LeArning and robuSt decision SupporT systems for agile mANufacTuring environments

Project Acronym:

ASSISTANT

Grant agreement no: 101000165

Deliverable no. and title	**D 4.1 - Requirements for production planning and scheduling**	
Work package	WP 4	Production planning and scheduling
Task	T 4.1	Assessment of key elements for flexible production
Subtasks involved		
Lead contractor	Institut Mines-Telecom (IMT) Alexandre Dolgui, alexandre.dolgui@imt-atlantique.fr	
Deliverable responsible	Institut Mines-Telecom (IMT) Simon Thevenin, simon.thevenin@imt-atlantique.fr	
Version number	v1	
Date	23/06/2021	
Status	Version 1.0	
Dissemination level	Public (PU)	

Authors

no.	Part. short name	Author name	Chapter(s)
1	PSA	Fabien D'Andras	4
2	IMT	Nicolas Beldiceanu	4, 5
3	IMT	Luo Dan	2, 4, 5, 7.2
4	IMT	Ramiz Gindulin	5
5	AC	Hristo Hristov	7.2
6	SE	Thomas Krause	4, 5, 7.2
7	FLM	Bart Meyer	4, 5
8	UCC	Helmut Simonis	4, 5, 7.2
9	SAG	Annelie Sohr	1, 4, 5, 7.2
10	IMT	Simon Thevenin	1, 2, 3, 4, 5, 6, 7.2
11	AC	Bavo Tielman	4
12	TUM	Sarah Wagner	3
13	UCC	Eduardo Vyhmeister	4.7
14	Biti	Per-Olov Östberg	3
15	LMS	Christos Gkournelos	3

Document History

Version	Date	Author name	Reason
v1	29.05.2021	see above table	Initial Version
v1	14.06.2021	see above table	Updated after the reviews of Thomas Krause and Sarah Wagner

Publishable Executive Summary

The industry 4.0 revolution currently underway foster mass customization and highly re-configurable production systems. In these complex production systems, planning and scheduling cannot rely on simple rules such as first in first out or the simple material requirement planning logic. Indeed, the resulting plans are sub-optimals, and they do not account for uncertainty properly. Companies must shift towards smart production planning and scheduling solutions, where an AI component can help the production manager to make the best use of data to suggest optimized plans and schedules. We propose the concept of intelligent digital twins, that enhance optimization approaches for production planning and scheduling with industry 4.0 technologies: internet of things, ontology, simulation, and machine learning. The resulting decision support system synchronizes with the shop floor, and they can properly account for various sources of uncertainty.

The work package 4 of ASSISTANT aims to develop such intelligent digital twin for production planning and scheduling, and this document aims to define their requirements and their architecture. First, based on an extensive literature review, we provided a state-of-the-art on the use of big data analytic, artificial intelligence (AI), Internet of Thing (IoT), and simulation in manufacturing operation management. Second, we define the requirements collected during multiple discussions with manufacturing use cases, and an in depth analysis of their shop floor operation. Finally, we provide the architecture for the modules developed in work package 4. This architecture rely on a data fabric that provide all relevant data. This data fabric is developed in work package 6, and the connection with the data fabric is defined in the architecture of ASSISTANT.

The development of the proposed intelligent digital twins for production planning and scheduling requires several breakthroughs in comparison with the current state-of-the-art. First, the requirements show the importance of model acquisition approaches for scheduling and production planning. Such an approach learns the constraints from data and accurate simulations of the shop floor. As a result, the constraint programming and mixed-integer linear programs commonly used in production planning and scheduling describe accurately the manufacturing processes. In addition, these models must incorporate uncertainties to properly hedge against unexpected events. However, the stochastic models do not scale well, and we will investigate appropriate optimization approaches to circumvent this issue.

Table of contents

List of Figures

List of Tables

1 Introduction

1.1 Motivation

Recent advances in technological development drive manufacturing systems to be adaptive, flexible, and reconfigurable. These advances foster the development of mass customization and more recently mass individualization, and they give a real competitive advantage to the industry. Indeed, they increase the service to customers in terms of product satisfaction and tight due dates. However, the production system becomes hard to manage, because the batch sizes tend to become smaller and smaller, the number of possible configurations of the shop floor increases, and many parameters vary significantly. In this context, to remain competitive, companies must make the best use of their resources, and thus plan and schedule their production carefully.

According to a recent survey [BARC, 2016], 74% of the companies still use Excel for production planning, and 33% rely solely on Excel to plan their production. Regarding scheduling, interviews conducted with ASSISTANT's use cases show that simple rules and human intuition remain the most common way to schedule production. As a human cannot analyze the huge number of scheduling alternatives, the resulting production schedule is sub-optimal. Advances in computer hardware and artificial intelligence algorithms allow a production planner to suggest optimized production plans and schedules. For instance, Thevenin et al. [2017] report that an optimization tool for short-term scheduling may reduce production costs (setup, raw material urgent deliveries, order rejection, tardiness) by 45% when compared to plans produced by hand. However, advanced planning and scheduling (APS) tools remain underused by the manufacturing industry. In addition, to realize their full potential, existing APS tools must be enhanced to take advantage of the massive amount of data generated on the shop floor, to integrate easily, to take advantage of new technologies fostered by industry 4.0 (IoT, simulation, big data analytics, etc.), and to adjust automatically to the constant changes on the shop floor. To overcome these shortcomings, we propose the concept of intelligent digital twins for production planning and scheduling.

The intelligent digital twin extends the classical digital twin to include prescriptive analytic. Prescriptive analytic in production planning and scheduling relies on optimization models (e.g., mathematical programming, and constraint programming), these models represent the production system through mathematical equations. An intelligent digital twin is an optimization model that automatically learns parts of the parameters and constraints to enhance its accuracy and to remain synchronized with the shop floor to make suggestions on how to proceed or even to take action autonomously. Such tools integrate machine learning techniques with optimization. On the one hand, statistical AI methods allow one to learn the parameters of the models to deal efficiently with uncertainties in the workshop. On the other hand, symbolic AI methods enable one to acquire explainable models to automatically learn the functioning of constantly changing production systems.

By fostering the adoption of AI for production planning and scheduling, and by increasing the accuracy of the production planning approaches, the resulting intelligent digital twin will have a strong impact on the manufacturing industry. Despite the short return on investment of prescriptive analytic tools, they are underused in the manufacturing industry. This situation may be explained by a lack of communications around these tools, or by the high initial investment costs. The implementation of prescriptive analytic tools requires high consulting costs to adapt software. Model acquisition from data will reduce these costs since the software will automatically adjust to the requirements of the shop floor. This will result in the large-scale adoption of prescriptive analytics in manufacturing. Automatic constraint learning will also help to better represent the production capacity during production planning. Classical approaches that rely on simple formulas to compare the load to the capacity, cannot account for the complex production processes encountered in today's shop floor (as resources can perform a wider range of tasks). We will investigate how machine learning can help to automatically learn the production capacity from the data. Finally, adaptive stochastic/robust optimization approaches provide production plans not only robust to various uncertainties, but that also select the states (resource usage, inventory level) to react appropriately when unknown parameters unfold. The intelligent digital twins for production planning and scheduling shall lead to production plans and schedules with the right level of agility.

1.2 Objective

This document aims to provide a precise description of the tools developed within WP4. The objective is to have all the elements required to ensure smooth development during the project. Of course, this architecture will likely evolve with the development of the tools during the project. As described in ASSISTANT's requirement engineering procedure available in the appendix, the definition of the architecture starts with the definition of the requirements.

The requirement analysis starts with a state-of-the-art on Artificial Intelligence (AI), Digital Twins (DT), Simulation, and IoT for production planning and scheduling. To focus the literature on the main contribution of the project, this state-of-the-art gives interested readers (within or outside of ASSISTANT's consortium) a clear view of recent development regarding the use of Industry 4.0 technology to enhance optimization approaches for production planning and scheduling. In addition, we provide the results of an in-depth analysis of the shop floor of two use cases of ASSISTANT (Atlas Copco and Siemens Energy). The document also reports the results of interviews conducted with ASSISTANT's use case providers to identify the needs of end-users.

Based on this analysis, the document lists the requirements for the intelligent digital twin for production planning and scheduling. From these requirements, we draft the architecture of our intelligent digital twins. The deliverable describes each module, the communications between the tools, the data flow between the modules, the programming language, the required libraries, and the research work required to build our vision.

1.3 Interaction with other deliverable

Deliverable D4.1 is mainly an input for the technical development in WP4. More precisely, it provides the specification, requirements, and the research agenda for the tools D4.2, D.4.3, D4.4. Deliverable D4.1 is based on the input from D7.1 that describes the manufacturing situation in planning and scheduling for the use cases in ASSISTANT. In addition, D4.1 was developed in synergy with D2.1, D3.1, D5.1, and D6.1. D2.1 describes the ethical and human-centric architecture of ASSISTANT. On the one hand, the interaction with T2.1 helped us to question ethics within tools from WP4. On the other hand, interactions with T2.2 the modules developed in WP4 fit within the global architecture of ASSISTANT. Finally, we synchronized the deliverable that provides the requirement and architecture for the other twins and the data fabric technical of ASSISTANT (D3.1, D5.1, D6.1). In particular, we defined a common methodology for the requirements engineering, and we synchronized the questionnaires and interviews with the end-users. These two documents are available in the appendix.

1.4 Structure of the deliverable report

Section 2 provides the state-of-the-art on production planning and scheduling in the industry 4.0 era. Section 3 positions WP4 within the ASSISTANT project, and it explains how the tools developed in WP4 will interact with the tools developed in other work packages. Section 4 provides the results of interviews conducted with end-users, a description of the two use cases that for WP4, and the requirements of our intelligent digital twins for production planning and for scheduling. This use case description complements deliverable D7.1 with a focus on production planning and scheduling. Section 5 describes the tools that will be developed in WP4. For each tool, we describe the research objectives, the input/output data, the communication with other modules tools, and the technology we planned to use for development. A conclusion ends the deliverable.

2 State-of-the-art on production planning and scheduling in the big data era

This section starts with a presentation of the classical function of production planning, before introducing the main concept of the intelligent digital twin for production planning. We then provide a state-of-the-art to on the main elements involved in the intelligent digital twin for production planning: The use of Internet of Things (IoT), big data analytic, Digital twins, simulation-optimization, and stochastic and robust optimization in production planning.

2.1 Definition, structure, and research overview for the production planning

Production planning and scheduling systems help companies match manufacturing performance with customer demands [Bonney, 2000]. These functions determine the global quantities to be produced (production plan) to satisfy the commercial plan and to meet the profitability, productivity, and delivery time objectives [Lolli et al., 2019]. Scholars often use hierarchical frameworks to describe the process of production planning and scheduling at different levels and planning horizons [Oluyisola et al., 2020]. Although the details for such framework differ in different studies, the core content remains the same, with a division between the long-term, medium-term, and short-term [Bonney, 2000, Oluyisola et al., 2020, Garetti and Taisch, 1999, Jacobs et al., 2011].

Figure 1 depicts such production planning and scheduling frameworks. The decision process includes multiple sub-processes (production planning, capacity planning, rough-cut capacity planning, ...). This decomposition was defined before the democratization of computers, and it allowed humans to plan by hand. The first software for production planning and scheduling (like enterprise resource planning (ERP) systems) followed this historical decomposition, and they provide a set of functionality, where each functionality corresponds to one of these sub-processes. As this decomposition is sub-optimal and inconvenient, the literature suggests integrating these decisions with integrated processes (e.g., sales and operation planning), and new software followed (e.g., MRPII fosters the integration of procurement, production, and capacity planning). With the increase of computation power and the development of optimization approaches, decision support tools for production planning tend to integrate all the decisions and data at a given planning level.

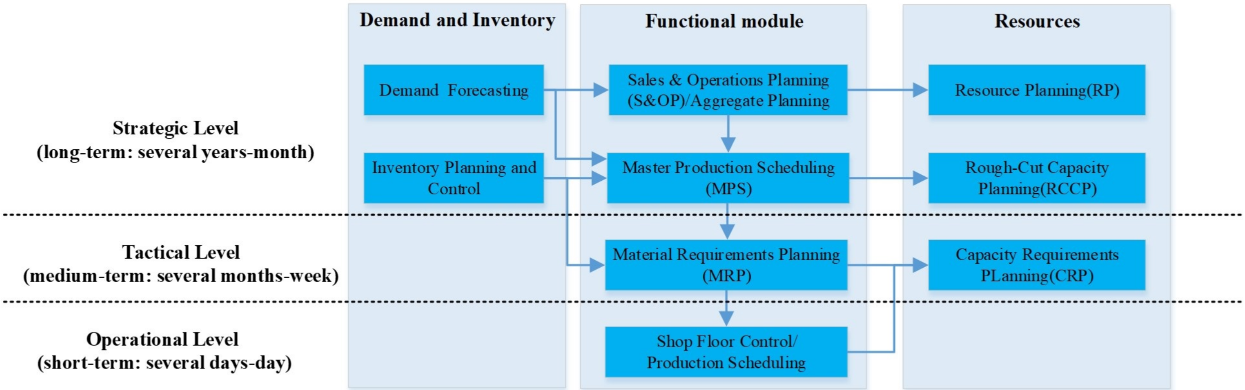


Figure 1: The framework of the Production planning and scheduling system.

Looking at the different time horizon in Figure 1, the strategic level adopts a long-term, aggregated view of manufacturing operations [Oluyisola et al., 2020]. The process begins with sales and operations planning (S& OP) or aggregate planning, and also considers the master production scheduling (MPS). The tactical level considers medium-term planning, which is called materials resource planning (MRP). The operational level concerns day-by-day, shift-by-shift detailed scheduling, which is for the short-term. We explain below the three functional levels of production planning in detail.

Aggregate production planning balances the overall demand with the available supply resources (production, distribution, procurement, and finance). This process was suggested recently to unify plans traditionally produced independently by different departments (production, distribution, procurement, and sales) [Pereira et al., 2020]. S&OP is performed monthly, at an aggregated level (based on product family), and for a planning horizon of up to a few years (since the planning horizon must include the resource (machine, workers) lead time) [Noroozi and Wikner, 2017]. The input of S&OP includes demand data (volumes per product family per planning period) as well as some metadata (such as forecast uncertainty) from demand management (DM), and future available aggregate capacity from resource planning (RP) [Oluyisola et al., 2020].

Master production scheduling (MPS) generates the production target for each end-item by period (typically monthly), whereas S&OP considers product families. In recent production planning systems, MPS integrates Rought-Cut Capacity Planning (RCCP) [Rossi et al., 2017], where planners check that the capacity of critical resources (bottleneck, labor, critical material) is sufficient to meet the production target. If this is not the case, the planners may increase capacity through overtime, temporary workers, subcontracting, or they may reduce the production target.

Materials requirements planning (MRP) combines the MPS records with the bill of materials data and inventory data to calculate the requirements of the components and parts. Using the results of MPS as the input, MRP makes recommendations on the weekly release replenishment orders for materials for a planning horizon of a few months. Based on the production capabilities and lead times which dictate the capacity requirements planning (CRP) process, it is possible to release detailed material and capacity plans with a shorter time horizon (typically weekly) [Oluyisola et al., 2020]. These plans are revised frequently, and the outputs of this stage are production plans and replenishment orders for materials, which in turn are the inputs for the operational stage [Dolgui and Prodhon, 2007].

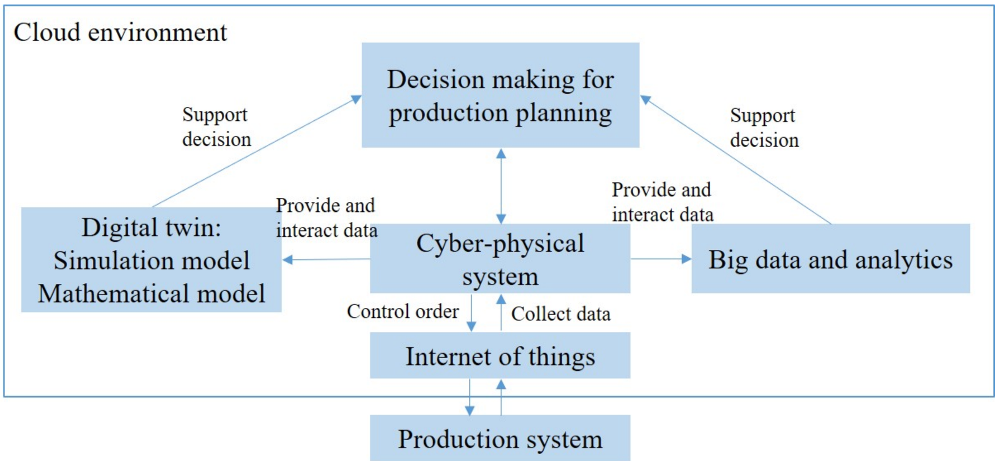


Figure 2: The overview of the production planning in Industry 4.0.

In recent years, the demand for customized products leads to various uncertainties in the supply chain, like delays in deliveries and unpredictable demands. The current supply chain is characterized by high complexity, high flexibility, mass customization, dynamic conditions, and volatile markets [Bonney, 2000]. In this context, companies must upgrade the production planning and scheduling tools to respond to the dynamic and diversified market changes. Industry 4.0 technologies can help to improve the production planning process. This includes the digital twin (DT), cyber-physical systems (CPS), internet of things (IoT), big data (BDA), analytics/artificial intelligence (AI), and cloud manufacturing (CMg) [Ivanov and Dolgui, 2020, Ivanov et al., 2020]. Figure 2 presents the relationship between production planning and these frontier technologies. Typically, IoT devices collect data from the production system to create a cyber-physical system. This data can be used in predictive analytic and prescriptive analytic to support production planning decisions. The rest of this section presents the current state of the art on the use of industry 4.0 for production planning and their impact on classical prescriptive analytics tools for planning such as simulation and optimization.

2.2 Internet of Things

IoT originated from radio frequency identification devices (RFIDs) proposed by MIT Auto-ID Labs in 1999 [Ashton et al., 2009]. IoT is the crucial basis for realizing cloud manufacturing, digital twins, and big data analysis [Hwang et al., 2016]. The International Telecommunications Union (ITU) defines IoT as intelligent connectivity for anything at anytime and anywhere [Atzori et al., 2010].

The core function of IoT in production is to acquire real-time data from the shop floor and its environment. With IoT technologies, a product can be equipped with a uniquely identifiable code, and it can be monitored and tracked by using sensors and wireless sensor networks [Fang et al., 2016]. The key technologies of IoT are RFID and wireless communication technologies. RFIDs enable tracking and distinguishing every single product. Wireless communication technologies embedded in intelligent devices enable real-time access to data on the status of products. Finally, IoT collects various data (e.g., the information of sound, light, heat, electricity, mechanics, chemistry, biology, and location) by global position system, infrared sensors, laser scanner, gas sensors, and other devices [Tao et al.].

IoT is exploited industrially by supply chains at various levels and stages of manufacturing, such as

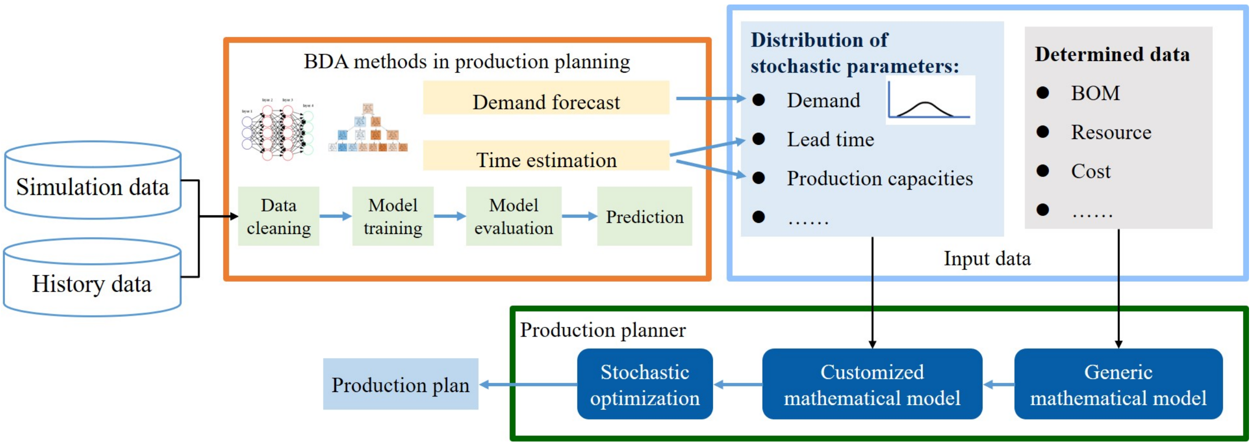


Figure 3: BDA methods for the production planning.

logistics and inventory management, assembly lines, and after-sales services [Fang et al., 2016]. IoT increases the accuracy and the flexibility of production planning by providing data from physical systems [Bueno et al., 2020, Rauch et al., 2018]. For instance, [Tao et al., 2017, Zuo et al., 2018] shows that RFID reduces the uncertainty of inventory shrinkage due to damage and thieves. Typically, the data collected by IoT devices that can help production planning include the demand of customers, the inventory level of materials, the capacity of the workshop, and the status of suppliers. With the accurate collection of the data in real-time, IoT helps production planning become more automatic and intelligent. As a result, production planning can respond quickly to various events such as machine breakdowns, urgent incoming customer orders, late material delivery.

2.3 Big Data and Analytics applied in production planning

With the development of IoT, along with the growing number of software systems in the factory, a massive amount of data is collected, and it can support decision-making for production planning [Bonney, 2000, Sun et al., 2019]. Based on the massive data collected by IoT, BDA/AI tools predict the values of the input parameters required to plan the production such as demand, production yield, supply/product lead times, process duration, production capacity [Lolli et al., 2019, Gonzalez-Vidal et al., 2019]. The processes to implement BDA/AI tools in production planning include data collection and cleaning, predictive models, model training, validation, and testing [Cadavid et al., 2020]. While BDA/AI tools and the amount of data improve the accuracy of the forecast, a forecast will never be correct. Nevertheless, BDA/AI tools can compute the variability of the parameters to account for the uncertainty. Accounting for uncertainty leads to plans that are better implementable in practice. Figure 3 shows the position of BDAs to feed parameters values and their probability distribution to optimization models. Despite the growing interest in BDA/AI, the exploitation of big data in production planning is still immature compared to other fields like IT, finance, and E-commerce [Lamba and Singh, 2017]. The application of big data and analytics requires a combination of understanding and knowledge about the domain and the right BDA algorithms. Therefore, companies have collected massive data, but they cannot currently get the best value of this data.

The choice of the data source is an important decision when training a machine learning (ML) model. There are six main data sources relevant in data-driven smart manufacturing [Sharp et al., 2018, Tao et al., 2018b, Lu, 2014] :

Table 1: Big Data Analytics based time estimation.

