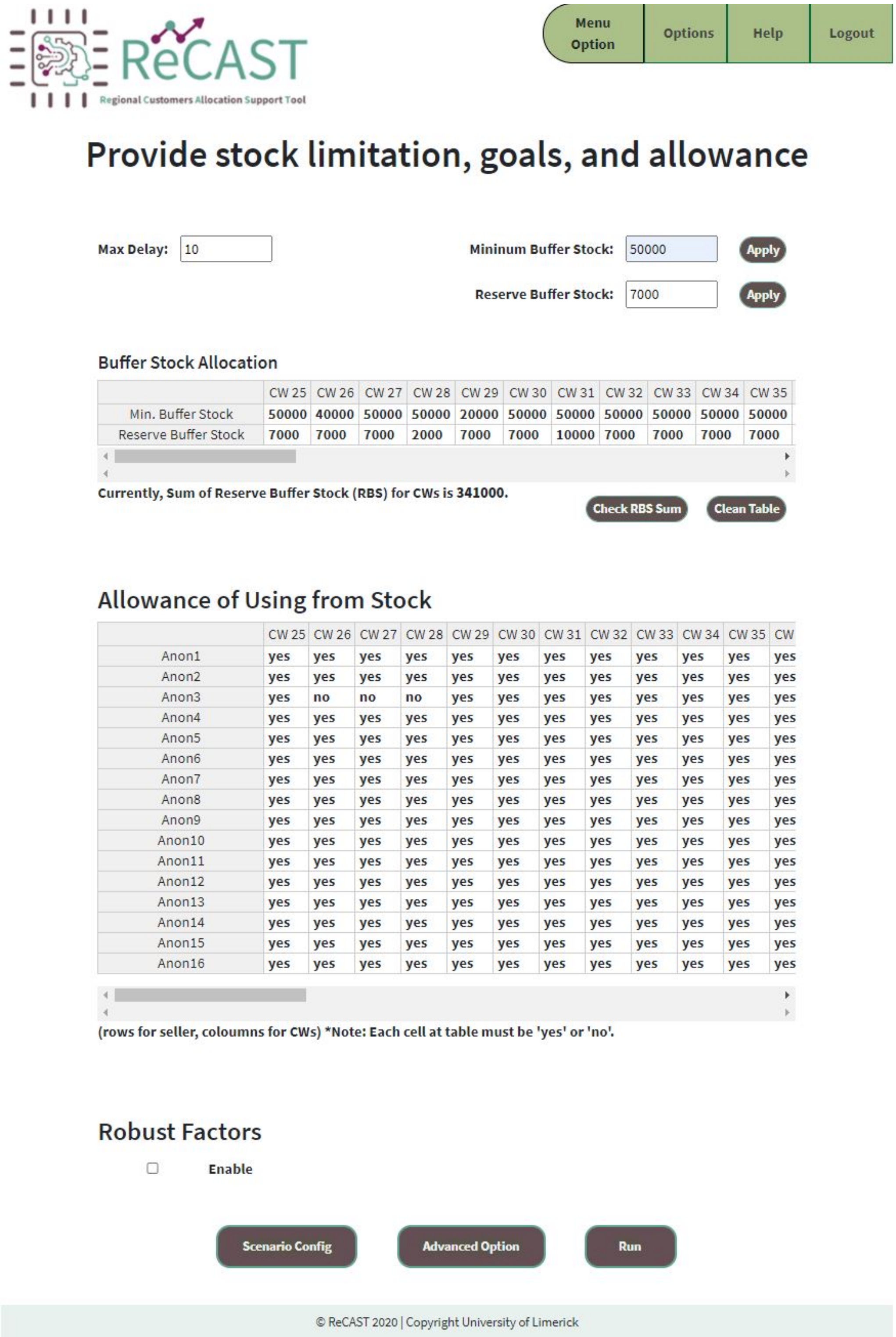
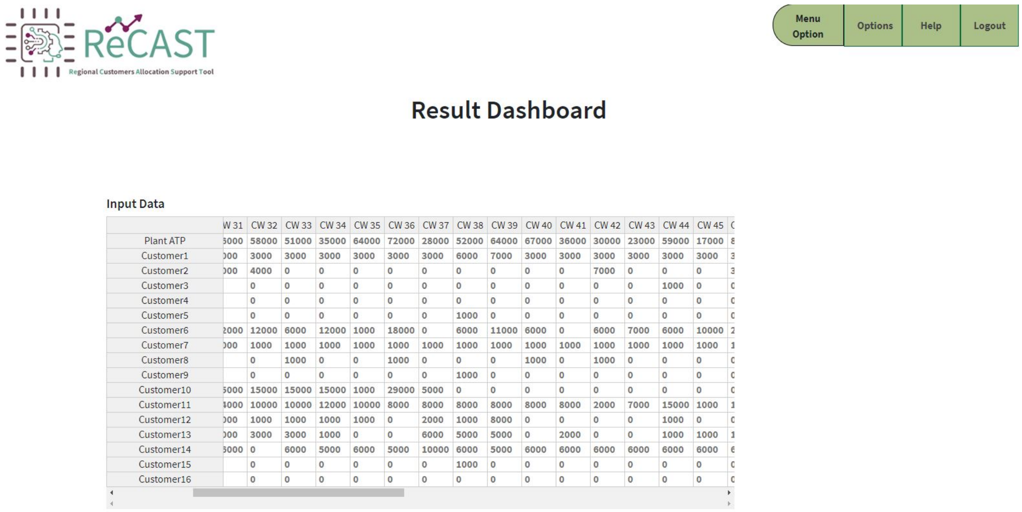


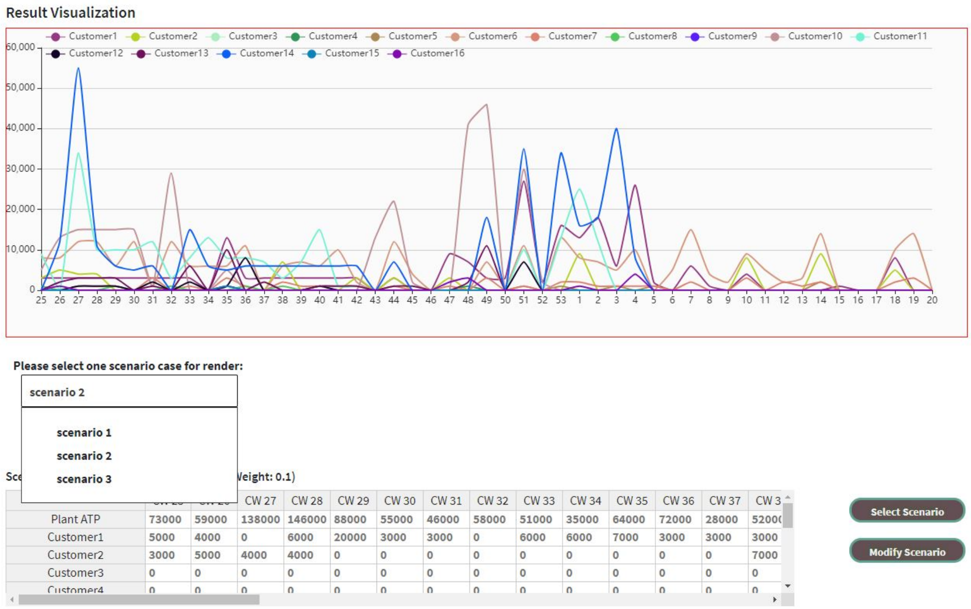
Figure 7.9: ReCAST business situation inputs and configuration page.

After a couple of seconds, the MIP programming model described in Chapter 6 is run, then the ReCAST generates and visualizes solutions. Input data and visualization (see Figures 7.10 on the next page and 7.11 on the following page, respectively) were developed to simplify the understanding to the planner. Users can compare different customers according to scenarios. In the scenario's table (See Figure 7.12 on the next page), they can see the allocation plans developed by ReCAST based on the defined scenarios.



	W 31	CW 32	CW 33	CW 34	CW 35	CW 36	CW 37	CW 38	CW 39	CW 40	CW 41	CW 42	CW 43	CW 44	CW 45	C
Plant ATP	000	58000	51000	35000	64000	72000	28000	52000	64000	67000	36000	30000	23000	59000	17000	8
Customer1	000	3000	3000	3000	3000	3000	3000	6000	7000	3000	3000	3000	3000	3000	3000	3
Customer2	000	4000	0	0	0	0	0	0	0	0	0	7000	0	0	0	3
Customer3		0	0	0	0	0	0	0	0	0	0	0	0	1000	0	C
Customer4		0	0	0	0	0	0	0	0	0	0	0	0	0	0	C
Customer5		0	0	0	0	0	0	1000	0	0	0	0	0	0	0	C
Customer6	2000	12000	6000	12000	1000	18000	0	6000	11000	6000	0	6000	7000	6000	10000	2
Customer7	000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1
Customer8		0	1000	0	0	1000	0	0	0	1000	0	1000	0	0	0	C
Customer9		0	0	0	0	0	0	1000	0	0	0	0	0	0	0	C
Customer10	5000	15000	15000	15000	1000	29000	5000	0	0	0	0	0	0	0	0	C
Customer11	4000	10000	10000	12000	10000	8000	8000	8000	8000	8000	8000	2000	7000	15000	1000	1
Customer12	000	1000	1000	1000	1000	0	2000	1000	8000	0	0	0	0	1000	0	C
Customer13	000	3000	3000	1000	0	0	6000	5000	5000	0	2000	0	0	1000	1000	1
Customer14	5000	0	6000	5000	6000	5000	10000	6000	5000	6000	6000	6000	6000	6000	6000	6
Customer15		0	0	0	0	0	0	1000	0	0	0	0	0	0	0	C
Customer16		0	0	0	0	0	0	0	0	0	0	0	0	0	0	C

Figure 7.10: ReCAST result page part 1 of 3.



			CW 27	CW 28	CW 29	CW 30	CW 31	CW 32	CW 33	CW 34	CW 35	CW 36	CW 37	CW 3
Plant ATP	73000	59000	138000	146000	88000	55000	46000	58000	51000	35000	64000	72000	28000	5200(
Customer1	5000	4000	0	6000	20000	3000	3000	0	6000	6000	7000	3000	3000	3000
Customer2	3000	5000	4000	4000	0	0	0	0	0	0	0	0	0	7000
Customer3	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Customer4	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 7.11: ReCAST result page part 2 of 3.

After comparing the results between scenarios, the planner can select and export the most appropriate allocation plan based on their insights. The exported file is compatible with the destination software module. Thus, users upload the output of ReCAST to the

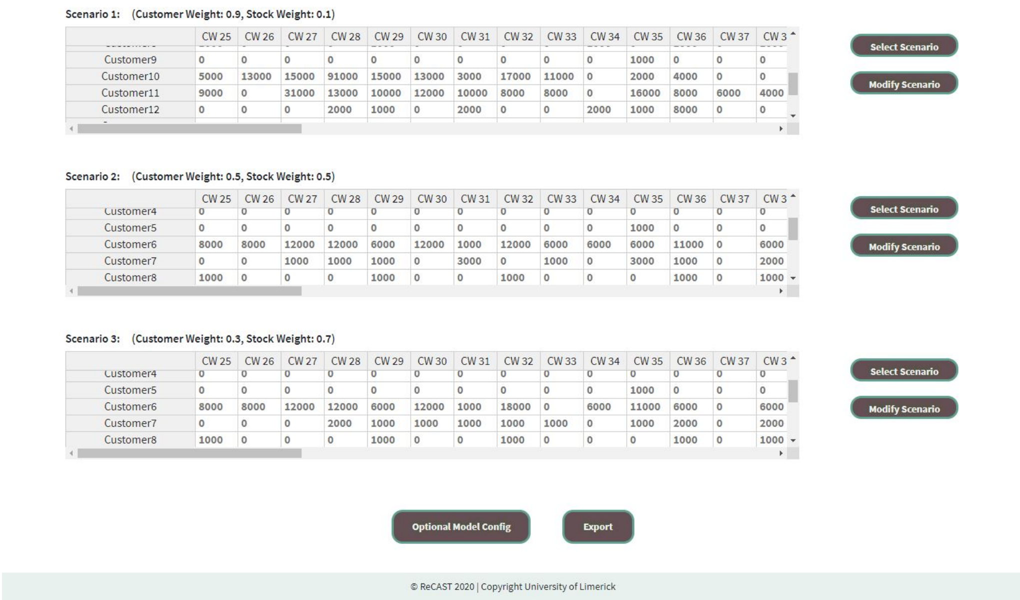


Figure 7.12: ReCAST result page part 3 of 3.

advanced planning system. It will update the whole planning system automatically.

The prototype version of ReCAST aims to show the overall functionality of the task of allocation planning. Further features were also designed but were not developed and executed within the ReCAST prototype. For instance, data bases, modification, sensitivity analysis, and robust model (see Figure 7.9 on page 191). Note that the robust model in this figure is an incomplete part of developing a robust optimization model for the instability of ATP. The robust optimization could be considered as a future extension of the decision support tool. During development and testing, several bugs were identified, which we briefly discuss this process in the next section.

7.4 Test and Debugging

After the initial development of ReCAST, its performance was tested. Thus, we performed two sets of testing, internal tests and tests with ZLPs. Within the internal tests, we used five real cases from the case study and tested ReCAST with various combinations of inputs and scenarios. Table [2] 7.1 on the next page shows an example of a test

[2]SP defines the product type, PU shows the packing unit value of the product, RB_Shift indicate whether the user changes the ‘Reserve Buffer goal’ or all are the same value, and MB_Shift indicates whether the user changes the ‘Minimum Buffer_Stock’ or all are the same values.

Table 7.1: Internal testing of ReCAST.

SP #	**Date**	**Test Date**	**Case #**
product SP	181218	200917	01-Mar
Mode	**Consideration**		**Customer #**
Normal	Test the initial stock value		3
PU	**Customer Weights**	**CW**	**Initial Stock Value**
500	0.9	50-25	535500
Max Delay	**Reserve Buffer goal**	**RB_ Shift**	**Minimum Buffer_Stock**
10	200000	No	100000
MB_Shift	**All Use of Stock = 1**	**Bug**	**Rationality of Solutions**
No	Yes	YES	Infeasible

table. This table aims to capture the input data, add configurations, and identify bugs. In this example, we identified two issues. First, there was an issue with the parser data for selecting the right Current Week (CW). The number of weeks in each year is 52 or 53. When the selected CW starts with bigger values and then goes to the next year (for instance, from week 50 in 2019 to week 25 in 2020), the developed module for the parser misses some data. Second, the minimum buffer stock constraint also makes the model infeasible in some cases. This bug was also identified within this example, which led to the reformulation of the mathematical model discussed in Chapter 6.

To perform the industry tests, we conducted the following steps:

- **Organized and gathered users**: We selected and communicated with planners and SC experts. In the selected team there were two ZLPs from Asia, two ZLPs from Europe, one ZLP from the US, one ZLP coordinator, and the allocation process owner. This team supported us to examine ReCAST.

- **Introduced ReCAST within two online meetings**: We held two meetings with the selected team. In these meetings, we introduced ReCAST and provided a complete how-to-use tour. Additionally, a ReCAST user manual and examples were forwarded to the ZLPs. The overall outputs of these meetings showed that ReCAST is easily understandable for users. They could grasp the concepts and definition of parameters. It showed that ReCAST's allocation decision variables were defined in alignment with the planners' cognitive decision processes. However, some further suggestions were also collected that could be considered in future steps of the platform development.

- **Supported ZLPs during testing**: During testing, ZLPs dealt with issues on the definitions of some variables which were resolved by holding online meetings and sending information by emails. ZLPs in Asia could not reach the web-application due to network security in the Asia office which were resolved by using a different network. Consequently, all of the testing team tested ReCAST completely. Note that due to confidentiality agreement, the ReCAST prototype did not store any test examples used by ZLPs to test the prototype.

- **Held feedback meetings to collect ZLP experiences and suggestions**: Two meetings were held to collect the ZLPs' feedback. In these meetings, we received their suggestions and discussed the quality of solutions. These are discussed in the next section.

7.5 Feedbacks and Benefits of ReCAST

ReCAST reads parameters from specific AM-UI data and planners' inputs were based on their understanding from the business situation. Besides, it generates plans based on scenarios defined by planners. In ReCAST, planners have to define seven types of input variables. Here we aim to structure feedback. To document the feedback, we defined the following questions which we aim to answer here based on the meetings that were held.

1. Is importing and entering data understandable and convenient for users?

2. Do the defined inputs cover all the decision variables and parameters within the planner's decision making?

3. Are the results interpretable by users?

4. What is the evaluation of users regarding the quality of solutions?

In the first part, we gathered and discussed the relevant issues and proposed solutions as follows:

- **Sometimes inserting the allocation table does not work and reloading is needed.**
  Solution: This issue is because ReCAST was developed as a prototype version and its only aim is to deliver the tool for testing the mathematical model and interactions with planners. In the final product development, these issues will be solved as it must be developed under the consideration of the case study's IT department.

- **Definition of MAX Delay was confusing.**
  Solution: Max Delay in the prototype version was defined for each product. This update in the mathematical model does not affect the whole model and it is easy to implement. The max delay defines the penalty in the mathematical model. In previous monthly meetings and management meetings, the possibility was discussed that updating penalty functions could tailor the model based on business requirements. Currently, the penalty function is linear for each product. In future, it could be defined and tested for different functions (e.g., logarithmic) for each customer as recommended. This slight modification can improve a planner's understanding regarding Max Delay and improve allocation plans.

- **Adding a new scenario causes a previous scenario to be deleted.**
  Solution: This issue is mostly related to the prototype version, but provides an idea of how the final product could be improved. In the final product, predefined scenarios could be developed and tested, which will reduce the amount of time that planners are required to input data.

- **Allocation planning is done on one product, the aim should be to do this on several products at the same time.**
  Solution: Considering multiple products in one allocation plan is possible from the mathematical side of ReCAST. However, according to the business under-

standing of this project, there might be differences between input variables of each allocation plan that might take more time. However, if considering multiple products means reducing the effort to download and upload of various files from AM-UI, this could be done in the final product. The ReCAST prototype is a separate tool, but as discussed it could be integrated with AM-UI which means each allocation situation case could easily be imported into ReCAST. Thus, it is recommended to consider multiple products in the final integrated ReCAST. However, solving multiple products in one run may not be an efficient approach.

- **The current ReCAST prototype works for one product with one of the ZLPs mentioned that it takes more time to enter ReCAST's inputs in comparison to manually modifying allocation plans.**
Solution: First, this claim is highly dependent on the size of the case. If the allocation situation has three customers and the issue belongs to a short week horizon, it might be faster to maintain the plan manually. However, in the performed test cases by ZLPs in Asia, the number of customers is more than 30, which makes the calculation harder and above human cognitive capabilities. Secondly, the ReCAST prototype was tested with only two tutorial sessions for ZLPs. Having more tutorials and experiences could drastically increase the familiarity with the tool. Thirdly, the main advantage of ReCAST is that it changes the role of planners from just planning to providing insights on the allocation plans. In addition, it avoids human errors and biases while providing optimal allocation plans.

- **In the current prototype, all customer demands have to be considered in allocation planning. It was mentioned that it would be worth to have a new feature in ReCAST to select specific customers for allocation planning.**
Solution: This issue is very relevant and important. The only point here is that adding this feature does not require any update in the mathematical model. It only relates to data parsing of the tool that could be updated easily. To cater for this, the demands of filtered customers should be diminished from ATP and stock. Mathematical model updates itself automatically with the size of the input data.

Besides, it should be evaluated how removing one customer may affect the overall ATP to avoid developing an infeasible solution.

- **Regarding the quality of solutions, it was mentioned by different ZLPs that ReCAST solutions apparently focus on the centre of selected time horizon. It means that the allocated quantities appear to be larger in the middle of the selected planning horizon.**
  Solution: From the mathematical method used, this issue does not seem to be possible and sufficient feedback would need to be provided by planners, but we do not think this is an issue. It is possible that in the test cases that the CMADs are more centric. Besides, updating penalty functions related to Max Delay could improve this issue. This issue should be considered in future development by further testing and analysis.

- **Another issue highlighted in the current ReCAST implementation is that allocations occur early in the CMAD.**
  Solution: The mathematical model in ReCAST aims to satisfy the customer as soon as possible. Thus, when it can, it allocates all the order quantity on the first date. This approach was considered in the model based on initial meetings in the model development phase. However, considering an upper limit for allocation quantity or forcing the model to divide orders into weeks is completely applicable to the current mathematical model without the need for drastic update or changes. Thus, with a low remodeling effort the model could be updated regarding further constraints and decisions.

Beyond the observed issues, there is a list of suggestions that could be considered in the future development of ReCAST.

- **Would it be better to also consider the percentage of target allocation quantities between each seller and each weekly bucket, not only considering overall coverage in the whole planning horizon.**
  Solution: This suggestion clarified a new concept in decision making. It is mostly

considered in Asia and we did not see it in the the case study branch in Ireland. It shows that in some cases ZLPs consider the share of each customer's demand in comparison to other customers. This feature needs further investigation to define the logic in this decision. It could be easily included in the mathematical model were a lower bound of variables could be used.

- **Adding a buffer value in the result dashboard**
  Solution: This feature was considered when gathering the requirements and during the development of the prototype. Due to time limitations, visualization of this information could not be done during the project. Moreover, we use output from ReCAST to distinguish between the allocated quantity from ATP and the allocated quantity from stock. This modification page provides further information requested by ZLPs. Thus, in the final product, various data and visualization could be applied based on ZLP requirements.

- **Further variables were suggested to be considered such as customer priority and customer average weekly Run Rate (RR)**
  Solution: In some terms, customer priorities are considered in the variable, "allowance of using from stock". However, further investigation should be considered. For instance, ZLP could identify these prioritized customers or machine learning algorithms could prioritize them.

In addition to the above suggestions, we discussed whether ReCAST is useful for ZLPs or not, and if they feel that an automatic decision support tool such as ReCAST is required? The following feedbacks are provided within this discussion:

- At current, ReCAST is a working prototype, and needs further development to become a final product. To reach this goal, testing of the tool by ZLPs, discussion with IT experts and recommendations by planning experts are required.

- The calculation is very fast in the current scenario setting.

- ReCAST is user-friendly and it is easy to use.

- The logic behind the ReCAST is good and it would reduce the training required for CLMs/ZLPs as it provides a structured approach to an ill-defined problem area.
- To sum up, ZLPs suggested that a tool to automate the current allocation process would be beneficial. A tool that combines manual and automatic allocation such as ReCAST would considerably improve the as-is process.

In conclusion, the ReCAST prototype was tested by planners and feedback was obtained. This prototype shows how the case study could further optimize their planning system by digitalization and using advanced analytics approaches. Response times could be shortened and scalability greatly improved. Furthermore, it clarifies how the planners' roles can shift from handling calculations and operations to applying insight and business strategies and it reduces planning cost. As discussed in the previous chapter, Section 6.2 on page 163, ZLPs spend 30 minutes for each allocation planning which the tests show could be handled in 5 minutes by ReCAST.

7.6 Conclusions

Operations Research (OR) analytics requires to be complemented with a final product that delivers a decision support to the end user within a company. Data analysis and visualization are two key elements required to connect OR analytics to end users in a complex planning system like APS. As demonstrated here, these decentralized decision support tools can integrate well with a centralized planning system (APS), which can lead to more efficient planning as shown here. This approach is more maintainable. In addition, it changes the role of planners from doing repetitive tasks to more analytical decision making that needs human cognitive capabilities.

In this chapter, we discussed the development steps of a decision support tool called Regional Customer Allocation Support Tool (ReCAST). ReCAST is a web-based application with an OR core which aims to support planners in the calculation of customer allocation plans. The tool was developed as a prototype version and tested using industry users. The results of the test and deployment supported the obtained approach and showed its benefits.

In summary, ReCAST empowers planners' daily allocation planning with advanced mathematical techniques and digitalization. ReCAST facilitates the decision-making process, digitizes customer allocation planning, reduces human errors, avoids human decision biases, increases the quality of allocation plans, reduces planning costs, and improves work-life quality of human planners.

Chapter 8

Conclusion and Future Researches

8.1 Introduction

This thesis focused on carrying out research on improvements to an Advanced Planning System (APS) in the semiconductor supply chain. The APS is complex and comprises of various functions and interactions. Within this thesis, we used Model Based System Engineering (MBSE) languages to communicate with stakeholders, model complexities, define research questions, and develop quantitative tools, as presented in Chapter 4. Using these modeling MBSE languages, we analyzed the root causes of nervousness in APS and understood where manual intervention was used. For our investigation of nervousness, we used a multi-paradigm simulation approach presented in Chapter 5 and developed a decision support tool for planners in customer allocation, which where described in Chapters 6 and 7.

This chapter summarizes and discusses the results of the research presented in this thesis (Section 8.2 on the facing page). The main scientific contributions are highlighted in Section 8.3 on page 208. Managerial insights are derived in Section 8.4 on page 209. Finally, the limitations of the presented research are detailed in Section 8.5 on page 211. From these, directions for future research are discussed in Section 8.6 on page 212.

8.2 Summary and discussion of results

This thesis deals with disruption, instabilities, and inefficiency in master planning and demand fulfilment within a semiconductor supply chain planning system. The studied planning software modules comprise of automatic algorithms that interacts with the production system, where human planners manually intervene. The algorithmic processes investigated were master planning and demand fulfilment.

For this purpose, Chapter 3 discusses our gained understanding of the planning system processes and its complexities. Chapter 3 details the output of interviews with experts, investigations of the planning system's documents, reviews of the literature, and studied tutorials from the case study. Following on from this chapter, in Chapter 4, we demonstrate the use of MBSE languages for investigating Supply Chain (SC) disruption and formulate possible improvements of a complex planning system. We developed a pathway using a common language with stakeholders, which supported us to understand the planning system, develop hypotheses, and propose quantitative solutions. Within this research, we emphasized the importance of using systematic approaches within disruption management and as a requirement for the development of quantitative research methods. Through the use of three MBSE languages, we evaluated the benefits of using these to describe Discrete Event Logistics Systems (DELS), such as manufacturing or supply chain systems. Moreover, the use of these MBSE languages resulted in accurate identification and careful definition of problems within the supply chain planning system.

Based on acquiring a system understanding, we extracted two challenges, one in demand fulfilment and the other in master planning. The first challenge was an Early Warnings (EW) Key Performance Indicator (KPI) for which we found the root causes of disruption. For this purpose, we developed a hypothesis regarding the nervousness of Available To Promise (ATP). The root causes of this nervousness were found to be in a deviation between planned and execution in logistics. To find the root causes of disruption and to test our hypothesis, we used a multi-paradigm simulation modeling framework. Following validation of the hypothesis using simulation, we carried out a data analysis of APS to fully prove completely our hypothesis. The result of this part of the research is presented in Chapter 5.

The second problem in APS relates to the intervention of human planners in allocation planning and demand fulfilment. To keep the flexibility of APS, in cases of tight supply or demand, the fluctuation of plans has to be updated by human planners. This manual intervention is mainly required due to the hierarchy of the planning system, where there is no KPI or measurement method to evaluate the developed allocation plans. Therefore, due to the existence of several alternatives for the selection of an allocation plan, the planner more than lightly may not select the optimal alternative. Thus, they try to satisfy the plan with simplistic approaches. To solve this issue and improve allocation planning, we developed a multi-objective Mixed-Integer Programming (MIP) model. The model improves the way planners consider an allocation plan by adding a new tool that optimizes their decision making.

This tool changes the role of planners from manually calculating the allocation plans to planners who incorporate business strategies in selecting the most appropriate evaluated scenario. This tool is for the allocation of products to customers in an allocation situation, which is called the Regional Allocation Support Tool (ReCAST). ReCAST aims to improve customer satisfaction and keep the stability of inventory. A prototype ReCAST was developed as a web application which was tested by planners in industry. Results of this study are discussed in Chapters 6 and 7.

More specifically, the research questions that where addressed in this thesis are:

Research Question 1: Evaluate how model-based system engineering languages could support disruption management within a complex supply chain system?

Disruption management is a continuous process in supply chain or manufacturing which requires concrete methodological steps. Identifying inefficiencies in planning, finding solutions in cross disciplinary teams, and investigating solutions was facilitated by using MBSE methods. Within this study, we showed that the MBSE model is the sole source of truth regarding capturing the interactions and processes of a supply chain planning system. Having a referential MBSE of planning and supply chain systems in the time of disruption will support SC operators and experts. In addition, we showed that the methodological MBSE provides a common language between researchers and industry partners, improves system understanding, and supports the structuring of quantitative method development. Thus, the use of MBSE methodologies for problem solving or referential MBSE for system mapping supports DELS within disruption management projects.