Paper	Application	Parameter	BDA-method	If consider planning model?	If compare with traditional method?
Cadavid et al. [2020]	Food industries	The proportion of production losses (Yield)	Linear model with stepwise selection, regression tree, bagged tree, random forest, gradient boosting, ridge regression, lasso regression, elastic net, and spline regression	No	No
Meidan et al. [2011]	Semiconductor manufacturing	Cycle time	Selective naive Bayesian classifier (SNBC)	No	No
Wang et al. [2018]	Semiconductor wafer fabrication systems (SWFS)	Cycle time	Density peak based radial basis function network (DP-RBFN)	No	No
Mori and Mahalec [2015]	Eyeglasses (a flowshop manufacturing environment)	Lead time	Hybrid Bayesian network	No	No
Gyulai et al. [2018]	Steel production	Production time	Linear regression, regression tree, random forests, support-vector regression	No	Yes
Lingitz et al. [2018]	Semiconductor manufacturer	Lead time	Random forest	No	No
Öztürk et al. [2006]	Hypothetical manufacturing environment (Simulation)	Lead time	Regression tree	No	Yes
Alenezi et al. [2008]	Multi-resource, multi-product systems	Order flowtimes	Support vector regression	No	Yes
Schuh et al. [2019]	Demonstration Factory Aachen	Transition time	A methodology for databased identifying influencing factors in order specific	No	No

1. *Management data*: Historical data coming from the company's information systems such as the ERP systems, MES, Customer Relationship Management system (CRM), etc.
2. *Equipment data*: Data coming from IoT technologies.
3. *User data*: Consumer information collected from e-commerce platforms, social media, etc. This type of data also encompasses feedback given by workers or experts that will be used to train the machine learning model.
4. *Product data*: This includes data originating from products or services either during the production process or during their use by the final consumer.
5. *Public data*: Data available in public databases from universities, governments, or from researchers.

We review below the literature on BDA tools for demand forecasting, before surveying the works on time estimation. Demand forecasting is crucial for manufacturing companies since it provides a basis for production planning. However, demand forecasting is difficult because customer demands often fluctuate due to economic trends, market competition, etc [Kück and Freitag, 2021]. Compared to traditional methods, machine learning methods, such as artificial neural networks (ANN) [Kourentzes, 2013, Kourentzes et al., 2014], support-vector machines (SVM) [Lu, 2014, Villegas et al., 2018], bayesian networks (BNs), random forests, have shown promising results in current studies, and they have surpassed in accuracy and performance the classic methods. However, the application of these forecasting methods in production planning is still not very wide. The planner's experience is still the main source for demand forecasting and production planning [Lorente-Leyva and Alemany, 2020].BDA-based time estimation is promising to adjust different time-related parameters to current working conditions.

The time estimation includes the prediction of lead time, cycle time, production time, and even the yield (when it is related to the time). Table 1 summarizes the literature on BDA-based methods to predict time-related parameters in production planning, and it gives the application, the predicted parameter, the method, whether or not the study consider simultaneously planning and prediction, and if the paper compares their approach with classical methods. Only a few works consider lead time prediction in the research community [Cadavid et al., 2020]. Lingitz et al. [2018] compare the performance of different ML algorithms for the prediction of lead times. However, the authors do not consider high variance processes. Meidan et al. [2011] evaluate different ML algorithms, but they solely consider waiting times. Alenezi et al. [2008] show that support vector machines perform better than neural networks to predict order flow time. However merely simulated data and no real shop floor information was used. Finally, Schuh et al. [2019] give a framework as well as a study on real industry data on how ML algorithms can be used for the prediction of the transition time.

IoT technologies motivate the BDA applications with equipment and product data [Correa et al., 2020, Hajjaji et al., 2021], but accessing IoT data in the planning system remains a challenge. This issue can be addressed by creating digital twins, that collect IoT data scattered in various systems, and automatically clean and integrate the data. While various studies provided tools and methods to create digital twins [Tao et al., 2018a, Zheng et al., 2019, Lu et al., 2020], this still represents a research challenge. Companies need to build general domain models to integrate interactive platforms, as well as the data connection between the physical and virtual systems.

2.4 Digital twin applied in production planning

The digital twin and the cyber-physical systems can provide decision-making support, dynamic production planning, and real-time visualization by building the virtual duplicate for the physical system [Shao and Helu, 2020]. As a result, digital twins are powerful tools to support optimization, prediction, re-planning, reporting and visibility within production planning. One challenge for production planning tools in the context of CPS is to enhance its adaptability, automation, and efficiency to deal with large-scale problems and more complex systems. More specifically, this section analyzes research of the digital twin applied in production planning in the context of Industry 4.0.

Besides mathematical models, simulation models are one of the most used quantitative approaches for modeling and decision-making in production systems. In the Industry 4.0 context, new paradigms arise to collect and store large amounts of data in real-time and throughout productive and logistical operations, and they enabe the development of the digital twins concept and related approaches [Agostino et al., 2020]. Shafto et al. [2010] gave one of the first public definitions of a digital twin in 2010. While the essence of digital twins is simulation models, a DT is very different from the traditional simulation model. The DT is multiphysics, multiscale, probabilistic, and ultra-fidel. A DT reflects the state of a corresponding twin in a timely manner based on the historical data, real-time sensor data, and physical model. With the development of industry 4.0, the concept of DT has been expanded. Nowadays, DT includes not only the simulation model but also the mathematical model and the data model. Furthermore, people pay attention to the entire system of digital twins.

When it comes to DT, another important concept to mention is Cyber-Physical System (CPS). The cyber-physical system is a set of embedded systems which communicate and interact with each other in a communication network the data and information about each asset come from the CPS on the shop floor [Geisberger and Broy, 2012]. CPS is the main source of data and

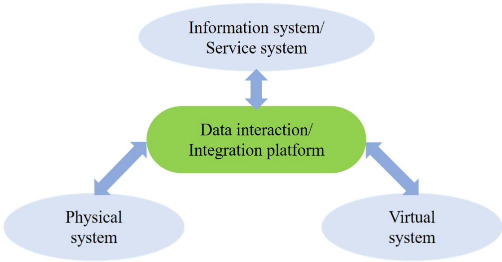


Figure 4: The conceptual model for the digital twin.

information for the digital twin. In the context of production technologies, the CPS is called a cyber-physical production system (CPPS) [Weyrich et al., 2017]. The information and data collected from CPPS can be used to create the digital twin for production planning. Generally, the CPPS collects hardware, software, and real-time information [Biesinger et al., 2019]. DTs can support decision-making in every stage or each level of production planning systems. For aggregate production planning, DTs can achieve multi-level data sharing, traceable data flows, and integration with demand forecasting, inventory control, and MES/ERP systems [Yu et al., 2018]. For MPS, DTs can support capabilities concerning real-time and dynamic production planning systems, with distributed and collaborative decision-making through MES, MPS/ERP, and CPS integration [Rossit et al., 2019]. In MRP, the DT model and CPS can help to achieve automatic optimization, prediction, and re-planning for MRP [Lin et al., 2018], and extending MRP with real-time calculations, early reports, traceability, and visibility[Shao and Helu, 2017].

There are many frameworks for the DT and CPS, but they share the same core elements shown in Figure 4. Through the construction of the platform, the physical system, virtual system, and information system can be interconnected with each other. In the initial stage of the research about DTs, researchers mainly proposed the digital twin framework for entire supply chain management issues. With the deepening of research, researchers began to focus on frameworks dedicated to production planning and scheduling. However, as scheduling is more sensitive to real-time data, most works concern scheduling problems, and few studies discuss mid-term and long-term production planning. Furthermore, with the proposed digital twin framework, there are few quantitative analyses and case applications for production planning. Table 2 presents papers that propose the DT frameworks, and it provides the author's viewpoint, and the core methods, and the considered application.

2.4.1 State of the art on data-driven simulation for production planning

The essence of digital twins is simulation models. In production planning, discrete even simulation models entities and the flow of events through time. Manufacturing systems are very different from a company to the next, and it is not possible to create a generic simulation model for manufacturing. The construction of simulation models for large-scale production systems requires knowledge from business experts, and it is time-consuming. To reduce the time of building simulation models, researchers have proposed a data-driven method to automatically build simulation models[Liu et al., 2019, Zhang et al., 2019b]. These tools can reduce the total modelling time from several months to several weeks [Wang et al., 2021, Wy et al., 2011] and they reduce errors in the modelling process.

Figure 5 shows how the data-driven modeling and simulation technology works. First, we extract original information from the data information systems and standardize these data. Second, we further classify and associate the data to build a structured data model. Third, based on

Table 2: Literature review about DT framworks.

Paper	Application	Viewpoint	Core methods/focus	Case study
Tao et al. [2018a, 2019]	Product design	Product	Big data, Cyber and physical convergence	The power transformer and bicycle, no data
Ivanov et al. [2019], Ivanov and Dolgui [2020]	Digital supply chain twins	Supply chain	Additive Manufacturing, blockchain; Big Data Analytics	No
Qi et al. [2019]	Digital supply chain twins	Supply chain	Five-dimension model, enabling technologies, enabling tools	No
Tao et al. [2018b]	Smart Manufacturing	Manufacturing system	Lifecycle of manufacturing data, framework	Silicon wafer production line, figures of implementation interface
Lu et al. [2020]	Smart Manufacturing	Manufacturing system	Review,connotation, reference model, applications and research issues	No
Rossit et al. [2019]	Smart Manufacturing	PPC	Review,Cyber-physical system	No
Agostino et al. [2020]	Smart job shop	PPC	Cyber-physical system	Scheduling in a job shop of a Brazilian supplier for the automotive industry
Zhang et al. [2019b]	Smart shop-floor	Workshop	Cyber-physical system	Scheduling of the blisk machining, data
Ding et al. [2019]	Smart shop-floor	Workshop	Cyber-physical system, operations control	Interface of operations control, no data

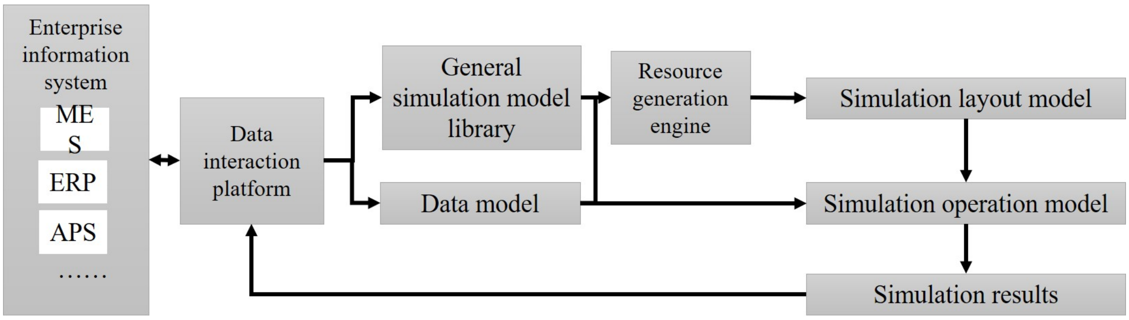


Figure 5: The process of data-driven automatic modeling and simulation method.

original simulation objects, we personalize the internal logic and attributes of objects to build the general simulation model library, which meets the needs of the particular industry. Fourth, with the help of a resource generation engine, we can use the objects in the general model library to generate the model layout quickly and automatically based on the layout data. Lastly, driven by real-time data, we can obtain a specific simulation operation model, and run it to get simulation results.

In the simulation operation phase, when production demands or production layout change, the traditional offline simulation takes several hours to update the data and adjust the model manually. On the contrary data-driven simulation update the data and adjust the model automatically and quickly. Therefore, with the data-driven automatic modeling and simulation technology, production planning respond to uncertain events quickly. The data-driven modelling and simulation technology is also one of the important technologies in the digital twin [Zhang et al., 2019a, Wang et al., 2021]. Within the scope of our knowledge, there is little literature on the use of data-driven automatic modeling and simulation technology in production planning.

2.5 Optimization for the production planning in the industry 4.0 era

This section discusses prescriptive analytics methods, that combine machine learning and optimization to prescribes the best course of actions to optimize the plan.

2.5.1 Optimization model for production planning and scheduling

Optimization models for production planning problems involve inventory management and lot-sizing problems. The lot-sizing problem aims at determining production and procurement quantities and their timing [Yano and Lee, 1995]. Research on production lot sizing dates to the beginning of the 20th century, and since then, several problems have been addressed and a large number of modelling approaches and algorithms have been proposed [Buschkühl et al., 2010]. With the deepening of research, the focus of research on the lot-sizing problem gradually changed [Louly and Dolgui, 2013, Hnaien et al., 2016, Schemeleva et al., 2018, Tavaghof-Gigloo and Minner, 2020] from single-product single-period single-machine systems to complex multi-product multi-period multi-machine systems [Cunha et al., 2018]. One of the most generic versions of lot-sizing problem for production planning is the multi-echelon multi-item capacitated lot-sizing problem (MMCLP). The MMCLP is to recommend when to produce as well as the sizes of the production lots to minimize the expected total cost (including inventory holding costs, fixed setup costs, unit production costs, extra capacity cost). These decisions are made based on the demand, the bill of material, the production capacity, and the lead time.

Mathematical optimization is the most appropriate tool for production planning. The lot-sizing models have attracted a lot of work from the operation research community. Researchers propose several reformulations, cuts, and solution algorithms such as Lagrangian Relaxation, cutting planes, etc. Tempelmeier et al. [Tempelmeier and Helber, 1994, Tempelmeier, 2006] and Helber et al.[Helber, 1995, Helber and Sahling, 2010] have done a series of studies about the decomposition approaches and Lagrangian Relaxation-based heuristic algorithms for the multi-level capacitated lot-sizing problem. This solution offers opportunities for the improvement of large problem instances.

An important part of operation management is optimization of production schedules. There are a large number of mathematical models that are used for scheduling optimization. Pinedo [Pinedo, 2018] gives an overview of most common scheduling models, including flow-shop and job-shop models. To solve scheduling optimization problems a variety of approaches is used, including constraint optimization and a selection of metahuristics. The overview of constraint optimization approaches for solving constraint problems is given in [Baptiste et al., 2001]. Bewoor et al. [2018] used particle swarm optimization to optimize production schedule of the foundry. In many cases, the hybrid approaches, that combine two or more algorithms, are used. Giacomello et al. [2013] and Luna et al. [2019] used hybrid approaches to solve pump scheduling optimization. Giacomello et al used the combination of linear programming with greedy algorithm and Luna et al used a genetic algorithm with knowledge-based mechanisms. Oliveira et al. [2011] used a combination of genetic algorithm with mathematical programming to optimize scheduling problem for the process industry. Babukartik and Dhavachelvan [2012] used ant colony optimization and cuckoo search the job scheduling problem. The general overview of hybrid approaches used for the scheduling problems with additional resource constraints can be found in [Pellerin et al., 2020]. It should be noted, that majority of scheduling optimization papers consider problems with only one optimization objective. [Mokhtari and Hasani, 2017] and [Soto et al., 2020] are examples of works that examine multi-objective scheduling problems.

Mathematical models are usually constructed by researches themselves in attempts to describe a selected problem, including planning and scheduling problems, as accurate as possible. The model acquisition is using available data to construct constraint models automatically, with minimal input from a researcher. This allows to significantly speed up the process of creating adequate models for the industry.

The area of learning symbolic formulae and models can be divided in the following categories:

- Formula discovery which covers equation and conjecture discovery [Brence et al., 2021, Larson and Cleemput, 2016, Aouchiche et al., 2005].
- Acquiring spreadsheet models from Excels tables or relational tables [Kolb et al., 2017, Paramonov et al., 2017].
- Model acquisition of combinatorial problems in the form of a constraint model or a MIP model [Beldiceanu and Simonis, 2016, Pawlak and Krawiec, 2017, Kumar et al., 2019a].

Additionally, the model seeker can generate models for some pure job-shop problems. Only recent initial work deals with acquiring scheduling models from a log of a set of events [Senderovich et al., 2019].

2.5.2 Simulation-optimization approaches

Simulation methods mainly include discrete event simulation (DES), agent-based simulation (ABS), and system dynamic (SD) simulation. These models are commonly used for facility resource planning, capacity planning, and job planning. Simulation can provide a detailed representation of the production process, and it can simulate the execution of a policy. Most simulation-optimization approaches use optimization methods (e.g., local search, gradient descent, genetic algorithms, …) to optimize the input parameter of the simulation. In this context, the simulation is embedded in the optimization approach to evaluate the costs associated with the input parameters. For instance, Lim et al. [2017] simulate the use of a dynamic inventory control policy under various sources of uncertainties, and optimize the parameters of the policy with a local search. Similarly, Liu et al. [2011] use a genetic algorithm that evaluates the expected cost of a production plan through a simulation. A major drawback of such approaches is the time-consuming solution evaluation by simulation, especially when multiple replicates are required to approximate the expected cost in an uncertain environment, or when the simulation is very detailed. An approach to circumvent this issue is to build surrogate models [e.g., Osorio and Bierlaire, 2013] to approximate the expected cost evaluated with the simulation. These surrogate models are leaned with machine learning from past simulation, and they are used to reduce the number of solutions evaluated through simulation.

The state-of-the-art optimization approaches for lot-sizing models commonly encountered in production planning relies on mathematical model solved with commercial solvers. This approach was also used in combination with simulation. In a simple framework, the simulation is only used to complete the decisions made by the analytical optimization model. For instance, Lim et al. [2006] use an optimization approach to set the capacity in the factory and a simulation model to compute the production plan. A more advanced setting is the recursive optimization-simulation approach, where the mathematical model is iteratively improved with the result of the simulation. For instance, Jung et al. [2004] solves a deterministic lot-sizing problem and iteratively adjusts the safety stock after evaluation in simulation that accounts for uncertain demand. This iterative approach was also recently applied for production planning in collaborative assembly lines [Vieira et al., 2021], and for production planning in a wafer fabrication production plant [Kim and Lee, 2016].

For more information on simulation-optimization approaches, the interested reader is referred to Figueira and Almada-Lobo [2014]. Overall, there is growing attention toward simulation optimization approaches, but their applications in production planning remain scarce. We believe that such approaches must be investigated, since a detailed simulation complement the optimization approaches, and ensure that the computed production plan is implementable on the shop floor. Stochastic optimization can be seen as an integrated simulation and optimization since it directly incorporates scenarios to describe the possible realization of uncertain parameters within the optimization model.

2.5.3 Optimization under uncertainty

While the first studies on lot-sizing considered that all parameters are known, in practice, none of the production planning parameters can be forecasted perfectly. Uncertainty may be defined as the difference between the amount of information required to perform a task and the amount of information already possessed [Galbraith, 1973]. Over the years many researchers attempted to formalize and model uncertainty in manufacturing systems [Sethi et al., 2002, Yano and Lee, 1995]. The production planning literature provides various approaches and models to cope with different forms of uncertainty. The main three uncertain parameters in production planning are demand, lead time, and capacity.

1. **Demand uncertainty** is critical for production planning, especially for manufacturers with long production lead times [Aouam et al., 2018]. Demand uncertainty has various forms, such as order size and due date.
2. **The lead time** refers to the number of periods between the placement of an order and its arrival. In production planning, we may distinguish between delivery lead time and production lead time. The first refers to the time required by suppliers to deliver components, whereas the second refers to the time between the release of an order to the shop floor and its shipping date. Delivery lead time uncertainty is common in practice and it is due to issues at the supplier production level or transport [Hnaien et al., 2016]. The reason production lead times are uncertain involves several factors, such as inaccurate capacity constraints modeling when building the production plan, machine breakdowns, stochastic variations on operation processing times, etc [Aghezzaf et al., 2010]. Some studies suggest modeling uncertain lead time with discrete support probability distribution built based on statistical data .
3. **Production capacity** uncertainty refers to issues to ensure the shop floor can satisfy the required production load. There may be uncertainty about the available resource capacity due to machine breakdown or employee absenteeism, and uncertainty in the capacity consumption for an operation due to variable process duration, or product quality if the shop floor reworks or redoes bad quality parts. Another major source of problems is that the optimization models for production planning only approximates the capacity roughly to produce a feasible plan. In practice, even when a good scheduling tool is used, the resources may have idle times. In addition, in flexible production plants, it is difficult to estimate which resource will perform each task before doing the production schedule. While capacity uncertainty leads to infeasible plans, very few works consider production planning under capacity uncertainty, when compared with the cases of demand and lead time uncertainty.
4. **Yield uncertainty** occurs when bad quality parts cannot be re-work or replaced by a new one. This situation occurs for operation with long processing time such as aluminum casting, or in multi-echelon systems, where producing an additional part is impossible when the components are not available. Yield uncertainty is also common in the disassembly

of end-of-life items since the quality of components is only observed once the item is disassembled.

The classical approaches compute the lot sizes under the assumption that all parameters are deterministic, whereas safety stock, safety lead times, and safety capacities are computed separately to hedge against the uncertainty. With the improvement of computation power and new development in optimization approach, it is nowadays possible to integrate the uncertainty directly in the optimization problem with stochastic optimization (SO) approach [Spall, 2005]. That is, random variables appear in the formulation of the optimization problem itself, which involves random objective functions or random constraints. Consequently, the research recently moved from the initial deterministic to non-deterministic lot-sizing models [Tavaghof-Gigloo and Minner, 2020]. Many studies consider a single uncertainty parameter [Yano and Lee, 1995, Zikopoulos, 2017, Kroer et al., 2018, Afsar et al., 2020], but more scholars have paid attention to the consideration of multiple uncertain parameters in recent years. For instance, demand and lead time are sometimes considered together[Tang et al., 2019, Köchel and Thiem, 2011, Song and Dinwoodie, 2008]. Finally, a large variety of methods were proposed to solve lot-sizing problems, such as fuzzy logic, scenario-based stochastic optimization, robust optimization, and game theory [Su, 2017, Cunha et al., 2018, Carvalho et al., 2018, Simon et al., 2021, Zarei et al., 2021].

2.6 Limitation and future direction

Based on this literature review, we identified several gaps in the literature regarding, and they represent the main contribution to research of the WP4 of ASSISTANT.

Future research direction regarding data collection and integration for production planning and scheduling:

1. The integration of information from different systems remains a challenge because data from heterogeneous sources must be reconciled. Other difficulties include the use of different standards in information systems and data interaction. Solutions to overcome this integration issue include service-oriented architectures [Niknejad et al., 2020] and blockchain [Korpela et al., 2017] for the flexibility and security of data transmission, and ontologies Kumar et al. [2019b] to map different data models. Nevertheless, future work is required to facilitate the integration of the information collected from IoT devices, software, and between information systems from different shop floors. This requires the development of data format standards, protocols for system interaction, and data management procedures that ensure safety and reliability. There is also a need to develop tools to automatically clean the data, and to detect and fix incoherent information (e.g., the level of inventory in the ERP and computed from RFID).
2. As IoT collects a large amount of data and interconnects the virtual and actual systems, it leads to large and complex information systems with heavy memory load and slow calculation. Reducing the complexity of the resulting system is an important research direction.

Future research direction in predictive analytics for production planning and scheduling.

1. Existing research mainly focuses on demand forecasting, and they only seek to forecast a single parameter. Few works consider machine learning approaches to predict the joint distribution of multiple parameters, whereas production planning parameters may be related (e.g., the demand and the cycle time).
2. More research is required to provide the best way to apply generic machine learning tools

in the production planning context. The research of BDA in the Manufacturing system is still at the preliminary stage. Some researchers study how to use BDA in the supply chain. But they only test different BDA methods, and they do not provide a breakthrough in forecasting models.

3. Research on the optimization of production planning model based on BDA. Few papers further consider how to use the forecast results to optimize the production planning model, and what kind of quantitative impact will it have. In the research about production planning, no one compares the difference between BDA based production planning methods and traditional production planning methods

Future research direction in prescriptive analytics for production planning.

1. Solving the complex lot-sizing problem under uncertainty is hard, especially in the dynamic decision framework, where the production setups are updated as the information unfolds. The existing works are limited to small-scale instances in a simple environment [Thevenin et al., b]. To solve large instances, with multi-echelon BOM a large planning horizon, heuristic algorithms must be provided. For instance, Thevenin et al. [a] showed that the two-stage approximation provides a good heuristic to the static-dynamic decision framework when the demand is uncertain. However, more research is needed to solve a large time horizon, and the use of a fix-and-optimize approach possible research direction. In addition, methods must be developed to handle the dynamic decision framework.
2. While most approach assume the probability is known, this will never be true in practice, and the distribution can only be estimated. Distributionally robust optimization is an interesting class of approach that optimize for the expected cost of the worst case distribution [Zhang et al., 2016], and its application to production planning must be further explored.
3. The data-driven model acquisition is an interesting area of research for production planning and scheduling. Regarding scheduling, as every shop floor is different, model acquisition removes the need to create a model for each workshop or whenever the shop floor changes. Only recent initial work deals with acquiring scheduling models from a log of a set of events. The model acquisition could also be used to improve the accuracy of planning by communication with a detailed simulation.

3 Position of Work Package 4 in ASSISTANT

According to the requirement engineering procedure, this section defines the scope of WP4 and its interactions with other tools in ASSISTANT (called system context). Work package 4 aims to provide the intelligent digital twin for production planning and scheduling. Fig 6 below highlights the scope of WP4 from the ASSISTANT's concept picture [Beldiceanu et al., 2021].

The production planning intelligent digital twin aims to manage the capacity and the supply requirement. This twin suggests the production quantities per period for each item, as well as capacity adjustment (shift length, temporary worker, ...), and deliveries from suppliers. Within ASSISTANT, classical approaches for planning will be enhanced to account for uncertainties and to learn functions that represent the capacity constraints in complex shop floors thanks to simulation. The production scheduling intelligent digital twin takes as input the production requirements from production planning, and it assigns a resource to each operation and sequences the operation on the resources. The intelligent digital twin for production planning and scheduling interacts with the other components developed in ASSISTANT:

- **The Data Fabric (WP6)** offers the data sharing mechanisms among the different decision

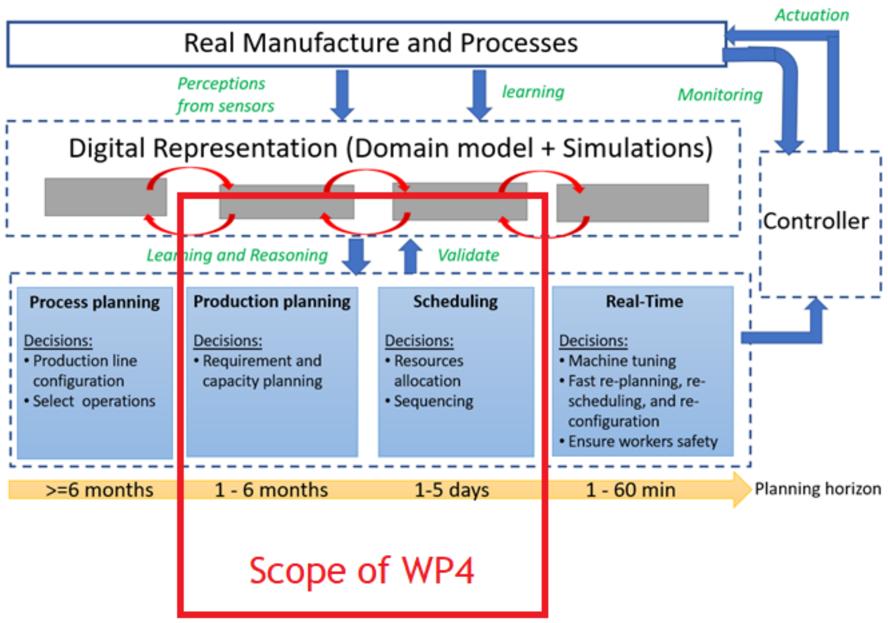


Figure 6: Scope of WP4 within the ASSISTANT concept [Beldiceanu et al., 2021]

levels (Process, Production, Schedule, Execution). It cleans, aggregates, stores and integrates into domain models all data that can be consumed by the intelligent digital twins. In addition, the data fabric provides data analytic tools useful to the digital twin for production planning and scheduling. For instance, tools to learn probability distribution from historical data.

- **The Digital Twin for Execution (DTE)** which will be implemented within WP5, is responsible for the shopfloor control level. DTE requires information from the Process Planning and the Production Scheduling levels. Operation's details, bill of processes, initial task allocation among the available resources, and production sequence are some of the required data. A feedback channel will be provided also from DTE for updating the execution progress and triggering the replanning functionalities of Process and Production digital twins. DTE deployment aspects are presented in deliverable D5.1 of ASSISTANT.
- **The Digital twin for process planning (DTPP)** will be developed in WP3. The DTPP helps engineers to design their production processes. This twin considers a single production line, and it selects the best processes, resources, and tools to process a part on the line. While the process planning intelligent digital twin assigns resources (machines/equipment) to tasks, it makes very different decisions from the production scheduling module. The intelligent digital twin for scheduling selects the machine (among the machines present in the factory) to perform each operation, but a machine may be the entire production line. In addition, an operation is not the same concept in scheduling and process planning. More precisely, an operation in the schedule refers to the processing of a lot by the production line, and scheduling does not look at the detailed task assignment inside of the production line. The processing time of an operation in scheduling is the duration required to process an entire lot on the assembly line, and it depends on the takt time of the line (which is decided in process planning).

Figure 7 shows the data flow within ASSISTANT for the data consumed by the intelligent digital twin for production planning and scheduling. The data fabric is the central data provider in ASSISTANT. We follow a service-oriented architecture, and each module of the intelligent twin for production planning and scheduling access and store the data in the data fabric via REST APIs. The data from external software (ERP, MES, CRM, HRM) is fed to the data fabric via some use

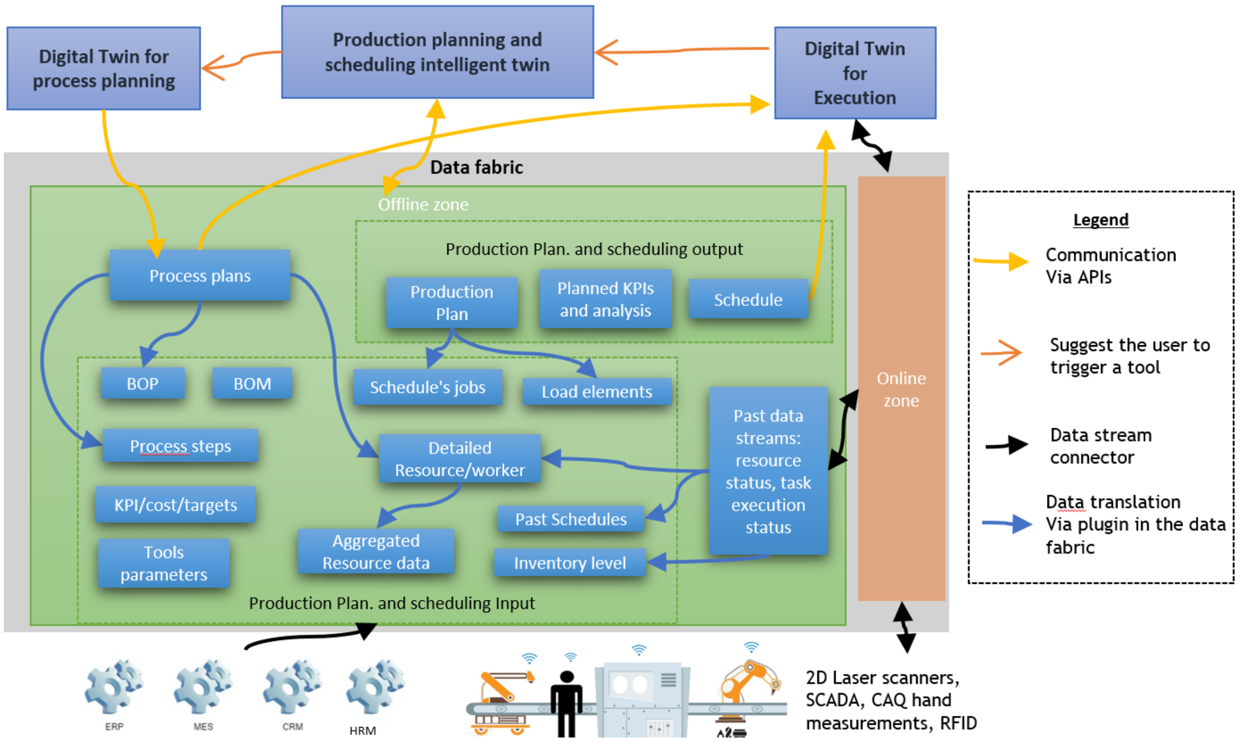


Figure 7: Data flow within ASSISTANT for the data consumed by WP4

case specific connectors developed in WP7, and it is accessible through the domain model of the data fabric. As some data require to change granularity to be consumed by a specific module, the data fabric provides plugins to translate the data in the right format. Once the data is produced, it is translated in the right format, and it is available in the data fabric. For instance, as explained earlier, the computation of the duration of a job in the scheduler requires multipling the lot size decided in the production planner by the takt time computed in the process planner. Note that the same data may come from different sources, and the data fabric provides tools for data aggregation. Following the chain of the planning, the tools developed in WP4 consume data produced by the tools developed DTPP and DTE.

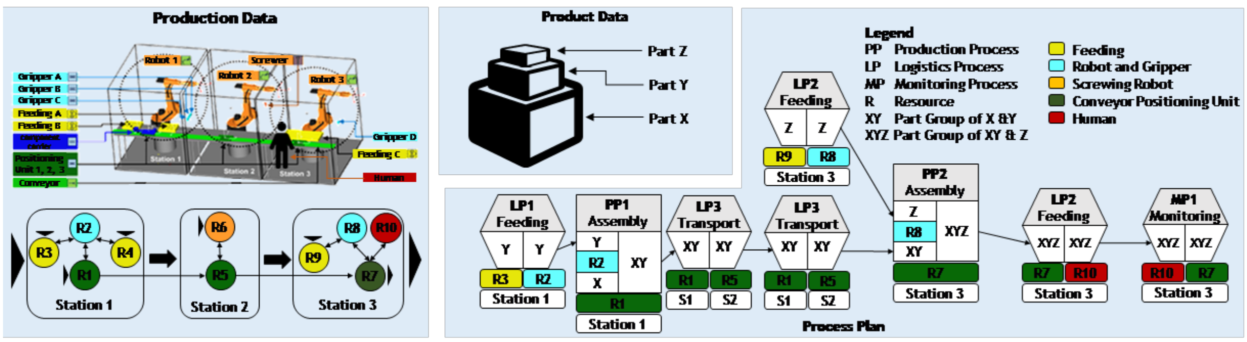


Figure 8: Exemplary Process Plan as the Main Result of Process Planning

The DTPP provides the process plans that are translated further into inputs for production planning and scheduling. They consists of all processes and their sequences required to produce and monitor a product within a given production system and selected resources. Fig. 8 provides an exemplary process plan which serves as the basis for process planning and scheduling. Within WP4, the bill of processes, the process duration of a lot, the detailed resource/tools available on the shop floor (machines, workers, or production lines, ...) required to process a lot are extracted using the process plan. The translation of the process plans towards information requested by the intelligent twin for production planning and scheduling takes place via the data fabric.

DTE monitors the execution of the operations on the manufacturing line with IoT devices. The data collected during the execution are used (after translation) within the intelligent digital twin for production planning and scheduling. This data includes actual resources available, the inventory level, and historical data required to predict future parameter values: process duration, records on machine break down and their duration, records on product defects. For instance, this data helps to improve the accuracy of the simulation or scheduling tool by providing precise process duration from the shop floor. In addition, the data recorded by WP5 can be translated into schedules that were implemented in the past, and this data is consumed by the model acquisition for scheduling. Similarly, WP4 provides through the data fabric the production schedule to DTE, and this production schedule defines the item production sequence on each resource.

Finally, DTE will suggest the user re-scheduling the production if the schedule is outdated (e.g., a machine is broken), and the production planning module wills suggest the user modifying the process plan if the production capacity cannot be respected.

4 Requirements of the intelligent digital twins for production planning and scheduling

This section lists the requirements for the intelligent digital twins for production planning and production scheduling. Following the requirement engineering procedure given in the appendix, we conducted interviews with the three use cases of ASSISTANT to elicit the requirement. In addition, we had multiple discussions to get an in-depth understanding of the shop floor of two use cases that will validate our solution: the compressor production plant of Atlas Copco, and the blade and vane production shop floor of Siemens Energy). This section provides a summary of the interview results, the description of the two production shop floors, and the requirements for the modules developed in WP4.

4.1 Interview results

The interviews with the three industrial partners were synchronized with T3.1, T5.1, and T6.1, and the question list is available in the online supplement. We provide below a summary of the discussion.

Current situation: Among the three use cases, only PSA is currently using software to optimize production planning and scheduling. At *Atlas Copco* and *Siemens Energy*, production planning is currently done manually (with Excel). The production plan is done in a daily planning meeting. Based on the relevant information (delivered components at the beginning of the day, demand, ...), the planners decide what to produce. At *Siemens Energy*, a simulation is also available, that takes the excel sheet as input, and it helps the user validates the plan and schedule. Similarly, there is no tool for scheduling in *Atlas Copco* nor *Siemens Energy*, the shop floor simply follows a FIFO rule. While FIFO is far from being optimal in a complex production environment, this rule is easy to implement on the shop floor. A more sophisticated production schedule would require equipment for the workers to follow on the shop floor, for instance, a screen in the shop to tell the user what to do next.

Problem with the current tools: *Atlas Copco* and *Siemens Energy* reported that this manual decision-making process is cumbersome because production planning is complex and must account for the availability of the parts, the amount of work in progress, the customer orders to fill today, the availability of operators and resources, ... Often, the plan is outdated after a few hours due to unforeseen events. For instance, if a component is not delivered on time the production cannot take place, or when an urgent order is released by a customer. In such a situation, the human planner does the firefighting to update the plan but alone (not in a meeting). *Atlas Copco* reported they would like to go for fully automated production planning and scheduling, whereas *Siemens Energy* made it clear that they only want a tool to suggest decisions to a planner whereas the planner makes the actual decision. *PSA* mentioned that their production planning tools are difficult to use because they are too far from their shop floor (they do not represent their shop floor precisely). Another difficulty is to transfer the production schedule to the shop floor since currently, the production planner gives the information through the cell phone. Finally, all three use cases reported issues regarding the integration between the shop floor and the software. For instance, at *Siemens Energy*, the MES and the simulation are not connected. *Atlas Copco* mentions some issues regarding the lack of integration with the data. For instance, a production cell might put all its energy to complete a sub-assembly on time, and they realize later that the next production step cannot be performed because some components required for assembly were not delivered by the suppliers.

Fears regarding the use of AI for production planning and scheduling: The use cases reported no fear from the end-user regarding the use of AI for production planning and scheduling, as long as a human is responsible for the ultimate decision (AI only suggest a production plan). On the contrary, production planning problems are too complex to solve by hand. The planners understand that an AI can investigate much more possible plans than a human, and the management is happy because the shop floor will be more efficient. Regarding the workers on the shop floor, they will not see the difference between a plan created by an AI or by a human. The only possible bad consequence to consider is that the workload of the workers may increase. In addition, *PSA* highlighted the importance to ensure the safety of the network and software.

Expectation regarding the use of AI for production planning and scheduling: The three use cases mention the importance of providing accurate production plans. The production plans must be implementable on the shop floor with the available resources (materials, machines, workers, ...). *Atlas Copco* highlighted the importance to be able to select the optimization target (e.g., costs versus meeting due dates), as the planner may emphasize different targets depending on the situation of the shop floor. Finally, *Atlas Copco* and *PSA* mentioned the importance of a tool that can help to deal with and react to unforeseen undesirable events.

Management of uncertain events: All three use cases reported issues with uncertainties on the shop floor. The first issue reported by the three use cases are problems related to machine breakdowns, because they lead to major disruptions of the operations on the shop floor, and they may have critical consequences. In addition, *PSA* and *Atlas Copco* reported issues with the deliveries of parts from suppliers. Finally, *Atlas Copco* reported issues with the quality of the produced part.

4.2 Use case description

This section introduces the two use cases provided by the industry partners of the consortium. Section 4.2.1 details the Atlas Copco use case regarding production planning and scheduling. Section 4.2.2 gives the Siemens Energy use case. Both use cases are the base to derive the requirements in section 4.3- 4.7.

4.2.1 *Atlas Copco* use case

The Atlas Copco (AC) use case focuses on the Airtec plant that produces rotors for the compressors. AC does not sell components, only full compressors, so the Airtec plant has only internal customers. AC operates with a make-to-stock strategy because the lead times are long (3-5 weeks). For a few products, the lead time is acceptable, and they are made to order. The shop floor is large (approx. 200 m $\times$ 200m) with 116 machines. The Airtec plant produces 120 different rotors, leading to 300 different compressor variants, by using different casings and other parts. The weight and size of a rotor vary a lot (300g to 300 kg). The shop floor runs 24h per day (3 shifts). AC can also run the production during the weekend to overcome difficulties or to increase capacity.

Production steps: Figure 9 represents the production steps to create the rotors.

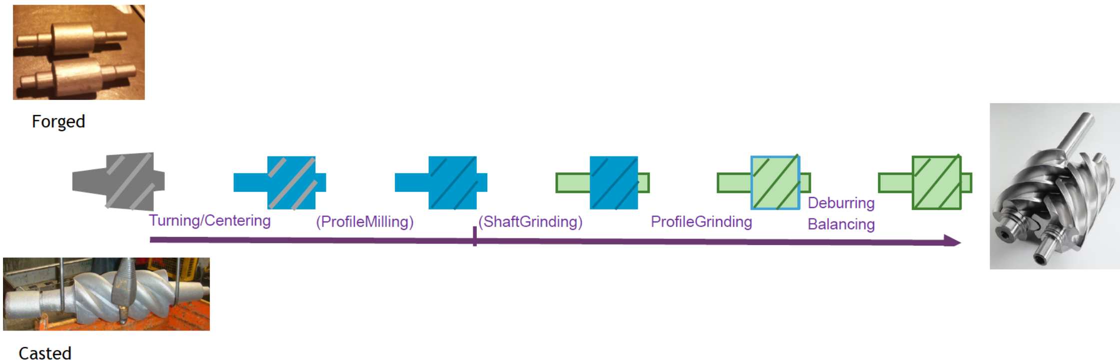


Figure 9: Atlas Copco's production steps.

The production starts from forged steel parts or cast-iron parts. Then, each step of the routing removes some material from the casted pieces bought from a supplier. The following process steps are required to produce a rotor:

- Turning removes most of the material.
- Milling shapes the rough profile
- Shaft grinding finishes the shafts
- Profile grinding finishes the complex profiles
- Finishing operations are similar to grinding. They are more accurate but a lot slower (so remove as much material as possible on the previous steps).
- Assembly with other components includes components that do not impact the quality and are therefore bought from suppliers.

The routing varies depending on the item to produce. The input material may be cast or forged iron. Cast Iron are closer to the end-item (it requires less production step, and it is cheaper to produce), but for some end-items, forged iron is not strong enough. Each operation lasts 10-20 minutes. Processing time for one part depend on the complexity and the size of the part. Every detail in the rotor takes time. The shop floor is divided into several lines, and the lines may use different technologies. The bottleneck depends on the technology.

Machines/resources: Each of the production steps is executed on different CNC machines, and in each step, there can be several parallel machines. There are operators to check the quality, to make sure there are enough raw materials and pallets, and to do the changeovers/setups. However, the loading-unloading is fully automated, and an operator can attend several machines. Besides, operators are cross-trained able to work on several machines. For the operator, AC has a skill/qualification matrix per operator. For the moment, it contains only yes/no, but they are working towards something more specific (70% qualification, experience, …). The skills are related to the quality check because the operator must know what must be changed to go back to spec. There is currently no KPI regarding the optimal use of an operator. It would be interesting to rotate, so they stay fresh in memorizing how to do a step.

Material flow: When changing the type of rotor to process, a setup (15 min-3h) must be performed on the machines. This setup time depends on the production sequence on each machine. Therefore, the production is done per batch of (50-200 items) of the same type. It takes 2 or 3 shifts per batch. There are buffers between steps, but a batch can overlap between 1 step to the next. The operator moves a pallet from one machine to the next as soon as it is finished, but the cycle times are not all equal. In some rare cases, a batch is split (in time or over several machines), and they must be recombined afterward. If AC splits a batch, the finished part goes to the warehouse and all the elements are in the same emplacement in the stock (any male rotor match with any female). Nevertheless, splitting the batch leads to a changeover and must be avoided.