In this research, we used three different MBSE languages (BPMN, OWL, and SysML) to investigate the complex functions, interactions, and manual interventions within a supply chain planning system. These languages were used through a proposed pathway of using MBSE in the disruption analysis of a supply chain planning system. By using such a proposed pathway, we demonstrated how the issues in the planning system are identified and how the developed hypothesis are supported by MBSE. The results in Chapter 4, provided support for the use of MBSE languages as a methodology for the analysis of DELS, which in our case was a complex supply chain system.

We feel that these modeling approaches provided a common language between stakeholders to better define the research questions, the methods used, allowing the evaluation of the proposed hypothesis. In addition, the developed models were used as conceptual models for the development of quantitative tools. Thus, in this research, we provided the use of the three MBSE languages for identifying issues, communicating with stakeholders, developing conceptual models, and developing quantitative tools within the case study. The results of this thesis proved the importance of using an MBSE language for the initial steps of disruption management within DELS, such as supply chain or manufacturing systems. Note that selecting an MBSE approach depends on the problem and objectives of a study.

Research Question 2: What are the root causes for Early Warnings (EW) nervousness which results from ATP instability in the case study supply chain planning?

Performance evaluation of the APS in this case study had a high level of complexity. Although an EW KPI highlights issues with supply chain plans, it does not reveal how these can be resolved. To identify the root causes of an observed disruption in EW, a hypothesis needed to be developed and tested.

In this challenging environment, where testing in an alive running system can cause poor customer service, a simulation model was developed to capture the important aspects of the case study's APS. The model captured the important aspects of reality to allow experimentation in a risk-free environment. Within this context, we developed a model that imitates the interactions between planning modules and its production system. The model was built using a multi-paradigm simulation framework that comprises of agent-based and discrete event simulation entities.

The proposed simulation modeling approach provided capabilities in the analysis of the complex causalities of a supply chain planning system. Within this simulation, we simulated the material flow, information flow (planning system), and the interaction of these two flows. The presented simulation model in Chapter 5 is based on the use of agent-based and discrete-event simulation entities combined into one model. This simulation modeling framework can facilitate the simulation of complex supply chain systems. Evaluating a complex supply chain system using this framework can support one in developing a digital twin, and the framework has the additional capability of modeling human intervention in the planning system.

By implementing the simulation model with real data, we proved the source of disruption within ATP nervousness, which demonstrated the competency of the proposed models and framework. Within this research, we proved the difference between what is planned and what is executed in a logistic route which resulted in nervousness within the planning system creating an EW KPI. This incompatibility causes instability in ATP which leads to nervousness in EW, which may result in the late delivery of an order, unless planners manually intervene. The output of the simulation model can guide planning experts to tune parameters and to modify the planning system in a cost effective manner.

Research Question 3: How to move from low-level operational manual alterations by planners to more strategic decisions through the use of analytical

decision tools?

There is a misunderstanding that APS can fully replace human activities with automatic algorithms (Wiers and Kok 2017). The algorithms indeed calculate plans based on a simplified definition of planning and scheduling. Although these algorithms calculate and run automatically, human planners are essential to input data, run, control, or modify the inputs and outputs of automatic planning and controlling modules. In industrial applications, planning systems support human planners, where human planners run planning systems through their use of human cognitive capabilities. Thus, human planners are a critical element in planning systems that interact with information and production systems. Effects of human decisions in the operation of planning systems are rarely investigated, but it is obvious that supporting human planners' decision-making provides advantages to the performance of planning systems.

Within this research, we developed a decision support tool for human planners that supports them in maintaining allocation plans to customers within APS. The allocation planning that they carry out is an important aspect of planning within the case study. Due to the incapability of human cognition in the calculation of thousands of scenarios, planners often choose an easy but applicable decision. In addition, it is not possible for the APS system to evaluate a human planner decision as there is no benchmark or means to evaluate them. These challenges were addressed within this research by replacing a manual human planner's alteration of an allocation plan with mathematical optimization and the development of a web application decision support tool.

As discussed in Chapter 6, the allocation planning decisions in the case study are made in a decentralized manner, which means the Customer Logistic Managers (CLM) defines the allocated quantities manually based on their business situation. In Chapter 6, we propose a new multi-objective MIP model to allocate quantities to customers. The mathematical model requires input from APS and planners to develop the optimal allocation plan. The model was tested on various real data from industry and its solution quality was proved by supply chain experts.

The ReCAST decision support tool was presented in Chapter 7. ReCAST is an analytically based decision support tool to plan the allocation of products to customers using a mathematical optimizer as its core element. ReCAST is a web application which is accessible by planners anywhere in the world through the use of an internet connection. This helps keep the allocation planning decentralized but in one structured platform that could be easily connected to an APS. The prototype of the ReCAST was developed and tested within industry, where the results of these tests and its deployment showed the benefits of analytical tools like ReCAST for complex planning systems like APS in the case study.

8.3 Contributions

The research presented in this thesis has three main contributions to the current state of the art of APS and supply chain planning systems. First, Chapter 4 presents the use of MBSE for disruption management of DELS, which provides evidence for the development of a common language for the description of systems. Such a tool would support the exchange of information across multidisciplinary stakeholders, supporting the development of the correct quantitative tool to address the correct problem. In this thesis, we demonstrated this here using a complex supply chain planning system. MBSE is represented as an initial step in a methodology for analysis and for the development of quantitative tools for the improvement and evaluation of such systems. Moreover, the presented pathway and discussion of the three used MBSE languages provides insights for language and method selection within similar studies.

Second, in Chapter 5 we proposed a new framework for simulating a planning system. The simulation approach imitates the interactions between the planning system and manufacturing steps. The use of this framework can evaluate the performance of planning systems and identify the incompatibility within supply chain systems. The proposed multi-paradigm model can be used for the maintenance of planning systems. Within the case study, the results of the simulation analysis supported parameter tuning within APS. Modeling the interaction between the planning system and the flow of parts in the manufacturing system introduces a new perspective on APS maintenance. Evaluation of ATP nervousness due to the lack of incompatibility between planning software modules and planning and production is one of the main applications of the simulation model framework presented in this research. This model demonstrates the importance of internal interaction between different software modules within the planning of complex supply chain systems.

Third, in Chapters 6 and 7 we demonstrated the use of analytical decision tools for the improvement of the planning system with APS, as a decentralized tool, in the case study. The availability of data in APS is often neglected and therefore many planning processes require manual intervention of planners. We modeled the allocation decision making using a a multi-objective MIP model. We showed how a mathematical optimization model, can support planners and change their role from routine manual operational calculations to a provider of optimal solutions that are more aligned with the strategies of the company. In addition, from the core MIP model we developed a web application as a decentralized decision support tool and tested this in an industrial setting.

8.4 Managerial insights

The results of the research presented in this thesis have various implications for industrial practitioners and decision-makers. First, research and development projects in complex APS require a shared understanding that is developed using a common modeling language. MBSE can store complex DELS and value chains as a single source of truth. MBSE can transfer knowledge in DELS, such as supply chain systems, in standardized models. These computer-based models could be improved and maintained more efficiently than descriptive documents or presentations from the real system. Choosing a proper MBSE approach and developing a referential model is an extremely large undertaking, but we feel with the adoption of Industry 4.0 concepts, it will progress.

Moving towards a referential MBSE model for DELS provides various benefits for research and development and improvement projects as described in this thesis. To reach this goal, DELS needs to be divided based on organizational structure, functions, and networks. Then these subsystems should be mapped with a selected MBSE language.

The developed models are connected through the selected language and finally the referential model should be maintained and used. These models can gradually be developed within various research and development projects. This, however, would require a standardized road map and a selected MBSE language. For instance, a BPMN model of software modules and their processes could be developed simultaneously with a BPMN of planners' interactions with software modules. Then these two models could be connected according to a standardized road map to finally form the single source of truth of the system.

This comprehensive model could be abstracted in quantitative research like simulation models, which will make the research and development project more efficient, maintainable, and capable of being connected with other research projects. Note that deploying MBSE requires a change management perspective. Beyond research and development, MBSE can be applied to the whole organization, which leads companies to better digitization and sharing of knowledge.

Second, based on the developed MBSE and simulation models, we showed the importance of parameter tuning within APS. Through early warning analysis conducted by simulation modeling and data analysis, we emphasize the importance of an internal model (see the Base Model in Figure 3.8 on page 52) for the performance of APS. We demonstrate the importance of these parameters on stability of the plans. Thus, they need to be maintained through research, process modification, and parameter tuning. Simulation modeling is an efficient approach to maintaining APS. In the long term, this approach could be extended to an APS digital twin, which allows decision-makers to examine various scenarios of APS in a risk-free environment. For example, during our meeting (see Appendix A.1 on page 233) with the case study experts, we saw some recommendations for mixing demand planning with capacity planning and replacing them with mathematical optimization. In this regard, if there is a simulation model of APS, the new recommended system could be tested within this simulation model.

Third, due to the availability of data within APS, the APS algorithms are capable of being improved through the use of analytical decision tools. As shown in this thesis, most parts of APS use a simplified rule-based algorithm that then requires human intervention. These two aspects of APS need to be improved. Advanced quantitative decisions like mathematical optimization could increase the performance of APS by either supporting human planners or replacing simple rule-based algorithms. Providing user-friendly decision support tools is the right means of attempting to decrease the impact of human decision biases on APS.

8.5 Limitations

Regarding the results reported in Chapter 4, there are, of course, limitations to the research reported here. The first is that the gained knowledge and insight obtained with each modeling approach had an impact on the output of the next used MBSE methodology. As previously discussed, we developed OWL, BMPN, and SysML, respectively.

Second, MBSE is a wide and vast area of knowledge, and here we were restricted to the domain of our case study. The aim of these tools for us is to assist in modeling to reveal the source of disruption in the system based on the developed hypothesis.

Third, MBSE approaches have several features that we did not need to cover in our case study. For instance, SysML contains nine different diagrams that individually or together could be analyzed. Finally, here our aim was to evaluate MBSE in the initial stages to help understand disruption in SC systems and to assist in the support of quantitative methods, where the MBSE methodologies were used to communicate with the different stakeholders within the system.

Fourth, regarding the simulation model, we abstracted APS according to the developed hypothesis due to the limitation of time. These models could be developed capturing more realistically to model closer APS functionality. These simulation models could be equipped with a visual dashboard to make supply chain adjustment easy for users.

Fifth, in any planning system, the role of human planners in disruption and management is necessary. Considering a planners' effect on APS needs comprehensive research, which we did not consider within the developed simulation model.

Sixth, the ReCAST prototype could have been developed further from the requirements we gathered, which we could not implement due to the limitation of time. Elements such as sensitivity analysis, further data visualization, and robust optimization within the prototype dashboard could be developed..

8.6 Directions for Future Research

There are many possibilities to further develop the research presented in this thesis. Following on from the presented research in Chapter 4, we propose the development of a standardized modeling language to improve APS maintainability and extension. We recommend to develop a MBSE that contains various levels as follows. The aim of the following categorization is to provide a standardized structure for the development of a MBSE model.

- First level is the planning modules in APS which contains all software with their functionalities and interaction discussed in Chapter 3.
- Second level is a model of human planners, which consists of a planner network, responsibilities, and type of decision. The level aims to model the human intervention within the supply chain system.
- Third level is databases which show the available data for each software module, the flow of data within the processes, and timing of using data. The quality of data in each database could be added to this level.
- Fourth level is a production network which maps the manufacturing routes, type of production, capacity, yields, and connections. This level aims to represent the material flow within the supply chain.
- The last level is the business environment like third parties, customers, and their interaction with supply chain systems.

Having a model on all these levels could be used as a single source of truth within the supply chain. This model can be accessible for updates within the supply chain or updated regularly by human resources. This referential MBSE of a supply chain system provides benefits for managers of the supply chain system. For instance, it could be used for the development of conceptual models for simulation modeling in order to resolve issues, development of digital twins, or improving supply chain systems

The second research direction relates to simulation modeling. The simulation model can aim to solve the following challenges, while it could be expanded as a digital twin of APS. A list of research directions for simulation of APS is as follows:

- Simulate the divisional model algorithm to test replacing the ATP calculation rule-based algorithm with mathematical optimization. Proposing a mathematical model for ATP calculation that is compatible with available data and concepts. However, to test the model, it should be examined within various scenarios that could be generated by simulation modeling of the divisional model.

- Rule-based demand fulfilment also could be replaced by combinatorial optimization. This model also could be tested by a simulation model using current demand fulfilment that would compare the delivery performances.

- It is also interesting to demonstrate the impact of human planners like ALM, ZLP, CLM, and demand planners on the performance of APS and automatic algorithms. This analysis could be examined through simulation models. Thus, understanding the consequences of the planner's decision-making on other parts and the algorithm of the system is an open avenue for further research.

The third suggested research area is the development of a new KPI for the delivery performance of automatic demand fulfillment. As discussed, EW only captures the deviation in delivery times while delivery quantities could change too. Besides, EW could not show the shift in the number of order lines that split within i2DF. Thus, proposing a new EW or upgrading the current EW to represent a deviation in quantities and presentation of information regarding order splits could improve the quality of this KPI.

Fourth, ReCAST is a decentralized optimization application that was proved within this study. Use of decenteralized planning modules which store data centrally, that could make them available to APS can improve the performance of APS and make it more traceable. On a general scale, the use of decentralized optimization can satisfy various objectives within hierarchical planning. Computationally, this approach is efficient. These methods of optimization application could be maintained easier while it optimizes all inputs of APS.

Fifth, evaluating the balance between decentralized and centralized applications is an open research stream. While centralized optimization can decrease decision conflicts within a supply chain, decentralized optimization can calculate local optimal decisions to APS based on the structure and hierarchy of the whole supply chain. Defining to what extend each of these approaches should be used in APS is a matter of debate.

Finally, ReCAST can be improved by various improvements such as:

- Considering product allocation by ALM.
- Applying machine learning for customer segmentation.
- Adding KPIs to allocation plans to improve scenario selection by planners.
- Improving visualization to ease scenario selection.
- Integration with another planning module (Possibly AM-UI).
- Minimize deviation between the new allocation plan with what was planned before.

Bibliography

Achter, Sebastian et al. (2017). “On agent-based modeling in semiconductor supply chain planning”. In: *2017 Winter Simulation Conference (WSC)*. IEEE, pp. 3507–3518.

Albers, Albert and Quentin Lohmeyer (2012). “Advanced systems engineering–towards a model-based and human-centered methodology”. In: *International Symposium Series on Tools and Methods of Competitive Engineering, TMCE*, pp. 407–416.

Albers, Albert and Christian Zingel (2013). “Challenges of model-based systems engineering: A study towards unified term understanding and the state of usage of SysML”. In: *Smart Product Engineering*. Springer, pp. 83–92.

Alexander, D et al. (2007). *Quicklook final report*. Tech. rep. Technical Report, George Mason University.

Alexander Seitz Phillip Kriett, Martin Grunow (2013). *Test of Hypotheses and Development of Solutions Improving the Concept of Negative Early Warnings in the Order Management System of the Semiconductor Company Infineon Technologies AG*. Infineon internal study with Technical University of Munich.

Anylogic (2020). en. Library Catalog: www.anylogic.com. URL: https://www.anylogic.com/ (visited on 05/29/2020).

Arena (2020). URL: https://www.arenasimulation.com/ (visited on 07/02/2020).

Babarogić, Sladjan et al. (Dec. 2012). “Multi-period Customer Service Level Maximization under Limited Production Capacity”. en. In: *International Journal of Computers Communications & Control* 7.5, pp. 798–806. ISSN: 1841-9836. DOI: 10.15837/ijccc.2012.5.1075. URL: http://univagora.ro/jour/index.php/ijccc/article/view/1075 (visited on 01/29/2019).

Batarseh, O G, E Huang, and L McGinnis (Feb. 2015). “Capturing simulation tool and application domain knowledge for automating simulation model creation”. en. In: *Journal of Simulation* 9.1, pp. 1–15. ISSN: 1747-7778, 1747-7786. DOI: 10.1057/jos.2014.9. URL: https://www.tandfonline.com/doi/full/10.1057/jos.2014.9 (visited on 05/26/2020).

Batra, Gaurav et al. (2018). *Right product, right time, right location: Quantifying the semiconductor supply chain.* URL: https://www.mckinsey.com/industries/semiconductors/our-insights/right-product-right-time-right-location-quantifying-the-semiconductor-supply-chain.

Bayar, Nawel et al. (2016). “Using immune designed ontologies to monitor disruptions in manufacturing systems”. In: *Computers in Industry* 81, pp. 67–81.

Beery, Paul T (2016). *A model-based systems engineering methodology for employing architecture in system analysis: developing simulation models using systems modeling language products to link architecture and analysis.* Tech. rep. NAVAL POSTGRADUATE SCHOOL MONTEREY CA MONTEREY United States.

Borshchev, Andrei (2013). *The big book of simulation modeling: multimethod modeling with AnyLogic 6.* AnyLogic North America.

BPMN ISO Standard Document (2013). *ISO/IEC 19510:2013.* en. Library Catalog: www.iso.org. URL: https://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/06/26/62652.html (visited on 05/28/2020).

BPSIM (June 2018). *BPSim.org.* URL: http://bpsim.org/ (visited on 04/06/2018).

Cameron, Bruce and Daniel Mark Adsit (2018). “Model-Based Systems Engineering Uptake in Engineering Practice”. In: *IEEE Transactions on Engineering Management.*

Cano-Belmán, Jaime and Herbert Meyr (Jan. 2019). “Deterministic allocation models for multi-period demand fulfillment in multi-stage customer hierarchies”. In: *Computers & Operations Research* 101, pp. 76–92. ISSN: 0305-0548. DOI: 10.1016/j.cor.

2018.09.002. URL: http://www.sciencedirect.com/science/article/pii/S0305054818302417 (visited on 09/26/2018).

Carlson, Robert C, James V Jucker, and Dean H Kropp (1979). "Less Nervous MRP Systems: A Dynamic Economic Lot-sizing Approach". In: *Management Science* 25.8, pp. 754–761. DOI: 10.1287/mnsc.25.8.754. URL: https://doi.org/10.1287/mnsc.25.8.754.

Chen, C., N. Srinivasan, and N. Syed (2005). "Method for Updating the Supply Plan used by an Available-To-Promise System". In: US Patent App. 10/612,257.

Chinosi, Michele and Alberto Trombetta (Jan. 2012). "BPMN: An introduction to the standard". en. In: *Computer Standards & Interfaces* 34.1, pp. 124–134. ISSN: 09205489. DOI: 10.1016/j.csi.2011.06.002. URL: http://linkinghub.elsevier.com/retrieve/pii/S0920548911000766 (visited on 04/13/2018).

Council, Supply Chain (2008). "Supply chain operations reference model". In: *Overview of SCOR version* 5.0.

D'Ambrosio, Joseph et al. (2019). "An MBSE Approach for Development of Resilient Automated Automotive Systems". In: *Systems* 7.1, p. 1.

De Sampaio, Raimundo JB, Rafael RG Wollmann, and Paula FG Vieira (2017). "A Flexible Production Planning For Rolling-Horizons". In: *International Journal of Production Economics* 190, pp. 31–36. DOI: 10.1016/j.ijpe.2017.01.003. URL: https://doi.org/10.1016/j.ijpe.2017.01.003.

Dekkers, Rob et al. (2012). "Entropy assessment of supply chain disruption". In: *Journal of Manufacturing Technology Management* 23.8.

Demirel, Güven et al. (2019). "Identifying Dynamical Instabilities In Supply Networks Using Generalized Modeling". In: *Journal of Operations Management* 65, pp. 136–159. DOI: 10.1002/joom.1005. URL: https://doi.org/10.1002/joom.1005.

Dittmann, Lukas (2018). "Advanced Planning in Semiconductor Manufacturing – Big Data Analysis to compare target and actual Data in operational Planning". MA thesis. Technical University of Dresden.

Ehm, Hans, Thomas Ponsignon, and Thomas Kaufmann (2011). "The global supply chain is our new fab: Integration and automation challenges". In: *2011 IEEE/SEMI Advanced Semiconductor Manufacturing Conference*. IEEE, pp. 1–6.

Ehm, Hans, Nour Ramzy, et al. (2019). "Digital Reference–A Semantic Web for Semiconductor Manufacturing and Supply Chains Containing Semiconductors". In: *2019 Winter Simulation Conference (WSC)*. IEEE, pp. 2409–2418.

Estefan, Jeff A et al. (2007). "Survey of model-based systems engineering (MBSE) methodologies". In: *Incose MBSE Focus Group* 25.8, pp. 1–12.

Estefan, Jeff A (2008). *Survey of Model- Based Systems Engineering (MBSE) Methodologies*. en. URL: http://www.omgsysml.org/MBSE_Methodology_Survey_RevB.pdf.

Feldmann, Stefan, Konstantin Kernschmidt, and Birgit Vogel-Heuser (2014). "Combining a SysML-based Modeling Approach and Semantic Technologies for Analyzing Change Influences in Manufacturing Plant Models". en. In: *Procedia CIRP* 17, pp. 451–456. ISSN: 22128271. DOI: 10.1016/j.procir.2014.01.140. URL: http://linkinghub.elsevier.com/retrieve/pii/S2212827114004181 (visited on 04/12/2018).

Fleischmann, B., H. Meyr, and M. Wagner (2015). "Advanced Planning". In: *Supply Chain Management and Advanced Planning*. Ed. by H. Stadtler, C. Kilger, and H. Meyr. Berlin Heidelberg: Springer-Verlag, pp. 71–95.

Fleischmann, Bernhard and Herbert Meyr (2003). "Planning Hierarchy, Modeling And Advanced Planning Systems". In: *Handbooks in Operations Research and Management Science* 11, pp. 455–523. DOI: 10.1016/S0927-0507(03)11009-2. URL: https://doi.org/10.1016/S0927-0507(03)11009-2.

Forsberg, Kevin and Harold Mooz (1991). "The relationship of system engineering to the project cycle". In: *INCOSE International Symposium*. Vol. 1. Wiley Online Library, pp. 57–65.

Foster, John (2005). "From simplistic to complex systems in economics". In: *Cambridge Journal of Economics* 29.6, pp. 873–892.

— (2011). "Economic systems". In: *Philosophy of Complex Systems*. Elsevier, pp. 509–530.

Friedenthal Sanford, Moore Alan, and Steiner Rick (Nov. 2014). "OMG Systems Modeling Language (OMG SysML™) Tutorial". In: *INCOSE International Symposium* 18.1, pp. 1731–1862. ISSN: 2334-5837. DOI: 10.1002/j.2334-5837.2008.tb00914.x. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/j.2334-5837.2008.tb00914.x (visited on 04/12/2018).

Geier, Sebastian (2014). *Demand Fulfillment bei Assemble-to-Order-Fertigung: Analyse, Optimierung und Anwendung in der Computerindustrie*. Wiesbaden: Springer Gabler.

Gelders, Ludo F and Luk N Van Wassenhove (1982). "Hierarchical Integration in Production Planning: Theory and Practice". In: *Journal of Operations Management* 3.1, pp. 27–35. DOI: 10.1016/0272-6963(82)90019-5. URL: https://doi.org/10.1016/0272-6963(82)90019-5.

Gössinger, Ralf and Sonja Kalkowski (Dec. 2015). "Robust order promising with anticipated customer response". In: *International Journal of Production Economics*. Current Research Issues in Production Economics 170, pp. 529–542. ISSN: 0925-5273. DOI: 10.1016/j.ijpe.2015.05.026. URL: http://www.sciencedirect.com/science/article/pii/S0925527315001784 (visited on 10/08/2018).

Grillo, H. et al. (Jan. 2018). "A Fuzzy Order Promising Model With Non-Uniform Finished Goods". en. In: *International Journal of Fuzzy Systems* 20.1, pp. 187–208. ISSN: 1562-2479, 2199-3211. DOI: 10.1007/s40815-017-0317-y. URL: http://link.springer.com/10.1007/s40815-017-0317-y (visited on 02/06/2019).

Gruchmann, Tim and Thomas Gollmann (2017). "Simulation-based Modeling And Analysis of Schedule Instability in Automotive Supply Networks". In: *Operations Research*

Proceedings 2015. Ed. by Karl Franz Dörner et al. Vol. Operations Research Proceedings (GOR (Gesellschaft für Operations Research e.V.)) Cham, Switzerland: Springer, pp. 453–458.

Halldórsson, Árni et al. (2007). "Applying advanced planning systems for supply chain planning: three case studies". In: *International Journal of Physical Distribution & Logistics Management*.

Harmon, Paul (2016). *The State of Business Process Management 2016*. Tech. rep. URL: http://www.bptrends.com/bpt/wp-content/uploads/2015-BPT-Survey-Report.pdf.

Hasachoo, N and R Masuchun (2016). "Reducing Schedule Nervousness in Production and Operations under Non-stationary Stochastic Demand: The Case of an Airline Catering Company". In: *2016 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*. Institute of Electrical and Electronics Engineers (IEEE). Bali, Indonesia, pp. 941–945. DOI: 10.1109/IEEM.2016.7798016.

Haskins, Cecilia (2011). "4.6. 1 A historical perspective of MBSE with a view to the future". In: *Incose international symposium*. Vol. 21. 1. Wiley Online Library, pp. 493–509.

Hegde, S. R. et al. (2004). "Decomposition System and Method for Solving a Large-scale Semiconductor Production Planning Problem". In: US Patent 6,701,201.

Herrera, Carlos, Sana Belmokhtar-Berraf, et al. (2016). "A Reactive Decision-Making Approach to Reduce Instability in a Master Production Schedule". In: *International Journal of Production Research* 54, pp. 2394–2404. DOI: 10.1080/00207543.2015.1078516. URL: https://doi.org/10.1080/00207543.2015.1078516.

Herrera, Carlos and André Thomas (2009). "Simulation of Less Master Production Schedule Nervousness Model". In: *IFAC Proceedings Volumes* 42, pp. 1585–1590. DOI: 10.3182/20090603-3-RU-2001.0554. URL: https://doi.org/10.3182/20090603-3-RU-2001.0554.

Herzog, Erik, Asmus Pandikow, and AB Syntell (2005). “SysML–an assessment”. In: *Syntell AB, SE* 100, p. 55.

Hettinger, Lawrence J et al. (2015). “Modelling and simulation of complex sociotechnical systems: Envisioning and analysing work environments”. In: *Ergonomics* 58.4, pp. 600–614.

Ho, C-J (2005). “Examining Dampening Effects for Alternative Dampening Procedures to Cope with System Nervousness”. In: *International Journal of Production Research* 43, pp. 4009–4033. DOI: 10.1080/00207540500147935. URL: https://doi.org/10.1080/00207540500147935.

Hofmann, M., J. Palii, and G. Mihelcic (Aug. 2011). “Epistemic and normative aspects of ontologies in modelling and simulation”. en. In: *Journal of Simulation* 5.3, pp. 135–146. ISSN: 1747-7786. DOI: 10.1057/jos.2011.13. URL: https://doi.org/10.1057/jos.2011.13 (visited on 09/05/2018).

Huan, Samuel H, Sunil K Sheoran, and Ge Wang (2004). “A Review and Analysis of Supply Chain Operations Reference (SCOR) Model”. In: *Supply Chain Management: An International Journal* 9, pp. 23–29.

Huldt, Tomas and Ivan Stenius (2019). “State-of-practice survey of model-based systems engineering”. In: *Systems Engineering* 22.2, pp. 134–145.

INCOSE, A (2014). “A world in motion: systems engineering vision 2025”. In: *International Council on Systems Engineering*.

Ivanov, Dmitry, Alexandre Dolgui, et al. (2017). “Literature review on disruption recovery in the supply chain”. In: *International Journal of Production Research* 55.20, pp. 6158–6174.

Ivanov, Dmitry, Boris Sokolov, and Alexandre Dolgui (2014). “The Ripple effect in supply chains: trade-off ‘efficiency-flexibility-resilience’in disruption management”. In: *International Journal of Production Research* 52.7, pp. 2154–2172.

Joannou, Demetrios et al. (Mar. 2019). “A Model-Based Engineering Methodology and Architecture for Resilience in Systems-of-Systems: A Case of Water Supply Resilience to Flooding”. en. In: *Water* 11.3, p. 496. ISSN: 2073-4441. DOI: 10.3390/w11030496.

URL: https://www.mdpi.com/2073-4441/11/3/496 (visited on 10/17/2019).

Johannessen, Stig (2005). "Supply Chain Management and the Challenge of Organizational Complexity—Methodological Considerations". In: *Research Methodologies in Supply Chain Management*. Springer, pp. 59–73.

Jung, Hosang (2010). "An Available-To-Promise Model Considering Customer Priority and Variance of Penalty Costs". In: *The International Journal of Advanced Manufacturing Technology* 49, pp. 369–377. DOI: 10.1007/s00170-009-2389-9. URL: https://doi.org/10.1007/s00170-009-2389-9.

Jupyter Notebook (Jan. 2021). URL: https://jupyter.org/index.html/ (visited on 01/19/2021).

KBSI (Sept. 2018). *Welcome to KBSI.com*. URL: https://www.kbsi.com/ (visited on 04/09/2018).