Uncertainties: The main sources of uncertainties on the shop floor are demand and production capacity:

- **Yield uncertainty:** Sometimes there are bad parts, and this decreases the batch size. It is not possible to add a new part in the batch - AC would have to wait too much for the delivery of the casted material. Nevertheless, this is not a big issue since the proportion of bad parts is below 1%.
- **Demand uncertainty:** The demand is known quite long in advance. AC has a 3-month prediction of the demand, but this prediction is not very accurate (30% deviation)
- **Lead time uncertainty:** Suppliers are reliable (they deliver on time, and there is no problem with the quality of components). However, production lead time is a problem. The customer would like a 1-hour accuracy, but in reality, the lead time can vary from 1-3 days late.
- **Capacity uncertainty:** The number of machine breakdowns depends on the technology/age of the machine. Roughly there is a breakdown once a week per machine group with a 2 days downtime. If the production plan consumes 70/80% of the available capacity, the shop floor can adapt to unforeseen events. However, when the demand is high, the shop floor runs close to 100% of the capacity, and it cannot recover from a machine breakdown. AC tries to move toward predictive maintenance to avoid problems

Scheduling currently: Scheduling and production planning are done manually (with Excel). There is a daily production planning meeting. Based on the delivered components at the beginning of the day, the planners decide what to produce. Production planning is complex and must account for the availability of part/ Work in process/ Customer orders to fill today/ availability of operator/resource. The current replenishment strategy uses constant lot size (revised once a year). To hedge against uncertainties, AC uses safety stock, and safety lead time. There is also a safety process duration. The process duration includes some margin to account for possible deviation. Therefore, you can finish a batch quicker than plan. The schedule follows a FIFO rule. FIFO is easy to implement on the shop floor, it is easy for the operators to follow on the shop floor. If a different production planning approach is used, a tool must be used to inform, such as screen in the shop to look at the screen and know what to do. Often, the schedule is outdated after a few hours due to unforeseen events (for instance, the bearing did not get delivered on time, and the schedule must change asap). In such a situation, the same planner does the firefighting but alone (not in a meeting).

Computation speed: When the schedule must be changed, the strategy is to finish the current batch before doing something else. Therefore, the software can take 1 hour for computation.

Objectives: The main objective is to deliver on time. In addition, it is important to minimize throughput time, the lead time, and to reduce the work in progress (because the shop floor is full). AC aims for cost optimization and an increase in agility and flexibility.

4.2.2 *Siemens Energy* use case

The selected Siemens Energy (SE) use case comprises three workshops: coating, drilling, completion, and testing. Each of these workshops contains several production steps as shown in 10, 11, and 12. To simplify the use case, a subset consisting only of the production steps has been selected shown in yellow in the figures.

Production steps: An operation may require different tools, and these tools will be different for different blades. This includes for instance the tools to fix the blades on the machines. Except for testing, the operation in each resource group requires a single step. However, multiple routings are possible to create a blade or a vane. For instance, Figure 11 shows that the process of eroding followed by drilling can be replaced by only drilling (but this second alternative requires more drilling time). Similarly, Figure 10 shows that the inner coating can be performed with a black coating machine or a white coating machine, but the outer coating is always done with a white coating machine. Not all machines in each resource group are eligible to perform all jobs. Some machines are not technologically able to perform a task. In addition, some machines might be technologically able to perform a task but have never been tested and thus these machines would not be eligible. The process duration for an operation depends on the machine. It is quite stable and does not vary much.

Machines/resources: Each of these production steps can be performed by several parallel machines which we call a resource group. In each resource group, the machines can be different. Especially for coating, machines can use different technologies (black and white coating technologies). In total, the shop floor includes between 30 and 50 machines. The completion step includes heat treatment, and this machine (furnace) processes several blades simultaneously in a batch. All the other machines process a single operation at a time. The machines require operators with specific qualifications for processing. For instance, workers with specific skills are required for the setup. The use case involves 7 workers split over 3 worker's profiles (laser expert, coating expert, and quality expert), and 5 qualifications. Note that one operator can operate multiple machines at the same time. Sometimes the workers are the "bottleneck", and no worker is available to perform a task. They are currently scheduling with FIFO. If a different scheduling strategy is used, they will adhere to the schedule mostly, and they will not change the sequence of tasks.

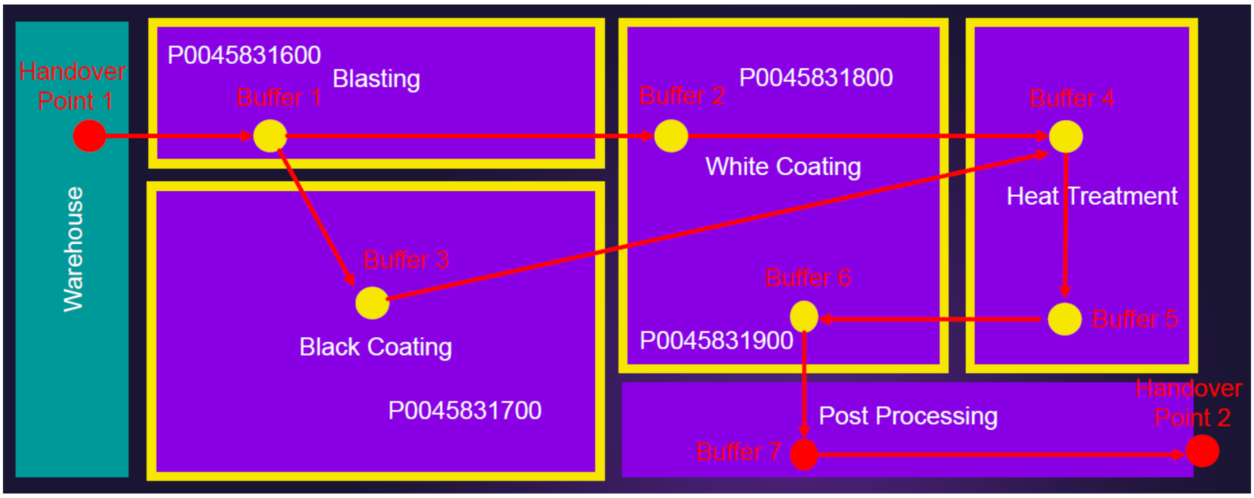


Figure 10: Siemens Energy's coating shop floor organization.

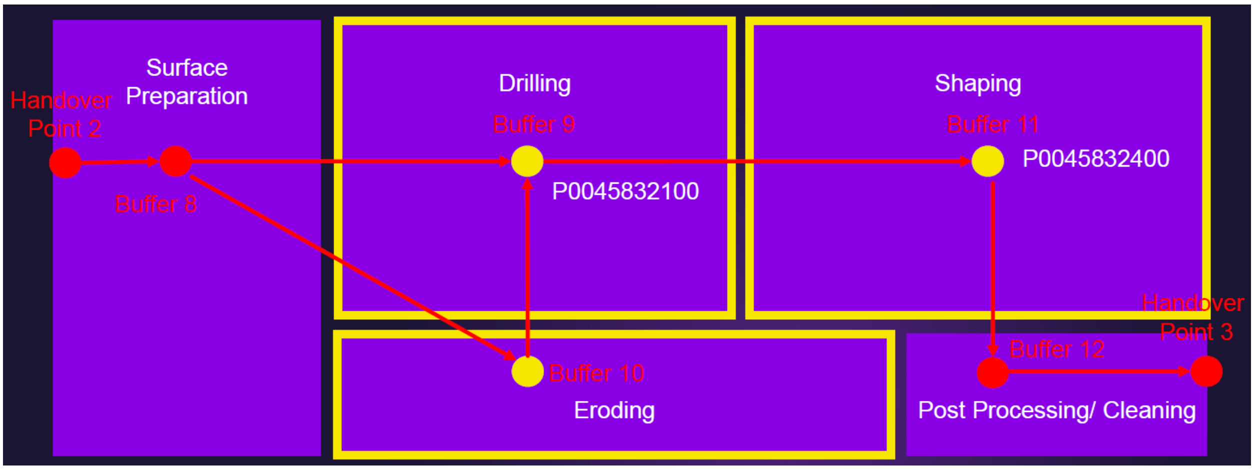


Figure 11: Siemens Energy's drilling shop floor organization.

Material flow: The processing of the operation on a machine requires a setup times that does not depend on the production sequence. This setup time is performed for each individual blade and not per lot. For instance, smaller and bigger blades that need to be fixed on a support at very small tolerance. We will assume that the necessary tools are available for each machine, such as lifting devices, cleaning tools, and fixtures to position the blades and vanes within the corresponding machines. Currently, the shop floor runs the operations per batch, and only complete load elements (containing all relevant blades/vanes) are moved to the next buffer. Operators process the items in batches because of customer needs. All blades from the same customer order line item are batched together. However, SE would like to investigate the possibility to do partial preemption (e.g. 50 %) or even single single-piece flows.

Uncertainty: The main source of uncertainty for the use case is machine breakdowns. Machine breakdowns do not happen often, but they can have a big impact. SE will provide a failure rate and a failure duration.

Scheduling currently : As described in D7.1 SE plans currently manually, but a a simulation is available to validate the plan. The schedule follows a FIFO approach.

Computation speed: Computation speed is not an issue for SE, and the software may take several hours to suggest a plan.

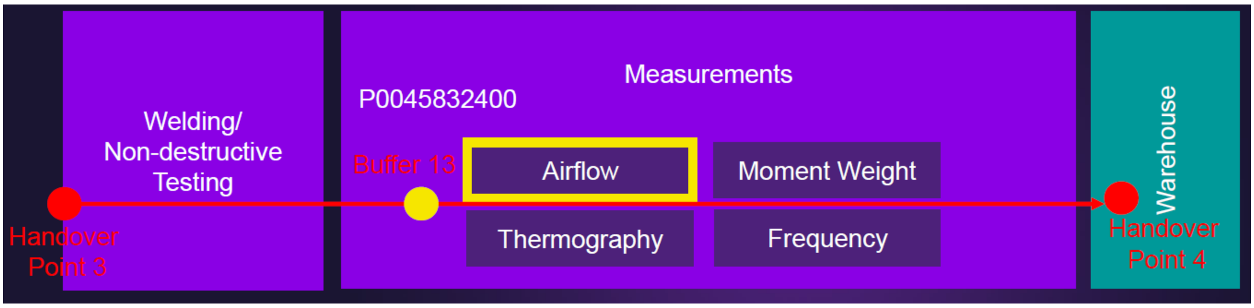


Figure 12: Siemens Energy's testing shop floor organization.

Objectives: Each product to be manufactured has a due date, it can be a finished or an unfinished material. The production orders are may have different priorities (not all due date are equal) based on the business type/order type. In addition, the production must respect the arrival date of the raw materials required for production. In the production planning system currently in place at Siemens Energy, the explosion of the BOM structure provides the due dates for all subsequent manufacturing operations, by reading the routings (processes and process steps). Depending on the scenario, the simulation has either to validate an existing plan (created by human planners), or the simulation has to calculate the best possible start date for all exploded load elements. Nevertheless, these due date on lower level BOM levels are not so important in a day-to-day basis as the customer requested due dates for products to be delivered.

4.3 Requirements on automated production planning

High-level requirement: To remain competitive, companies must operate with efficient production plans to reduce the lead times they offer to their customers, deliver on time, and reduce the operating costs (inventory, changeover costs, ...). Companies that have adopted mass customization are facing difficulties to provide efficient production plans. First, the number of end-items leads to a large number of components (and thus a large number of suppliers), and complex production systems with hundreds of resources. In such circumstances, manually produced plans are often of poor quality, and the human planner needs a tool to help them create the production plan.

The tool provided in WP4 must be generic enough to be applicable in a wide range of environments. In particular, it must apply to the complex situations encountered in the two use cases (see Section 4.2, and D7.1), with alternative operations, capacity restrictions, multi-echelon production system.

RP.1	**Generic model**	**M24**
Category	Functional requirement	
Description	The production planning model must be generic enough to plan in a wide range of manufacturing environments. Following interviews with use case providers, and in-depth analysis of their operations, we must ensure the model can deal with the flexible bill of materials, alternative components, resource capacity constraints, and multi-echelon assembly system.	
Reason	Thanks to the aggregation of items and resources, production planning models are generic enough to handle various situations. To facilitate the use of the production planning modules in the industry, the automated production planning functionality must be applicable in a wide range of manufacturing environments.	

The use cases reported various issues related to uncertainty (see section 4.2). While uncertainty in production planning may concern demand, lead time, capacity, and yield, we will restrict our study to the most critical parameters for our use case (see section 4.2): the demand and the capacity. Note that machine breakdowns or employee absenteeism (a major source of uncertainty reported by the use cases) lead to capacity uncertainty at the production planning level.

RP.2	Uncertainties	M24
Category	Functional requirement	
Description	The planning tool must rely on stochastic or robust optimization to hedge against uncertainty. To account for the dynamic decision process, the tool may employ adjustable robust optimization or multi-stage stochastic optimization. Following the discussion with use case providers, we focus on capacity and demand uncertainty.	
Reason	Today's manufacturing environment is highly uncertain due to the large number of end-items, it is often difficult to predict precisely the demand, the supplier lead times, production yield, and production quality, …. Often the production plans are outdated after a few hours, and this may have large consequences on the performance of the company.	

An AI component for planning cannot account for the details of the shop floor, because it plans for long planning horizons. Indeed, it would be extremely difficult to create an efficient plan accounting for each specific task to perform over a planning horizon of several years (e.g., D7.1 reports that simple computation of the performance of a plan takes 900 seconds in a detailed simulation). More importantly, the precise demand is unknown for the entire planning horizon (as reported by both use cases during our discussion), and looking at specific tasks based on the wrong forecast would lead to wrong decisions. Consequently, the AI component must consider aggregated items, resources, and periods. The production is planned with a granularity of a day or a week. The items and resources are aggregated into families. Nevertheless, this aggregation leads to error [Taal and Wortmann, 1997]. For instance, the resource consumption is computed for each resource group, and production planning approaches do not allocate specific resources to each operation. This aggregation prevents managing the capacity accurately. Consequently, a production plan may not respect the production capacity once implemented in practice. The discussion with our use cases showed the importance and the difficulty to manage the capacity properly (see D7.1) to create valid plans, that are implementable on the shop floor. Therefore, we will investigate how to learn the capacity constraints in the mathematical model through machine learning based on the output of a precise data-driven simulation.

RP.3	Learn from lower-level tools	M24
Category	Functional requirement	
Description	Despite the item and resource aggregation required to make long-term decisions, the tool should be able to represent the available production capacity precisely. These capacity constraints can be learned by analysing the detailed simulation run output. More precisely, the simulation can calculate the execution of the plan based on given product, process, resource, load data, such as start dates, make-or-buy ratios, shift models. The output of the simulation tells the actual capacity consumption for a given production plan, and based on an analysis of several simulation run, we will investigate the possibility to predict the required capacity given the production quantities. This learning can be conducted in an iterative manner, where the planner provides a different plan in each iteration.	
Reason	A valid production plan must make the best possible use of the resources, but it must be implementable on the shop floor. Therefore the automated production planning tool should be able to account precisely for production capacity. Manufacturing shop floors are becoming complex, with hundreds of resources able to perform a wide range of tasks. The simple capacity computations based on average resource consumption per item are not adapted anymore.	

The software cannot be responsible for manufacturing decisions. The planner must remain in charge, and he must be able to drive the creation of the plan depending on the situation. Our use cases requested (see D7.1, and based on the discussion we had during the interviews) the possibility to control the tool and to change the parameters depending on the encountered situation. For instance, in some situations, the planner wants to ensure on-time delivery, in other situations, it is preferable to use the capacity as best as possible.

RP.6	Manual production planning and scheduling	M18
Category	Human-centric requirement	
Description	The human planners should be able to add filtering constraints on the main KPIs to modify the output of the automatic planner.	
Reason	The production manager must remain in charge.	

Finally, based on D7.1 and the discussion we had during interviews the computation time for planning is not critical. For instance, SE would accept a response time of 12h, while AC requires less than 1h of computations.

RP.4	Fast computations	M30
Category	Performance Requirement	
Description	The automated production planning functionality should provide a plan in a reasonable amount of time. Following discussion with use cases, the computation time must not exceed an hour. Ideally, the tool should provide a preliminary result every few minutes.	
Reason	Users want tools with fast response time.	

4.4 Requirements on automated scheduling

High-level requirement: Operating a complex production system requires scheduling efficiency. Most companies operate with simple rules because better scheduling approaches require a lot of manual efforts, to improve the operation of the shop floor, the industry must adopt automated scheduling tools. This tool is also explicitly requested by use cases in D7.1.

RSc.1	Model acquisition	M24
Category	Functional	
Description	WP4 must provide a tool to optimize production schedule. A production schedule affects the tasks to the production resources, and it gives the production sequence on each resource. The tool will include automated schedule generation functionality. The tool will be provided for the flexible job shop scheduling problem (FJSP) because this production environment is generic enough to encompass most situation encountered in manufacturing industries.	
Reason	To automate the generation of production schedule.	

Scheduling must account for the specific business rule of the shop floor. As each shop floor is different and in constant evolution, the design of an automated scheduling tool requires optimization experts to design a shop floor specific decision model. To facilitate the use of automated scheduling approaches, the software should be able to self-adapt to specificity and changes in the shop floor. There we will investigate the possibility to acquire the model automatically from data.

RSc.1	Model acquisition	M24
Category	Functional	
Description	The automated scheduling tool should be usable in a wide range of manufacturing environments with limited expert work. In other words, the software should be able to self-adapt to the specificity and the changes in the shop floor through communication with the digital twin. Each of these adaptions must be validated by a user because autonomous decision-making capability is not in sync with German regulations when humans are involved in the manufacturing process.	
Reason	Automatic generation of scheuling models for optimization.	

Similarly to the planner, the scheduler must have reasonable computation time, and be controllable through a graphical interface.

RSc.2	Fast computations	M30
Category	Performance Requirement	
Description	The automated production planning functionality should provide a plan in a reasonable amount of time. Following discussion with use case providers, the computation time must not be above 1h.	
Reason	Users want tools with fast response time.	

RSc.3	Manual production scheduling	M18
Category	Human-centric requirement	
Description	The human planners should be able to add filtering constraints on the main KPIs to modify the output of the automatic scheduler.	
Reason	The computer software cannot be responsible for manufacturing decisions. The planner must remain in charge, and he must be able to drive the creation of the schedule depending on the situation.	

4.5 Requirement for simulation

High-level requirement: The development of a detailed simulation of the shop floor is an important requirement for WP4 (see D7.1). First, the simulation can validate decisions made by the automated planner and the automated scheduler, by including all the details of the shop floor. The simulation will play an important role in the development of the learning mechanism of the algorithm developed in WP4. The tools will first learn to plan/schedule appropriately by communicating with the simulation, before being used on the shop floor. Therefore, WP4 must provide a detailed material flow simulation to compute the feasibility and the performance of a production plan. To ensure the accuracy of the simulation and shorten the development costs. The simulation must be data-driven.

RSim.1	Semi-automatic model generation	M18
Category	Functional Requirement	
Description	The simulation module must allow for a (semi-)automatic simulation model generation from the static production plant data (as specified in the domain model). Manual steps should be limited to a minimum, e.g. moving components on the graphical factory layout sheet of the simulation tool to get a better overview and understanding of the plant.	
Reason	The domain model in the data fabric is the agreed source of truth in ASSISTANT.	

The simulation must be able to simulate the production plan and the production schedule generated by AI components.

RSim.3	Simulation scenario in static decision framework	M18
Category	Functional Requirement	
Description	A simulation scenario provides a value for each decision to make. In a static decision framework, the simulation module must allow for an automatic simulation scenario generation from decisions made by the production planning and scheduling module over the entire planning horizon (as specified in the domain model).	
Reason	Validate plans and schedules	

RSim.6	Scenario results	M18
Category	Functional Requirement	
Description	The simulation module must return the results as specified by the domain model. Where different business targets are available, condensed results according to the selected business targets need to be provided, e.g. by calculating weighted sums.	
Reason	Simulation results contain a huge amount of data. Only a subset of this data is relevant for the decision process. The assessment of the result data may also differ depending on the selected business target.	

As described earlier, managing uncertain event is important in Industry 4.0 production systems. To ensure the production plan and the production schedule remain feasible in an uncertain environment, the simulation must account for the variation of the parameters.

RSim.2	Uncertainty	M18
Category	Functional Requirement	
Description	Where required, the simulation module must use the functions of the the simulation tool to account for variations. If not available, the simulation module itself must offer the possibility to evaluate different stochastic experiments.	
Reason	Robust decisions are required.	

Methods to plan in a stochastic dynamic environment produce policies that tell the decision to make depending on the state of the production shop and its environment. Such approaches do not provide a fixed plan for the next month as the shop floor manager are used to work with. To validate the policy, the user must simulate its execution.

RSim.4	Simulation scenario in dynamic decision framework	M18
Category	Functional Requirement	
Description	A simulation scenario provides a value for each uncertain parameter and each decision to make. In a dynamic decision framework, the decisions for period t+1 are made once the value of the uncertain parameter up to period t is revealed. The simulation module must allow for an automatic simulation scenario generation from the dynamic production planning and scheduling decisions. In other words, the simulation must provide the state value to the production planning module, and receive the decision to execute in each period.	
Reason	An agile factory should update its production plan when new information is available. In a dynamic decision framework, given the updated information available in each period, dynamic production planning and scheduling decisions are selected in each period by the production planning and scheduling modules or by user interaction.	

Similarly to the planner and the scheduler, the simulation must have reasonable computation time, and be controllable through a graphical interface.

RSim.5	Scenario execution	M18
Category	Functional Requirement	
Description	The simulation module must offer an interface to execute and cancel a simulation run.	
Reason	The simulation will be triggered by the production planning and scheduling modules or by user interaction.	

RSim.8	Fast computations	M18
Category	Performance Requirement	
Description	The simulation module must run the applied simulation tool in the fastest mode (provided by the tool).	
Reason	Fast response time is required.	

4.6 Requirements on data

High-level requirement: The intelligent digital twin requires a precise and update digital picture of the factory which is built from heterogeneous data sources. To facilitate access to this data and the integration of the tools developed in WP4 on the shop floor, the data must be integrated into a domain model. We give below all detailed requirements for the data fabric and the domain model for production management.

The tools of WP4 require data from various sources (ERP, MES, supplier data, ...), whose data models are likely very different (different scope, terminology, units, ...). For instance, our discussions with AC showed the difficulty to collect data across various software. To facilitate the integration of the tool in the manufacturing environment, a domain model must be provided to access the data from a single interface.

RD.1	Data integration	M18
Category	Functional Requirement	
Description	The domain model must provide a unified access to heterogeneous data. The tool of WP4 will have a standard domain model interface to access all required data. This domain model must reuse existing standard as much as possible.	
Reason	Integration of heterogeneous data sources	

Given the strong connection between the domain models of the different WPs and industry cases, a common format and tooling for domain modelling is identified as a requirement of ASSISTANT. The tools for production planning and scheduling require different levels of aggregation for the data. For instance, production planning considers the global capacity per resource, whereas scheduling requires knowing the available specific resources and their status.

RD.2	Data flow	M18
Category	Functional Requirement	
Description	The domain model must provide tools/rules to automatically aggregate data (define resource group, ...). For instance, scheduling accounts for each machine individually, whereas production planning uses the global capacity per resource group as input.	
Reason	Communication between the modules of WP4.	

The decision support tool must see what decisions must be taken, and how to evaluate the quality of these decisions. The user must also have a clear understanding of the KPIs used by the optimizer for transparency.

RD.6	Design for optimization	M18
Category	Functional Requirement	
Description	The domain model should differentiate KPIs, decisions variables, constraints, and parameters.	
Reason	the domain model must tags the parameters to facilitate decision support.	