Kernan, Brian and Con Sheahan (Mar. 2010). "Development and construction of an ontology to represent simulation data for a generic enterprise". In: *Applied Ontology* 5.1, pp. 29–46. ISSN: 15705838. DOI: 10.3233/AO-2010-0073. URL: https://search.ebscohost.com/login.aspx?direct=true&db=a9h&AN=48776307&site=ehost-live (visited on 07/17/2018).

Kleijnen, Jack PC (2005). "Supply Chain Simulation Tools and Techniques: A Survey". In: *International Journal of Simulation and Process Modelling* 1, pp. 82–89.

Koca, Esra, Hande Yaman, and M Selim Aktürk (2018). "Stochastic Lot Sizing Problem With Nervousness Considerations". In: *Computers & Operations Research* 94, pp. 23–37. DOI: 10.1016/j.cor.2018.01.021. URL: https://doi.org/10.1016/j.cor.2018.01.021.

Kotzab, Herbert et al. (2006). *Research methodologies in supply chain management*. Springer Science & Business Media.

Laue, Ralf and Christian Mueller (June 2016). "The Business Process Simulation Standard (BPSIM): Chances And Limits". In: ECMS, pp. 413–418. ISBN: 978-0-9932440-2-5. DOI:

10.7148/2016-0413. URL: http://www.scs-europe.net/dlib/2016/2016-0413.htm (visited on 04/10/2018).

Law, Averill M (2013). *Simulation modeling and analysis.* Vol. 5. McGraw-Hill New York.

Law, Kincho H., Jack C.P. Cheng, and Moumita Das (Sept. 2015). "An ontology-based web service framework for construction supply chain collaboration and management". In: *Engineering, Construction and Architectural Management* 22.5, pp. 551–572. ISSN: 0969-9988. DOI: 10.1108/ECAM-07-2014-0089. URL: https://www.emeraldinsight.com/doi/full/10.1108/ECAM-07-2014-0089 (visited on 08/31/2018).

Lečić-Cvetković, Danica, Nikola Atanasov, and Sladan Babarogić (Dec. 2010). "An Algorithm for Customer Order Fulfillment in a Make-to-Stock Manufacturing System". en. In: *International Journal of Computers Communications & Control* 5.5, pp. 783–791. ISSN: 1841-9836. DOI: 10.15837/ijccc.2010.5.2238. URL: http://univagora.ro/jour/index.php/ijccc/article/view/2238 (visited on 01/29/2019).

Lee, Hau L, Vineet Padmanabhan, and Seungjin Whang (2004). "Information distortion in a supply chain: the bullwhip effect". In: *Management science* 50.12_supplement, pp. 1875–1886.

Levy, David L (1995). "International Sourcing and Supply Chain Stability". In: *Journal of International Business Studies* 26, pp. 343–360. DOI: 10.1057/palgrave.jibs.8490177. URL: https://doi.org/10.1057/palgrave.jibs.8490177.

Li, Qinyun and Stephen M Disney (2017). "Revisiting Rescheduling: MRP Nervousness and The Bullwhip Effect". In: *International Journal of Production Research* 55, pp. 1992–2012. DOI: 10.1080/00207543.2016.1261196. URL: https://doi.org/10.1080/00207543.2016.1261196.

Lin, Chao-Hsien, Sheue-Ling Hwang, and Eric Min-Yang Wang (2007). "A reappraisal on advanced planning and scheduling systems". In: *Industrial Management & Data Systems.*

Liston, Paul et al. (2010). “An evaluation of SysML to support simulation modeling”. In: *Conceptual Model. Discrete-Event Simul*, pp. 337–354.

MacDonald, John R et al. (2018). “Supply chain risk and resilience: theory building through structured experiments and simulation”. In: *International Journal of Production Research* 56.12, pp. 4337–4355.

Madni, Azad M and Michael Sievers (2018). “Model-based systems engineering: Motivation, current status, and research opportunities”. In: *Systems Engineering* 21.3, pp. 172–190.

Madni, Azad M. and Shatad Purohit (2019). “Economic Analysis of Model-Based Systems Engineering”. In: *Systems* 7.1, p. 12.

McGinnis, L et al. (2011). “Ontologies and simulation: a practical approach”. In: *Journal of Simulation* 5.3, pp. 190–201. ISSN: 1747-7778. DOI: 10.1057/jos.2011.3. URL: http://link.springer.com/10.1057/jos.2011.3.

Meistering, Malte and Hartmut Stadtler (2019). “Stabilized-cycle Strategy for a Multi-Item, Capacitated, Hierarchical Production Planning Problem in Rolling Schedules”. In: *Business Research* 12, pp. 1–36. DOI: 10.1007/s40685-019-0089-3. URL: https://doi.org/10.1007/s40685-019-0089-3.

Meyr, Herbert (Jan. 2009). “Customer segmentation, allocation planning and order promising in make-to-stock production”. en. In: *OR Spectrum* 31.1, pp. 229–256. ISSN: 1436-6304. DOI: 10.1007/s00291-008-0123-x. URL: https://doi.org/10.1007/s00291-008-0123-x (visited on 10/08/2018).

Modelio (Dec. 2018). *Modelio Open Source - UML and BPMN free modeling tool.* en-gb. URL: https://www.modelio.org/ (visited on 04/12/2018).

Mönch, Lars, Reha Uzsoy, and John W Fowler (2018a). “A Survey of Semiconductor Supply Chain Models Part I: Semiconductor Supply Chains, Strategic Network Design, and Supply Chain Simulation”. In: *International Journal of Production Research* 56, pp. 4524–4545. DOI: 10.1080/00207543.2017.1401233. URL: https://doi.org/10.1080/00207543.2017.1401233.

— (2018b). "A survey of semiconductor supply chain models part I: semiconductor supply chains, strategic network design, and supply chain simulation". In: *International Journal of Production Research* 56.13, pp. 4524–4545.

— (2018c). "A Survey of Semiconductor Supply Chain Models Part III: Master Planning, Production Planning, and Demand Fulfilment". In: *International Journal of Production Research* 56, pp. 4565–4584. DOI: 10.1080/00207543.2017.1401234. URL: https://doi.org/10.1080/00207543.2017.1401234.

Mönch, Lars et al. (Aug. 2011). "A survey of challenges in modelling and decision-making for discrete event logistics systems". en. In: *Computers in Industry*. Special Issue: Grand Challenges for Discrete Event Logistics Systems 62.6, pp. 557–567. ISSN: 0166-3615. DOI: 10.1016/j.compind.2011.05.001. URL: http://www.sciencedirect.com/science/article/pii/S0166361511000662 (visited on 05/28/2020).

Mordecai, Yaniv and Dov Dori (2013). "6.5. 1 I5: A Model-Based Framework for Architecting System-of-Systems Interoperability, Interconnectivity, Interfacing, Integration, and Interaction". In: *INCOSE international symposium*. Vol. 23. 1. Wiley Online Library, pp. 1234–1255.

Moscoso, Philip G, Jan C Fransoo, and Dieter Fischer (2010). "An Empirical Study on Reducing Planning Instability in Hierarchical Planning Systems". In: *Production Planning and Control* 21, pp. 413–426. DOI: 10.1080/09537280903454172. URL: https://doi.org/10.1080/09537280903454172.

Mousavi, Behrouz Alizadeh, Radhia Azzouz, and Cathal Heavey (2019). "Mathematical Modelling of Products Allocation to Customers for Semiconductor Supply Chain". In: *Procedia Manufacturing* 38, pp. 1042–1049.

Mousavi, Behrouz Alizadeh, Radhia Azzouz, Cathal Heavey, and Hans Ehm (2019a). "A Survey of Model-Based System Engineering Methods to Analyse Complex Supply Chains: A Case Study in Semiconductor Supply Chain". In: *IFAC-PapersOnLine* 52.13, pp. 1254–1259.

Mousavi, Behrouz Alizadeh, Radhia Azzouz, Cathal Heavey, and Hans Ehm (2019b). "Simulation-Based Analysis of the Nervousness within Semiconductors Supply Chain Planning: Insight from a Case Study". In: *2019 Winter Simulation Conference (WSC)*. IEEE, pp. 2396–2407.

Munirathinam, Sathyan and Balakrishnan Ramadoss (2016). "Predictive models for equipment fault detection in the semiconductor manufacturing process". In: *IACSIT International Journal of Engineering and Technology* 8.4, pp. 273–285.

Nikolaidou, Mara et al. (2015). "Simulating SysML models: Overview and challenges". In: *2015 10th System of Systems Engineering Conference, SoSE 2015*. ISBN: 978-1-4799-7611-9. DOI: 10.1109/SYSOSE.2015.7151961.

Oliveira, Josenildo Brito, Renato Silva Lima, and José Arnaldo Barra Montevechi (2016). "Perspectives and Relationships in Supply Chain Simulation: A Systematic Literature Review". In: *Simulation Modelling Practice and Theory* 62, pp. 166–191. DOI: 10.1016/j.simpat.2016.02.001. URL: https://doi.org/10.1016/j.simpat.2016.02.001.

Onggo, B. S. S. (Mar. 2009). "Towards a unified conceptual model representation: a case study in healthcare". en. In: *Journal of Simulation* 3.1, pp. 40–49. ISSN: 1747-7778, 1747-7786. DOI: 10.1057/jos.2008.14. URL: https://link.springer.com/article/10.1057/jos.2008.14 (visited on 04/12/2018).

Onggo, B. S. S. et al. (May 2018). "A BPMN extension to support discrete-event simulation for healthcare applications: an explicit representation of queues, attributes and data-driven decision points". In: *Journal of the Operational Research Society* 69.5, pp. 788–802. ISSN: 0160-5682. DOI: 10.1057/s41274-017-0267-7. URL: https://doi.org/10.1057/s41274-017-0267-7 (visited on 04/09/2018).

Onggo, Bhakti SS (2014). "Agent-based simulation model representation using BPMN". In: *Formal languages for computer simulation: Transdisciplinary models and applications*. IGI Global, pp. 378–400.

Panahifar, Farhad et al. (2018). "Supply chain collaboration and firm's performance". In: *Journal of Enterprise Information Management*.

Pavlov, Alexander et al. (2017). "Hybrid fuzzy-probabilistic approach to supply chain resilience assessment". In: *IEEE Transactions on Engineering Management* 65.2, pp. 303–315.

Plattner, Hasso et al. (2013). "Available-To-Promise on an In-Memory Column Store". In: US Patent 8,601,038.

Polwin, Robert (2018). "Influence of the Order Batch Structure on Early Warnings". MA thesis. University Duisburg-Essen.

Ponsignon, Thomas (July 14, 2006). "The Bullwhip Effect in the Semiconductor Industry on the Example of Infineon Technologies". Master's thesis. DOI: 10.13140/2.1.3498.6241.

Ponsignon, Thomas and Lars Mönch (2014). "Simulation-based performance assessment of master planning approaches in semiconductor manufacturing". In: *Omega* 46, pp. 21–35.

Prasad, Sameer and Jayavel Sounderpandian (2003). "Factors influencing global supply chain efficiency: implications for information systems". In: *Supply Chain Management: An International Journal.*

Productive4.0 (2017). *Productive 4.0 - A European co-funded innovation and lighthouse project on Digital Industry.* en-US. Library Catalog: productive40.eu. URL: https://productive40.eu/ (visited on 06/02/2020).

Protégé (2020). URL: https://protege.stanford.edu/ (visited on 05/29/2020).

Proudlove, N. C., S. Bisogno, B. S. S. Onggo, A. Calabrese, and N. Levialdi Ghiron (Dec. 2017). "Towards fully-facilitated discrete event simulation modelling: Addressing the model coding stage". In: *European Journal of Operational Research* 263.2, pp. 583–595. ISSN: 0377-2217. DOI: 10.1016/j.ejor.2017.06.002. URL: http://www.sciencedirect.com/science/article/pii/S0377221717305209 (visited on 10/08/2019).

Proudlove, NC, Stefania Bisogno, Bhakti SS Onggo, Armando Calabrese, and N Levialdi Ghiron (2017). "Towards fully-facilitated discrete event simulation modelling: Addressing the model coding stage". In: *European Journal of Operational Research* 263.2, pp. 583–595.

Pujawan, I Nyoman and Alison U Smart (2012). "Factors Affecting Schedule Instability in Manufacturing Companies". In: *International Journal of Production Research* 50, pp. 2252–2266.

Pundoor, Guruprasad and Jeffrey W Herrmann (2006). "A Hierarchical Approach to Supply Chain Simulation Modelling Using the Supply Chain Operations Reference model". In: *International Journal of Simulation and Process Modelling* 2, pp. 124–132.

Quante, Rainer, Herbert Meyr, and Moritz Fleischmann (2009). "Revenue management and demand fulfillment: matching applications, models and software". en. In: *Supply Chain Planning: Quantitative Decision Support and Advanced Planning Solutions*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 1–32. ISBN: 978-3-540-93775-3. DOI: 10.1007/978-3-540-93775-3_3. URL: https://doi.org/10.1007/978-3-540-93775-3_3 (visited on 10/31/2018).

RAMÍREZ, PANTOJA (2015). "Reducing The Number Of Negative Early Warnings In The Order Management System Of The Semiconductor Industry". MA thesis. Universidad Politécnica de Madrid.

Ramos, Ana Luísa, José Vasconcelos Ferreira, and Jaume Barceló (2011). "Model-based systems engineering: An emerging approach for modern systems". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 42.1, pp. 101–111.

Rashid, Muhammad, Muhammad Waseem Anwar, and Aamir M. Khan (Aug. 2015). "Toward the tools selection in model based system engineering for embedded systems—A systematic literature review". In: *Journal of Systems and Software* 106, pp. 150–163. ISSN: 0164-1212. DOI: 10.1016/j.jss.2015.04.089. URL: http://www.sciencedirect.com/science/article/pii/S016412121500103X (visited on 10/17/2019).

Rosing, Mark von et al. (2015). *Business Process Model and Notation-BPMN*.

Sabri, Yasmine (2019). "In pursuit of supply chain fit". In: *The International Journal of Logistics Management*.

Santa-Eulalia, Luis Antonio, Sophie D'Amours, and Jean-Marc Frayret (2012). "Agent-based simulations for advanced supply chain planning and scheduling: The FAMASS methodological framework for requirements analysis". In: *International Journal of Computer Integrated Manufacturing* 25.10, pp. 963–980.

Sargent, Robert G (2013). "Verification and validation of simulation models". In: *Journal of simulation* 7.1, pp. 12–24. DOI: 10.1057/jos.2012.20. URL: https://doi.org/10.1057/jos.2012.20.

Schneeweiss, Christoph and Kirstin Zimmer (2004). "Hierarchical coordination mechanisms within the supply chain". In: *European Journal of Operational Research* 153.3, pp. 687–703.

Seitz, Alexander (2012). "Analysis of the negative Early Warnings of Infineon's Order Confirmation System". MA thesis. Technical University of Munich.

— (2017). "Data driven approaches increasing robustness, accuracy, and service levels of industrial demand fulfilment". PhD thesis. TECHNICAL UNIVERSITY OF MUNICH.

Seitz, Alexander and Martin Grunow (Feb. 2017). "Increasing accuracy and robustness of order promises". In: *International Journal of Production Research* 55.3, pp. 656–670. ISSN: 0020-7543. DOI: 10.1080/00207543.2016.1195024. URL: https://doi.org/10.1080/00207543.2016.1195024 (visited on 06/06/2018).

Seitz, Alexander, Martin Grunow, and Renzo Akkerman (2016). "Data Driven Supply Allocation to Individual Customers Considering Forecast Bias". en. In: *SSRN Electronic Journal*. ISSN: 1556-5068. DOI: 10.2139/ssrn.2813835. URL: https://www.ssrn.com/abstract=2813835 (visited on 06/06/2018).

Silver, Gregory A et al. (2011). "An ontology for discrete-event modeling and simulation". In: *Simulation* 87.9, pp. 747–773.

Sivadasan, Suja et al. (2013). "Reducing Schedule Instability by Identifying and Omitting Complexity-Adding Information Flows at the Supplier–Customer Interface". In: *International Journal of Production Economics* 145, pp. 253–262. DOI: 10.1016/j.

ijpe.2013.04.043. URL: https://doi.org/10.1016/j.ijpe.2013.04.043.

Stadtler, Hartmut (2005). "Supply chain management and advanced planning—-basics, overview and challenges". In: *European journal of operational research* 163.3, pp. 575–588.

Stadtler, Hartmut, Christoph Kilger, and Herbert Meyr (2015). *Supply Chain Management and Advanced Planning: Concepts, Models, Software, and Case Studies.* 5th. Berlin Heidelberg: Springer-Verlag. DOI: 10.1007/978-3-642-55309-7. URL: https://doi.org/10.1007/978-3-642-55309-7.

Steele, DANIEL C (1975). "The Nervous MRP System: How to Do Battle". In: *Production and Inventory Management* 16, pp. 83–89.

Steger-Jensen, Kenn et al. (2011). "Advanced planning and scheduling technology". In: *Production Planning & Control* 22.8, pp. 800–808.

Studer, Rudi, V Richard Benjamins, and Dieter Fensel (1998). "Knowledge engineering: principles and methods". In: *Data & knowledge engineering* 25.1-2, pp. 161–197.

SysML ISO Standard Document (2017). *ISO/IEC 19514:2017.* en. Library Catalog: www.iso.org. URL: https://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/06/52/65231.html (visited on 05/28/2020).

Tiacci, Lorenzo and Stefano Saetta (2012). "Demand Forecasting, Lot Sizing and Scheduling on a Rolling Horizon Basis". In: *International Journal of Production Economics* 140, pp. 803–814. DOI: 10.1016/j.ijpe.2012.02.007. URL: https://doi.org/10.1016/j.ijpe.2012.02.007.

Uzsoy, Reha, John W Fowler, and Lars Mönch (2018). "A survey of semiconductor supply chain models Part II: demand planning, inventory management, and capacity planning". In: *International Journal of Production Research* 56.13, pp. 4546–4564. DOI: 10.1080/00207543.2018.1424363. URL: https://doi.org/10.1080/00207543.2018.1424363.

Vallejo, Carlos, David Romero, and Arturo Molina (2012). "Enterprise integration engineering reference framework and toolbox". In: *International journal of production research* 50.6, pp. 1489–1511.

Van Der Zee, Durk-Jouke and Jack GAJ Van Der Vorst (2005). "A modeling framework for supply chain simulation: opportunities for improved decision making". In: *Decision sciences* 36.1, pp. 65–95.

Vernadat, François (2020). "Enterprise modelling: Research review and outlook". In: *Computers in Industry* 122, p. 103265.

Vespignani, Alessandro (2012). "Modelling dynamical processes in complex socio-technical systems". In: *Nature physics* 8.1, p. 32.

Vidoni, Melina C and Aldo R Vecchietti (2015). "A systemic approach to define and characterize Advanced Planning Systems (APS)". In: *Computers & Industrial Engineering* 90, pp. 326–338.

Virtual Factory Lab (2020). *Virtual Factory Lab | Georgia Institute of Technology | Atlanta, GA*. URL: https://factory.isye.gatech.edu/ (visited on 04/28/2020).

Vogel, Sebastian and Herbert Meyr (Oct. 2015). "Decentral allocation planning in multistage customer hierarchies". In: *European Journal of Operational Research* 246.2, pp. 462–470. ISSN: 0377-2217. DOI: 10.1016/j.ejor.2015.05.009. URL: http://www.sciencedirect.com/science/article/pii/S0377221715003811 (visited on 06/06/2018).

W3C (2012). *OWL - Semantic Web Standards*. URL: https://www.w3.org/OWL/ (visited on 06/01/2020).

Wagensonner, Lukas (2018). "Modeling and Assessment of Available-to-Promise Nervousness: A Case Study in the Semiconductor Industry". MA thesis. Technical University of Munich.

Wagner, Andreas (1999). "Causality in complex systems". In: *Biology and Philosophy* 14.1, pp. 83–101.

Walden, David D et al. (2015). *Systems engineering handbook: A guide for system life cycle processes and activities*. John Wiley & Sons.

Wiers, Vincent CS and A Ton G de Kok (2017). *Designing, Selecting, Implementing and Using APS Systems*. Springer.

Willard, Brian (2007). "UML for systems engineering". In: *Computer Standards & Interfaces* 29.1, pp. 69–81.

Williams, H Paul (2013). *Model building in mathematical programming*. John Wiley & Sons.

Wu, Teresa, Jennifer Blackhurst, and Peter O'Grady (2007). "Methodology for supply chain disruption analysis". In: *International journal of production research* 45.7, pp. 1665–1682.

Yang, Lan, Kathryn Cormican, and Ming Yu (2019). "Ontology-based systems engineering: A state-of-the-art review". In: *Computers in Industry* 111, pp. 148–171.

Yildiz, Hakan et al. (2016). "Reliable supply chain network design". In: *Decision Sciences* 47.4, pp. 661–698.

Zaletelj, Viktor et al. (Oct. 2018). "A foundational ontology for the modelling of manufacturing systems". In: *Advanced Engineering Informatics* 38, pp. 129–141. ISSN: 1474-0346. DOI: 10.1016/j.aei.2018.06.009. URL: http://www.sciencedirect.com/science/article/pii/S1474034616301264 (visited on 08/27/2018).

Zarour, Karim et al. (2019). "A systematic literature review on BPMN extensions". In: *Business Process Management Journal*.

Zdravković, Milan et al. (Nov. 2011). "An approach for formalising the supply chain operations". en. In: *Enterprise Information Systems* 5.4, pp. 401–421. ISSN: 1751-7575, 1751-7583. DOI: 10.1080/17517575.2011.593104. URL: http://www.tandfonline.com/doi/abs/10.1080/17517575.2011.593104 (visited on 08/24/2018).

Zhou, Honggeng et al. (2011). "Supply chain integration and the SCOR model". In: *Journal of Business Logistics* 32.4, pp. 332–344.

Zoryk-Schalla, Anastasia J, Jan C Fransoo, and Ton G de Kok (2004). "Modeling the planning process in advanced planning systems". In: *Information & management* 42.1, pp. 75–87.

Appendix A

Appendix

A.1 List and aim of meetings

This thesis was carried out in collaboration with the case study. Here we list the meetings we organized and held with the company in carrying out this research. The types and purpose of the meetings are as listed below. Table A.1 on page 235 lists these meetings.

1. Review Meetings (Online): A series of meetings held with the project manager at the case study every Friday. These meetings aim to update the project plan, discuss contributions and challenges, and organize tasks. It started in January 2018.

2. Monthly Meetings (Online): A series of meeting held with the Lead Principal Supply Chain at the case study, [redacted] and the [redacted]. These meetings aim to report progress and discuss scientific challenges with shareholders. It started in September 2018.

3. Management Meeting (Online): A series of meetings held quarterly with the case study managers. These meetings were designed to align project directions and report progress to the case study strategic managers. It started in September 2018.

4. Supply Chain [redacted] (Online): A weekly meeting in the case study held every Fridays. These group meetings were all researchers studying supply

chains at the case study presented their research topics and shared their issues. Participated from August 2018 to January 2019.

5. Meetings with Planners (Online/In-person): A series of on-demand meetings with human planners in the case study company. These meetings were held to investigate their decision-making process and understand decision conditions and formulate identified issues. Later we held a series of meetings to introduce and test the developed decision support tool (ReCAST). These meetings were held according to demands.

6. Meeting with the case study's supply chain experts (Online/In-person): A series of meeting with the case study supply chain experts on simulation modeling, allocation planning and KPI evaluation. These meetings were held according to demands.

7. Meeting with the case study IT expert: Discussion on deployment and requirements of ReCAST with the case study IT experts. These meetings were held according to demands.

8. Two-week meeting: This was held at the case study over a two week period where we studied the case study supply chain planning system and challenges. In these two week series of group and individual meetings were held. July 2018.

A.2 Simulation model

The developed simulation model comprises of abstracted various elements and modules in the planning system. In this appendix, these elements are briefly described. The main agent of the simulation model shows all agents in the simulation model (see Figure A.1 on page 236).

As shown in Figure A.2 on page 236 the simulation model reads customer data from databases. The "orderstbl" consist of order ID, customer ID, requested product type, RMAD, and requested quantity.

Table A.1: List of meetings that UoL held with Industry Partner for project Productive 4.0.

	Title	Location	Start Date	Reoccurrence	Aims
1	Review Meetings	Online	Jan 2018	Every Friday	These meetings aim to update the project plan, discuss contributions and challenges, and organize tasks
2	Monthly Meetings	Online	Sep 2018	Monthly	These meetings aim to report progress and discuss scientific challenges with shareholders.
3	Management Meetings	Online	Sep 2018	Quarterly	These meetings designed to align project directions and report progress to the case study strategic managers.
4	Supply Chain Innovation [redacted]	Online	Aug 2018	Weekly until Jan 2019	These group meetings were held with all researchers at the case study to discuss their issue and exchange other topics.
5	Meeting with Planners	Online or In-Person	Jan 2018	Based on demands	These meetings held to investigate their decision-making process and understand decision conditions and formulate identified issues. Later we held a series of meetings to introduce and test developed decision support tool (ReCAST).
6	Meeting with the case study's supply chain experts	Online or In-Person	Jan 2018	Based on demands	To Investigate the case study Planning system and discuss simulation modeling, allocation planning and KPI evaluation.
7	Meetings wirth the case study IT experts	Online	Jan 2019	Based on demands	Discussion on deployment and requirement of ReCAST with the case study IT experts.
8	Two-Week Meeting at the case study-[redacted]	Munich	July 2018	Two week visit	For studying supply chain planning system and challenges in the case study. In these two week series of meeting with individual meetings held.

Input demands in the customer's table reads and prioritize in order promising and demand planning A.3 on page 237. Collection here is used to store a list of variables used to model the planning horizon.

In capacity planning, there are sets of parameters which show the available capacities. These parameters are updated by production agents within master planning. These agents are simplified, where master planning is modeled in much more detail as shown in Figure A.4 on page 238. Functions in this agent are coded for the specific part of master planning. The result of ATP deviation is stored in Excel files for further analysis outside of the AnyLogic software.

For example, "maATP" is a function for calculation of ATP in MA dimension. It shows how the collections (representation of RAM in APS) update by getting data from other agents (see Figure A.5 on page 238). The code in this function is shown in properties of function in this figure.

In physical production, each production type (FE, BE, logistic) consist of several

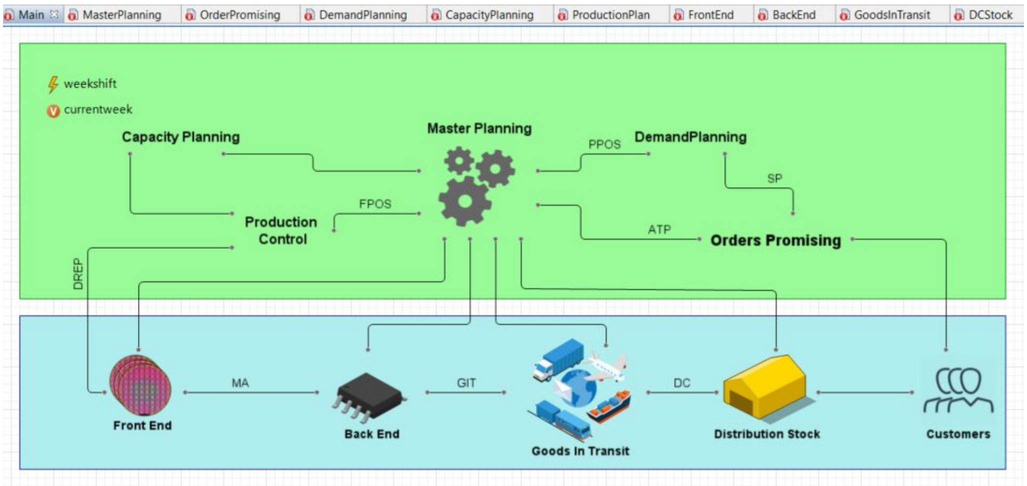


Figure A.1: Main agent in AnyLogic.

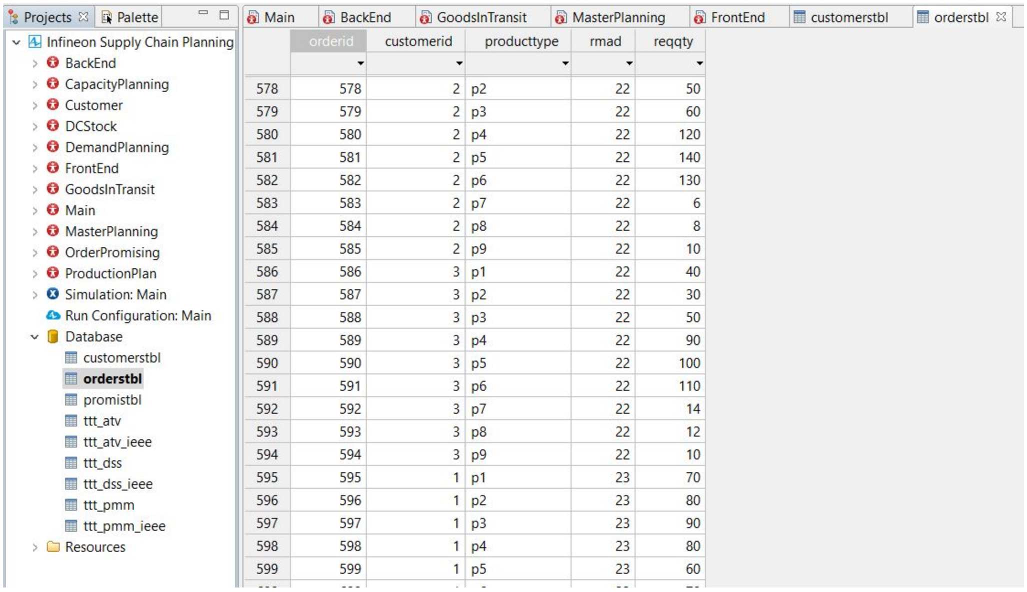


Figure A.2: Database tables in AnyLogic.

agents. Figure A.6 on page 239 show the front end and properties of service time for each FE. Figure A.7 on page 239 show the back end sites and selected properties for update events. The logistic system also shown in Figure A.8 on page 239. It identifies all the modeled routes for moving product from the back end to distribution centres. The properties of "empDistGIT1" also shows the distribution applied for service time of shipment which is used by the the case study databases.

Finally, the simulation run dashboard is shown in Figure A.9 on page 240. This dashboard could be extended by adding visualization and input parameters.

The source file of the simulation model attached shows the resources used in this

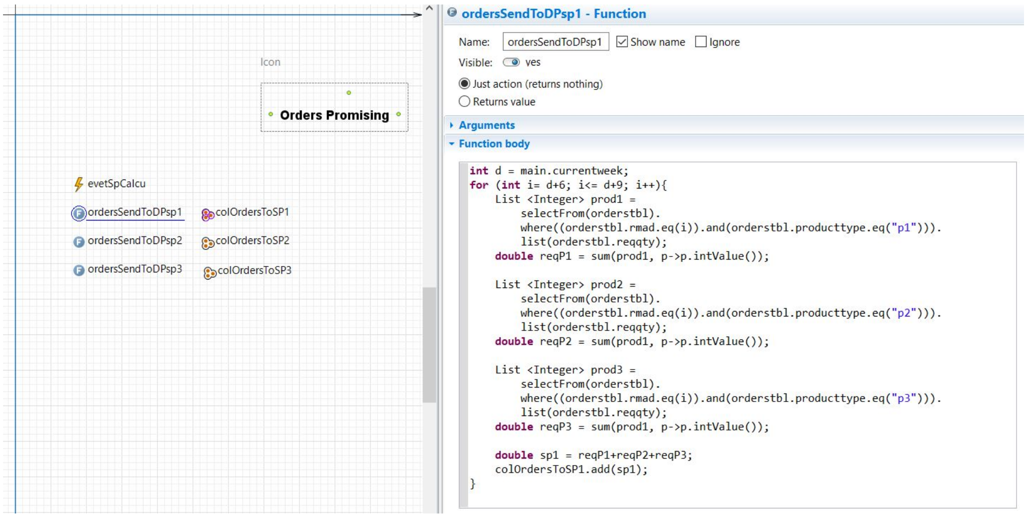


Figure A.3: Order Promising agent and functions in AnyLogic.

thesis. For further investigation of modeling and function codes please refer to this model.

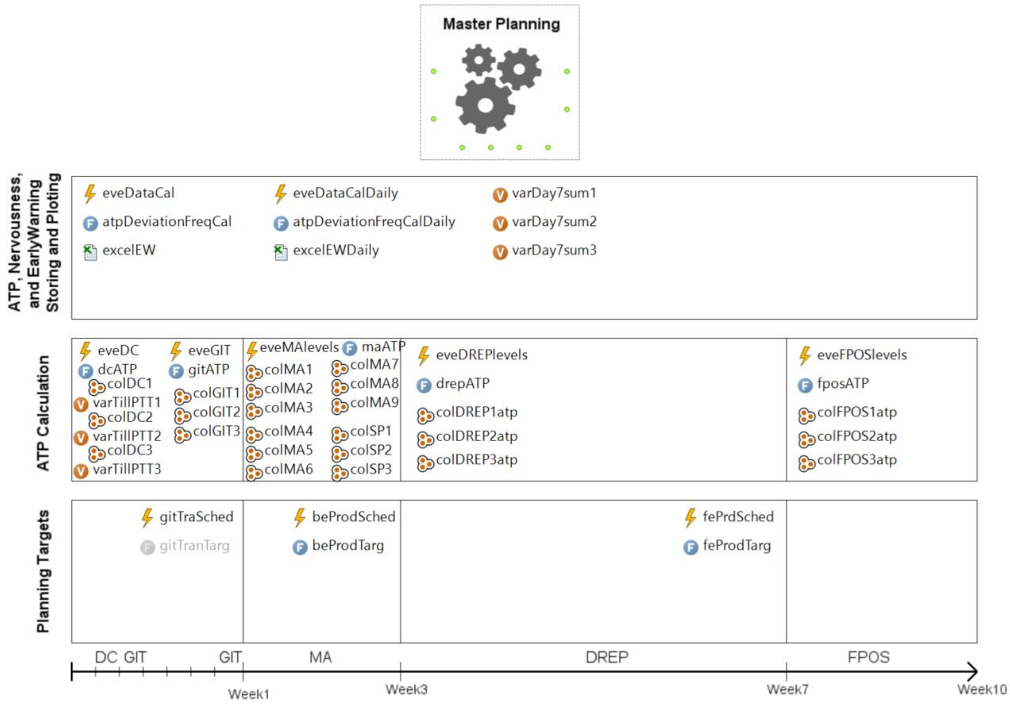


Figure A.4: master planning agent in AnyLogic.

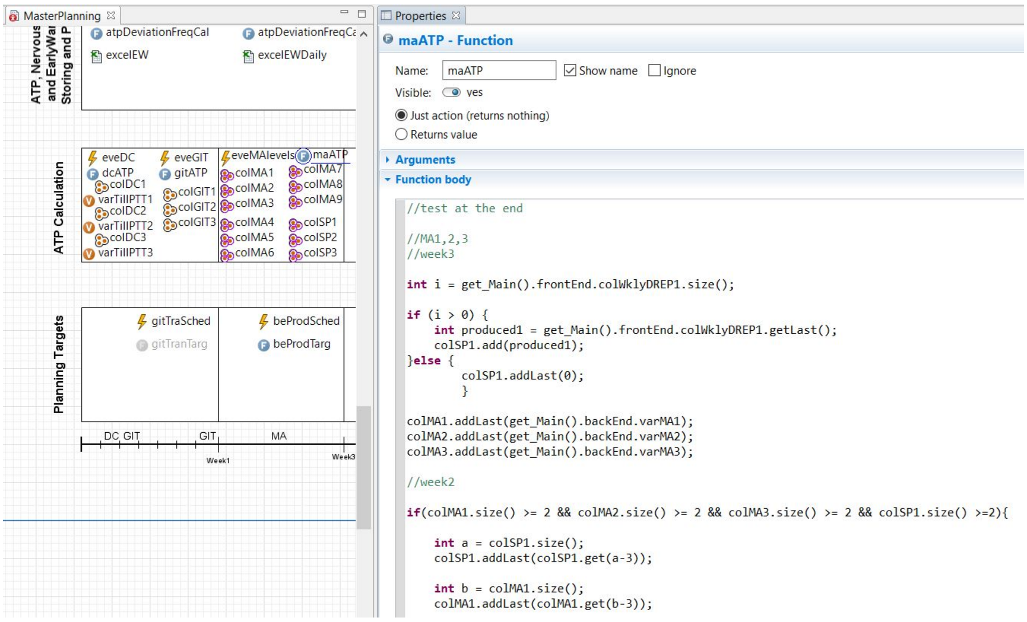


Figure A.5: maATP function in master planning.

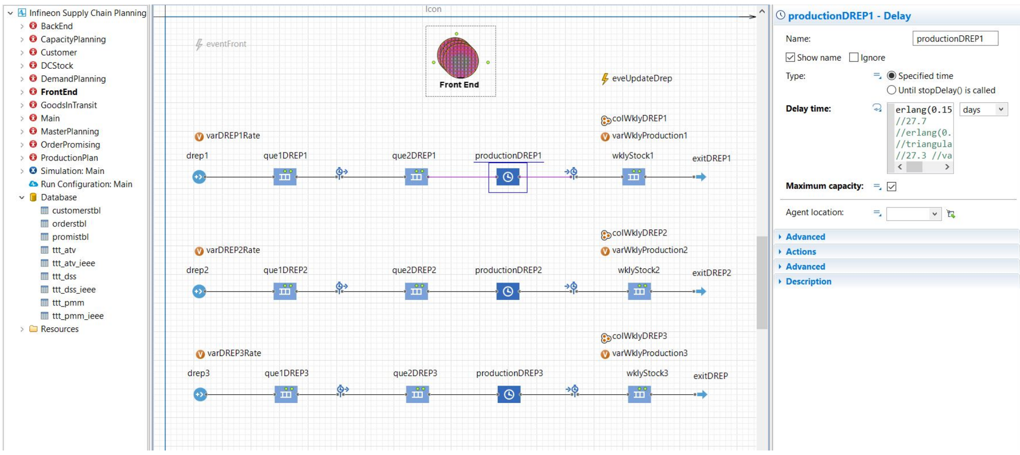


Figure A.6: Front end agent in physical system.

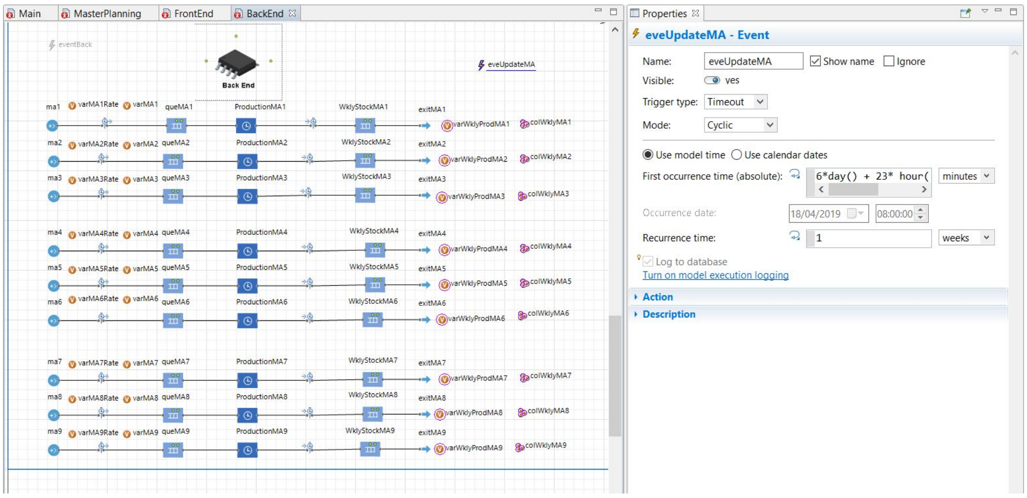


Figure A.7: Back end agent in physical system.

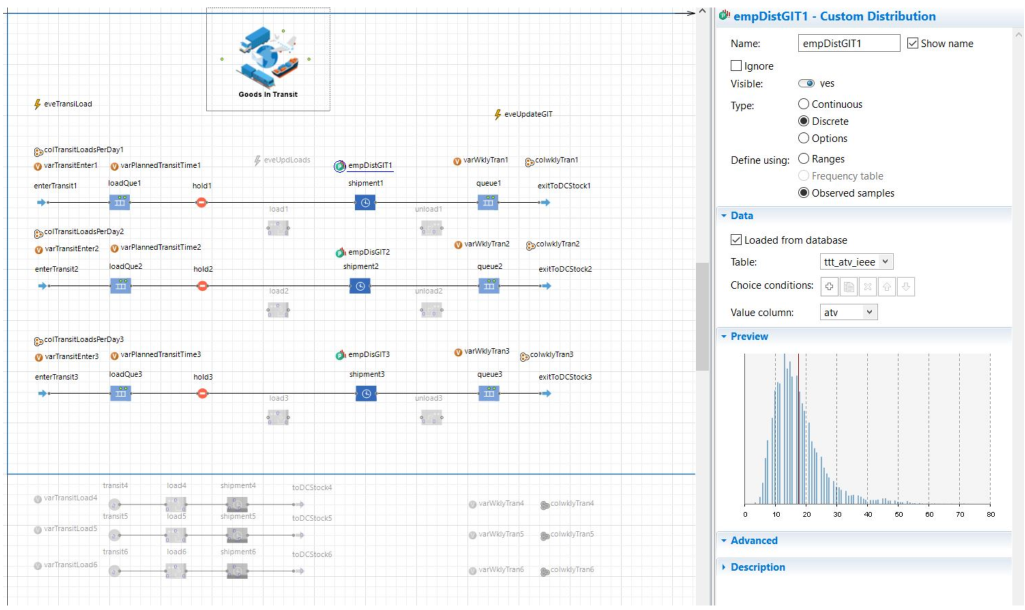


Figure A.8: Logistic agent in physical system.

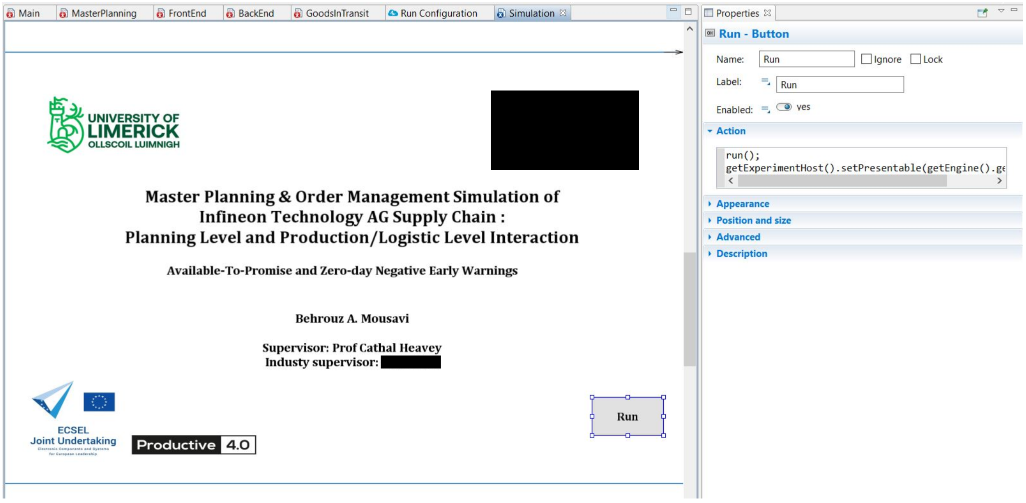


Figure A.9: Simulation dashboard.

A.3 Verification results of simulation model

As discussed in Section 5.7 on page 141, the verification of the simulation model was performed based on the proposed scenarios and results for RH 14 presented as a sample of the results. Here we provide two other selected Rolling Horizon 10 and 20 to provide further results in this regard. Figures A.10 on the facing page and A.11 on page 242 are the plotted average deviation in ATP for RH 10 and 20 based on the proposed scenarios.

A.4 Validation and experimentation results

The frequency values of planned delivery days for the whole RH belonging to the first logistic route is discussed and presented in Section 5.9 on page 149. Figures A.12 on page 243 and A.13 on page 243 are for logistic routes 2 and 3.

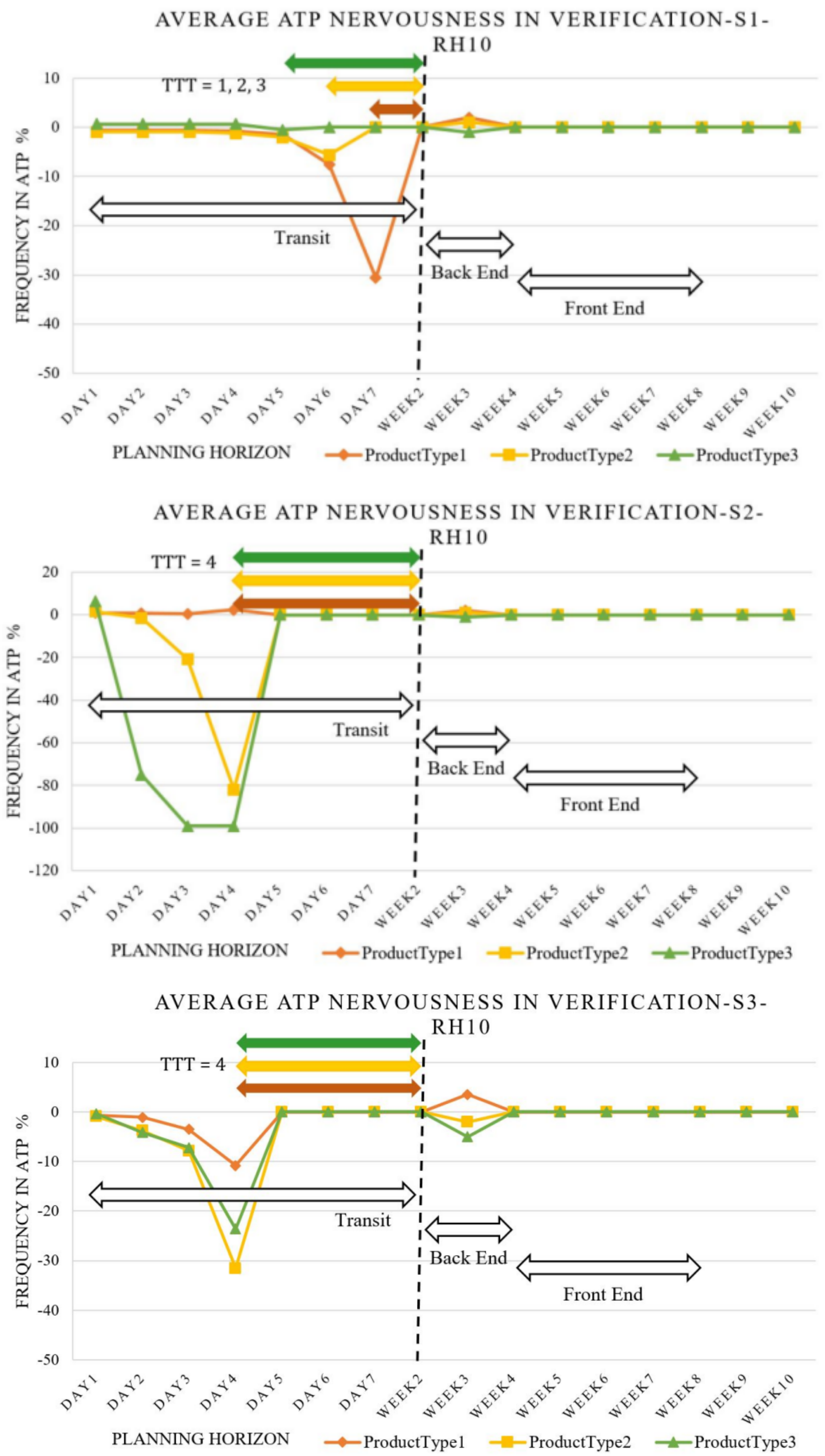


Figure A.10: Results of verification according to the values in Table 5.1 for RH 10.

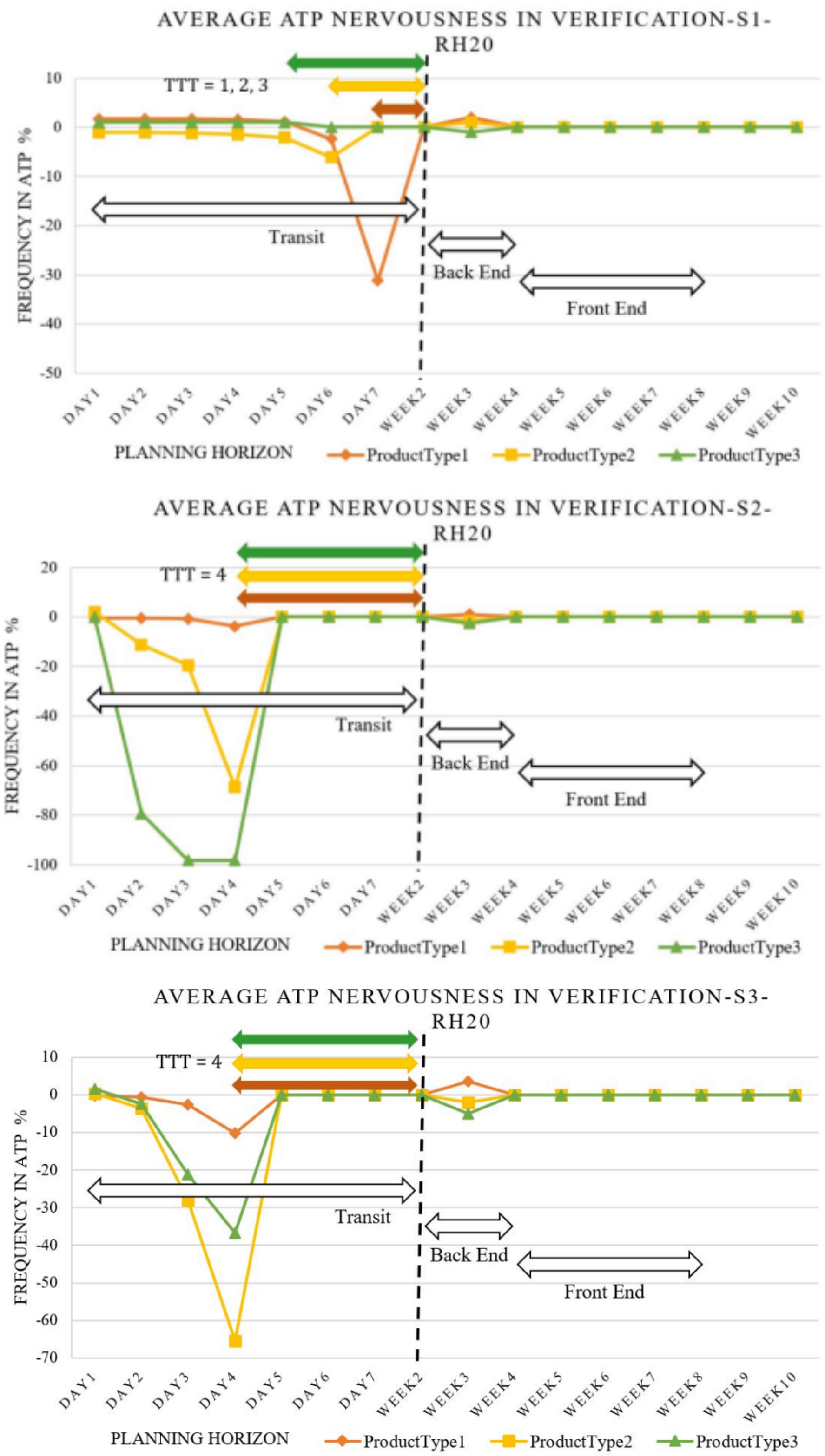


Figure A.11: Results of verification according to the values in Table 5.1 for RH 20.

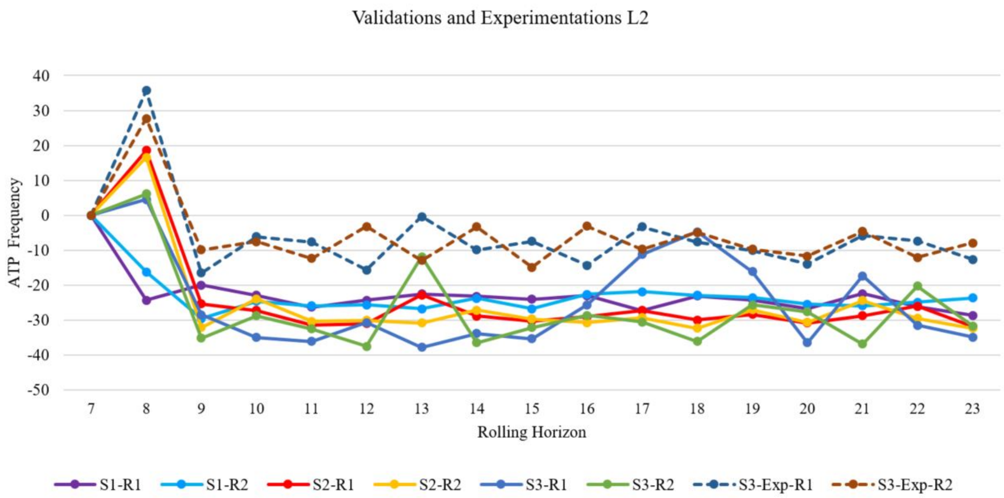


Figure A.12: Experimentation and validation outputs comparison for Logistic Route 2.

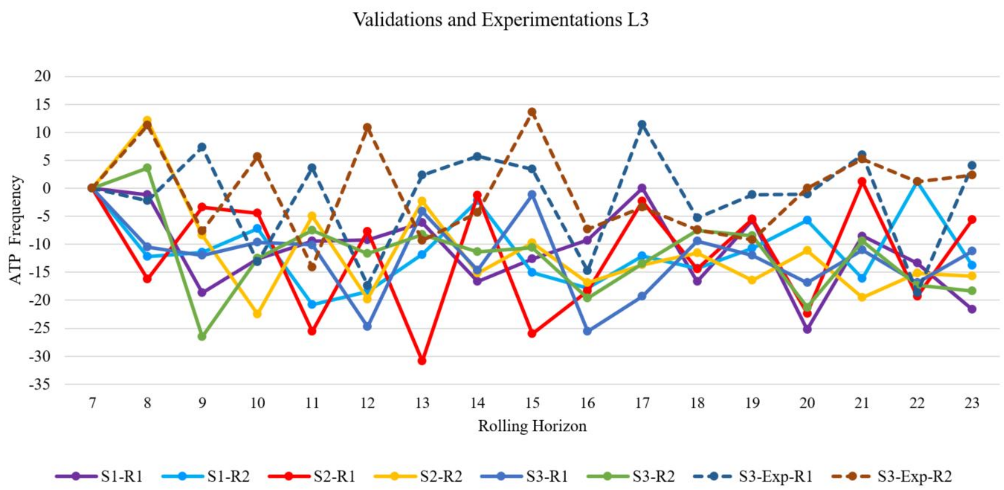


Figure A.13: Experimentation and validation outputs comparison for Logistic Route 3.