Production planning and scheduling require an estimate of future values for some parameters (demand, delivery lead times, ...). The data fabric must facilitate the prediction of these parameters.

RD.3	Predictive analytic	M18
Category	Functional Requirement	
Description	The domain model must recognize known versus unknown data. The unknown data have not been discovered yet, and their values must be predicted. The intelligent twin for production planning and scheduling must provide machine learning tools to predict the values of these parameters based on historical data.	
Reason	prediction of parameters	

The use case providers mentioned the importance to have a reliable production plan able to cope with various sources of uncertainties. To account for possible uncertain events, the tool must be able to characterize the uncertainty as best as possible (through a probability distribution or an uncertainty set).

RD.4	Handling probabilistic data	M24
Category	Functional Requirement	
Description	The tools must identify unknown parameters with large variation, and they must allow the user to infer a probability distribution or a range of values for parameters that vary significantly. Therefore, the domain model must provide an interface for domain experts to build learn a probability distribution or an uncertainty set based on the historical data.	
Reason	Characterize the uncertainty as best as possible (through a probability distribution or an uncertainty set).	

ASSISTANT fosters a human centric approach, and the user must also be able to manipulate the data.

RD.5	User friendly	M12
Category	Functional Requirement	
Description	The human production planers must be able to understand, extend, and modify the domain model.	
Reason	The domain model provides the data interface for tools of WP4. To reduce the development costs of tools provided within WP4, manufacturing experts should be able to setup their domain model.	

4.7 Ethical requirements

Even though the present document covers the technical requirements established for WP4, the first revision of ethical considerations was performed to have a broad understanding of the impact the developed tools could have on the manufacturing users. This initial evaluation is subject to the analyses performed by WP2 (tasks explicitly T2.1 and T2.2). A first analysis performed over WP4 in relation to risk conditions for human security and oversight resulted in a clear establishment that there is NO impact on fundamental human rights, human well-being, or democracy within this WP. Therefore most of the ethical considerations driven by WP4 should be focused on technical robustness, privacy and data governance, transparency, and accountability. (based on the Trustworthy requirements.). Lawful local concerns will be carried out by the system as system constraints (e.g. overwork and shifts maximum time), so the AI component will manage regulations.

RE.1	Respect work regulations for the employee	M30
Category	Functional requirement	
Description	The tools that make decision regarding employee scheduling must ensure work regulation are properly incorporated in the models. This include for instance the respect of the maximum amount of work hours per weeks, vacations, sick leaves, or other regulatory conditions that can impact the legal frameworks.	
Reason	The AI components must respect work regulations.	

The AI components of WP4 may allow the user to prioritize a customer or a supplier over another, and this could be seen as unfair to the clients or the supplier. This point is not seen as an issue, as the AI component will be fair to the manufacturer. In practice, the implementation of tools at specific manufacturers may require anonymous client/suppliers (represented with numbers), but we will not enforce this restriction within ASSISTANT. The decision is left to the manufacturer who implements the tool within its factory. Therefore, any fairness concern that could arise in relation to production scheduling (WP4 outputs) would be managed externally to the AI by users (given override options). Furthermore, since tools will not use any sensitive information that could consider diversity or discriminatory concerns, scheduling preferences would follow users current trends and not societal biases. Nevertheless, as production planning and scheduling decisions have a critical impact on the factory economic performance, we must identify a responsible for AI components failure.

RE.2	Ethical requirement	M8
Category	Human-centric requirement	
Description	For each module (simulation, model acquisition, optimization for scheduling, automatic planner, see section 5) developed within WP4 must be assigned one responsible for each cause of failure directly linked to algorithmic failing. Such failures are situations where the AI component does not provide a result, or it provides a result of poor quality. Within WP4, such a failure occurs when the production plan or the production schedule is not implementable on the shop floor, or when they are associated with bad KPIs values. Furthermore a clear establishment of responsibility on: (1) Poorly constructed algorithm, (2) data is of bad quality, (3) human use the tool poorly. Will be done	
Reason	Compliance with Accountability.	

Another critical concern regarding ethics is the security of the data. Fake information provided to the systems may lead to the wrong decisions and considerable economical impact. While the security of the data itself will be ensured within the data fabric, each module of WP4 must ensure secure communication with the data fabric and control over the access to the module. Furthermore, UI interfaces will be limited to organize information and interact with information (not modify it) and, therefore, access to data managing would not be performed throughout any component within WP4. Finally, as checked in the following requirement, Further security considerations will be set in place.

RE.3	**Security of the data**	**M36**
Category	Ethical requirement	
Description	Each module developed within WP4 must require to log in, and state-of-the-art protocols must be used to ensure secure communication with the data fabric.	
Reason	Ensure the safety of the data and safe communication with the data fabric.	

Running the production planning and scheduling tools on corrupted data (by accident or due to an attack) may have a large impact on the economical performance of the company.Ideally, the tool must be robust against attack or misinformation.

RE.4	**Robustness of the AI component**	**M30**
Category	Ethical requirement	
Description	Participant of WP4 may investigate how to identify if a user set a parameter/data to a wrong value. What could be the effect on the system?	
Reason	Ensure safety against attack and misinformation.	

Following transparency requirements (linked with the communication), the user must be informed that they interact with an AI component. This consideration could be necessary if the information is sent directly to the shop floor to be used, and therefore, any inadequacy of the tasks requested (given by know-how considerations) could help to mitigate erroneous outcomes (and prevail failing conditions).

RE.5	**Inform user on the interaction with an AI component**	**M18**
Category	Ethical requirement	
Description	The plans and schedules created entirely or partly with an AI-system must be tagged to inform they were created by an AI. In addition, the chatbots must inform the user that they are interacting with an AI.	
Reason	Comply with regulation of Transparency and Accountability	

To ensure accountability and transparency, all decisions suggested by the AI will be tagged. This will allow to evaluate, in case of need, users preferences and modifications in case the best options derived by the AI are not chosen to be implemented.

RE.6	**Log decisions**	**M36**
Category	Ethical Requirement	
Description	The systems must log the suggested decision, and the decisions validated by the user for 6 months. The system must log all input & output data of all simulation runs, production planning run, and scheduling run for a specified period (e.g., 6 months). Also any additional logging data by the used simulation tool shall be stored.	
Reason	Accountability	

To comply with transparency, the learning process for the model acquisition in scheduling and planning must be reproducible, and the system must be able to produce the information that led to a specific output of the AI component.

RE.6	**Tag data used for training, prediction, and optimization**	**M36**
Category	Ethical Requirement	
Description	A tag must be associated with each data used during the learning of a specific AI component, during the building of the model, or during prediction.	
Reason	Ensure we can re-evaluate the algorithms in case it is required, this allows reproducing failure to improve and explaining the decisions.	

To comply with human agency and oversight, the AI components should not make autonomous decisions that may have an impact on the workers. Therefore a manual planning interface must be defined for the user to control the AI component and validate any decision before actuation.

RE.7	**Human validation**	-
Category	Human-centric requirement	
Description	The tools developed in WP4 cannot transfer production plans or production schedule for actuation unless they are validated by a user.	
Reason	The computer software cannot be responsible for manufacturing decisions. The planner must remain in charge.	

For the explainability of the decisions, the user should be able to visualize the decision model: the decision variables, the objective functions, and the constraints. This implies that users would be able to understand the decisions made

RE.8	**Model visualization**	**M18**
Category	Ethical Requirement	
Description	The user must be able to visualize the acquired model. The visualization must clearly explain how the tool makes decisions (with the variables, constraints, and objective function) so an optimization expert can explain why the tool makes a particular decision. Ideally, the tool should also be able to explain the decisions to a non-expert. For instance, the tools may explain which variables, constraints affected the most decisions. The tools may also explain why another decision was not selected. For instance, the user input a plan, the system tells why the solution was not selected (which constraints are violated, which cost increase).	
Reason	Explainable	

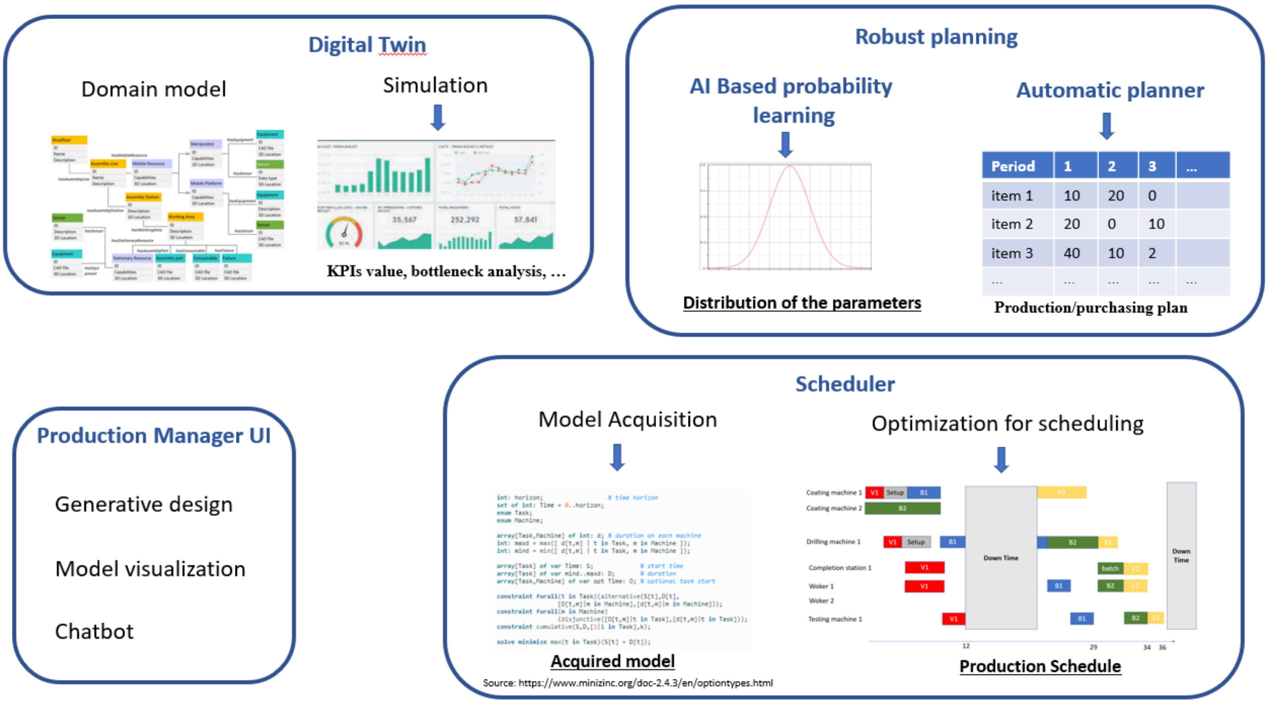
Finally, the tool must comply with the following requirement from the GDPR. Nevertheless, this requirement concerns the input data to the tool, and use case providers will be responsible to identify the data, and use the anonymization tools provided in the data fabric.

RE.9	**Model visualization**	-
Category	Ethical Requirement	
Description	The data used, collected, or stored may not include the name of a user or a worker. They must be anonymized. In addition, the data should not allow the identification of a single person and should link towards personal information.	
Reason	GDPR compliance	

5 Description of the tools and research agenda

WP4 aims to develop an intelligent digital twin for production planning and scheduling. As described in Section 3, the tool will also be usable in a digital shadow, but the user will be responsible to provide accurate data and to communicate the schedule to the shop-floor. To provide explainable models (see requirement RE.9), the AI components for planning and scheduling will rely on symbolic AI, constraint programming, and mixed-integer programming. These approaches proved their efficiency for planning and scheduling, but they provide explainable models with clear objective functions and constraints. Fig 13 shows the modules developed within WP4 to create an intelligent twin for production planning and scheduling, and these components are described below

Automated planner: This tool helps the user to create a production plan. The production plan can be created at two granularity levels. For material requirement planning, the plan gives the production quantity for all items (end-item and items in progress) to release on the shop floor in each period, the number of components to orders to suppliers, the amount of production



Period	1	2	3	...
item 1	10	20	0	
item 2	20	0	10	
item 3	40	10	2	
...	...	...	...	...

Figure 13: Modules developed within WP4

subcontracted, and the extra capacity required for each resource. This tool is used every week or every day depending on the need of the company, and these decisions are made to ensure the demand is met on time, the inventory level is minimized (which lead to a reduction of the production lead time), and the setup times and costs are reduced. The automated planner takes as input the demand per period, the available capacity, the bill of material, and various costs. For master planning, the production plan provides the number of end items to produce, the number of resources and employees to acquire/hire, and the subcontracting quantities. Master planning is done every month or every quarter depending on the company. The tool must hedge against uncertainties, and the production plan must be robust to machine break down, late component deliveries, and changes in demand ...

AI-based probability distribution learner: This tool helps the user understand the behavior of scheduling/production planning parameters that vary significantly, and whose variations have a lot of impact on the optimal plan/schedule. The resulting joint probability distribution explain to the user how to reduce the variability of a parameter. These probability distribution is also an input of the robust planner and the simulation. The user selects a parameter to analyze and the concept that may explain the value of this parameter. The tool will output the conditional probability distribution of the parameter. For instance, the user may want to analyze the available production capacity on a resource group. He can select the parameters that explain the value such as the last maintenance on the resource, the production load, or the operator. The tool will compute a conditional probability distribution for the available capacity based on the available historical data.

Optimization for scheduler: This tool helps the user to create a production schedule. The scheduler takes as input a set of jobs, where each job corresponds to a production lot released by the planner (i.e., a production lot for a given item). To ensure adherence to the plan, each job has a release date and a due date to respect. A job requires several operations to perform on different resource groups, and the duration of each operation depends on the quantity released by the planner. The scheduler assigns the resources (machines, operators, ...) to the operations, and it sequences the operations on the machines, thus each operation is given a start and end time. The scheduler optimizer is used several times per day (as soon as the current schedule is outdated), it takes as input the current status of the factory and the production plan, and it outputs a production schedule.

Model acquisition: The scheduling optimization model models must account for various industry-specific constraints, and facilitate the development and maintenance of the scheduling model. The model acquisition tool create a constraint programming model, that is later user by the optimizer to suggest schedules. The model acquisition tool is run during the first use of the scheduler and every time a major change occurs on the shop floor (new machine, new product, ...). The model acquisition tool takes as input the data the user selected as relevant and schedules implemented in the past. Then it generates a constraint programming model that describes the constraints the schedule must satisfy to be feasible.

Domain model: The digital twin should provide an accurate picture of the shop floor with all information required to manage the operations. Therefore, the digital twin includes a domain model that extends the generic domain model of WP6 with the specific data structure required by the intelligent digital twin for planning and the intelligent digital twin for scheduling. It integrates data from various sources and aggregates the data at the right level for the simulation, the scheduler, and the planner. The domain model presents a rich data model understandable by the end-user. Thus, the user can access all manufacturing data through the domain model. The domain model is also the main data source for all modules of WP4, and it is the main communication bridge between tools developed in other work packages. More information on the data flow is given in the next section.

Simulator: A discrete-event simulation allows the user to analyze the impact of a production plan or a production schedule, and it can be used in a dynamic environment to evaluate policies. The simulation is used several times a day to analyze the decision made by the user (eventually with the help of the automated planner or optimization for scheduling). The simulation provides a very detailed view of the shop floor, and it allows computing the start and end date of each task, various KPIs, and to recognize the bottlenecks. The classical input data for the simulation is the production plan (quantity released per period and due date, number of hour of each resource type available per period), or the production plan and production schedule (quantity released per period and due date, precise resource available in each period, affectation of the operations to the machines, and sequencing of the operations). When used in a dynamic environment, the simulation evaluates a policy, and it communicates with the production planning tool directly to access the decision in each period based on the latest information on the status of the shop floor (demand, machine break down, supply delivery lead time, …).

Production manager UI: The production manager UI provides the interfaces for the user to interact with the tools developed in WP4. This interface includes three elements: the generative design interface; the model visualization interface; and chatbots. First, the generative design interface allows the decision-maker to interact with the automated production planner and production scheduler. The user can specify the characteristics of the solution he wants to create: provide upper or lower bound for the KPIs, assign weights to the optimization objectives, enter additional constraints on the plan/schedule. Second, the user interface provides a model visualization tool that allows communication between the model acquisition tool and the business expert. The user can visualize the acquired model for full transparency on how the AI module makes decisions, the user can also manually modify some constraints if they were not correct. The model acquisition module may also ask the user to confirm some constraints. Finally, we will investigate the use of a chatbot for the user to query the domain model.

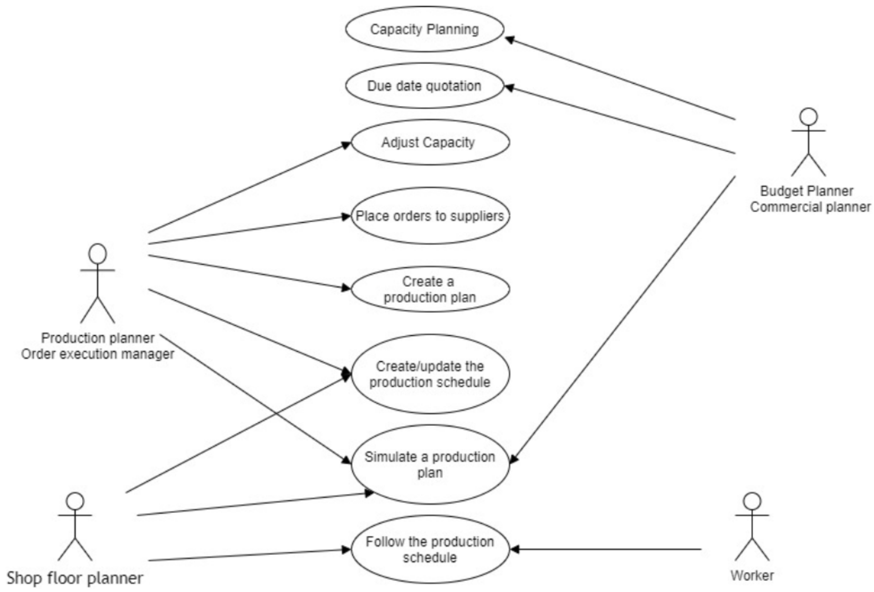


Figure 14: Use case diagram

To fulfill requirement RE.2, a responsible for component failure due to poorly constructed algorithm is assigned to each module: T4.3 leader for the simulation; T4.4 leader for the AI-based probability learner and automatic planner; T4.5 leader for the model acquisition and the optimization for the scheduler. The responsible will follow the procedure developed in WP2 to ensure the robustness of their AI components. In particular, the responsible will benchmark the tools on synthetic and realist data sets. These tests will clarify the conditions under which the algorithms function properly and their success rate. To fulfill requirement RE.4, the responsible may investigate which variables or data provide the most risk to make the AI component fail to monitor this data closely. They might also evaluate the consequence of a wrong decision for the manufacturing system.

These modules are developed and delivered continuously during the project: D4.2 (M18) provides the domain model and simulation; D4.3 (M24) provides the predictive analytic elements: the model acquisition and the AI-based probability learner; D4.4 (M30) provides advanced optimization for automated decisions: the automatic planner and optimization for the scheduler.

The rest of this chapter describes the tools in detail. The sections precisely explain the novelty that will be developed within the ASSISTANT project. These novelties rely on (1) the learning of constraint within the automated planner and automated scheduler through an actor-critic communication with the simulator; (2) the integration of stochastic or robust optimization approach to deal with uncertainties. Section 5.1 and 5.2 provide a use case diagram and the information flow between the different modules. Section 5.3 describes the generative design interface and model visualization developed in T4.2. Section 5.4 describes the digital twin, including model the domain model and simulation developed in task T4.3. Section 5.5.2 describes the production planning module developed in T4.4, and it explains how the AI-based probability distribution learning is used within WP4. Note that a generic AI-based probability learning module is developed within WP6 (as it also useful for WP3 and 5). Finally, Section 5.6.2 describes the model acquisition, and scheduling optimization developed in T4.5.

5.1 Use case scenarios

Figure 14 provides the use case diagram of the intelligent digital twins for production planning and scheduling. The roles and the responsibilities of the employees involved in production planning and scheduling decisions differ from one company to the next. The above use case diagram is based on the roles identified by *Siemens Energy* (see D7.1):

- **Budget planner:** Employee who plans the manufacturing volume to run the manufacturing operations of the defined scope from a technical perspective for a given time frame, most cases for a fiscal year.
- **Commercial manager:** Employee who reviews the technical manufacturing plan and assigns a corresponding financial budget to run the manufacturing operations of the defined scope for a given time frame, most cases for a fiscal year.
- **Production planner:** Employee who is overseeing the execution of production orders from a customer/order management perspective.
- **Shop floor planner:** Employee who plans manufacturing operations for a smaller subset of the defined scope, usually for a shorter time frame, e.g., for a month, a week or for a specific shift.
- **Worker:** Employee who carries out the different tasks on the shop floor. The workers must get a convenient work schedule, and know what tasks he or she will work on.

The functionalities of the described roles are:

- **Capacity planning**: Decision on the s the shift model, qualification of existing machines within the resource group. Under normal conditions, temporary workers might be hired for given time spans.
- **Due date quotation**: Commit to a delivery date for each customer order.
- **Adjust Capacity**: Adjust the number of shifts per week or change the length of the work shift, for instance with extra hours during the weekend. Define the make-or-buy ratio for each period.
- **Place orders to suppliers**: Place raw material and components orders to suppliers (which supplier, when to deliver, what quantity).
- **Create a production plan:** plan the production (how much to produce in each period), compute the expected inventory costs, the setup costs, and the probability of shortages.
- **Create/update a production schedule:** manage the detailed operations on the shop floor. What operation is performed in each moment and with which resources?
- **Simulate**: simulate the decision (shift length, number of qualified resources, production load per period, task assignment to resources, task sequence on the resources, outsourcing, …) on the digital replicate of the factory.
- **Follow the production schedule:** Visualize the production schedule to be implemented on the shop floor. Note that WP4 will not provide tools to update the schedule in real-time (this is the goal of WP5)

5.2 Information flow

Figure 15 shows the data flow between the different modules of WP4. Note that all the data is transferred through the data fabric (and its domain model), and there is no direct data transferred between tools. The only direct communication between the module of WP4 aims to trigger the run of a module that responds to tell computations are done, and the data is available in the domain model. The data fabric provides APIs to store and retrieve data; these APIs insert

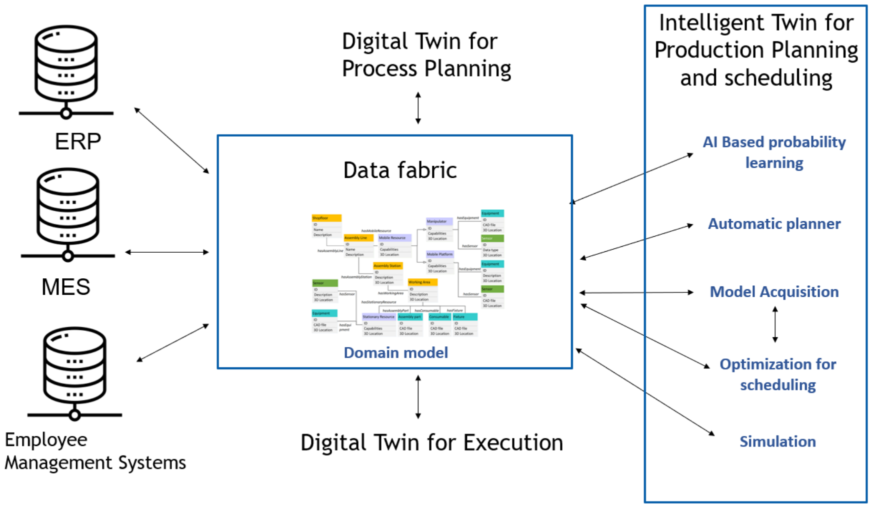


Figure 15: data flow in WP4

and retrieve the data from the domain model. The data fabric will provide tools to transform the data when required. For instance, the production planner provides a set of production lots and depending on the factory, these production lots will become one or multiple jobs in the scheduler (one per unit of item). Transferring all data through the data fabric will ensure accountability since it creates logs of all the computation results. We will use JSON to exchange data from and to the data fabric, but each module may provide additional input/output data formats (CSV, XML, ...). To fulfill requirement RE.3, secure communication between the tools will be ensured with REST API, the communication will be encrypted with public keys protocols, and the API will require authentication. Table 3 details the specific data communicated between the modules.