A.5 Data analysis in R

In this appendix, a part of the source code of data preparation for the simulation model is presented. It is based on the R language and aims to extract information from TMA data. The outputs of this code provide clean data to be used in simulation model. To reuse this code, the TMA data should be uploaded with the name 'TMA'. And the dates should be updated. The reason we only present part of the code is to avoid duplication that conveys the core elements of the code. Note that the full code is provided with the thesis materials.

```

## ATV explanatory diagrams.
```

```
# This part of the code aims to visualize the delivery delay of each
 product.
O_Div = numeric(nrow(TMA))
for(i in 1:nrow(TMA)) {
  if ((TMA$DIV[i] == "ATV") &&
      (TMA$`TT Total (Actual)` >= 25)){
      O_Div[i] = c(TMA$`TT Total (Actual)`[i])
  }
  else O_Div[i] = c("0")
}
plot(O_Div)
index_ATV = which(TMA$DIV =="ATV")
Product_ATV = ceiling( ((TMA$`TT Total
              (Actual)`[index_ATV])/24))
plot(Product_ATV)
hist(Product_ATV, breaks = 100)

## Plotting delays based on days...
# for specific product type and DC
# centre in chosen planning horizon on data on Import SAP

indx = which( (TMA$`ShipTo XXX Loc`=="DCA")
              & (TMA$DIV =="ATV")
              & (TMA$`T0 - SAP Import Date` >=
              as.POSIXct("2019-06-13 00:00:00")) &
                (TMA$`T0 - SAP Import Date` <=
                as.POSIXct("2019-06-19 00:00:00")))

plot(TMA$`T0 - SAP Import Date`[indx],
      (TMA$`TT3 (Actual)`[indx]),
     xlab = "Calandar Time",  ylab = "Actual TT3 (hour)",
     main = "DCA - ATV - TT3" )
abline(h = 10, col="blue")


```

```
## Portion of delays per days in different time frame
# It is similar to histogram.
in_time <- paste("")
indx_Tot <- which( (TMA$`ShipTo XXX Loc`=="DCA") &
                (TMA$DIV =="ATV") &
                (TMA$`T0 - SAP Import Date` >=
                as.POSIXct("2019-05-01 00:00:00")) &
                (TMA$`T0 - SAP Import Date` <=
                as.POSIXct("2019-08-20 00:00:00")))

len_indx_Tot = length(indx_Tot)

indx_01 = which(ceiling(
          TMA$`TT Total (Actual)`[indx_Tot]/24) == 1)
len_indx_01 = length(indx_01)

indx_02 = which(ceiling(
          TMA$`TT Total (Actual)`[indx_Tot]/24) == 2)
len_indx_02 = length(indx_02)

indx_03 = which(ceiling(
          TMA$`TT Total (Actual)`[indx_Tot]/24) == 3)
len_indx_03 = length(indx_03)

indx_04 = which(ceiling(
          TMA$`TT Total (Actual)`[indx_Tot]/24) >= 4)
len_indx_04 = length(indx_04)

for(i in 1:4){
  nam <- paste("len_indx_0", i, sep = "")
  print(get(nam) / len_indx_Tot)
  }

## Analyzing the difference of ATT 1,2, and 3.
indx = which( (TMA$`ShipTo XXX Loc`=="DCA")
```

```
                & (TMA$DIV =="ATV")
                & (TMA$`T2 - Pickup by Forwarder` >=
                as.POSIXct("2019-05-13 00:00:00")) &
                  (TMA$`T2 - Pickup by Forwarder` <=
                as.POSIXct("2019-05-19 00:00:00")))

plot(TMA$`T2 - Pickup by Forwarder`[indx],
      ceiling((TMA$`TT Total (Actual)`[indx])/24),
     xlab = "Calandar Time (Week)",
     ylab = "Actual TT (hours)" )
abline(h = 24, col="blue")

  ##based on pick yup date + TT3 actual mines target

indx = which( (TMA$`ShipTo XXX Loc`=="DCA")
                & (TMA$DIV =="ATV")
                & (TMA$`T2 - Pickup by Forwarder` >=
                as.POSIXct("2019-06-21 ")) &
                  (TMA$`T2 - Pickup by Forwarder` <=
                  as.POSIXct("2019-06-22 ")))

plot(TMA$`T2 - Pickup by Forwarder`[indx],
      (TMA$`TT3 (Actual)`[indx]) - (TMA$`TT3 (Target)`[indx]) ,
     xlab = "Calandar Time",
     ylab = "Actual Transit Time (per hour)" )
abline(h = 24, col="blue")


##clean TT2 from negative value

indx_TT2_Clean = which((TMA$`ShipTo XXX Loc`== "DCA")
                        & (TMA$DIV == "ATV")
                        & (TMA$`TT2 (Actual)` > 0))
print(TMA$`TT2 (Actual)`[indx_TT2_Clean])
hist(TMA$`TT2 (Actual)`[indx_TT2_Clean])
```

```


## Centeral mean calculation on days/month
rec_Prob <- c()
seq_day = seq.Date(from = as.Date("2019/05/01 00:00:00") ,
to = as.Date("2019/07/31 00:00:00"), by = "week" )

for(i in 1:length(seq_day)-1) {
  nam_date <- seq_day[i]
  nam_date_nxt <- seq_day[i+1]
  indx_Tot <- which( (TMA$`ShipTo XXX Loc`=="DCA") &
                       (TMA$DIV =="ATV") &
                       (TMA$`TT2 (Actual)` > 0) &
                       (TMA$`T0 - SAP Import Date` >=
                       as.POSIXct(nam_date)) &
                       (TMA$`T0 - SAP Import Date` <
                       as.POSIXct(nam_date_nxt)))

  indx_1 = which(TMA$`TT2 (Actual)`[indx_Tot] == 1)
  len_indx_1 = length(indx_1)

  indx_2 = which(TMA$`TT2 (Actual)`[indx_Tot] == 2)
  len_indx_2 = length(indx_2)

  indx_3 = which(TMA$`TT2 (Actual)`[indx_Tot] == 3)
  len_indx_3 = length(indx_3)

  indx_4 = which(TMA$`TT2 (Actual)`[indx_Tot] == 4)
  len_indx_4 = length(indx_4)

  indx_5 = which(TMA$`TT2 (Actual)`[indx_Tot] == 5)
  len_indx_5 = length(indx_5)

  indx_6 = which(TMA$`TT2 (Actual)`[indx_Tot] == 6)
  len_indx_6 = length(indx_6)
```

```

  indx_7 = which(TMA$`TT2 (Actual)`[indx_Tot] == 7)
  len_indx_7 = length(indx_7)

  indx_8 = which(TMA$`TT2 (Actual)`[indx_Tot] == 8)
  len_indx_8 = length(indx_8)

  indx_9 = which(TMA$`TT2 (Actual)`[indx_Tot] == 9)
  len_indx_9 = length(indx_9)

  indx_10 = which(TMA$`TT2 (Actual)`[indx_Tot] == 10)
  len_indx_10 = length(indx_10)

    for(i in 1:10){
      nam <- paste("len_indx_", i, sep = "")
      #print(get(nam) / len_indx_Tot)
      b <- get(nam)/length(indx_Tot)
      rec_Prob <- c(rec_Prob, b)
      #add to matrix
    }
}
print(rec_Prob)
mat_rec_Prob = matrix(rec_Prob,
              nrow = 10 , ncol = length(seq_day))
write.csv(mat_rec_Prob, "DCA_ATV_Daily_TT2.csv")

## Exteract cleaned data for correlation
indx_cor <- which( (TMA$`ShipTo XXX Loc`=="DCA") &
                     (TMA$DIV =="ATV") &
                     (TMA$`TT1 (Actual)` > 0) &
                     (TMA$`TT2 (Actual)` > 0) &
                     (TMA$`TT3 (Actual)` > 0))

clean_TT1 = TMA$`TT1 (Actual)`[indx_cor]
clean_TT2 = TMA$`TT2 (Actual)`[indx_cor]
```

```
clean_TT3 = TMA$`TT3 (Actual)`[indx_cor]

A <- data.frame(clean_TT2, clean_TT3)
mat_cor <- cbind(clean_TT2, clean_TT3)

write.csv(clean_TT2, "TT2_correlation.csv")
write.csv(clean_TT3, "TT3_correlation.csv")

cor(clean_TT2, clean_TT3, use = "complete.obs")
cor(clean_TT2, clean_TT3, method =
      c("pearson", "kendall", "spearman"))

## Extract clean Data for emprical
#distribution for simulation validation
indx_EmpDis1 <- which( (TMA$`ShipTo XXX Loc` == "DCA")&
                         (TMA$DIV == "ATV") &
                         (TMA$`TT Total (Actual)` > 0 ))

clean_TTT_ATV = ceiling((TMA$`TT Total
                          (Actual)`[indx_EmpDis1])/24)

write.csv(clean_TTT_ATV, "TTT_ATV.csv")

indx_EmpDis2 <- which( (TMA$`ShipTo XXX Loc` == "DCA")&
                         (TMA$DIV == "DSS") &
                         (TMA$`TT Total (Actual)` > 0 ))
clean_TTT_DSS = ceiling((TMA$`TT Total
                          (Actual)`[indx_EmpDis2])/24)
write.csv(clean_TTT_DSS, "TTT_DSS.csv")
indx_EmpDis3 <- which( (TMA$`ShipTo XXX Loc` == "DCA")&
                         (TMA$DIV == "PMM") &
                         (TMA$`TT Total (Actual)` > 0 ))
clean_TTT_PMM = ceiling((TMA$`TT Total
                  (Actual)`[indx_EmpDis3])/24)
```

```
write.csv(clean_TTT_PMM, "TTT_PMM.csv")
```

Listing A.1: Sample of R code used for data preparation of simulation.

A.6 Data Science modeling for root causes

This appendix provides the source code in Python for analysing the effects of transit times between the Back-Ends and the Distribution Centers capturing information on on Negative Early Warning (nEWs). In this model, three different types of data from the case study Technology AG is imported to generate the results. This model is based on the Transit Monitoring and Analysis (TMA) database detailing nEWs. The main outputs of this model are two excel files which contain relations between nEWs and logistic routes. The model is used to visualize some explanatory diagrams which are discussed in Section 5.10 on page 154.

```

### 1- Pandas,Numpy,and Matplotlib are the used libraries.
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline


### 2- Read the three relevant datasheet.
df_TMA=pd.read_excel(io='TMA405_List_Of_Deliveries_CW33.xlsx')
df_nEWs = pd.read_excel(io= 'nEW_ex_znEW_CW33.xlsx')
df_TMA101 = pd.read_excel(io='TMA101_Delivery_List_BO4_CW32_CW34.xlsx')

### 3- TMA405 data preparation
# read the relevant columns of dataframe
df_TMA_Clean = df_TMA[['ShipFrom Loc','ShipTo XXX Loc',
                'Delivery No', 'PO Number','Delivery Quantity',
                'TT Total (Target)', 'TT Total (Actual)']]
print('The TMA dataframe shape is: ', df_TMA_Clean.shape)
```

```
df_TMA_Clean.head(5)

#Calculation the diffrence between Actural Transit Time
# (ATT) and Traget Transit Time (TTT)
df_TMA_Clean ['Deviation in Transit'] =
        np.where((df_TMA_Clean['TT Total (Target)'] <
        df_TMA_Clean['TT Total (Actual)']) &
        (df_TMA_Clean['TT Total (Actual)'] > 0), 1, 0)

df_TMA_Clean['Deviation Percentage'] =
        np.where((df_TMA_Clean['TT Total (Actual)'] > 0),
        np.round(100*(df_TMA_Clean['TT Total (Actual)']-
        df_TMA_Clean['TT Total (Target)'])/
        (df_TMA_Clean['TT Total (Target)'])),0)
df_TMA_Clean.head(5)

# plot the average percentage of change in ATT vs TTT
# deviation percentage
df = df_TMA_Clean.loc[df_TMA_Clean['Deviation in Transit'] == 1]
List_Late_Delivery = np.array(df['Delivery No'])

labels_01 = 'On-Time Delivery','Late Delivery'
result_01= df_TMA_Clean.groupby(['Deviation in Transit']).size()

Late_del_num = result_01[1]
OnTime_del_num = result_01[0]



fig, ax = plt.subplots(figsize=(10,6))
ax.pie(result_01, labels=labels_01,autopct='%1.1f%%')
ax.axis('equal')

print(Late_del_num, "of", Late_del_num + OnTime_del_num,
        "deliveries have ATT bigger than TTT ")
plt.title('Comparison of deliveries in selected
```

```
        logistic route\n On-Time or Late delivary',
          bbox={'facecolor':'0.8', 'pad':5})
plt.show()

### 4- TMA 101 data preparation
df_TMA101.columns = df_TMA101.iloc[0]
df_TMA101= df_TMA101.drop([0], axis= 0)
df_TMA101_Clean = df_TMA101[['ShipFrom\nCountry',
                'ShipTo\nXXX Loc', 'Delivery\nNo',
                'Product\nSP Material', 'Delivery\nDate',
                'Delivery\nQuantity' ]]
print('The size of TMA101 is: ', df_TMA101_Clean.shape)

### 5- Search for SP
df_TMA101_Clean['Delivery\nNo'] =
    pd.to_numeric(df_TMA101_Clean['Delivery\nNo'],
        errors='coerce')
df_TMA101_Clean['Delivery\nQuantity'] =
    pd.to_numeric(df_TMA101_Clean['Delivery\nQuantity'],
        errors='coerce')

# Create a new column to identify the relevant SP

df_TMA101_Clean['has_Value'] =
    df_TMA101_Clean['Delivery\nNo'].
    isin(List_Late_Delivery)

df_merge_101v405 = pd.merge(df_TMA_Clean, df_TMA101_Clean,
                    how='inner',
                    left_on=['Delivery No',
                    'Delivery Quantity'],
                    right_on=['Delivery\nNo',
                    'Delivery\nQuantity'])
df_merge_101v405_Delay =
    df_merge_101v405.loc[df_merge_101v405['has_Value']
```

```
    == True]

df_01 = df_TMA101_Clean.loc[df_TMA101_Clean['has_Value']
        == True]
List_Late_SP = np.array(df_01['Product\nSP Material'])

if np.size(List_Late_SP) == np.size(List_Late_Delivery):
    print('All the', np.size(List_Late_SP),
        'Delivery_NO matched with SP values')
else:
    print('Some of the Delivery numbers did
            not match with SP values.',
          '\n please check the date range
          of the selected databases')

### 6- nEWs data preparation
df_nEWs_Clean = df_nEWs[['Old CMAD','Old Qty',
        'New CMAD','New Qty','SP Item', 'Creation Date']]
print('The BO Universe dataframe size is: ',
        df_nEWs_Clean.shape)

### 7- Search for SP in nEWs
df_nEWs_Clean['nEWs_dealy'] = (df_nEWs_Clean['New CMAD'] -
    df_nEWs_Clean['Old CMAD']).dt.days
df_nEWs_Clean['relative_nEWs_dealy'] =
    (df_nEWs_Clean['Old CMAD'] -
    df_nEWs_Clean['Creation Date']).dt.days

df_merge_TMAnEWs = pd.merge(df_merge_101v405_Delay,
                        df_nEWs_Clean, how='inner',
                        left_on=['Delivery Quantity',
                        'Product\nSP Material'],
                        right_on=['Old Qty', 'SP Item'])

df_merge_TMAnEWs_cleaned=df_merge_TMAnEWs[['ShipFrom Loc',
```

```
    'ShipTo XXX Loc','TT Total (Target)',
    'TT Total (Actual)','Delivery Quantity',
    'Delivery No', 'Deviation in Transit',
    'Deviation Percentage','Delivery\nDate', 'has_Value',
    'SP Item', 'Old CMAD','Old Qty','New CMAD','New Qty',
    'Creation Date','nEWs_dealy', 'relative_nEWs_dealy']]

df_merge_TMAnEWs_cleaned.to_excel("output.xlsx")

num_days_relat_nEWs = 2

df_final_cleaned_ZD_nEWs =
    df_merge_TMAnEWs_cleaned.loc[df_merge_TMAnEWs_cleaned
        ['relative_nEWs_dealy']<num_days_relat_nEWs]
print('The size of ZD-nEWs set is:',
    df_final_cleaned_ZD_nEWs.shape)

print('The size of Final set is: ',
    df_merge_TMAnEWs_cleaned.shape)
df_final_cleaned_ZD_nEWs.to_excel('output_ZD_nEWs.xlsx')

### 8- Explanatory results of the model
num_Zd_nEWs = len(df_final_cleaned_ZD_nEWs)
df_TMA_Clean.groupby(['Deviation in Transit']).size()

labels_02 = 'On-Time Delivery','Late Delivery',
            'Late Delivery Caused ZD-nEWs'

Late_del_num = result_01[1] - num_Zd_nEWs
OnTime_del_num = result_01[0]
result_02 = [OnTime_del_num, Late_del_num, num_Zd_nEWs]


fig, ax = plt.subplots(figsize=(10,6))
ax.pie(result_02, labels=labels_02,autopct='%1.1f%%')
```

```
ax.axis('equal')

plt.title('Comparison of deliveries in selected
        logistic route of TMA\n'+ 'percentage that
        arrive on-time, late, or late delivery
        that cause ZD-nEWs',
          bbox={'facecolor':'0.8', 'pad':5})
plt.show()

labels_03 = 'Late Delivery',
            'Late Delivery Caused ZD-nEWs'

Late_del_num = result_01[1] -num_Zd_nEWs
result_03 = [Late_del_num, num_Zd_nEWs]

fig, ax = plt.subplots(figsize=(10,6))
ax.pie(result_03, labels=labels_03,autopct='%1.1f%%')
ax.axis('equal')

plt.title('Percentage of late delivery in TMA that
            caused Zero Day nEWs',
          bbox={'facecolor':'0.8', 'pad':5})
plt.show()

df_nEWs_ZD = df_nEWs_Clean.loc[df_nEWs_Clean
            ['relative_nEWs_dealy']<
            num_days_relat_nEWs]
labels_04 = 'Zero Day nEWs',
            'Zero Day nEWs by selected logistic route'

total_ZD_num = len(df_nEWs_ZD)- num_Zd_nEWs
#print(total_ZD_num)
result_04 = [total_ZD_num, num_Zd_nEWs]

fig, ax = plt.subplots(figsize=(10,6))
```

```
ax.pie(result_04, labels=labels_04,
        autopct='%1.1f%%')
ax.axis('equal')

plt.title('Percentage of Zero Day nEWs caused\n
            by the selected logistic route in TMA data',
          bbox={'facecolor':'0.8', 'pad':5})
plt.show()

### End of the notebook
```

Listing A.2: Jupyter Notebook source code in Python for data analysis the relation between nEWs and logistic route.

A.7 ReCAST Requirement Analysis

RECAST
REQUIREMENTS SPECIFICATION

Regional Customer Allocation Support Tool

Zhikang Tian

Behrouz A. Mousavi

Cathal Heavey

[redacted]

May 2020

1. INTRODUCTION

1.1. Purpose of Document

After carrying out the preliminary survey of ReCAST software requirements, this requirement specification is certain. In the requirement specification, we focus on the analysis of user types, user’s expected functions, user interfaces, user interactions, and system performance, so that the members of the project could quickly understand the system.

So far, we've been building simple prototypes and meeting to stakeholders for validating requirements and contacting project managers, ZLP, developers, and project consultants for getting prototype feedbacks, and modifying some of the user interaction operations and business processes based on the previous prototypes.

On the specification, the system specification and the subsequent design and development work provided.

Intended audience: customers, business requirements analysts, testers, user documentation writers, project managers.

1.2. Project Background

The enterprise information management system is an indispensable part of modern enterprise management, such as ERP or APS system. These systems play a role in resource allocation for enterprise production planning, thus improving productivity and maximizing profits.

However, when demand exceeds supply, there is a shortage to end customers., the problem comes out – how to decide to allocate supply to customers. Based on ATP, APS provides allocation planning to customer’s order. But that planning decision is made from APS with stochasticity and fuzziness in both orders and demands. Therefore, in real practice, it is impractical only to adopt decisions from APS without human intervention. Because customer relationships and negotiations need to be accomplished case by case, and the strategy is always being updated. Also, the decisions are made hierarchically according to the structure of the value chain.

Thus, it needs the combination of human intervention and heuristics algorithms to achieve the flexibility of planning to find feasible optimal decisions regarding allocations.

To facilitate the above problems, Universtiy of Limerick propose a decision support tool based on mathematical optimization called ReCAST (Regional Customer Allocation Support Tools). The ReCAST support ZLPs to allocate regional products to customers in time of tight supply. The benefits of ReCAST are avoiding planners bias decisions, allocate product in a faster time, provide scenarios according to marketing or inventory strategies, increasing the customer satisfaction, decrease ATP consumption fluctuation, control the inventory, and making the allocation process tractable and measurable. ReCAST in fact is a web-based user decision support tool based on a mathematical optimization model..

2. RECAST PROJECT

2.1. Project Brief

- The aim is to obtain optimal Target Allocation (TA) to be used within AM-UI as benchmarks by the ZLPs during their allocation planning.

- The generated optimal solutions by ReCAST are based on the ZLP input configuration. These configurations comes from insights that ZLPs have or gain from marketing, products, business situation, and etc.

- Convenient and easy-to-use design: ZLP is required to manually import a local Excel file, run the mathematical optimization model according to the tooltip, and finally output TA (result) to the Excel file

2.2. ReCAST Context

SCOR Model consists of 5 management processes: Plan, Source, Make, Deliver, and Return. The planning process is divided into 5 sub-processes, which are Demand Planning, Capacity Planning, Supply Planning, Production Management, and Order Management as shown above. Production Program and Business Scenario follow the planning process.

For Order Management: AM-UI is used for allocation maintenance. The tool DF from JDA is used for Demand Fulfillment and SAP is used for the final Order Management Proces. The ReCAST is supporting ZLPs for developing and implementing tight allocation plans that add to AM-UI. .

Generally, in case study order management system there are four types of users: Supply Planner, ALM, ZLP and CLM. In terms of the user of ReCAST, ZLP is the primary user because the purpose of ReCAST is to get the optimal TA scenario, and it belongs to the business of ZLP.

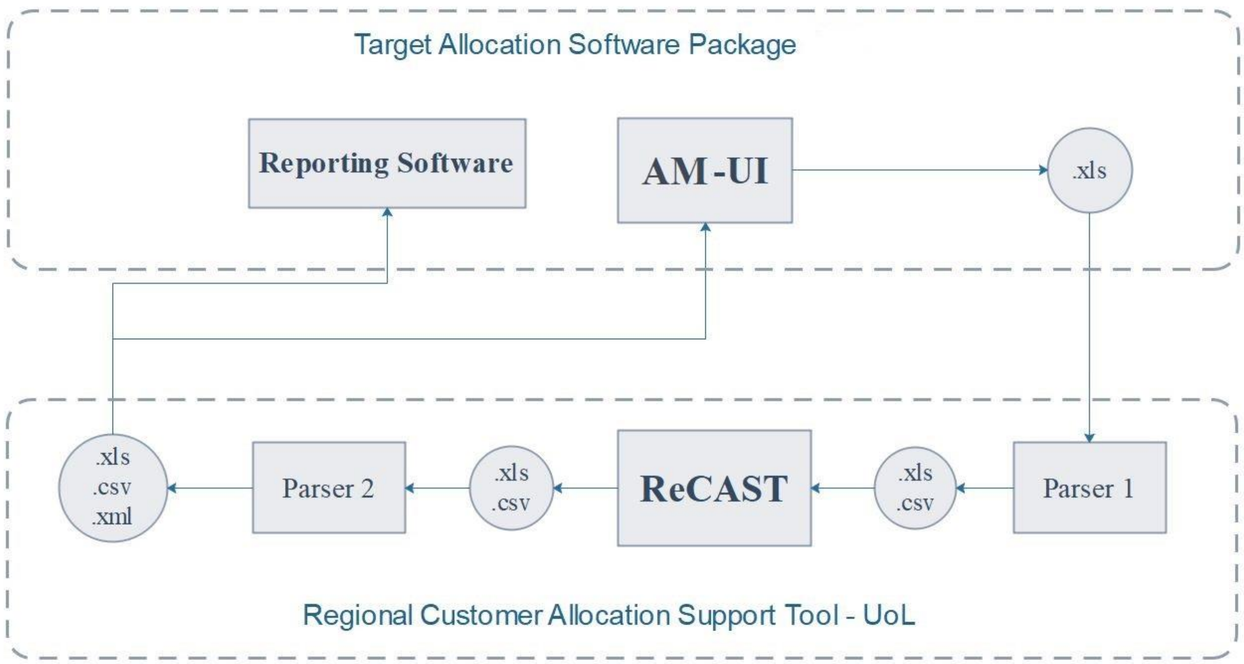


Figure 2. ReCAST connects to Others Module at APS

To execute the obtained optimal solution, the allocation plan should be inserted into AM-UI. To avoid such manual work, the ReCAST output file should be parsed into another format to fit the import style of destination software. The output of ReCAST not only add values to AM-UI, but also it could be used by any other software of the global planning system of the case study for providing reports.

2.3. Business Logic

When customer demand expands, it can logically be considered as a decrease in supply. As a consequence of this, the product allocation for ZLP is going tight. To solve the problem, first of all, ALM will check the inventory, CAP will assist ALM to transfer goods from other warehouses to alleviate the current situation, and also Product Marketing (PM) will help ALM to provide other substitutes to meet product allocation.

But if it still not be resolved, ALM will rise a tight allocation flag to ZLP. Next, ZLP needs to find an optimal scenario – the Target Allocation. TA maximize customer satisfaction while under the available inventory for allocation, whichis known as the ATP, the current and future supply picture that could be allocated. ReCAST was born under this situation for finding the TA.

The following presents graphically step to take to achieve the ReCAST business goal: derive the Target Allocation.

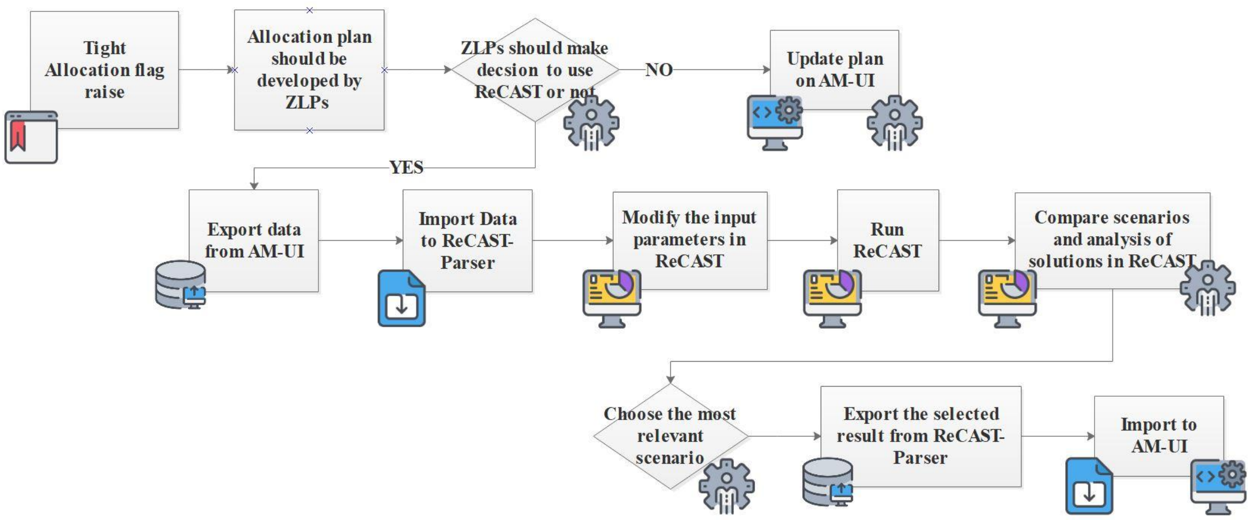


Figure 4. ReCAST Business Process

A tight allocation flag firstly is raised by ALM. Then ZLPs should develop an allocation plan: if the plan is obvious, they can update on AM-UI directly. Also, they can derive TA plan with the help of ReCAST:

1. Extract data from AM-UI (format should be an Excel file).
2. Import Data to ReCAST.
3. Configure ReCAST mathematical optimal model parameters.
4. Run the configured ReCAST model.
5. Compare scenarios and modify parameters to get the best scenario.
6. Export the TA scenario as an Excel file for ReCAST output.
7. ZLP manually updates AM-UI by importing an output Excel file to AM-UI.

2.4. Business Form

This section describes the system input and system output.

- Excel File Extracted from AM-UI (System Input)

  For the first step for running ReCAST, a user needs to import Excel file with the following format.

Seller	Measures	CW47	CW48	CW49	CW50	CW51	CW52	CW1	CW2
[redacted]	Plant ATP								
	ATP vs. Net Target Alloc	[redacted]							
	Sum AP Forecast								
	Sum Target Alloc								
	Sum Delivered								
	Sum Orders (RMAD)								
	Sum Conf Orders (CMAD)								
	AP Forecast vs Net TA								
	ATP vs Net Target Alloc (C								

Seller	Measures	17/11/201	24/11/201	1/12/201	8/12/201	15/12/201	22/12/201	29/12/201	5/1/201
[redacted]	Consignment Inv.	0	0	0	0	0	0	0	0
	Orders (RMAD)	[redacted]							0
	Confirmed Orders (CMAD)								0
	Cum Coverage (RMAD)								0
	Delivered								0
	Min. Run Rate								0
	Target Allocation								0
	Consignment Inv.								0
	Orders (RMAD)								0
	Confirmed Orders (CMAD)								0
	Cum Coverage (RMAD)								0
	Delivered								0
	Min. Run Rate								0
	Target Allocation								0
	Consignment Inv.								0
	Orders (RMAD)								0
	Confirmed Orders (CMAD)								0
	Cum Coverage (RMAD)								0
	Delivered								0
	Min. Run Rate								0
	Target Allocation								0

Figure 5. System Input Excel File Format

It can be considered the file consisted of two parts, and there is an empty column between them for separation. The row circled by red rectangles are needed to be parsed and extracted:

1. Seller for WF00: Plant ATP.
2. Other Sellers: Min. Run Rate, Target Allocation.

- Excel Template for Uploading to AM-UI (System Output)

As the section 2.2 discussed, there are two ways to export the result of ReCAST to others module, the first one need to export to AM-UI for an update; another is for exporting to others reporting software in APS. For the first one, the route from ReCAST to AM-UI, according to the business process we have the following plan:

Generating an Excel file containing TA scenario for pushing back to AM-UI for updating. ReCAST needs to fill out the following Excel template as the system output. ZLP will upload the file manually to AM-UI.

			CW37
PRODUCT	DF_SELLER	MEASURE	09/05/2020

Figure 6. System Output Excel File Template

- **PRODUCT** Field Values can be either SalesProduct (SP) or Finished Product (MA)
- **MEASURE** Field values can be TARGET_ALLOCATION or MIN_RUNRATE
- **DF_SELLER** should be Leaf Sellers at which Allocations has to be maintained.
- Time horizon Fields should be named starting with CW and its content will be filled with scenario's result

2.5. Development Planning and Methodology

The project was initially followed by a waterfall model, the Gantt diagram was prepared for planning according to the model.

Figure 7. Gantt Chart for Waterfall Model

Looking back at the last month, we have identified most of the requirements of ZLPs. Next step, we need to analyze and further plan for other user types that might need ReCAST. Due to the limitations of the waterfall model, we cannot quickly respond to changes in software requirements, and writing code must wait until the requirements are confirmed before it can start, which results in a delay. Therefore, the project should be gradually converted from a waterfall model to an incremental development model. The development process includes many scrums, each of which is a five-step process:

1. Construct prototype and confirm requirements

With the cooperation of analysts and users, the basic requirements of the system are quickly determined, and a prototype should be implemented as soon as possible, and the prototype should meet the basic characteristics of the system as user's requirements described.

2. Quick analysis, draw the stator system module

Based on quick analysis, according to the feedback of stakeholders, the correctness of the prototype business is validated, and the system modules are divided.

3. Code running prototype

Quickly develop and test the identified modules.

4. Evaluation prototype

On the basis of operation, check and evaluate, analyze whether the operation effect meets the user's expectations, correct misunderstandings in the past interactions and errors in the analysis, and add new requirements.

5. Modify and accept

Modify according to the results of the evaluation prototype activities. If the user satisfaction is reached, the development of the module is completed.

The cycle repeats, the five processes are carried out in sequence, each process is a week or two weeks, and finally the integration test is carried out to deliver the product.

Although the development model has changed, the duration of the project is still following the Gantt chart, and the project will be delivered on time.

3. SYSTEM USERS

3.1. User groups / System's Actors

In ReCAST, the primary user is ZLP. Also, ALM, CLM and Supply Planner also relative to ZLP in business, which are defined as high-level user in ReCAST, and it might need to take them into consideration later. Currently, we have successfully validated the business logic for ZLP,

- Normal User: ZLP (Primary user)
- High-level User: ALM and Product Marketing.(Considered after ZLP's functions finished)

3.2. User Quantity

As the software designed for internal use, there is no big demand for performance because user quantity is low. For ZLP, based on the meeting we know they are not using ReCAST every day, generally once a week. And also, they wish their data before three months shall be erased, as they not usually login to ReCAST.

3.3. RCL User States

There are three states for an ZLP when they are using ReCASTS:

- State A: unlogined state: At Login page, Register page, Forget Password.
- State B: logined state: At Index page, Initial Scenario page (Import Data from Excel file), Task Information page. Including All pages at state C
- State C: working state: At Config Optimal Model page, ReCAST Analysis Result page, Scenario Modification page.

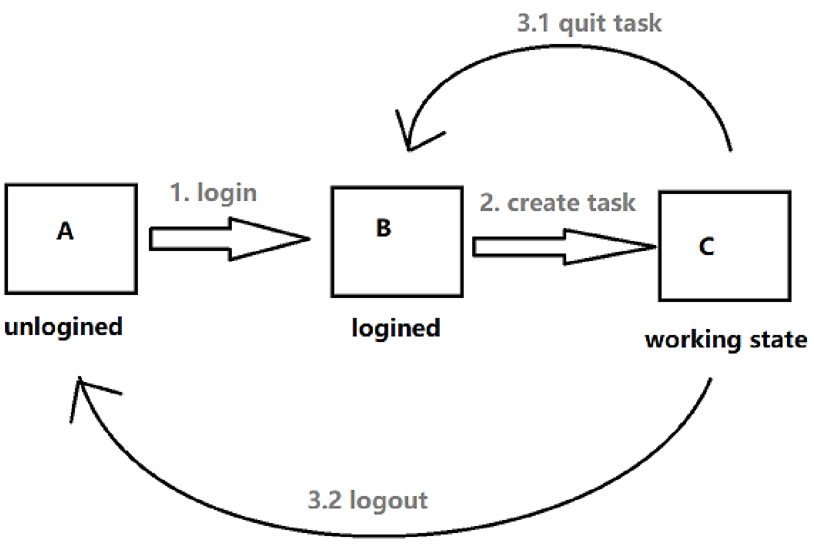


Figure 8. ZLP User States.

- ZLP State Transited (Webpage Jump Relations)

- A->B: From unlogined state to logined state, by triggering login event.
- B->C: By creating a task for running ReCAST. A task can be considered as a business logic flow for generating TA scenario.
- C->B: By quit the task or finish the task of ZLP currently doing.
- C->A: By logout.

4. FUNCTIONAL REQUIREMENTS

This section provides information on all known ReCAST tool functions and services. It is more relevant to the specific functional descriptions related to business logic and all user interactions between the system.

Use case diagram is essential to describe functional requirements, based on the system's actor, showing the interaction for user in the system also represents the relationship between each function. Each circle in the figure represents a use case for a system boundary that describes a usage scenario context to capture the requirements for user interaction with the system.

4.1. Use Cases Diagram – ZLP's functionalities

Here three use case diagrams are defined. In sequence, they describe all functionalities for ZLP under unlogined state, logined state and working state. ZLP can perform different functions in different situations.

- Unlogined State

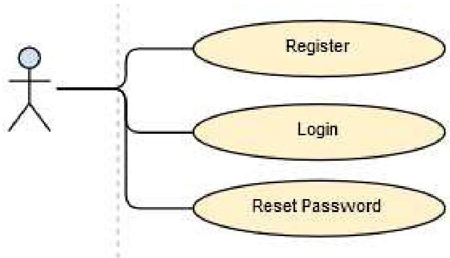


Figure 9. Use Case Diagram for ZLP at Unlogined State

At unlogged states, ZLP can only do the basic action to ReCAST: register. Login or reset password.

The following gives each of the use case description in detail, which demonstrate use case context scenarios, pre-conditions, use case outputs (post-conditions), event flows, user interactions, and so on. How ZLP interact with ReCAST in a different state.

Use Case 1	**Register**
Use Case Description	
Registration is one of the basic functions of the system. Every user entering the ReCAST system for the first time needs to register. The purpose is to identify the user to the legal identity.	
Actor Action	System Response
1 – Client (ZLP) clicks register	2 – ZLP's details check and stored in system
Alternative Route	
1(a) – ZLP registers client 1(b) – Details already on the system	2(a) – None 2(b) – If credentials already associated with an account ask for different details

Use Case 2	**Login**
Use Case Description	
Login is a way to identify the user's real identity, it is an essential part of system security. There are generally two ways to achieve user login: 1. Checking Registered user information in ReCAST This kind of user can authorize login through the traditional login authentication 2. SSO apply on existing users in ASP system This kind of users will be able to achieve rapid registration through SSO technology. In this way, the tedious account password detection steps during each login are avoided and the user experience is improved.	
Actor Action	System Response
1 – Client (ZLP) enter valid username and password	2 – Credentials are checked and log on welcome page
Alternative Route	
1(a) – Staff registers client 1(b) – Details already on the system 1(c) – User click ‘avatar’	2(a) – Client credentials fail, error message displayed

Use Case 3	**Reset Password**
Use Case Description	
ZLP request a one-time verification code by email, after receiving code, input new password and confirm password box for update.	
Actor Action	System Response
1 – Client (ZLP) enter valid verification code and password	2 – Update password on database

- Logined State:

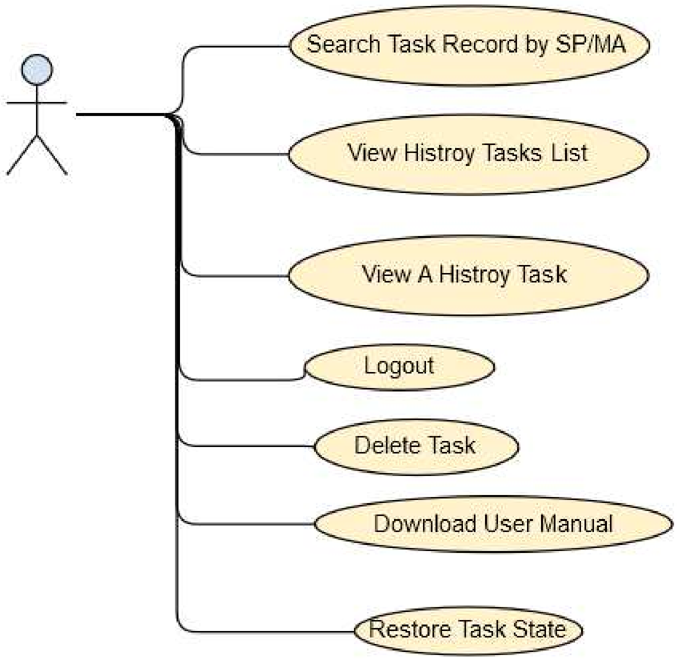


Figure 10. Use Case Diagram for ZLP at Logined State

In ReCAST, a task stands for a process of running ReCAST for generating TA scenario. Each task can only import only one Excel file, also only one product can be analyzed.

The above diagram defines 7 functions of ZLP at logined state. Each represents a different interaction between ReCAST.

Use Case 4	**Logout**
Actor Action	System Response
1 – Client (ZLP) click logout	1 – Website forward current page to login page, and logout user state at the back-end server. (ZLP transfer logined state to unlogined state.)

Use Case 5	**Download User Manual**
Use Case Description	
For the ZLP first time login to system, they don't have too many ideas about how to use ReCAST. A user manual can tell them how to use it.	
Actor Action	System Response
1 – Client (ZLP) click the download button.	2 – User manual file is downloaded.

Use Case 6	**Delete Task**
Use Case Description	
Two options are available for deleting history records: 1. ReCAST can automatically delete any histroy record more than three months. 2. ZLP can delete it manually	
Actor Action	System Response
1 – Client (ZLP) select task(s) and delete	1 – The task is deleted from database

Use Case 7	**Restore Task State**
Actor Action	System Response
Pre-condition	
The task must be created and saved before restore it.	
1 – Client (ZLP) select a task to restore	1 – ReCAST reload the task into the previous webpage where the user saved it. It can be at page of Modificatication, optimal parameter config page or ReCAST result page. (No need to run ReCAST math optimal model.)
Post-condition	
The task is restored as the previous state at the last time it.	

Use Case 8	**Search Task Record by SP/MA**
Use Case Description	
After ZLP click ‘History Scenario’ for importing data, webpage route to a search page, where there is one input box, both MA and SP can be input. (where the product ID starting with SP should be searched in box SP; similarly, if that starts with MA, should be searched by MA input box.) Then, ZLP clicks search button for jumping to search result page.	
Pre-condition	
1. User must be logined. 2.Input valid Product ID.	
Actor Action	**System Response**

1− ZLP input valid Product ID 2− ZLP click ‘Search’	2- Search result with specific product ID displayed on the search page (Only for their own account records).
Alternative Route	
1− ZLP givses an invalid input 2− ZLP gives a SQL Injection statement. 3− ZLP click ‘Back’	1- Prompts error message near to input box. 2- System should act same as 1. 3- Back to welcome page.
Post-condition	
Data for selected scenario is updated	
Exceptions	
1 – ZLP simulate a fake HTTP request with SQL Inection, as input content, sending to server for query others’ account data.	1 – Server side need to check validity of input string in case of the SQL Inject.

Use Case 9	**View A History Task**
Use Case Description	
User select one history task on search page and click ‘view details’. Website will jump to task information page whose display structure contains all input parameters. If the current task is a finished task, the last exported scenario should be displayed as well.	
Pre-condition	
1. User must be logined.	
Actor Action	**System Response**
1− ZLP click ‘view details’ button for a histroy task	1-Task information page displayed

Use Case 10	**View History Task List**
Use Case Description	
On the ReCAST History Task module, ZLP can view their history and also any history can be restored as the previous saved state so that ZLP can keep the analysis work continuous as before. It can save time and efforts for ZLP by avoid input all parameters again if they leave at halfway.	
Pre-condition	
1. User must be logined.	

Actor Action	System Response
1− ZLP choose import data from history tasks	All tasks for the ZLP are read from database and load into webpage.
Event	
1 –ZLP click view 'on-going task' - 2 –ZLP click view 'finished task' 3 –ZLP click view 'all task'	1 – display all on-going tasks. 2 – display all finished tasks. 3 – display all tasks

- Working State:

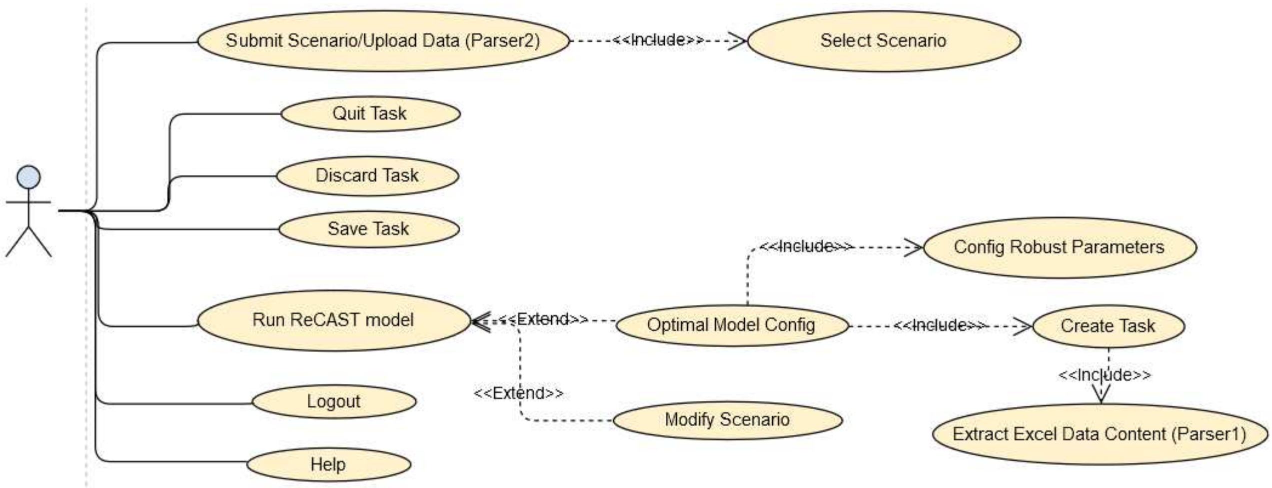


Figure 10 Use Case Diagram for ZLP at Working State

Among them, there are two type of relationship between some of use cases:

1. Extend: The use case indicated by the arrow, as shown in the figure, provides an interface that can be implemented by a use case that either arrow starts.

2. Include: The use case that starts with the arrow in the figure invokes the use case that is directly referred to.

For uses cases Run ReCAST model, it provides an interface to the other two use cases: Optimal Model Config and Modify Scenario, which means there are two ways for running ReCAST model. At optimal model config pages or modify scenario page, both can invoke ReCAST use case.

Also, we can say the use case 'Submit Scenario/Upload Data (Parser2)' include use case 'Select Scenario', which means the former can not be executed before the latter executed. Similarly, 'Optimal Model Config' include 2 use cases and both of them should be executed ahead.

Use Case 11	Extract Excel Data Content (Parser1)
Use Case Description	
The use case is used for importing data from AM-UI Excel file. Firstly, ZLP upload file and input parameters manually; and then, the file with parameter will be post to server. Server then check file and parameter's validity and extract all data from that file by an Excel file parser. Then, server responses result to bowser for updating image or showing error message.	
Pre-condition	
User must be logined.	
Actor Action	**System Response**
1−Client (ZLP) clicks Browse from PC and select AM-UI Excel file for uploading to server.	1− Excel Picture is updated or give an error message
Post-condition	
Excel image at webpage is updated	
Exceptions	
1 –ZLP input an invalue AM-UI format Excel file, when click 'next'.	1 –An alert information appears 'invalid Excel file Format, please upload a valid Excel file as the above structure'

Use Case 12	Create Task
Use Case Description	
Anytime when to run ReCAST model, a task should be created first. In case of ZLP confuses the past task activities and have a better understanding, the taskname and description should be marked for that. Because each time for running ReCAST may analysis different Product, so a task is the most basic business flow at ReCAST.	
Pre-condition	
1. User must be logined. 2. Excel image at webpage is updated (Must import The Input Excel File)	
Actor Action	System Response

1 – Client (ZLP) enter valid task name code and task description and CW, Sceario weight. 2− ZLP input Time Horizon and ZLP configure weight parameter 3− ZLP click 'Next'	1 – A ReCAST task is created 2 –Check the validity of input value for each input box and give an error message if necessary. 3 - Display navigation bar
Post-condition	
ZLP transfer into working state.	
Exceptions	
1 – ZLP input value out of range, when click 'next'	1 –An alert information appears 'invalid input information, re-input again'

Use Case 13	**Save Task**
Use Case Description	
When ZLPs want to leave at halfway, and they may do not want to re-input the previous parameter. They can click 'Save task' option at the navigation bar. And this task is persisted into database so that to be reloaded for next time when they come back.	
Pre-condition	
1. User must at working state.	
Actor Action	**System Response**
1− ZLP click 'save task' option	1-All Task information are saved in database

Use Case 14	**Discard Task**
Use Case Description	
When ZLPs want to discard the current task, might because a reason like import a wrong Excel file.	
Pre-condition	
1. User must at working state.	
Actor Action	**System Response**
1− ZLP click 'discard task' option	1-All Task information are discarded, and no data persist in database, although this task is saved previously.

Use Case 15	**Save Task**
Use Case Description	
The only different between Discard task and this use case is that: if user click ‘quit task’, any history task record at database will not be deleted.	
Pre-condition	
1. User must at working state.	
Actor Action	**System Response**
1- ZLP click ‘quit task’ option	1- Load to index page and nothing changed on database.

Use Case 16	**Help**
Use Case Description	
For assist ZLP use ReCAST, most of input box config a explanation help. In normal case, it will not open, only when user click ‘help’, in this case user’s mouse over the quesiton mark, it will come out the explanation.	
Pre-condition	
1. User must at working state.	
Actor Action	**System Response**
1- ZLP click ‘Help’ option	1-Toggle quesitons marks.
Event	
1 – ZLP’s mouse hover question mark	1 – There is a box appear on that place showing what is the meaning for this item, and giving the range for its value.

Use Case 17	**Select Scenario (Data Visualization)**
Use Case Description	
In order to display the visualized scenario, a data panel is applied for ZLP selecting scenario. Different customer represented by different color.	

Pre-condition	
1. User must at working. 2.All valid scenarios result come out.	
Actor Action	**System Response**
1−ZLP click different radio button on data visualization panel for switch scenario	1−Data visualization panel content changed. 2− The corresponding scenario button will be high-lighted and
Alternative Route	
1 –ZLP select a scenario by click ‘Select’.	1 –The button of selected scenario is high-lighted 2 –And data panel changed correspondingly
Event	
1 – ZLP’s mouse hovers any one value point on data panel.	1 – A tiny label come out showing its value, and the point will be bold and enlarged.
Post-condition	
A scenario is selected	

Use Case 18	**Submit Scenario/Upload Data (Parser2)**
Use Case Description	
The use case is used for exporting select scenario data to a template Excel file for uploading to AM-UI. After all scenario results displayed, ZLP select a scenario, then click ‘Export’. The browser will post data to server. And server will run parser 2 for generating an Excel file back to browser. Then also server will write all result and product information to database for the convenient use of the next time	
Pre-condition	
1.User must at working state. 2.A scenario is selected.	
Actor Action	**System Response**
1 – ZLP clicks ‘Export’	1 –Output Excel file 2 –All scenario data and product information will be written in database

Post-condition
Output expected Excel file (for importing AM-UI)

.

Use Case 19	Modify Scenario
Use Case Description	
On 'Modification' page, the table of scenario information can be modified by ZLP. A-ATP and A-Stock are editable but sum is a fixed number. (There is a relationship between these three parameters: A-ATP+A-Stock = Sum.) After that, user click 'Check', browser post data to server for runing ReCAST optimal model and then result data shall be responsed for updating data panel. If user do not click 'Check', there will be no data updated.	
Pre-condition	
1. User must be logined. 2.All valid scenarios result come out and a scenario is selected for modifying. (at 'Modification' page)	
Actor Action	**System Response**
1− ZLP can modify content at scenario table 2− ZLP click 'Check'	Data panel for dual visualization is updated; also, if user go back the previous page for scenario visualization, the corresponding scenario should be same as the latest version at dual visualization page.
Alternative Route	
1− ZLP can modify content at scenario table 2− ZLP click 'Back' 3− ZLP click 'Export'	2- Nothing updated. 3− do use case 'Submit Scenario' but there is no data changed.
Event	
1 – ZLP's mouse click a cell at scenario table (for column A-ATP or A-Stock) 2 – ZLP's mouse double-click a cell at scenario table (only for column A-ATP or A-Stock) 3 – ZLP click table header or the header of a row	1 – The cell will be high-lighted and . 2 – The cell shall be editable. 3 – The whole column or row will be selected in color of shadow.
Post-condition	
Data for selected scenario is updated	
Exceptions	

1 – ZLP input value out of range, when click ‘next’ 2 –ZLP input an invalue AM-UI format Excel file, when click ‘next’.	1 –An alert information/ alert window appears ‘invalid input value, try again’ 2 –An alert information appears ‘invalid Excel file Format, please upload a valid Excel file as the above structure’

Use Case 20	**Optimal Model Config**
Use Case Description	
To run ReCAST optimal model, there are some parameters should be filled firstly. In this case, ZLP can enable Rubost Factor. If it is enabled, use	
Pre-condition	
1. User must be logined. 2. AM-UI Excel file must be imported into ReCAST	
Actor Action	**System Response**
1− ZLP can modify content at scenario table 2− ZLP click ‘Check	Data panel for dual visualization is updated; also, if user go back the previous page for scenario visualization, the corresponding scenario should be same as the latest version at dual visualization page.
Alternative Route	
1− ZLP can modify content at scenario table 2− ZLP click ‘Back’ 3− ZLP click ‘Export’	2- Nothing updated. 3− do use case ‘Submit Scenario’ but there is no data changed.
Event	
1 – ZLP’s mouse click a cell at scenario table (for column A-ATP or A-Stock) 2 – ZLP’s mouse double-click a cell at scenario table (only for column A-ATP or A-Stock) 3 – ZLP click table header or the header of a row	1 – The number in cell and content background color will be high-lighted. 2 – The cell shall be editable. 3 – The whole column or row will be selected in color of shadow.
Post-condition	
Data for selected scenario is updated	
Exceptions	

1 – ZLP input value out of range, when click ‘next’ 2 –ZLP input an invalue AM-UI format Excel file, when click ‘next’.	1 –An alert information/ alert window appears ‘invalid input value, try again’ 2 –An alert information appears ‘invalid Excel file Format, please upload a valid Excel file as the above structure’

Use Case 21	**Run ReCAST Model**
Use Case Description	
There are many ways for users to run the ReCAST optimization program, including from the Configuration page and from the Modification page, but no matter which, they always trigger a click event to run it.	
Pre-condition	
1. User must at working state.	
Actor Action	**System Response**
1- ZLP initiate a click event for running ReCAST	System invoke Gurobi for running mathmetical model and render result to webpage at table and visualization panel.
Alternative Route	
1- ZLP can modify content at scenario table 2- ZLP click others link and quit the task.	2- Back to others pageand there is no data changed.
Post-condition	
ReCAST optional scenarios result displayed	
Exceptions	
1 – ZLP input invalid parameters for running.	1 –An alert information/ alert window appears ‘ReCAST can’t running. Please modify your parameters’.

5. NON-FUNCTIONAL REQUIREMENTS

Non-functional requirements here is referring to the operation requirements supporting ReCAST business running, especially to system quality.

5.1. Usability

Usability is important for a tool-based web application. It should be easy-to-use and user-prompted. This tool improves software usability by:

1. Provide user manual: The user can download the ReCAST user manual for usage guidance.

2. Enhanced user interaction: The webpage should prompt a little window while the user's mouse over question marks.

3. User interaction should be obvious and easy-to-understand.

4. If there is an interface for implementing the Single Site-On function, it may need to be implemented. But it needs to be discussed with AM-UI developer for getting AM-UI interface document / APS document. If there is no ready-to-use interface, it probably can not be implemented on time.

5.2. Security

1. The user's password must be hashed in a database.
2. The user uploaded file should be checked on the server-side for detection legality and safety.
3. While ZLP register on ReCAST, firstly must obtain the permission from the project manager (Controversial).
4. In case of SQL Injection for searching product by ID, an input check should be put on server side.

5.3. Performance

1. Optimal model running time as little as possible, maximum system response times should be less than 5 seconds. If it is too long, showing the process bar on screen.
2. The opening webpage should less than 2 seconds.
3. Ajax data dynamically pushed to the background must not cause the browser to fake death (Not Responding).
4.

5.4. Browser

Need to be compatible with IE6,7,8; Firefox3.5 or above; Typical Webkit kernel browsers such as chrome

6. TECHNICAL REQUIREMENTS

6.1. Technologies Overview

The software like auxiliary tools is more focus on their usability and business logic, rather than content display. That means the interaction requirements are the core of development.

User volume level: low (ZLP use it once a week).

Data volume level: low (ZLP's data before 3 months are deleted automatically).

Overall, ReCAST as a tool website, each webpage at ReCAST should as little as possible for improving performance. For meeting requirements, it should be developed as fast as possible. To implement business logic is the first priority followed by usability, performance and security in sequence.

6.1. Continuous Deployment (or Delivery)

For rapid response to demand, our development model should not always follow the waterfall model as its hypothesis is stable and non-changeable requirements. Therefore, it is necessary to deliver our product online in a certain period, so that user can use it from time to time and give the feedback on time for modifying requirements in time and improving user's satisfaction on the product.

Here we adopt the AWS cloud computing platform for continuous deployment. The up-to-date ReCAST development situation will be published on that for letting project members have a better understanding of ReCAST by access a specific IP address.

6.2. Website Module Segmentation

The website is divided into three modules: frontend, backend, and server. The frontend is used for presentation, the backend is used to provide business and data, and the server provides accessibility and connectivity.

The front-end shall be implemented by HTML+CSS+JavaScript; Website backend business shall be supported by the Django framework of Python; Server connectivity support by AWS, The Django built-in server uWSGI can be used for development.

Most of frontend user interaction implemented by Javascript framework Jquery.js and style design implemented by CSS framework Bootstrap. Also, for improving the scalability and maintainability of ReCAST, development frameworks adopt a more advanced version, technique stack proposal as the following:

Backend: Python 3.6 (Django 3.0)

Front-end: HTML5, CSS3, Javascript 5 (JQuery.js: 2.6; Echarts.js: 4.7; D3.js: 4.0;)

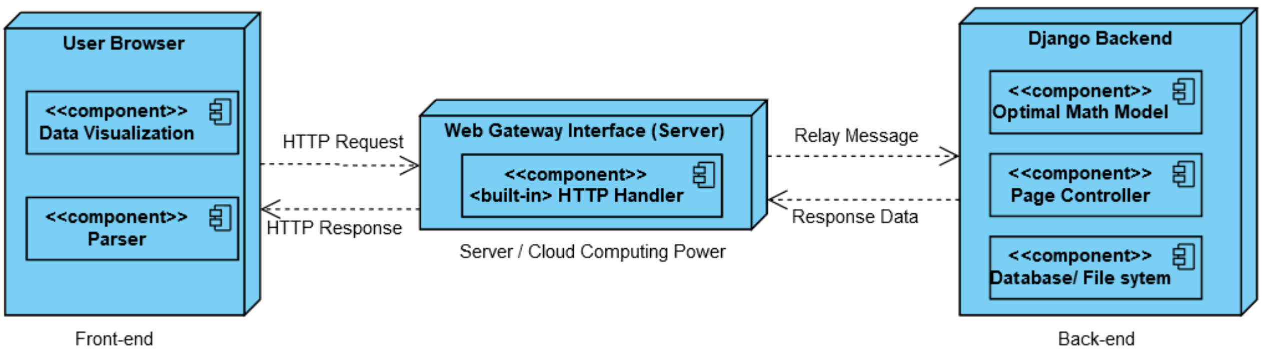


Figure 11System Modules Deployment Diagram

For some groups of functions at the system, we call them components, its main purpose to provide a specific business. The main components include Data Visualization and Parser component at the front-end, HTTP Handler component at the server, Database, Optimal Math Model and Page Controller.

The data visualization module provides ZLP scenario visualization so that ZLP can compare different scenarios results intuitively. It should be placed on the front end and implemented using JavaScript. Currently available options are: Echarts.js, D3.js, DataTable.js. etc.

The parser is the input and output module of the system, which provides functions for processing Excel. Among them, the input module can use DataTables.js to implement the display effect in the front end; the system output can use the pandas framework to generate the Excel file in the back end.

Django's built-in uWSGI can be used as a server to quickly implement access functions by combining with cloud computing platforms. And that is the HTTP access request processing function.

Optimal Math Model provide core business algorithm to generate Target Allocation scenarios by invoking the Gorubi mathematical API interface

Page Controller defines a webpage jump relationship (route logic) by the Django framework.

The database module provides system persistence ability, user’s history task records all stored in here. By default, Django provides a built-in database SQLite also with the ORM module to achieve persistence.

6.3. Maintenance requirements

Current server development environment: Linux Ubuntu (on AWS) + Django built-in database.