From \ To	Domain model	Simulation	Scheduler's optimization	Scheduler's Model Aquistion	Automatic Planner
Domain model / Data Fabric	-	• Load elements: end-item, quantity, release date, due date. • Available resources (machines and workers) along with their qualifications/skills, and calendar • The bill of materials (incl. alternatives) • The bill of processes (including alternatives) with the set of items, the set of processes, and the input item and output item for each process. • process steps with the input item, the required resource, resource type (batch machine or not), maximum batch size, process duration, setup duration, qualification of the operator for setup and process.	• Set of Jobs j, with the set of operation, product type, quantity, due date, release date, … • Set of Resources, with their type, speed • Machine Eligibility restriction matrix • routing of each job on the machines	Historical data corresponding to schedule implemented in the past. Including the input data (see from domain model to scheduler's optimizer) and output data (see from scheduler's optimer to domain model.	**Known data:** • Set of resources. • Flexible bill of material. • Various costs (setup, inventory, unit production costs, Extra capacity cost per resource …) • Inventory levels for end-items and components. **Uncertain data:** • Demand per period and per end-item (e.g., per week) • Resource consumption per operation • Resource minimal capacity. • Yield • Delivery and production lead times
Simulation	• Schedule: start/end date of each task, routing for each production load, workers allocation. • KPIs values: on-time-delivery, machine utilization, lead time, production cost, …				Information required to improve the accuracy of the production planning model: • KPIs value. • Adherence to the plan (list of orders produced late) • Actual resource consumption per period. • Machine utilization • Actual lead time per production load.
Scheduler's optimization	Sequence of operation on each resource. Resource (machine/worker) affected to each operation				
Scheudler's Model Aquistion	Constraint progamming model.				
Automatic Planner	• Production plan (quantity per period over the planning horizon) • Extra capacity required per period and per resource • Quantity subcontracted • Quantities to order to suppliers	• Load per period • Capacity per period and per resource (shift lenght)		-	-

Table 3: Information flow between the module of WP4.

5.3 Production Manager UI

The user interface includes three components: a generative design interface for production planning and scheduling, a model visualization tool, and a chatbots. All three components are described below.

Generative design: The user must be in control of their production planning and scheduling tools to not see it as a black box [Taal and Wortmann, 1997]. To fulfill requirements RP.6 and RSc.3, we will provide **generative design interfaces**, where the user can iteratively precise the values of the KPIs he wants, and the tool provides a solution within these targets. The KPIs at the production planning level may include the inventory costs, changeover costs, backorder costs, outsourcing cost, extra resource capacity cost, number of setups, percentage of on-time delivery, level of inventory. The KPIs at the scheduling level may include the due date adherence, the total setup time, and the flow time. The generative design interface includes:

- The possibility to precise a range for the KPIs, and to weigh or rank the optimization objectives. For instance, for scheduling, the user can ask for a solution with at most X jobs completed late, or at most X change over on resource X , …. For planning, the user can input the maximum amount of stock, ….
- The possibility to visualize the resulting production plan (production quantity for all items (end-item and items in progress) to release on the shop floor in each period, the number of components to order by suppliers, the amount of production subcontracted, and extra capacity required for each resource) or the production schedule (Gantt charts). Note that there seems to be a clear lack of a scalable and open-source Gantt chart library that could be reused. Therefore, we will develop the visualization only (not interactive) version for the scheduling aspect of WP4. The result is an SVG-based diagram.
- Tools to visualize the search space in a plot (with KPI scores as plot axes).
- Tools to show differences and similarities between two solutions. More precisely, the user is able to select the solution he or she wants to compare, and the tool highlights the main differences for the user to make a choice.
- Tools for solution visualization across a large number of scenarios of uncertain parameters (demand, lead time, capacity, process duration, and production yield). This will include the possibility to visualize the performance of the plan on the worst case, average, X^{th} percentile scenario.

To fulfill requirement RE.5 the production manager UI will inform that the plans and schedules are generated with the help of an AI.

Model acquisition visualization: We will provide tools to visualize the optimization model learned automatically from data. Some constraints have a natural visualization, and they will be displayed on the Gantt chart of the solution (precedence constraints, or cumulative constraints). Other constraints will be displayed in a textual format (constraint, formula) referring to the attributes of the concepts in the domain model (e.g., name of the attribute, subset of attributes with specific values, etc.).

Chatbot: Finally, we will investigate the value of a chatbot to answer operation management questions (e.g., when will the order X be scheduled?) related to production planning and scheduling. The chatbot will be based on predefined questions, and an AI system will match natural language questions to the predefined questions. Next, the tool will generate a SPARQL request to send to the domain model. Finally, the SPARQL answer will be translated to natural language. As a result, the chatbot will be able to communicate with the user thanks to the rich semantics included in the domain model. Note that the chatbot will not be implemented from scratch, we

Component	Suggested technology
Graphical User Interface	HTML, CSS, Java script, Angular (maybe), JSON, Java (maybe)
Visualization of constraints in model acquisition & model execution	CP-Viz [Simonis et al., 2010], SVG, PDF, JavaFX, Java
Chatbot: Language Recognition and generation	Python, RASA
Chatbot: Query Interface Python	SPARQL, OWL

Table 4: Technologies used in the production manager UI

will use existing tools, and tests their use for production planning and scheduling applications. To fulfill requirement RE.5, the chatbot will inform it is an AI when communicating with the user.

5.4 Digital twin

While so far data-based approaches and AI methods dominate in the production operation phase, simulations are currently primarily used in design and engineering. The vision of the digital twin refers to a virtual representation and a description of a component, product, system, infrastructure, or process by a set of well-aligned, descriptive, and executable models. It is a semantically linked collection of all relevant digital artifacts, including design and engineering data, operational data, and behavioral descriptions. It exists and evolves along the whole life cycle. The digital twin integrates the currently available and commonly required information and knowledge and is synchronized with the real twin if it exists. A simulation model represents the planned real system and calculates its properties or validates its behavior. With the advancement of simulation technology and the available computing power, these simulation models become more detailed and cover more aspects of the system under development. They thus represent a digital twin of the planned system, which leads to an extended understanding of the term digital twin [Rosen et al., 2019]. Note that we differentiate here the classical digital twin, and the intelligent digital twin proposed in assistant that include AI components for prescriptive and predictive analytics. WP6 will provide data acquisition, cleaning, storage and integration in the domain model. This section cover the extension of domain model for planning and scheduling, and the simulation.

5.4.1 Domain model

With the rise of industry 4.0 technologies, a large amount of data is permanently collected by various sensors, connected machines, systems, and digital models. Due to the evolutionary development of most factories – i.e. new machines and new technologies are permanently integrated into the legacy systems or existing structures of the production system – the data landscape in production systems is very heterogeneous and come from very different sources, such as MES, ERP, SCADA, machine data. Additionally, the data can appear in different formats, e.g., XML, JSON, CSV, text files, distinguishing between structured, semi-structured, or non-structured data. This data must be related to the respective data generators to create a domain model that enables and empower analytical components such as machine learning algorithms, simulations, and optimization algorithms. The domain model is a representation of all concepts and their relations in the domain that are useful for production planning and scheduling, for example, orders, resources, production steps, order, cycle time, etc., thus forming an ontology of the production domain. Optimally, each instance within the production, be it a machine, a product, a process, a material, or a worker, can be mapped and related to each other.

The domain model must have certain aspects to serve the decision support systems on the shop floor. As described in [Listl et al., 2020], the use of different layers and applications in decision support requires reusable, standardized, flexible, and extensible means for data exchange between them. This is an essential prerequisite for ensuring that decision-support solutions do not have to be rebuilt entirely for each factory but that certain parts can be reused across projects - i.e., following a library or framework approach. A generic domain model will be defined within WP6 along with various functionality required by WP3-6. In WP4, this generic domain model is extended with concepts specific to production planning, and specific to scheduling. In a later stage, the resulting model can be further extended with company-specific knowledge. Figure 16 shows an example of this organization.

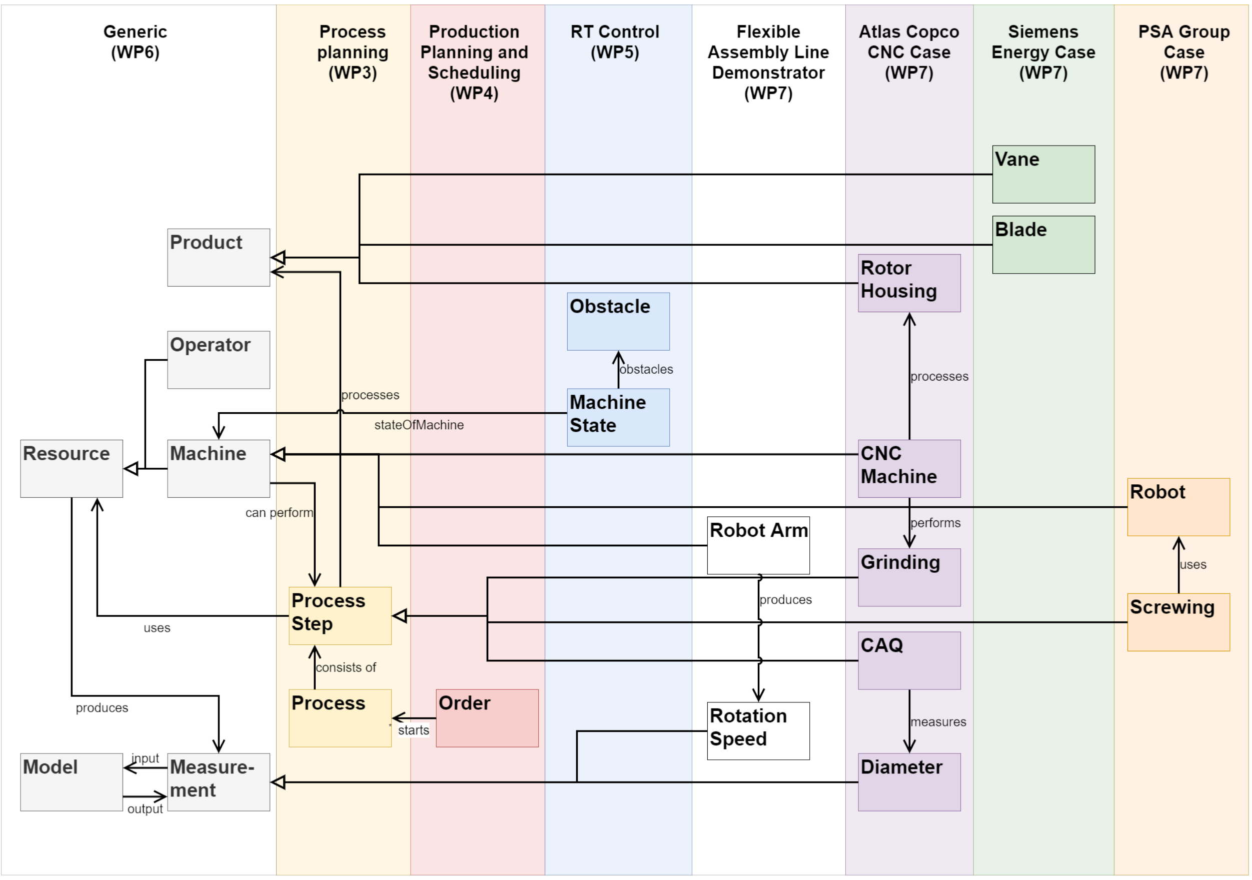


Figure 16: Organization of the domain model

The domain model is the access point to the data for WP4 production planning and scheduling tools, and it plays five major roles:

1. It is a mean to understand the system, and what information on the system is available (what resources, products, etc. exist?).
2. It provides unified access to heterogeneous data. The tools of WP4 require data from various sources (ERP, MES, supplier data, ...), whose data models are likely very different (different scope, terminology, units, ...). This features provides requirement RD.1.
3. Next to data, the domain model provides access to models, simulations, correlations between parameters, experimental setups, etc., and how they have been used in the past.
4. Modelling of known vs unknown data is another part of the domain model. It will allow indicating data that have not been discovered yet, and whose value must be predicted.

Component	Suggested technology
Domain model representation	OWL, RDF
Domain model tooling	Protégé
Domain model query language	SPARQL
Domain model query interface	HTTP endpoint, REST API
Tooling for link between domain model and data	Ontop

Table 5: Technologies used to provide a domain model and the associated tools

The prediction (done outside of the domain model) can be done based on historical data, or based on the data acquired after simulation runs. The AI-based probability learner is described more precisely in Section 5.5.1. This tool can learn the value of parameters with classical machine learning approaches, but it can also learn a distribution of parameters that vary significantly. For instance, the tool could return an empirical distribution based on historical data collected in a similar situation as the forecast. As the empirical data will over-fit, we will explore AI-based approaches to obtain distributions such as Bayesian networks. In some situations, historical data might not allow estimating the future value of the parameters precisely (because this data might be missing, or because the shop floor changed and historical data are not relevant). In such a case, the possible value for the parameter might be estimated more roughly through an uncertainty set (a range of values). This feature provides requirements RD.3 and RD.4.

5. The tools for production planning and scheduling require different levels of abstraction for the data. For instance, production planning considers the global capacity per resource group, whereas scheduling accounts for each machine individually. The domain model will provide tools/rules to automatically link different levels of abstraction (define resource group, …). This features provide requirement RD.2.

The domain model differentiates the known parameters, stochastic parameters, decision variables, and KPIs. These features will facilitate the model acquisition, and it provides requirement RD.6. As specified in requirements **??** and RE.6, the domain model will store the output of the simulation, scheduling, and production planning runs. The data model must include the parameter of the run, and whether the decision was validated by the user or not. To fulfill requirement RE.7, access to decisions non validated by human will be restricted to the tools of WP4 only, the data fabric may only transfer to the real-time controller the last validated decision. Finally, to comply with requirement RE.6 for each request to access the data, the domain model will store the reason (training, prediction, optimization) of the access along with the identifier of the tool and time.

Users will be able to query the knowledge graph in a unified way using a query language at the abstraction level of the knowledge domain. This means that there is no need to know the technical storage details of all data. Furthermore, tooling will be provided to edit the knowledge graph, for adding new types of data, information, and knowledge. For example, when a new sensor is added to a robot, the user must be able to make its metadata and data accessible in the domain model, thus adding its information and link to the data storage. These features fulfil requirement RD.5.

5.4.2 Simulation

To analyze the impact of the decisions on the shop floor, we will develop a discrete-event simulation. In material flow simulation, the logistics inside a production system are modeled to analyze the capacity of the factory and the production planning of the production with respect to efficiency, utilization, and in-time delivery, also considering failure situations. This simulation is able to take the output of production planning and compute its impact on the shop floor. To ensure fast computation (as required by RSim.8) over a large production planning horizon, the simulation performs the operational level decisions based on prioritization rules.

Data driven simulation The setup of these models usually requires a huge effort for both the data acquisition as well as the model generation itself [Rosen et al., 2020]. Using the data from the domain model, the simulation models should be generated automatically based on the factory layout, the production machinery, product and process specifications, and current order and production data. The behavior models can be taken from associated libraries of physical models combined with data-based models trained on historic production data. We will develop a wrapper for (semi-)automatic model setup from static production plant data available in the domain model. The user will be able to improve the automatically created simulation model by simple drag and drop of entities. To sum up, a simulation input (from the domain model) provides the necessary information for generating the simulation model, executing the simulation (simulator), and storing the simulation output data (for providing feedback into the domain model). To make the simulation applicable seamlessly inside the ASSISTANT AI and optimization algorithm, this wrapper will allow:

- the automatic model enrichment with dynamic production scenario data (decisions made by automatic planner and automatic scheduler). These simulation scenarios data is retrieved from the domain model.
- execution of a simulation run
- assessment and condensation of the simulation results. The simulation results are then stored in the domain model.

Figure 18 shows the input and output data for semi-automatic material flow simulation. This feature provides the requirements RSim.1, RSim.3, and RSim.5. Thanks to the flow of data coming from the real time digital twin into the data fabric, the data-driven simulation remain synchronized with the shop floor (e.g., it is aware of broken machines). Consequently, we may assume the simulation to be a perfect replicate of the shop floor. To provide requirement RSim.2, the simulation can be performed in a deterministic environment or it can account for stochastic parameters. To account for uncertain parameters, multiple simulation scenarios are generated with Monte Carlo methods.

To fulfill requirement RE.1, the simulator will only assign predefined shifts to the workers. The user will be responsible to define shifts that respect work regulations.

Integration The simulation will play different roles. First, it will generate training data for the scheduler's model acquisition since this data is not available in the use case. Currently, the use cases schedule the production with simple FIFO rules, and there is no record of the created schedule. For simplicity, we decided to record the schedule created by the simulation. Second, the simulation will communicate with the automatic planner to learn the capacity constraint functions. The objective is to have a tool to evaluate the feasibility of a production plan and to provide information on resource consumption and production lead time to improve the accuracy of the planner. In the static decision context, the planning module generates simulation input parameters for the entire planning horizon to be validated by the simulation. Feedback will be provided from the simulation module to the planning module, and this feedback will be used to learn the capacity constraint or production lead time. In a dynamic decision framework, given the updated information available in each period, dynamic planning decisions are selected in each period by the planning modules or by user interaction. To use the simulation in a dynamic decision framework, The planning module will send a request to simulate for a single period with

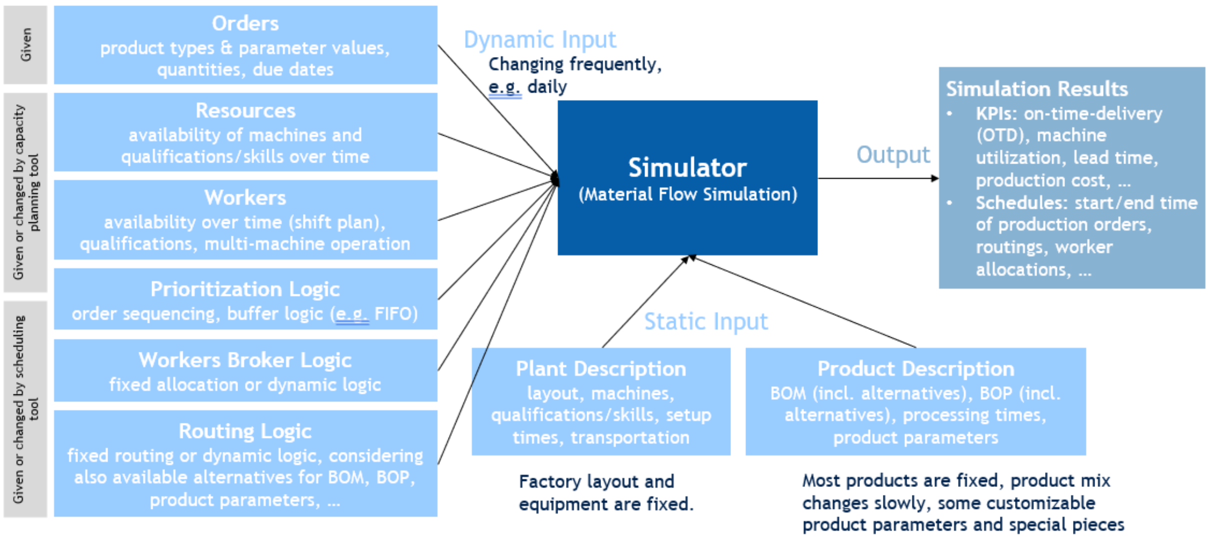


Figure 17: Data flow for the semi-automatic material flow simulation

a given input state and planning decisions. The simulation outputs the state at the end of the period. Third, we will evaluate the use simulation-optimization approach. In this context, the simulation provides a black box evaluation of schedules inside AI, black-box optimization, evolutionary algorithms, etc. Finally, the simulation will validate the plans and schedules proposed by production planning and scheduling components.

To facilitate the interface to the domain model, the production planning, and the production scheduling module, the simulator will provide APIs to input the simulation scenarios, and access the simulation results. Figure 18) shows the list of input data, and JSON or CSV are possible data exchange formats. Section 5.5.2 gives the detail of the communication between the simulation and the production planning module. Figure 18 provides a sequence diagram that represent a typical simulation run. The user can interact directly with the simulation to make manual decisions from scratch (i.e., the user create manually all the dynamic input described in Figure 18), or manually modify decisions suggested by the automatic planner. The human planner loads the data and decisions from the simulation interface, and it calls the simulation to visualize the results.

Technologies: We will rely on Tecnomatix Plant Simulation as an example for a proven tool on the market (and the preferred choice of SAG), SIMTalk as Tecnomatix Plant Simulation proprietary programming language for extended functionality inside the tool (to be used only very restrictively in ASSISTANT), and JSON or CSV for data exchange formats to the outside.