It may need the support of the supplier when the ReCAST is up and running – need more powerful could server for deploying ReCAST on that.

7. UI PROTOTYPE

An ZLP page relationship diagram is put forward based on the above business. According to the three states of ZLP, three corresponding modules are developed. Each module contains three to four pages. There are eleven pages, of which the Index page is not part of any module.

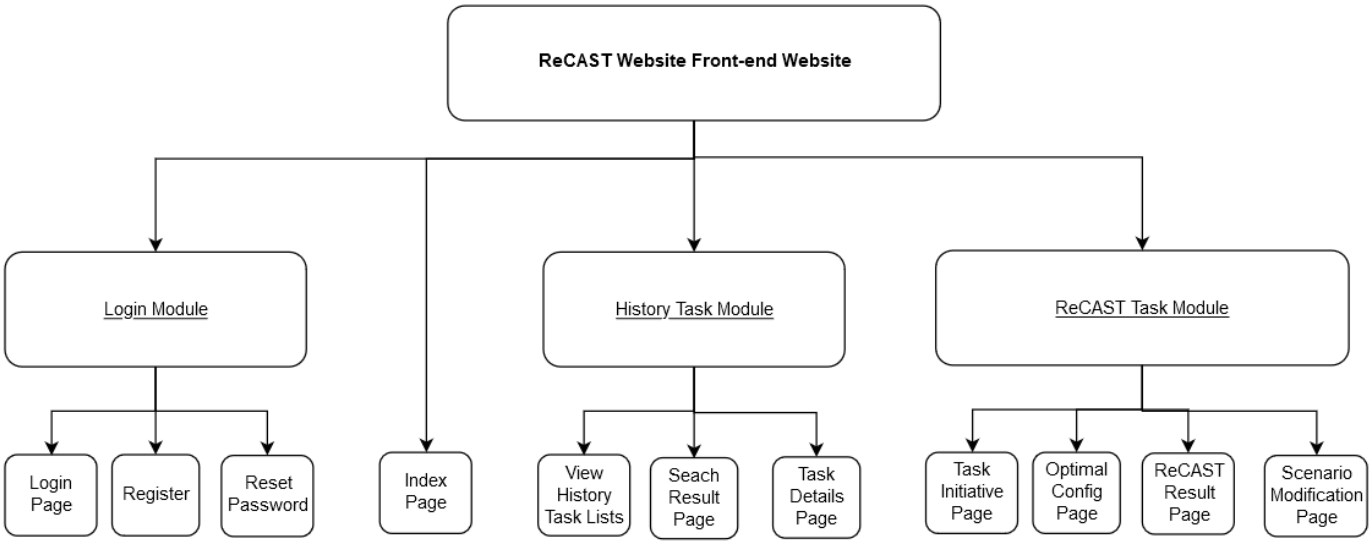


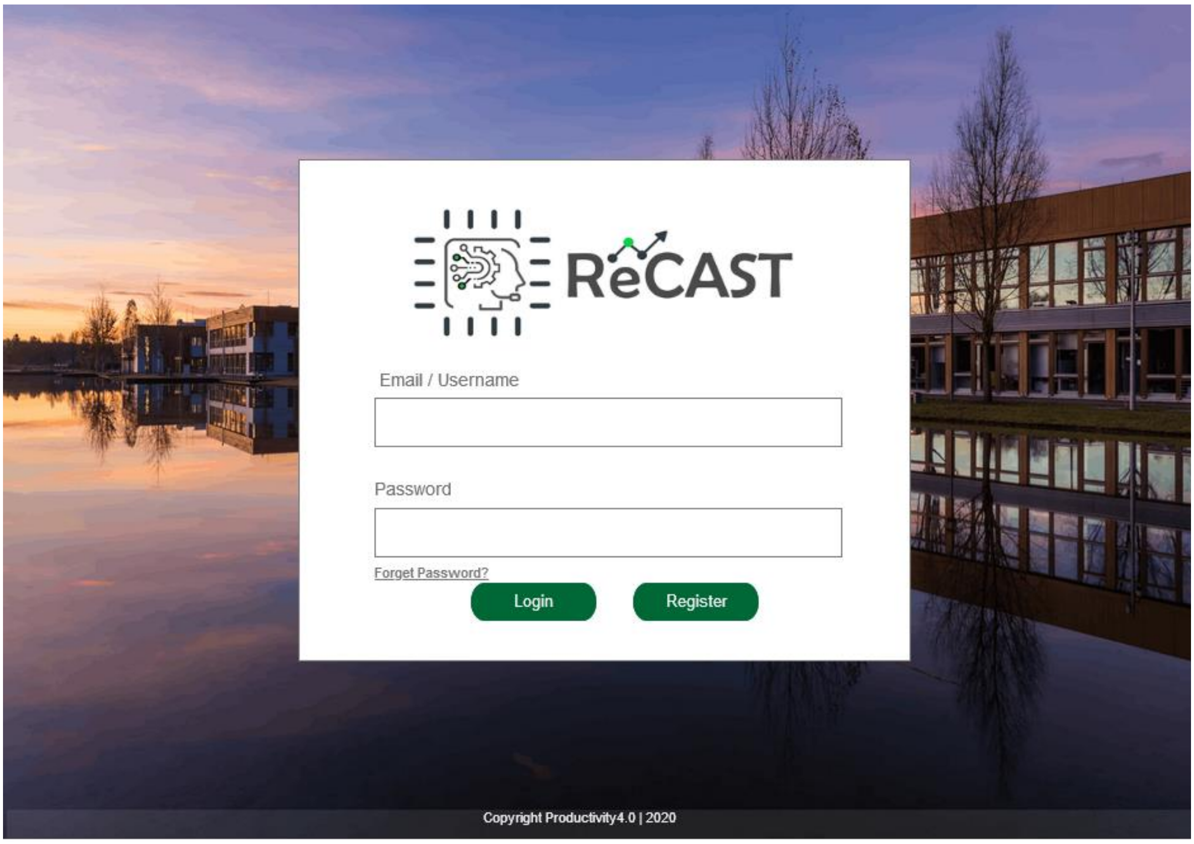
Figure 12ZLP's User Module and Visible Pages

In Login Module: It includes Login Page, Register Page, Reset Password Page.

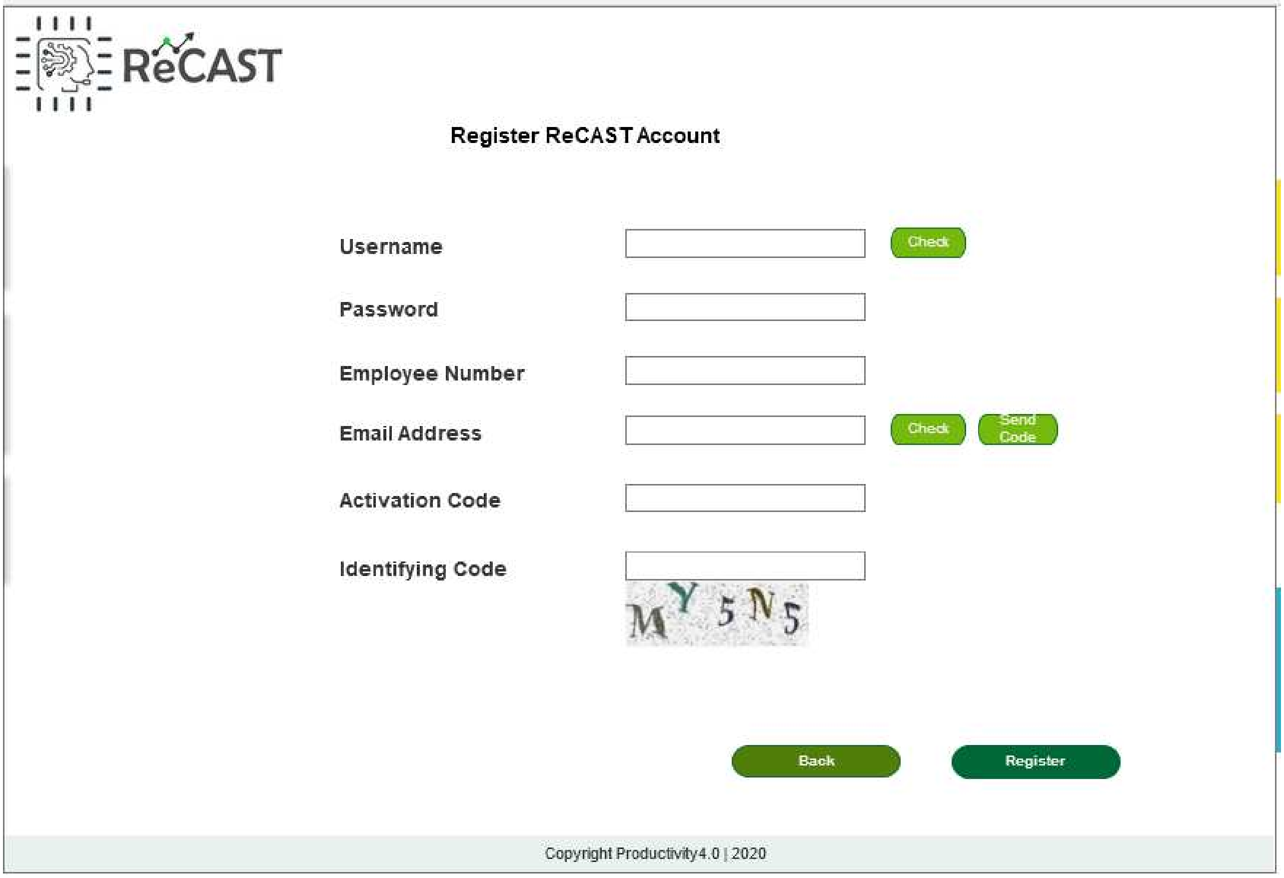
In History Task Module: View History Task Lists Page, Search Result Page, Task Details Page.

In ReCAST Task Module: Task Initiative Page, Optimal Config Page, ReCAST Result Page and Scenario Modification Page.

- **Login Page**

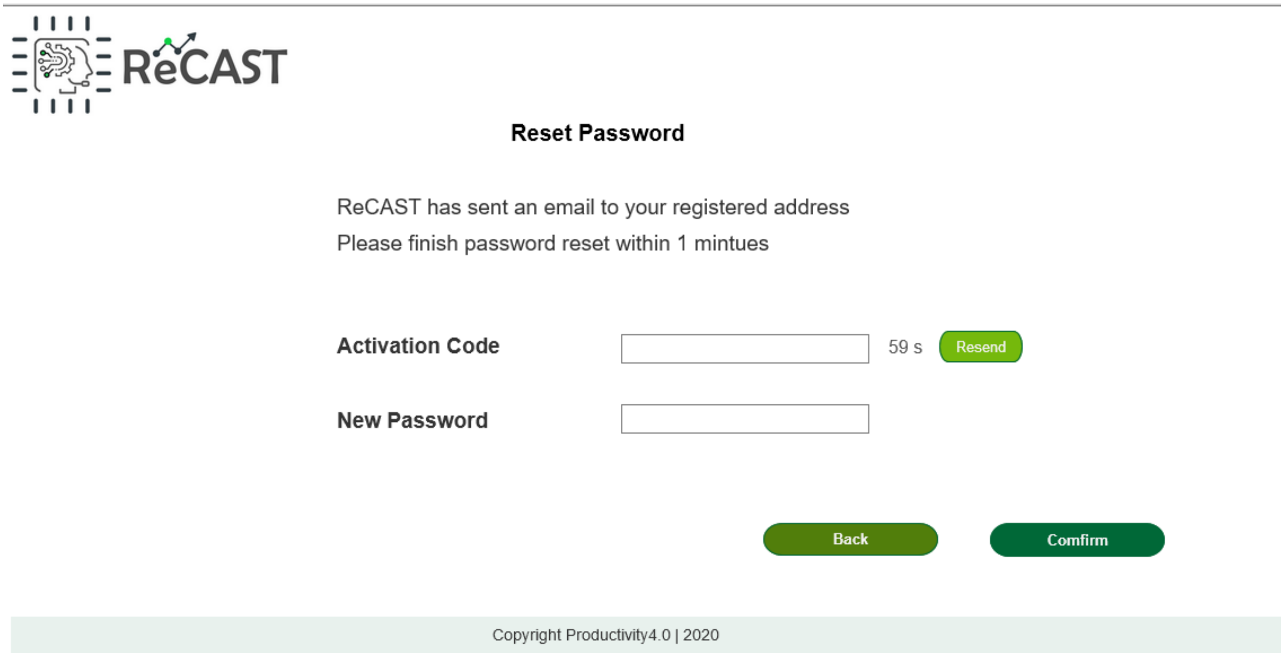


- **Register Page**,

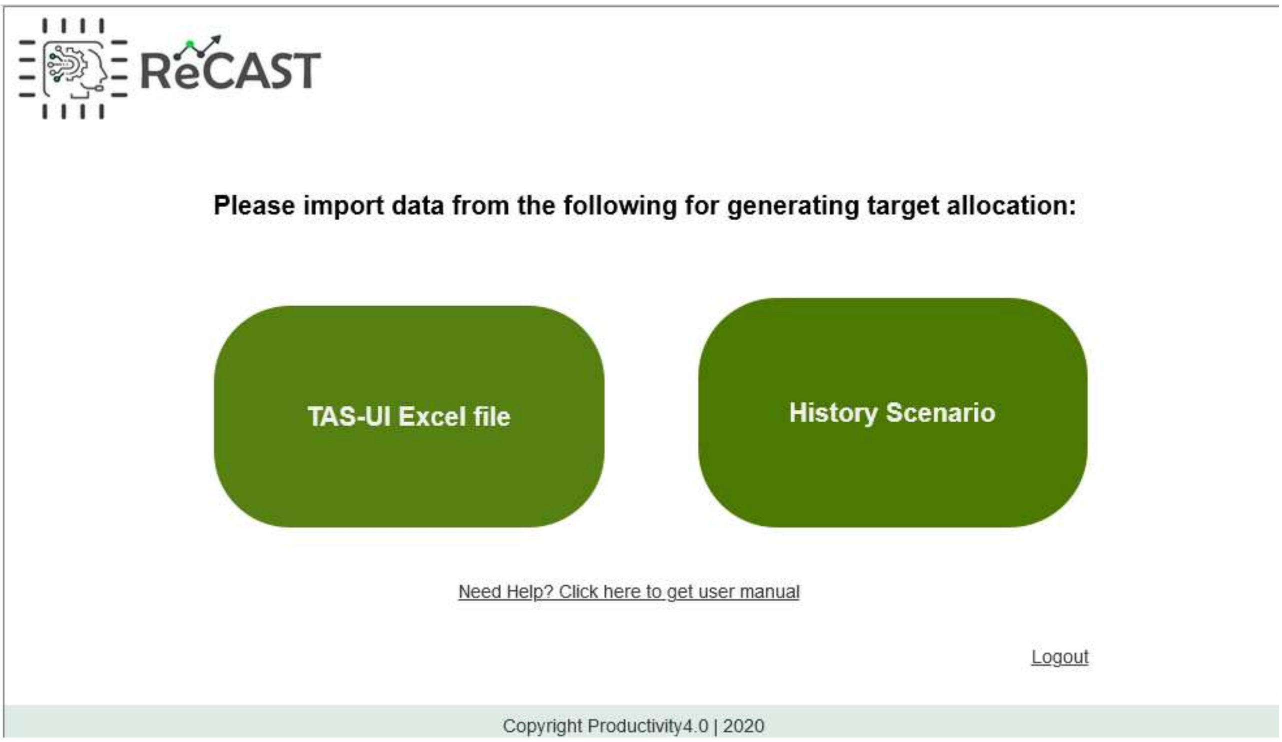


\

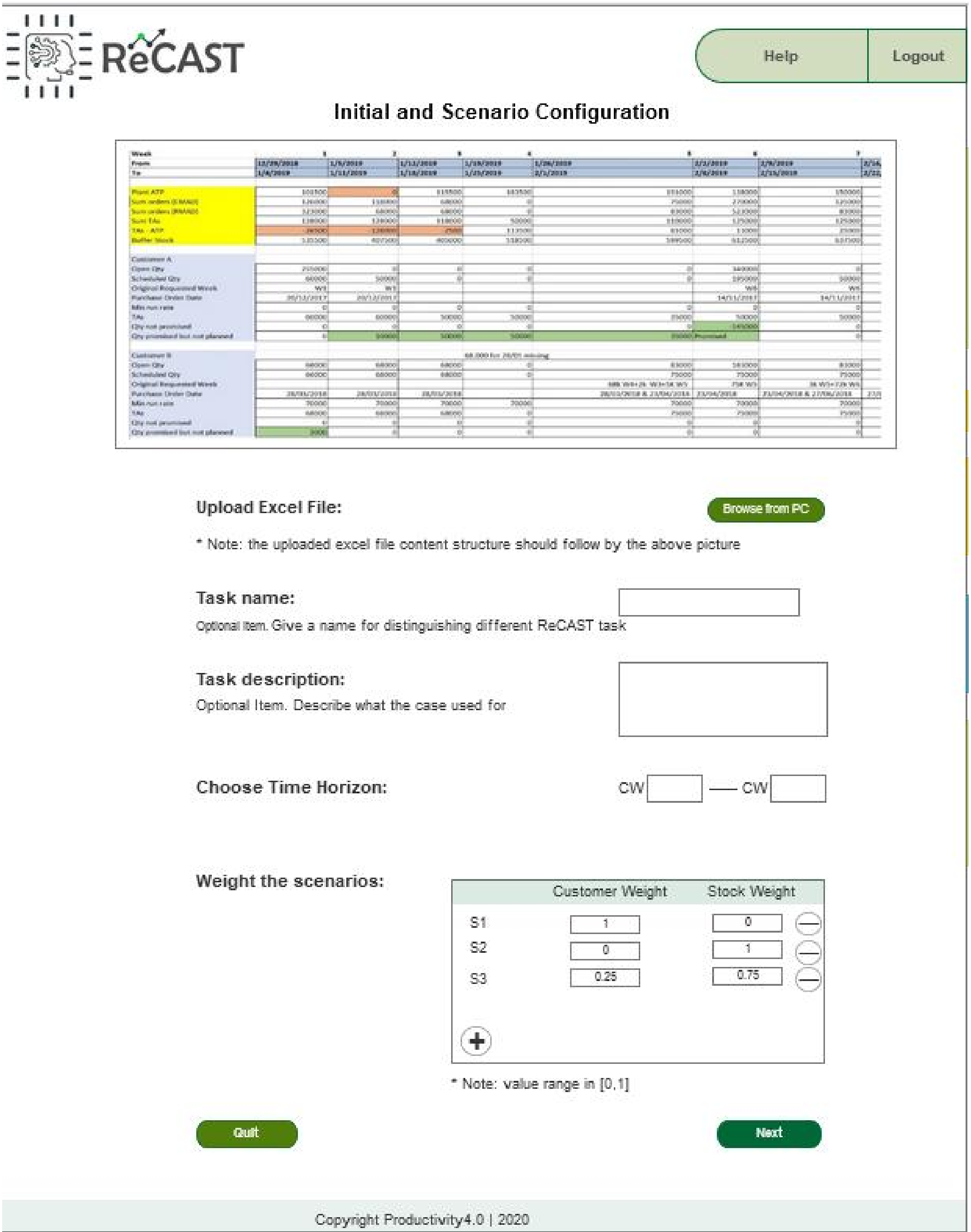
- **Reset Password Page.**



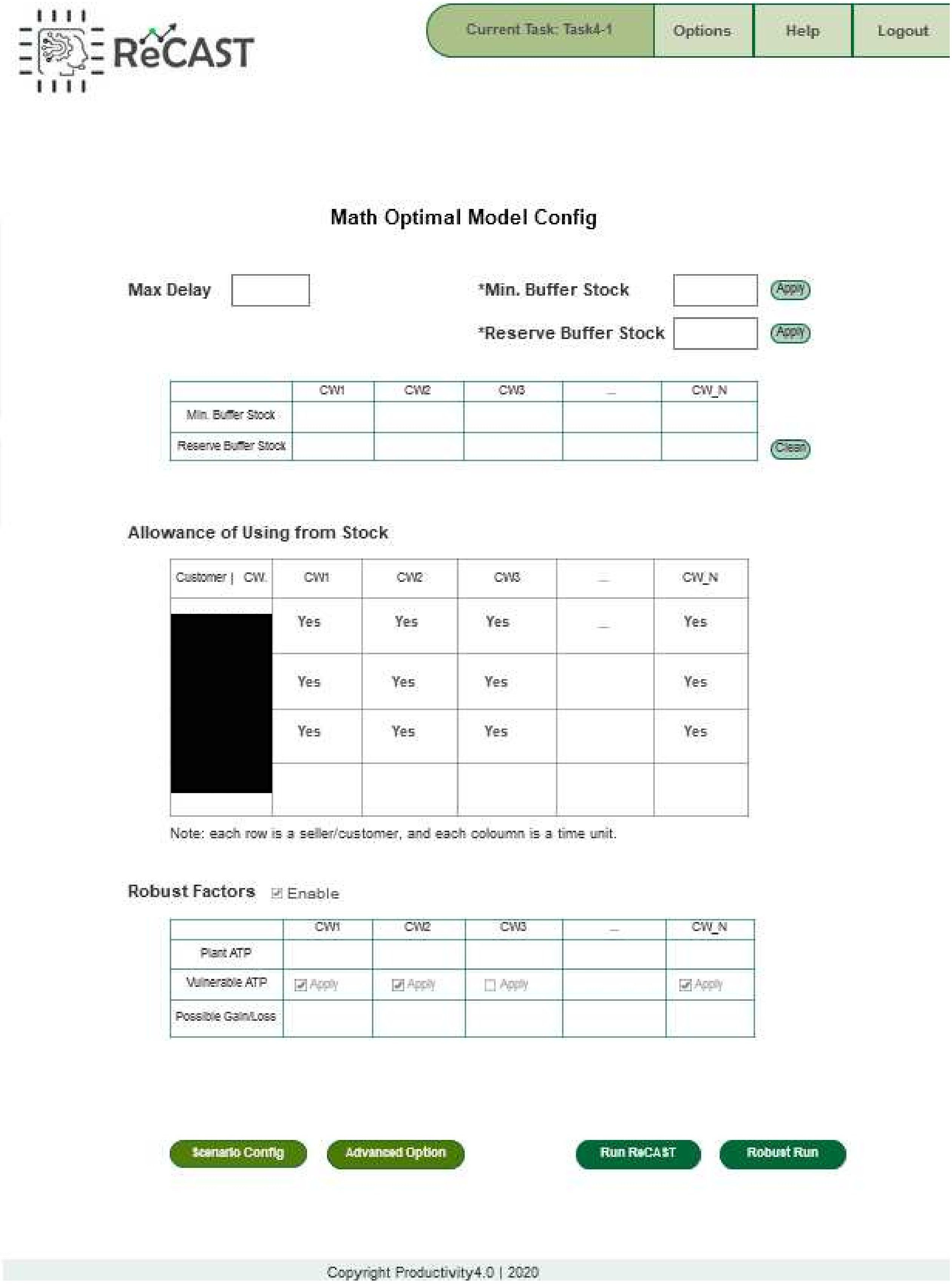
- **Inde Page**



- **Task Initative Page**



- **Optimal Config Page**



- , ReCAST Result Page

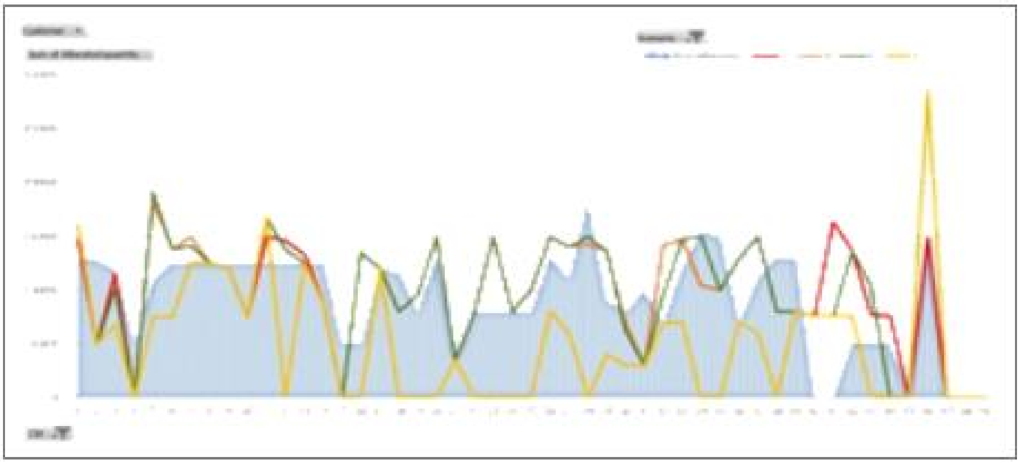
- **Scenario Modification Page**

ReCAST

Current Task: Task4	Options	Help	Logout

Modify Scenario

Dual Visualization



Scenario Informato

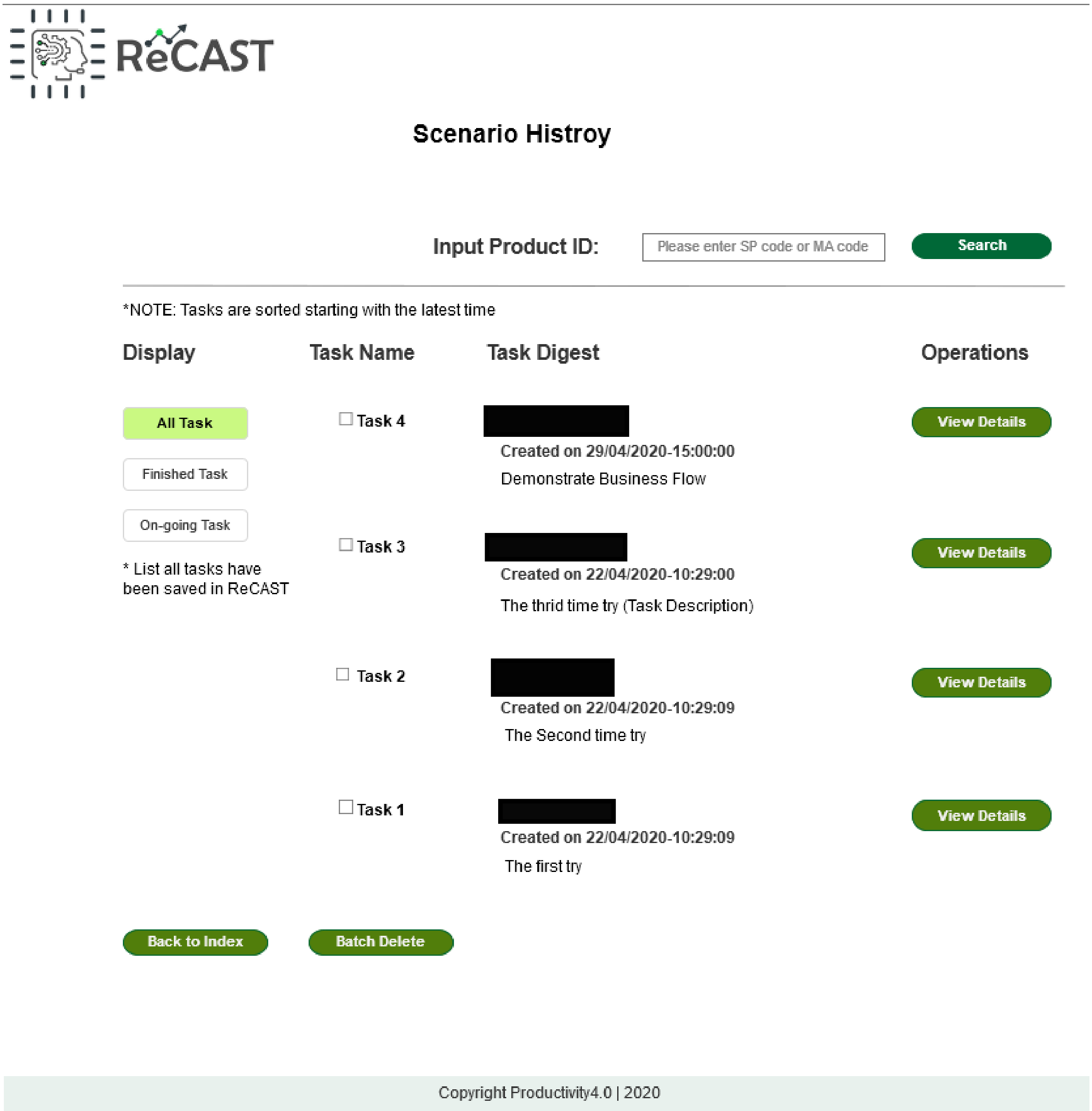
Customer | CW.	CW1			CW2			...	CW_N		
[redacted]	A-ATP	A-Stock	Sum	A-ATP	A-Stock	Sum	...	A-ATP	A-Stock	Sum
	90	10	100	90	10	100		90	10	100
	90	10	100	90	10	100		90	10	100
——										