Component	Suggested technology
Programming language	Python
Simulation Tool	Tecnomatix Plant Simulation
Python libraries	tbd
Communication	REST API

Table 6: Technologies used in the simulation

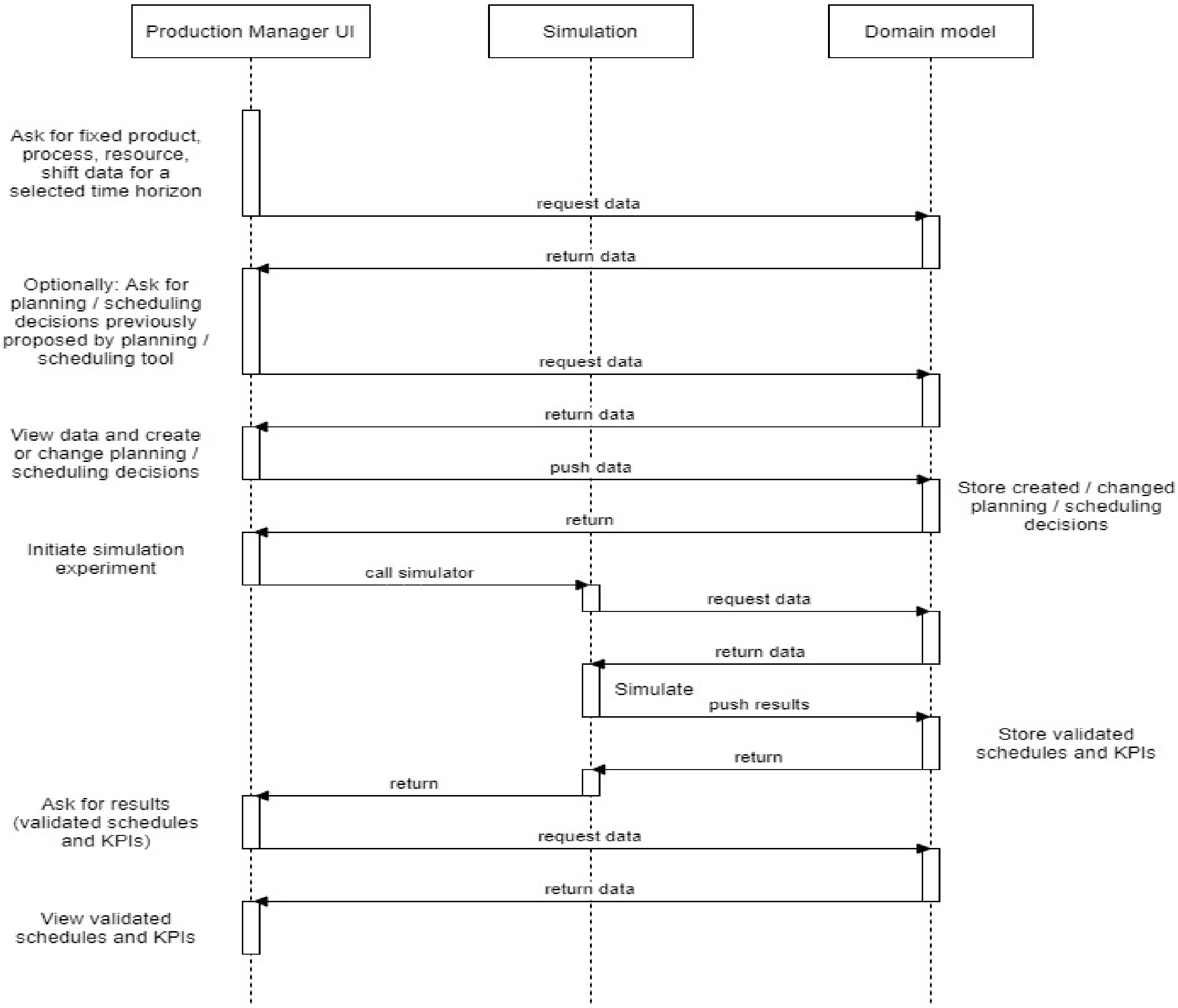


Figure 18: Sequence diagram for a simulation run

5.5 Production Planner

5.5.1 AI-based Probability learning

The AI-based probability learning module use the domain model to create a Bayesian network. The user selects the parameter to learn (unknown parameter) and the possible explanatory parameter (known). The Bayesian network is built from the relations in the domain model, and we learn the conditional probability with pair copula. In T4.4, this tool is adapted to learn the distribution of uncertain production planning parameters for material resource planning. The capacity uncertainty can be inferred from the machine breakdown represented by the mean time between failure, and mean failure duration. Assistant will seek to learn the distribution of this parameter from plans implemented in the past.

Figure 19 shows the sequence diagram that describes the communications between domain model and AI-based probability learner. The user can select the parameters he wants to learn (e.g., the demand for an item over a specific planning horizon) and the concepts that can explain the parameters to learn. The domain model will return the data and the link between the concept. These links can be used to build a Bayesian network before inferring probability distribution from historical data and store them in the domain model.

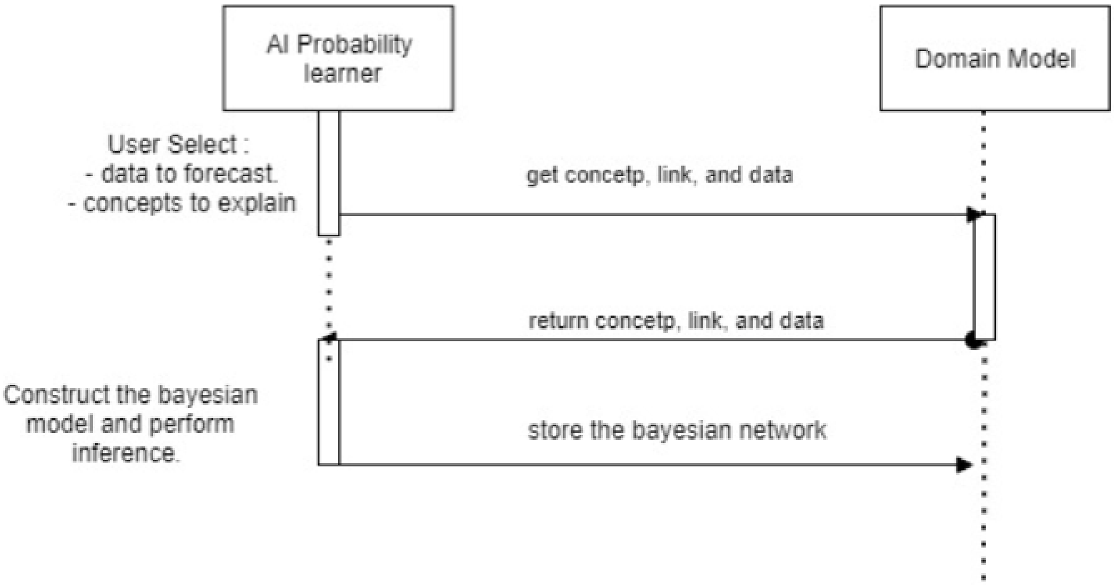


Figure 19: Sequence diagram for the interaction between the domain model and the AI-based probability learner.

5.5.2 Automated Production Planner

ASSISTANT's planner is a tool that automatically generates a production plan on a mid-term production planning horizon (e.g., few weeks, few months within 1 year) or a long-term planning horizon (1-2 years). That is, it automatically suggests when to produce/order as well as the sizes of the lots. The tool will help the planners to decide the requirements (make or buy decisions) and capacity planning (adjust the length of the shift/number of resources). The tool will account for uncertainty in demand and production capacity, to produce a robust production plan. In addition, we will use machine learning to better represent the capacity consumption and the production lead time.

Scope: The tool is based on a production planning model that considers aggregated items and resources with a granularity of a day, a week, or a month. To fulfill We will consider an extension of the multi-echelon multi-item capacitated lot-sizing problem (MMCLP) under uncertain parameters. The MMCLP is to suggest when to produce as well as the sizes of the production lots. The objective function is the expected total cost, and it includes inventory holding costs, setup costs, production costs, backlog costs, lost sale costs, and the extra capacity cost. This scope allows fulfilling requirement RP.1. To limit the scope of the project, The work presented here does not attempt to solve strategic capacity planning for master planning [e.g., Barahona et al., 2005] that aims to decide the number of production lines, etc.

Input data: The required input data for MMCLP are the demand, the bill of material, the production capacity, and the lead time. Note that this production planning model does not require any assumption on the type of shop floor (job shop, flow shop, etc.) because they consider aggregated data. The demand D_{it} for item i in period t can be represented with a parameter or a probability distribution. We assume (without loss of generality) that all customer demand is for end items only. If there exists a demand for components, we can create a dummy end-item corresponding to components reserved for shipping. The multi-echelon flexible bill of materials/processes gives the production structure of each item in the set I of items. We denote by I_e the set of end items and by I_c the set of components. Each item i can be acquired by alternative operations, and each operation o produces a_{oi} units of item i, it consumes b_{oi} units of component i, and consumes k_{or} units of resource r. Modelling operations leads to a very generic lot-sizing model that can include alternative production routing and make or buy decisions Begnaud et al. [2009]. The requirement plan must account for the production capacity. Each resource r in the set of resources R has a given capacity C_r. In each period t, the capacity of resource r can be expended, and each unit of extra capacity costs o_r. The component i produced in period t is available in period $t + L_i$, where L_i denotes the lead time of item i. This lead time may correspond to the time between the placement of an order to a supplier and its delivery, or to the number of periods between an order is released to the scheduler, and the period where the item is produced.

Output data: The tool will output the suggested production plan, including when to produce, how many items to produce, when to buy materials, and how many items to buy, and the amount of extra capacity required. More precisely, the tool will recommend a plan that includes:

- If a batch of operation o is performed in period t, and this is represented by a binary decision variable Y_{ot}.
- The quantity Q_{ot} of operation o to perform in period t.
- The amount e_{rt} of extra capacity required for resource r in period t.

Integration with the digital twin: The production planner will interact with the domain model and the simulation. These elements can enhance classical production planning approaches by providing data that can help enhance the accuracy of the model. Figure 20 shows the links between the production planner, the simulation, the scheduling tool, and the domain model. The *domain model* is the bridge between the physical system and the visual systems. It provides basic data from the physical system to the simulation model and production planning mathematical model, and feedbacks the production schedule to the physical system supporting decision-making. This data can help infer the value of the required parameters and the probability distributions of uncertain parameters. The *simulation models* validate the correctness of mathematical models, and make sure the production plan is implementable on the shop floor. The production planner will provide the production quantity per period to the simulation as well as the resource capacity. The simulation will inform on the possibility to adhere to the production plan. The simulation will tell if the capacity was exceeded on a specific resource, and at a specific period. The automated production planner will provide the size of the production batches to the *scheduler* as well as a targeted production period. In the scheduler, the release date corresponds to the start of the period, and the due date corresponds to the end of the period. The due date associated with non finished product in the scheduler is a soft due date (whereas violating the due date on end-item is heavily penalized), to ensure adherence to the production schedule, whereas the customer due date might be penalized strongly or even considered as hard deadlines. Note that the production planning module will communicate with the scheduler through the domain model.

The rest of this section describes how the uncertainty is incorporated into the production planning model, and how the communication between simulation and production planning model can help to learn the capacity constraint in the production planning model.

Uncertainty: To provide requirement RP.2, the tool will be able to deal with uncertain problems. In this work, we will develop stochastic optimization approaches to deal with uncertainties that are encountered by most companies. The parameter uncertainty must be described with a probability distribution given by the AI-based probability learner. Based on this distribution, the tool will generate a set of scenarios with Monte Carlo or advanced sampling approaches such as Quasi-Monte Carlo. For instance, uncertain demands can be represented by the set Ω of demand scenarios, where each scenario $\omega \in \Omega$ represents a possible realization of the demands over

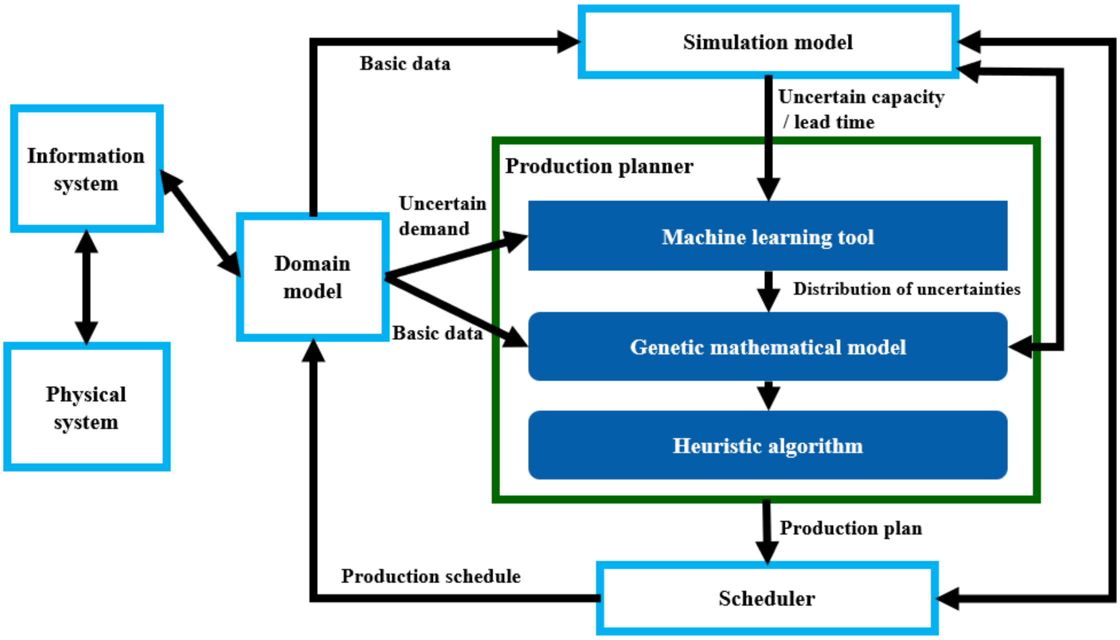


Figure 20: The integration of production planning in the digital twin.

the planning horizon, and it has a probability p_ω. The literature on production planning under uncertainty defines two different decision frameworks to deal with uncertain parameters. In the static decision framework, the decision is made for the entire planning horizon and they are frozen. On the contrary, in the dynamic decision framework, the production quantity in period t is decided dynamically after having observed the parameter in period $t-1$ (this leads to a multi-stage optimization problem). In the static-dynamic decision framework [Bookbinder and Tan, 1988], the setup decisions are frozen, whereas the production quantity is decided dynamically.

Uncertainty due to adherence to the plan in the scheduler/simulation: The planner releases a set of orders to the scheduler/simulation. In practice, due to the aggregation errors and the uncertainty in the capacity (machine breakdown) and process duration, the scheduler/simulation might not be able to follow the plan exactly (e.g., as simulation accounts for machine break down, we might discover that the suggested production qualities are too large to be produced in a period). From the point of view of the planner, in a *dynamic decision framework*, the non-adherence to the plan can be seen as yield uncertainty. The planner releases a production quantity in a period t, but the shop floor produces a different quantity. In period $t+1$, the planner observes the yield, and it decides the production quantities in period $t+1$. Note that the yield uncertainty depends on the decisions. If the production plan uses the capacity tightly, the scheduler will likely not be able to adhere to the plan. In addition, the planner could control yield uncertainty by providing a weight for each item to the scheduler (items with large weights are more important, and they are likely to be produced). However, decision-dependent probability leads to a non-convex stochastic program [Hellemo et al., 2018], which is hard to solve. We will investigate the best approach to solve this problem. In a *static decision framework* (or in the two-stage heuristic), the planner provides a plan for the entire horizon, and the simulation considers the production quantity as frozen (they are not modified dynamically). If the released orders are ultimately all planned, the non-adherence to the plan can be seen as stochastic lead time, where the lead time corresponds to the number of periods between the release of a production order to its production. For instance, if the production planner assumes the lead time is zero (i.e., a production order is performed in the same period it is released), the jobs will be released with a release date corresponding to the start of the period and a due date corresponding to the end of the period. The simulation might not be able to respect these release and due dates, and the actual lead time might differ from zero.

Note that when the simulation executes the plan, it may split a production order over several periods. In this case, the lead time may be considered as the time the last operation of the planned quantity is performed.

Learning the capacity constraint: To provide requirement RP.3, we aim to learn the capacity constraints in the mathematical model through machine learning based on the output of the simulation. The tool can run a simulation to get the capacity consumption associated with given production quantities. The challenge is to create a capacity constraint that is accurate but simple enough to solve the resulting model. The first step is to select the features to predict the capacity consumption. These features may include an estimate of the amount of each item in a product family and their estimated processing time, the number of alternative resources for an item, if two items or product families are produced the same day… We will use machine learning (ML) technics to learn the capacity constraint from these features. We will investigate linear regression, random forest. These ML methods lead to a trained model that can be translated to a mixed-integer linear program (MILP) Biggs and Hariss [2018] to predict if the capacity is violated or not. While these new constraints might increase the accuracy of the capacity consumption calculation, they also increase the number of variables and constraints in the production planning model. Therefore, it is important to select only the features that explain the most the actual capacity consumption. We might as well investigate if other types of constraints can be acquired in production planning (lead time, financial, …).

Optimization approaches: Mathematical optimization is the most appropriate tool for production planning. In fact, the lot-sizing models have attracted a lot of work from the operation research community. These researchers proposed several reformulations, cuts, and solution algorithms such as Lagrangian Relaxation, cutting planes,… However, the MILP approach does not scale well in the dynamic decision framework, where the production setups are updated as the information arises. The few works are limited to small-scale instances in simple environments. In fact, the simple MMCLP is NP-hard. We aim to solve large instances, with 5-10 echelons in the BOM, and a large planning horizon. To provide requirement RP.4, in the production planner, we will apply heuristic algorithms to get the optimal results within 30 min. [Thevenin et al., 2021] showed that the two-stage approximation provides a good heuristic to the static-dynamic decision framework when the demand is uncertain.

Figure 21 provides the sequence diagram for automated master planning. The user asks for the automated production planning from the Production manager UI where he can specify the various constraints on the plan. The planner will get the required data from the data fabric (domain model). Once the best plan is computed, the planner sends the production load per period (with their release date and due date) as well as the number of resources to the simulation. The simulation check that the plan is implementable, and it returns the actual capacity consumption per period. The planner improves the representation of the capacity function, and re-optimize. The loop continues until the plan is implementable in the simulation.

Finally, Table 7 provides the technologies we will use for the devlopment of the tool. We rely on Python for the numerous AI libraries available. To facilitate the exploitation of the result, we keep the solver selectable (depending on the business strategy a technology provider may prefer to rely on commercial or open-source solvers). However, we use CPLEX in our experiments since most benchmarks states it is one of the most efficient (Gurobi has similar performances).

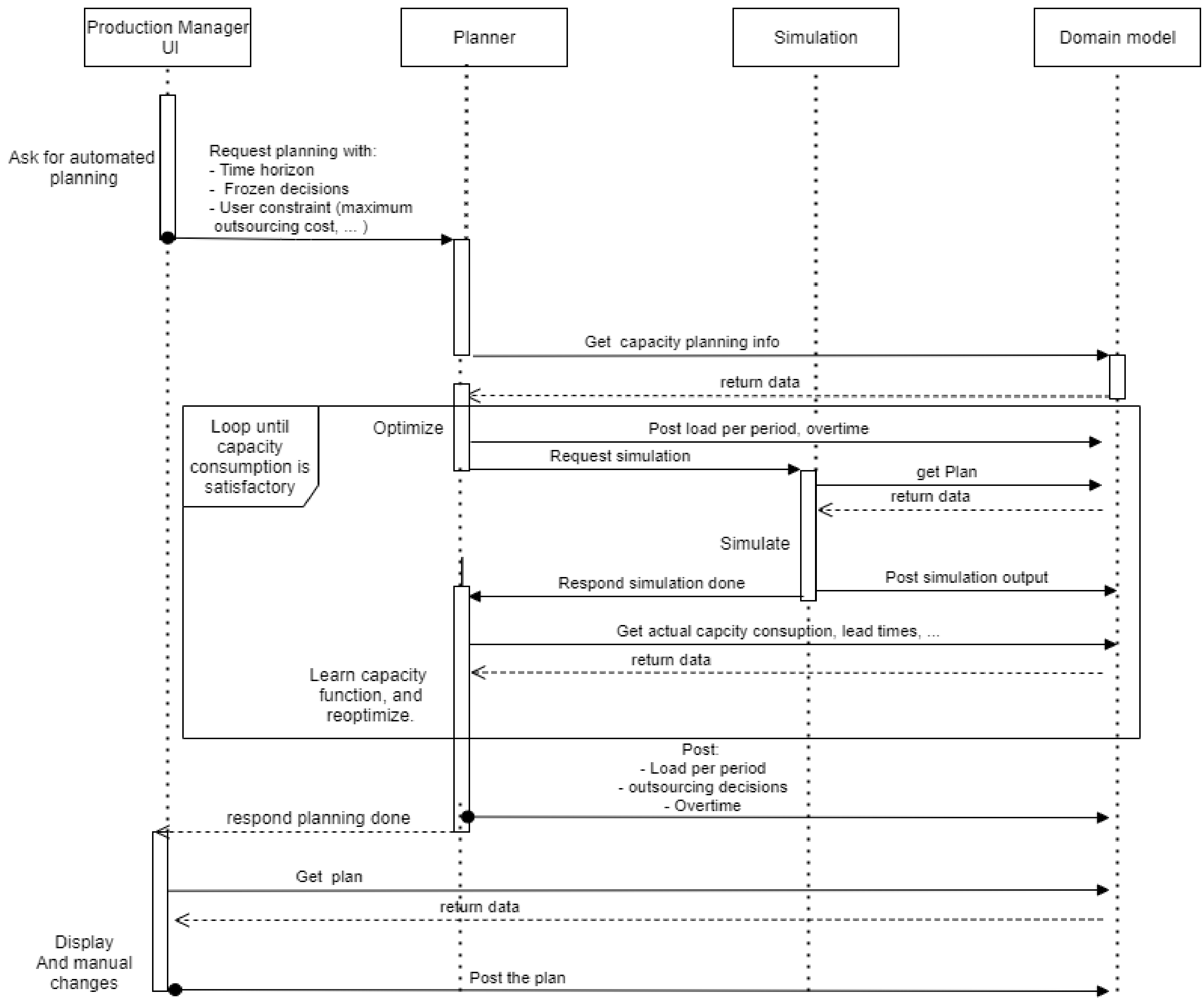


Figure 21: Sequence diagram for the interaction between the simulation and automatic planner.

Component	Suggested technology
Programming language	Python
Linear programming solver	Selectable, but experimentation will be carried out with CPLEX.
LP Modeling Library	Pulp
Sampling library	Lattice Builder
Python library	Numpy, Panda, Openpyxl, …
Communication	REST API

Table 7: Technologies used in the automated planner

5.6 Scheduler's optimizer and model acquisition

5.6.1 Scheduler's optimizer

ASSISTANT's scheduler is a tool that help production planners to generate a production schedule, and it will be provided for Flexible Job Shop scheduling (FJSP). The FJSP is to schedule a set of n jobs, where each job j consists of a set $\mathcal{O}_j = \{O_{j1} \dots O_{jm}\}$ of operations to perform on resource group $M_1 \dots M_m$, respectively. Each operation k of the entire set of operations $\mathcal{O}$ is associated

Component	Suggested technology
Programming language	C++ , Java
Solver	SICStus (and possibly solver interfaced to MiniZinc), IBM's CPOptimizer
Communication	REST API

Table 8: Technologies used in the scheduling optimizer

with a duration P_{kr} on machine $r \in \mathcal{R}$. In addition, each job j cannot start before its release date r_j, and it should ideally be completed before its due date d_j. Multiple industry specific constraints can be added to this model. The tool will rely on a constraint programming model solved with a solver.