Orders | CW.	CW1	CW2	CW3	...	CW_N
ATP				...	
Buffer Stock					
Order 1					
——					

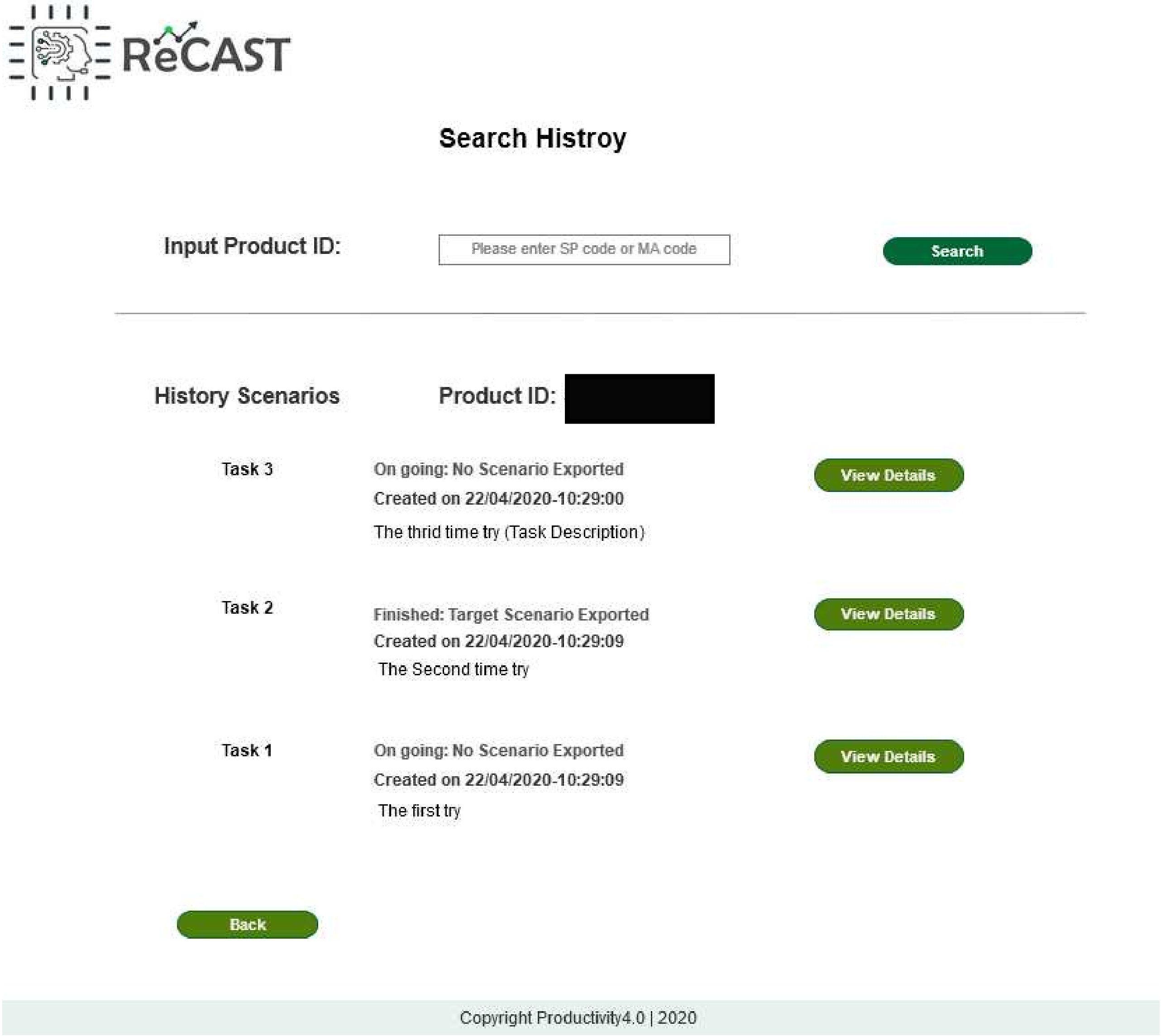
Check

Back Export

- **View History Task Lilsts Page**



- **Search Result Page**



- **Task Details Page**

ReCAST

Task Information Descriptions

Task Name: **Task 4**

Created on 22/04/2020-10:29:59

Product ID: [redacted]

Descriptions: Demonstrate Business Flow

Time Horizon: **CW 2 - CW 12**

Max Delay: 10

Scenarios weight:

	Customer Weight	Stock Weight
Scenario 1	**0.1**	**0.9**
Scenario 2	**1**	**0**
Scenario 3	**0.5**	**0.5**

Buffer Stock Config Table

	CW1	CW2	CW3	...	CW_N
Min. Buffer Stock	100	100	100	...	100
Reserve Buffer Stock	200	200	200	...	200

Allowance of Using from Stock

Customer | CW.	CW1	CW2	CW3	...	CW_N
[redacted]	Yes	Yes	Yes	...	Yes
	Yes	Yes	Yes		Yes
	Yes	Yes	Yes		Yes
....					

Robust Ractors

	CW1	CW2	CW3	...	CW_N
Plant ATP	0		80000		900
Vulnerable ATP	☐ Apply	☐ Apply	☐ Apply		☐ Apply
Possible Gain/Loss	100	-20			1

Exported Scenario: Scenario 2

Customer | CW.	CW1			CW2			...	CW_N		
	A-ATP	A-Stock	**Sum**	A-ATP	A-Stock	**Sum**	...	A-ATP	A-Stock	**Sum**
[redacted]	90	10	**100**	90	10	**100**		90	10	**100**
	90	10	**100**	90	10	**100**		90	10	**100**
....										

Back to List

Delete Task

Restore

A.8 ReCAST Sample Code

In this section, samples of mathematical optimization and decision support tool code are presented.

A.8.1 MATLAB code

This subsection presents MATLAB source code for the initial development of the mathematical model and sensitivity analysis. Note that this model was updated in the final Python model presented to the case study company. Here we aim to demonstrate the code in MATLAB and the proposed sensitivity analysis.

```

clear
close all
clc

%DEFINE PARAMETERS AND INPUTS

A = [ 101500  0 115500  163500  191000  138000  150000  154000  128000
      150000  93500 146500  140000  150000  150000  150000  150000
    150000  150000  150000  150000  150000  150000  150000  150000
    150000  163500  150000  136500  150000  152500  150000  147500
    150000  150000  150000  150000  150000  150000  150000  163500
    136500  150000  150000  162500  137500  159500  150000  150000
    ];
Atp = A';

Ord = [60000  50000 0 0 0 195000  50000 50000 45000 0 0 155000  50000
    15000 0 135000  50000 0 0 85000 35000 0 0 0 0 170000  0 0 149000  0
     0 0 149000  0 0 0 149000  0 0 0 0 149000  0 0 0 149000  0 0 0;
    66000 68000 68000 0 75000 75000 75000 75000 75000 75000 75000 75000
     75000 74000 0 0 70000 80000 80000 80000 0 80000 150000  80000
    80000 80000 60000 130000  40000 30000 30000 70000 70000 104000
    100000  70000 60000 80000 80000 76000 76000 76000 76000 76000 0 0 0
     0 0];
```

```

Ni = size(Ord,1);
Nd = size(Ord,2);
Nt = size(Ord,2);
X = ones(Ni,Nt);
BSmin = 400000*ones(Nt,1);
RB = 100000*ones(Nt,1);
MaxDelay = 49;

% Calculation of Penalty Function based on MAX-Delay
P = zeros(Ni,Nd,Nt);
for i=1:Ni
    for d=1:Nd
        if Ord(i,d) > 0
            idx_Max = min([d + MaxDelay-1, Nd]);
            idx_min = max([1, d - MaxDelay-1]);
            value_p = zeros(Nd,1);
            for m=idx_min:idx_Max
                value_p(m)= 1-(abs(d-m)/MaxDelay);
            end
            P(i,d,:) = value_p;
        end
    end
end

% DEFINE VARIABLES
AQ = sdpvar(Ni,Nd,Nt,'full');
AS = sdpvar(Ni,Nd,Nt,'full');
BS = sdpvar(Nt,1,'full');
Z = binvar(Nt,1,'full');
Idx_z = (Atp == 0);
eps = 1e-4;

% DEFINE CONSTRAINTS
Constraints = [ BS(1,1) == 535500,...
```

```
    (BS(2:Nt) == BS(1:Nt-1) + Atp(1:Nt-1) ...
    - squeeze(sum(sum(AQ(:,:,1:Nt-1),2),1))...
    - squeeze(sum(sum(AS(:,:,1:Nt-1),2),1))):'Buff',...
    (squeeze(sum(AQ,3)) + squeeze(sum(AS,3)) <= Ord):'Orde',...
    (squeeze(sum(sum(AQ,2),1)) <= Atp):'Reso',...
    (squeeze(sum(sum(AS,2),1)) <= BS):'Stoc',...
    (BSmin - BS <= 0):'BuMi',...
    (sum((Atp - squeeze(sum(sum(AQ,2),1)))) <= sum(RB)):'ResB',...
    0<= AS , 0<=AQ,...
    (squeeze(sum(AS,2)) == squeeze(sum(AS,3))):'AsAq'];

Constraints_bin = [
    Z(~Idx_z) <= ((squeeze(sum(sum(AQ(:,:,~Idx_z),2),1)))./Atp(~Idx_z))
   + eps,....
    Z(Idx_z) == 1,...
    AS <= reshape(repmat(((repmat(Z',[2 1]).*X).*1e5),1,Nt),Ni,Nd,Nt)];
     % test this for separ

% DEFINE OBJECTIVES

ObjAllocation = - (sum(sum(sum(AQ.*P)))+ sum(sum(sum(AS))));
ObjStock = - sum( (Atp - squeeze(sum(sum(AQ,2),1) )) - RB);

NormObjAllocation = (ObjAllocation+1191970.125)
   /(4911326.551-1191970.125);
NormObjStock = (ObjStock-2565999) /(2565999);

% SET OPTIONS FOR YALMIP AND SOLVER
options = sdpsettings('verbose',1,'solver','gurobi');
options.gurobi.IntFeasTol= 1e-5; %deafault
options.gurobi.FeasibilityTol= 1e-9;   %1e-9
options.gurobi.MIPGap = 1e-8;  % 1e-8
options.gurobi.OptimalityTol = 1e-9;  %1e-9
options.gurobi.TimeLimit = 200;
```

```
% SOLVE THE PROBLEM WITH ITERATION ON WEIGHTS
Nw = 9;

AQ_sol_wkly_C1= zeros(Nt,Nw+1);
AS_sol_wkly_C1= zeros(Nt,Nw+1);

AQ_sol_wkly_C2 = zeros(Nt,Nw+1);
AS_sol_wkly_C2 = zeros(Nt,Nw+1);

AQ_sol_wkly = zeros(Nt,Nw+1);
AS_sol_wkly = zeros(Nt,Nw+1);

BS_sol_wkly = zeros(Nt,Nw+1);
Z_sol_wkly = zeros(Nt,Nw+1);
% Total_alllocation = zeros(Nt,Nw+1);

sum_AQ = zeros(Nw+1,1);
sum_AS = zeros(Nw+1,1);

ObjAllocation_sol = zeros(Nw+1,1);
ObjStock_sol = zeros(Nw+1,1);

NormObjAllocation_sol = zeros(Nw+1,1);
NormObjStock_sol = zeros(Nw+1,1);
NormObj_sol = zeros (Nw+1,1);

DuBuff = zeros (Nt-1,Nw+1);
DuReso = zeros(Nt,Nw+1);
DuOrde = zeros(Ni*Nt,Nw+1);
DuStoc = zeros(Nt,Nw+1);
DuBuMi = zeros(Nt,Nw+1);
DuResB = zeros(Nt,Nw+1);
DuAsAq = zeros(Ni*Nt,Nw+1);
DuZ = zeros(Nt,Nw+1);

```

```
for n=1:Nw+1
    w =(n-1)/Nw;
    sol = optimize([Constraints;Constraints_bin],w*NormObjStock + (1-w)*NormObjAllocation,options);

    % Analyze error flags
    if sol.problem == 0
        AQ_sol_wkly_C1(:,n)= value(squeeze(sum(sum(AQ(1,:,:),2),1)));
        AS_sol_wkly_C1(:,n)= value(squeeze(sum(sum(AS(1,:,:),2),1)));

        AQ_sol_wkly_C2(:,n)= value(squeeze(sum(sum(AQ(2,:,:),2),1)));
        AS_sol_wkly_C2(:,n)= value(squeeze(sum(sum(AS(2,:,:),2),1)));

        AQ_sol_wkly(:,n)= value(squeeze(sum(sum(AQ(:,:,:),2),1)));
        AS_sol_wkly(:,n)= value(squeeze(sum(sum(AS(:,:,:),2),1)));

        BS_sol_wkly(:,n) = value(BS);
        Z_sol_wkly(:,n) = value(Z);

        sum_AQ (n,1) = value(squeeze(sum(sum(sum(AQ,3),2),1)));
        sum_AS (n,1) = value(squeeze(sum(sum(sum(AS,3),2),1)));

        ObjAllocation_sol(n,1) = value(ObjAllocation);
        ObjStock_sol(n,1) = value(ObjStock);

        NormObjAllocation_sol(n,1) = value(NormObjAllocation);
        NormObjStock_sol(n,1) = value(NormObjStock);
        NormObj_sol(n,1)= w*value(NormObjStock) + (1-w)*value(NormObjAllocation);

        ResultOfStock = value(reshape(repmat(((repmat(Z',[2 1]).*X).*1e6),1,Nt),Ni,Nd,Nt));

        % Sensitivity analysis of binary variable
        Constraints_bin_fxd = [
```

```
            (Z_sol_wkly(~Idx_z,n) <= ((squeeze(sum(sum(AQ(:,:,~Idx_z)
,2),1)))./Atp(~Idx_z))+ eps):'Zatp',....
            Z_sol_wkly(Idx_z,n) == 1,...
            (AS <= reshape(repmat(((repmat(Z_sol_wkly(:,n)',[2 1]).*X)
.*1e5),1,Nt),Ni,Nd,Nt)):'AlSt'];

        sol_bin_fxd = optimize([Constraints;Constraints_bin_fxd],w*
NormObjStock + (1-w)*NormObjAllocation,options);

        if sol_bin_fxd.problem == 0
            DuBuff(:,n)= dual(Constraints('Buff'));
            DuOrde(:,n)= dual(Constraints('Orde'));
            DuReso(:,n)= dual(Constraints('Reso'));
            DuStoc(:,n)= dual(Constraints('Stoc'));
            DuBuMi(:,n)= dual(Constraints('BuMi'));
            DuResB(:,n)= dual(Constraints('ResB'));
            DuAsAq(:,n)= reshape(dual(Constraints('AsAq')),Ni*Nt,1);

        else
            display('Hmm, dual problem not solved!');
            sol.info
            yalmiperror(sol.problem)
        end

        for j=1:Nt
            if Z_sol_wkly(j,n)== 0
                Z_s = 1 ;
            else
                Z_s = 0;
            end

            Constraints_bin_OneFxd = [Constraints_bin, Z(j)== Z_s];
            sol_bin_OneFxd = optimize([Constraints;
Constraints_bin_OneFxd],w*NormObjStock + (1-w)*NormObjAllocation,
options);
```

```
            if sol_bin_OneFxd.problem == 0
                DuZ(j,n)= NormObj_sol(n,1)- value(w*NormObjStock + (1-w
   )*NormObjAllocation);
            else
                display('Hmm, binary dual not solved');
                sol.info
                yalmiperror(sol.problem)
            end
        end
    else
        display('Hmm, something went wrong!');
        sol.info
        yalmiperror(sol.problem)
    end
end

for j=1:Nt
    Constraints_bin_OneFxd = [Constraints_bin, Z(j)== Z_sol_wkly(j,n)];
    sol_bin_fxd = optimize([Constraints;Constraints_bin_OneFxd],w*
   NormObjStock + (1-w)*NormObjAllocation,options);
end

% Plot Sensitivity Analysis
figure;plot(DuZ(1:Nt, :));
```

Listing A.3: MATLAB source code for initial development.

A.8.2 Python code

In this subsection, the developed Python model for the development of ReCAST core is presented. This model is used within the developed tool.

```
# 1- Definition of Indexes and parameters
# This part should be edited according to parser
from gurobipy import *
import pandas as pd
```

```python
import numpy as np
# This is just a sample creation model. It can load an excel file from
    local system.
df = pd.read_excel(io='181218_ZZZZZ extract_SP000646194.xls', header=
    None, skiprows=2, usecols="I:BE") #H:BF manualy added
df_main = df.iloc[[1,11,14,21]]
df_list = df_main.to_numpy().tolist()
confirmedOrder = {"customer1": df_list[2],"customer2": df_list[3]}
plantATP = df_list[0]
MBS = [400000]*len(df_list[0])
RBS = [100000]*len(df_list[0])
ATP_NTA = 535500
packingUnit = 500
scenarioList = [[1, 0],[0.9, 0.1],[0.8,0.2],[0.7,0.3],
                [0.6,0.4],[0.5,0.5],[0.4,0.6],[0.3,0.7],
                [0.2,0.8],[0.1,0.9],[0,1]]
maxDelay = 10

# defining the list of indices
time_periods = plantATP
allocation_time = plantATP
customers = list(confirmedOrder.keys())
atp = (np.array(plantATP)) / packingUnit
buffer_stock_min = np.array(MBS) / packingUnit
reserve_Buffer = np.array(RBS) / packingUnit
intial_Buffer_Value = ATP_NTA / packingUnit
order = []
for key in confirmedOrder:
    order.append(confirmedOrder[key])
orders = np.array(order) / packingUnit

# Strategic allocation decision for using from stock for specific
    customer or not
bin_usefrom_stock = [1]*len(orders)
for i in range(len(orders)):
```

```
    bin_usefrom_stock[i] = [1]*len(atp)
    # this is multidimensional list based on number of weeks and number of custmers
max_delay = maxDelay

# Defining penalty function of allocation later than requested data
penalty_coef = [[0]*len(atp) for i in range(len(atp))]
for time in range(len(atp)):
    idx_max = min(time + max_delay-1 , len(atp))
    idx_min = time
    value_penalty = [0]*len(atp)
    for acceptable_penalty_loop in range(idx_min,idx_max):
        value_penalty[acceptable_penalty_loop]= round((1-((abs(time-acceptable_penalty_loop) / max_delay))),3)
        #defined function here is linear and it could be shifted to exponentioal form to force the model penalty_coef[time] = value_penalty
penalty_coef_Stock = [[0]*len(atp) for i in range(len(atp))]

for time in range(len(atp)):
    idx_max = min(time + 1 , len(atp))
    idx_min = time
    value_penalty_Stock = [0]*len(atp)
    for acceptable_penalty_loop in range(idx_min,idx_max):
        value_penalty_Stock[acceptable_penalty_loop]= 1

    penalty_coef_Stock[time] = value_penalty_Stock

eps = 1e-4
# this is the value for relaxing solver about the binary constraint
big_M = 1e+5

# 2- Definition of Model & Variables
# The variable are allocated quantity from atp, allocated quantitiy from stock, buffer stock, and binary variable for using from stock
```

```
    or not

reCAST = Model('ReCAST')

var_Allocation_ATP = reCAST.addVars(len(orders), len(atp), len(atp),lb
    = 0, vtype = GRB.INTEGER, name = 'Var_Allocation_ATP')

var_Allocation_Stock = reCAST.addVars(len(orders), len(atp), len(atp),
    lb = 0, vtype = GRB.INTEGER, name = 'Var_Allocation_Stock')

var_BufferStock = reCAST.addVars(len(atp), vtype = GRB.INTEGER, name =
    'Var_BufferStock')

var_ReserveBuffer = reCAST.addVars(len(atp), vtype = GRB.INTEGER, name
    = 'Var_ReserverBufferStock')

var_z = reCAST.addVars(len(atp), vtype = GRB.BINARY , name = '
    useStockOrNot')

# 3- Definition of Constraints
# In this section the constraints defines in two sub section, linear
    constraints and binary constraints.

# 3-1 Linear Constraints

reCAST.addConstrs((var_Allocation_ATP.sum(i,r,'*') +
    var_Allocation_Stock.sum(i,r,'*') <= orders[i][r]
                 for i in range(len(orders)) for r in range(len(atp))),
     name = 'cons_orders');

reCAST.addConstr((var_BufferStock[0] == intial_Buffer_Value) , name = '
    con_Buffer_Initial');

reCAST.addConstrs((var_BufferStock[t] - var_BufferStock[t-1] +
                   var_Allocation_ATP.sum('*', '*', t-1) +
```

```
                    var_Allocation_Stock.sum('*', '*', t-1)  == atp[t-1]
                    for t in range(1, len(atp))), name = 'con_Buffer');

reCAST.addConstrs((var_Allocation_ATP.sum('*', '*', t) + var_ReserveBuffer[t] <= atp[t]
                    for t in range(len(atp))), name = 'con_Resources');

reCAST.addConstrs((var_ReserveBuffer[t] <= reserve_Buffer[t] for t in range(len(atp))), name= 'RBS_Goal' );

reCAST.addConstrs((var_Allocation_Stock.sum('*', '*', t) <= var_BufferStock[t]
                    for t in range(len(atp))), name ='con_stock');

reCAST.addConstrs((var_BufferStock[t] >= buffer_stock_min[t]
                    for t in range(len(atp))), name = 'con_buffer_stock_min');

reCAST.addConstrs((var_Allocation_Stock.sum(i,'*',t) - var_Allocation_Stock.sum(i,r,'*') == 0
                    for i in range(len(orders)) for t in range(len(atp)) for r in range(len(atp))),
                   name = 'con_AsAq');

# 3-2  Binary Constraints

z_indx_posi_list = [i for i in range(len(atp)) if atp[i] != 0]

reCAST.addConstrs( (var_z[t] - (var_Allocation_ATP.sum('*','*',t)/atp[t]) <= eps
                     for t in z_indx_posi_list), name = 'con_bin_allocation');

z_indx_atp0 = [i for i in range(len(atp)) if atp[i] == 0]

```

```
reCAST.addConstrs((var_z[t] == 1 for t in z_indx_atp0), name= '
    con_bin_z_eqOne');

reCAST.addConstrs((var_Allocation_Stock[i,r,t]- (bin_usefrom_stock[i][t
    ]*big_M * var_z[t]) <=  0
                    for i in range(len(orders)) for r in range(len(atp))
     for t in range(len(atp))),
                   name='con_bin_UseStockorNot');

# 4- Definition of Objectives
# The problem modeled as bi-objective optimization which normed by
    dividing to max values.

obj_Allocation = quicksum((var_Allocation_ATP[i,r,t] * penalty_coef[r][
    t]) + ((var_Allocation_Stock[i,r,t]) * penalty_coef_Stock[r][t])
                           for t in range(len(atp)) for i in range(len(
    orders)) for r in range(len(atp)) )

obj_ReserveStock = quicksum(var_ReserveBuffer[t] for t in range(len(atp
    )))

reCAST.Params.MIPGap = 1e-9 # -4 00
reCAST.Params.IntFeasTol = 1e-9 # -5 -9
reCAST.Params.FeasibilityTol = 1e-9 # -6 -9
reCAST.Params.OptimalityTol = 1e-9

for scenario in range(len(scenarioList)):
    weight_Allocation = scenarioList[scenario][0]
    weight_ReserveBuffer = scenarioList[scenario][1]

    reCAST.setObjective(weight_Allocation*obj_Allocation +
    weight_ReserveBuffer*obj_ReserveStock, GRB.MAXIMIZE)

    reCAST.optimize()
    print('--------obj=--------', reCAST.ObjVal)
```

```
    rows_reserve = ["Reserve Plan"]
    columns_reserve = df_list[1].copy()
    reserve_Plan = pd.DataFrame(columns=columns_reserve, index=
rows_reserve, data=0.0)

    for t in var_ReserveBuffer.keys():
        if(abs(var_ReserveBuffer[t].x>1e-6)):
            reserve_Plan.iloc[0,t] += np.round(var_ReserveBuffer[t].x)

    print('-----sum_reserve=', reserve_Plan.sum(axis=1)*
weight_ReserveBuffer)

    rows_ATP = customers.copy()
    columns_ATP = df_list[1].copy()
    allocation_ATP_Plan = pd.DataFrame(columns=columns_ATP, index=
rows_ATP, data=0.0)

    for i,r,t in var_Allocation_ATP.keys():
        if(abs(var_Allocation_ATP[i,r,t].x>1e-6)):
            allocation_ATP_Plan.iloc[i,t] += np.round(
var_Allocation_ATP[i,r,t].x*packingUnit,0)

    print(allocation_ATP_Plan)
    print('-----sum_AQ=', allocation_ATP_Plan.sum(axis=1))

    rows_stock = customers.copy()
    columns_Stock = df_list[1].copy()
    allocation_Stock_Plan = pd.DataFrame(columns=columns_Stock, index=
rows_stock, data=0.0)

    for i,r,t in var_Allocation_Stock.keys():
        if(abs(var_Allocation_Stock[i,r,t].x>1e-6)):
            allocation_Stock_Plan.iloc[i,t] += np.round(
var_Allocation_Stock[i,r,t].x*packingUnit,0)

```

```
    print(allocation_Stock_Plan)
    print('-----sum_AS=', allocation_Stock_Plan.sum(axis=1))
```

Listing A.4: Developed python code in Jupyter notebook for ReCAST development.

A.8.3 ReCAST tool development

Code for ReCAST was developed using Pycharm IDE. The full source code is available in the submitted materials to this thesis. Here we present a screenshot of the main module and a list of modules developed within ReCAST. As is shown in Figure A.14 on the facing page, the ReCAST model consists of four main modules named 'django1', 'ReCAST', 'Static', and 'template'. The Django code is related to backend development of ReCAST, related to all parsers of data and the optimization model. For instance, the optimization model is located in views.py. The static part is related to static features of the web-application in the front end of the tool. By using the static feature, it reduces RAM usage of the server and speeds up the working environment. The template module aims to keep all html codes within the front end development. Moreover, the git repository of the developed tool is available at:

https://github.com/BruceConstantine/ReCAST

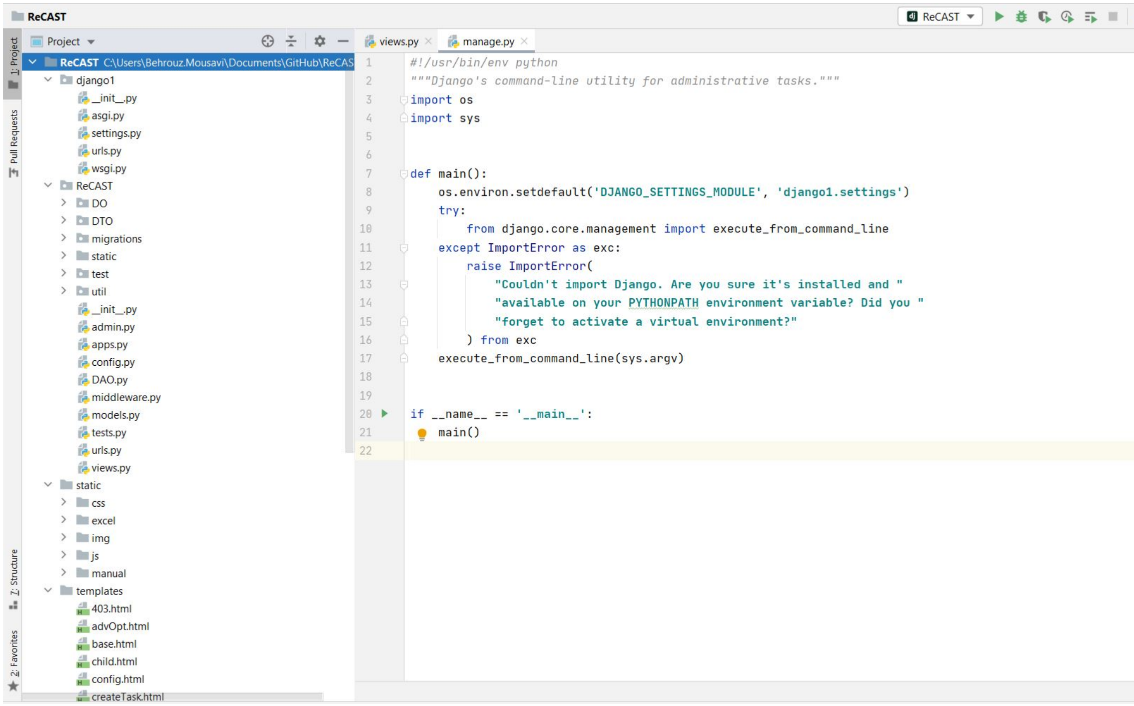


Figure A.14: Screen shot of Pycharm IDE used for ReCAST development.