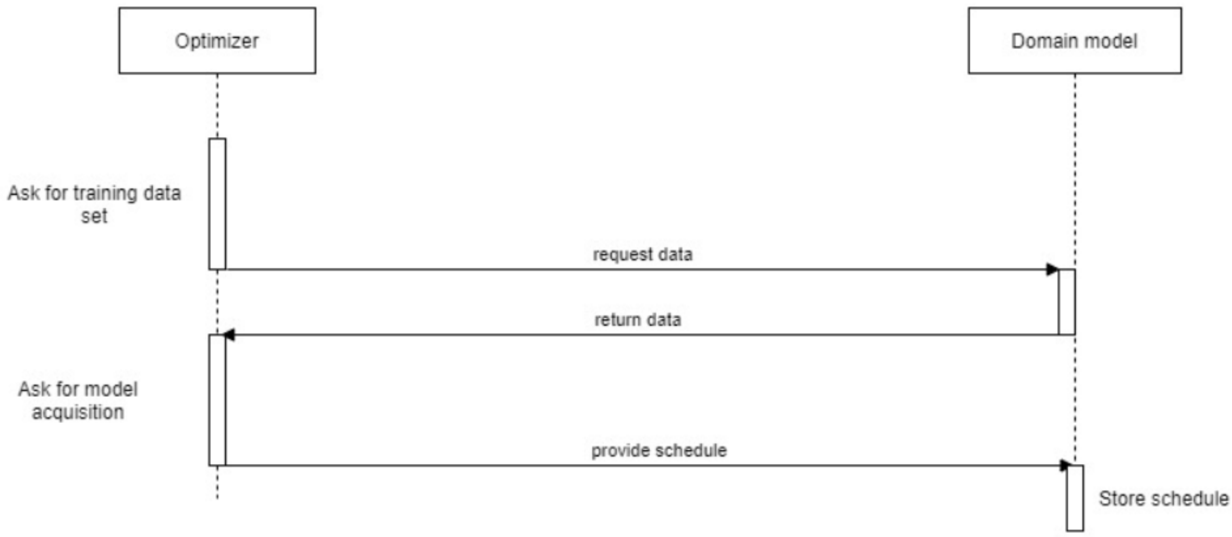


Figure 22: Sequence diagram for the interaction between the domain model and the scheduler.

Fig 22 provides the sequence diagram with the interaction between the optimizer and the domain model. The Optimizer will get the data to plan from the domain model. The input data will contain the same information as the information provided to build the model (but updated to current values). Running a solver will provide a production schedule that can be stored in the domain model.

Table 9 provides the technologies used to implement the solver. To provide requirement RSc.2, we will rely on the MiniZinc interface that can provide access to multiple state-of-the-art constraint programming solvers.

5.6.2 Model Acquisition

To fulfill RSc.1, the scheduling tool should be able to self-adapt to the production environment. Each manufacturing shop floor is unique, and an efficient production scheduling tool must account for the specific constraints encountered in a shop floor. The creation of the optimization model dedicated to a specific manufacturing shop is an expensive and cumbersome process. Modeling requires expertise in optimization and a good knowledge of business processes on the manufacturing shop floor. The optimization expert must discuss with the customers to understand the process. Since a shop floor manager usually has limited knowledge in optimization, the discussion is long and complicated, and additional requirements often arise once the model is ready,… We aim to simplify this process by automatically learning production scheduling parameters and constraints.

The model acquisition will acquire common constraints and parameter, such as:

- The formula to compute the processing time of operation on machine m.
- Machine eligibility restriction. An operation can only be performed by the subset of the set machines in each resource group.
- The formula to compute machine and sequence-dependent setup time on each resource.
- The minimum/maximum transition time (possibly 0) from one resource to the next.

Model acquisition will also acquire industry specific constraints, and this might include:

- The blocking constraints on resources. After processing an operation, some machine may not be able to process another operation until the operation starts on the next resource group. This constraint occurs because of the limited storage area between resource groups.
- The weight of each classical KPI in the objective function (setup costs, total tardiness, makespan, …
- A company-specific objective function.
- Tooling constraints (cumulative constraints): To process an operation, a machine might require a specific tool (or a worker with a specific skill) that comes in a limited number of exemplars. Often, the optimization model does not require to represent the tool occupation explicitly, because they are seldom blocking the processing. But the model must ensure that the number of tasks requiring tool t processed simultaneously is lower than the number of exemplars of t available on the shop floor. When each tool come in a single exemplar, then we have incompatibility constraints. Two operations o and o' are incompatible if they cannot be processed simultaneously.
- Limited raw material inventory. While the release date of the job might come from the production planning tools, they are also often associated with the arrival of the raw material and components required for production. In some cases, the processing of a job is not limited by a release date, but by the availability of raw material. More precisely, each operation consumes some units of raw material, and the production schedule must ensure that the amount of raw material consumed at any time is lower than the amount available at a time.
- Batching constraints. A batch is a set of items performed simultaneously on a resource (e.g., an oven). In manufacturing, the composition of a batch may be subject to various constraints (some parts might require a specific temperature, or a specific chemical environment).

The model acquisition is run only when the constraint programming model must be modified (change on the shop floor/first use/new product/…).

Input data: Model acquisition automatically build the scheduling model from data. From the model acquisition interface, the user selects the concept that he or she believes is required to compute the schedule. The user should limit the data to only use data (to avoid noise). The user also selects the relevant time frame to collect the past data, this time frame typically spans from the previous change on the shop floor (new product/new machine/…). These data may include:

- The schedule computed in the past (not actual data, but a plan made by hand, along with the data on product, machine, …) with the start and end date of the tasks on each resource. Ideally, the user should select plan schedule rather than the implemented one.
- Any data that can help to compute properties of the task

- The machines with their properties (disjunctive, setups, other characteristics, …)
- The task duration on each machine.
- The production load with the release date and due dates.
- The values of the KPIs in the previous schedule.

These inputs are given to the model acquisition as tables (that can be given in different formats including Json) describing feasible solutions of a production schedule. We may only provide one single solution. A table may describe one key element of the schedule such as the tasks, the resources as well as key parameters (e.g. cost setup). For instance, each entry of a table describing tasks typically gives the value of the different attributes of a task (e.g. identifier, start, duration, end, machine used, the quantity of resource used, type, earliest start, latest start, earliest end latest end, the tool used, the quantity of raw material used, quantity produced, due date, the cost for producing the task), while each entry of a table describing resources provides the values of the different attributes of a resource (e.g. identifier, type, speed, capacity, number of resources of this type, cost for opening the resource, cost of use per unit of time, unavailability periods, availability periods). For some problems, we may have one table describing renewable resources, as well as a table providing non-renewable resources. Note that the model acquisition should work even if there is information for a subset of the attributes. Besides providing such tables, we may have some metadata associated with the different tables providing information for interpreting a table and some of its entries:

- The main object is described by a table (e.g. tasks, machines).
- The attributes that should be computed by the solver, i.e. the output of the model, and the attributes that are fixed.
- The type of an attribute (cost, time point, duration, task, machine, type).

We list below the assumption about the possible inputs to interact with the model acquisition tool. We use here the word *possible* as not all these inputs could be possibly provided by the external modules. When some input is mandatory, we indicate that it is the case, but one should keep in mind that providing information that is not mandatory would help the model acquisition process.

1. **[input format: *some tables*] (mandatory)** We assume that the data is provided in some *table format* using one or several tables. Each table entry contains an integer value or a string. Dates should have been converted to some integer value.
2. **[one or several samples: *a Boolean and some marks*] (not mandatory)** One has to tell whether one provides *a single or several examples* from which the model acquisition tool should acquire a model. In the case of several examples (i.e. we give more than one schedule from which a model has to be acquired) one has to somehow mark these examples (by using, for instance, the first column of each table to indicate the sample number). By default, we may assume to have one single sample.
3. **[column interpretation: *one enumerated type for each column*] (not mandatory)** One may provide some information regarding the interpretation of each column. In the context of domain modelling, we may agree on a possible semantic for the information located in a given column of a table (e.g. instant, time interval, cost, amount of resource, temperature, speed, …). By doing this, we may better match a column with the argument of a scheduling constraint: the first argument of the disjunctive constraint expects instant, while it second argument expects intervals.
4. **[column names: *one string for each column*] (not mandatory)** To get an interpretable model, it would be good to provide some names for most columns: when such names are presented we will use them when reporting for a constraint found. If no name is provided, we may guess a potential name from 2) or just use the table name (if more than one table)

and the column number.

5. **[keys and functional dependency: *a set of keys and a set of functional dependencies*] (not mandatory)** We may also expect some information about the primary key (or compound keys) of the table, as well as columns whose value is a key referring to the key of some other table (e.g. in a table describing the tasks we have a column whose value represents the identifier of a resource the task uses; this identifier refers to the key of a table describing the different attributes of each resource). We may also get functional dependency (the fact that the value of a column of a table is uniquely determined by a subset of columns). In the context of functional dependency, we may ask for some way to restrict the scope of a functional dependency to a smaller subset of columns.
6. **[input and output of the generated optimization model: *a Boolean for each column of each table indicating whether that column is an input or an output*] (mandatory)** If we want to generate a model, we need to know what will be the input information and what will be the output computed by the model. For instance, in a scheduling problem, the start and end of each task will be typically computed by the model, and the task duration and task resource consumption may be fixed. Some other global parameters may be some implicit resource limit or the aggregated resource capacity (which is not given in the description of each resource), the number of tasks or the scheduling horizon. For solving a model we should assume that some incomplete set of tables are given (some columns will be empty, and running the model will provide values for such empty columns). Another point which is important to know is whether the tasks are preassigned to some resource (in this case this assignment will be an input of the model, otherwise the assignment of the tasks to the machines will also have to be found and we need to use a specific constraint: for instance, we will use the cumulative constraint if the tasks are preassigned to a resource and the cumulatives constraint if the tasks are not preassigned).

Output

The model acquisition provides a constraint programming model. We will investigate the possibility to validate these constraints by communication with different means (some metadata provided, an oracle, or a human). Here we assume that a human could potentially answer to a wide range of questions (but we should limit the number of questions we ask a human) or that an oracle is a program (e.g. a simulator) that could answer a fixed set of identified queries (but unlike a human, an oracle could answer a large number of queries). For instance, the model acquisition may ask to:

- Confirm it acquired the correct formulas (e.g., Task duration = speed · quantity)
- Confirm it model the resources appropriately (e.g., Machine M is a disjunctive/cumulative resource)
- Confirm that a set of elements belong to a subgroup (e.g., tasks $T_1, T_2, \ldots T_n$ run on the same machine group? Tasks $T_1, T_2, \ldots T_n$ are from the same job? Tasks $T_1, T_2, \ldots T_n$ were running on the same machine and at the same period?
- Ask questions about potential missing tasks (e.g., setup by a technician that was not described in the input data).

We list below the output and possible queries of the model acquisition tool:

1. **[an acquired model: *a set of constraints which was acquired given either as a MiniZinc model or as a SICStus program*] (mandatory)** In a preprocessing phase, the model acquisition tool will compute some constraints which are valid for a complete table (e.g., inequality which always holds for all rows of a table between two columns, constraint on the columns, …). As such constraints will be exploited to acquire further constraints, we

may ask whether these constraints are valid or are just artefacts.

2. **[validating some standard constraint on a full table: *a query asking whether a formula linking several columns of the tables is valid or not*] (not mandatory)** In a preprocessing phase, the model acquisition tool will compute some constraints which are valid for a complete table (e.g., inequality which always holds for all rows of a table between two columns, constraint on the columns, …). As such constraints will be exploited to acquire further constraints, we may ask whether these constraints are valid or are just artefacts.
3. **[confirming outliers table entries: *a query asking where some tables entries are incorrect or not*] (not mandatory)** When acquiring a formula that explains the values on each row of a given column with respect to the values of the same row of other columns we may find that some rows are outliers (on a specific row some data is wrong). We may ask to confirm this.
4. **[confirming the scope of a global constraint: *a query asking whether a global constraint linking several columns of the tables is valid or not*] (not mandatory)** When acquiring global constraints like for instance a chain of precedence, a disjunctive constraint one may ask whether the scope found for the constraint is correct or not (the scope being defined by some conditions on some attributes of the tasks: for instance, a set of disjunctive tasks will be identified both by the fact that they do not overlap, and that they are assigned to the same resource.
5. **[controlling the behaviour of the model acquisition tool: *a list of options interpretable by the model acquisition tool*] (not mandatory)** Some subparts of the model acquisition tool will be parametrized, for instance:
   (a) the module which acquires formulae can be parametrized to both specify which type of formula we may consider, as well as the order in which we should examine the different types of potential formulas. We may also specify the range of some coefficient of the formula, or the fact that some acquire inequalities should be sharp (or not).
   (b) the module which acquires global constraints in the context of scheduling (chain of precedence, disjunctive, cumulative, cumulatives, diffn, …) could also be parametrized to tell the subset of global constraints one may want to consider.
6. **[identifying constraints on implicit tasks: *a query asking whether a global constraint linking some functions derived from several columns of the tables is valid or not is valid or not*] (not mandatory)** If the tables we are given do not directly provide all information that is needed to identify some constraint of a model, we should have the possibility to somehow get such information. In the context of scheduling, a typical example would be the fact that some tasks (which have some kind of restrictions like maximum duration, needed resources) are never explicitly mentioned in the schedule; for instance, in a flow shop problem we may need to consider a task between the end of a given job on a machine and the start of the same job on the next machine; a second example also in the context of the flow shop would be that a job running on a certain machine may require some extra resource (e.g. a person) for doing some setup work during the first 15 minutes of the task. Even if we identify such candidate constraints, we should get some confirmation that they really hold (confirmation by a person or an oracle).

Basic steps of model acquisition: The model acquisition will consist of a set of modules that are processed sequentially:

- Step 1 will analyse each table in order to identify the types of the different attributes (e.g., Boolean, interval, set).
- Step 2 will search for functional dependency in each table: it will determine whether a given column of the table is functionally determined by a given subset of columns of the table.
- Step 3 will search for each identified functional dependency a formula which relate the

input parameters to the output parameter of the functional dependency. The formulae used will correspond to linear constraints but also to quadratic constraints as well as to constraints mentioning arithmetic operators such as min, max, mod, and div. We may also extract formulas corresponding to symbolic decision trees, where each internal node of a tree correspond to a condition on some attributes and each leave of the tree corresponds to a linear, quadratic or arithmetic formula. The motivation to acquire such symbolic decision trees is that some attribute value may be defined as a partition of cases where each case is described by a conditional part and by a formula. In the context of ASSISTANT (and scheduling), a typical example would be a situation where a cost or the value of an attribute of a task is a function defined by intervals: for instance the cost of a task may be defined by two distinct linear or quadratic functions, depending on whether the task terminates before some due date (in this case the cost first function encodes some storage cost), or after a due date (in this case the second function represents some penalty cost). For instance the duration of a task may be depend both of the period of the day where the task run (during working hours, during the night), but also on the machine to which the task is assigned.

- Step 4 will search for inequality constraints between columns of the table.
- Step 5 will interpret the content of each table.
- Step 6 will try to search for functional dependency constraints that can be inferred from two distinct tables. Note that cost constraints typically fall in this category where a cost may be determined from several tables. We illustrate this aspect with one example. For instance, the constraint giving the duration of a task t may be of the form $d[o] = m[o].s[m[o]]$, where $d[o]$ and $m[o]$ respectively correspond to the duration of a task o, and the machine to which task o is assigned, and $s[m[o]]$ to the speed of machine $m[o]$. In this example, the attributes $d[o]$ and $m[o]$ will be part of the table describing the tasks, while the attribute $s[m[o]]$ will be part of the table describing the resources.
- Step 7 will search for resource constraints. As in the model seeker, the idea is to project back the task on some attributes and check that each subset of projected tasks are for instance in disjunction.
- Step 8 will generate an executable model which can be run by a solver.

Figure 23 provides the sequence diagram for model acquisition. The users start by selecting relevant data by querying the domain model, and the model acquisition starts. Once a model is acquired it is validated by communication with the user. During T4.5, we will investigate if the validation questions can be answered by domain models, the simulation, or human (called an oracle in Figure 23). The possibility to answer these question with the simulation or domain model allow to ask a large number of questions, whereas asking a human require to reduce the number of questions, and make the questions understandable by a human. Beside answering the questions, the oracle may provide a pointer to data showing the assumption is not correct.

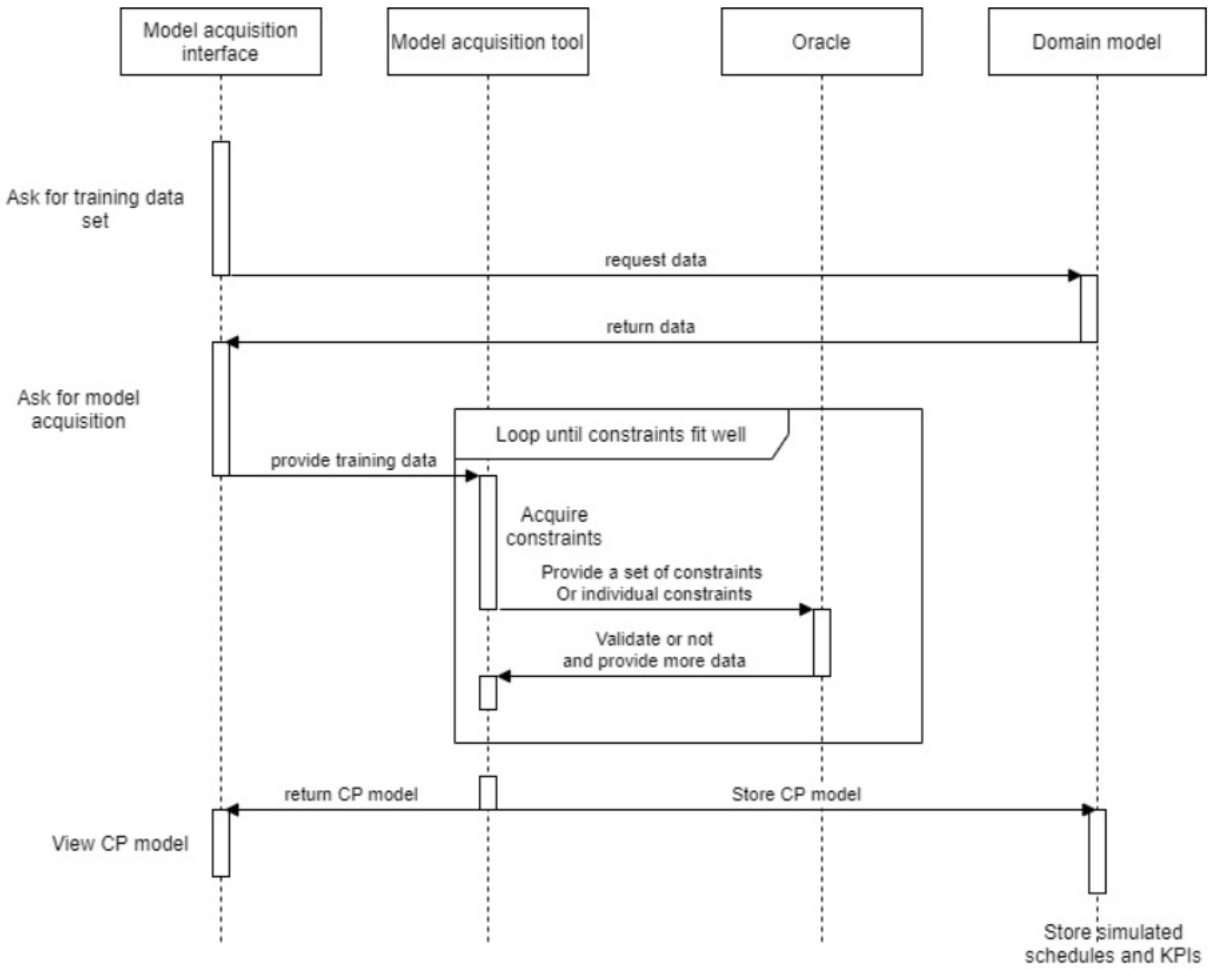


Figure 23: Sequence diagram for the model acquisition.

To enhance the already developed model seeker and make it applicable in a production scheduling context, the model acquisition tool of ASSISTANT will be developed in SICStus and Java. We may provide an interface to MiniZinc to access a wide range of Constraint Programming solvers. Nevertheless, IBM's CP Optimizer will be used to model some aspects of scheduling problems that are not covered by other Constraint Programming platforms, like optional or alternative process paths.

Component	Suggested technology
Programming language	Java
Model Acquisition	SICStus, Java
Solver	SICStus (and possibly solver interfaced to MiniZinc), IBM's CPOptimizer

Table 9: Technologies used in the scheduling optimizer

5.7 Use case validation

ASSISTANT promotes continuous integration in demonstrators and use cases. Therefore, the different tools will be integrated with other deliverables as soon as they are delivered. The deliverable will be released as follows:

- The digital twin will be delivered at month 18, and the contribution of WP4 to the digital twin includes the domain model and the simulation. With these elements, the use case will already be able to create production plans and schedules by hand and to validate the quality of these plans with the simulation.
- At month 24, we will release the model acquisition tool for scheduling, and the robust production planner. With these tools, the use case will be able to automatically generate a production plan and production schedule. However, the tools might not scale well, and they might not be able to solve large-scale instances.
- At month 30, we will release tools that include advanced optimization techniques to solve large-scale production planning and scheduling problems.
- At month 36, after use case validation, an updated version of the tools will be released.

To respect requirement RE.2, each use case will define a responsible for AI failure due to bad quality data, and a responsible for AI failure due to poor use of the tool by humans.

6 Conclusion

This document provides the road map to develop the intelligent digital twins for production planning and scheduling. Starting from interviews with use case providers, we built the requirements for the tools developed in WP4 by following the requirement engineering procedure. Based on these requirements, the partners involved in the work package agreed on the precise structure of the developed tools. The requirement analysis shows that the use of AI to improve the accuracy of decision models for production planning and scheduling is a promising area of research. On the one hand model acquisition can acquire complex scheduling problems automatically from data, and this technics can also be used to better learn capacity consumption in production planning. In addition, modeling uncertainties within such models will yield robust and flexible production plans. This document provides a rough description of the technology used and the communication required between the tools. The next step in the project is the development of these tools, starting with the domain model and the simulation, before implementing the planning module and scheduler.

7 Appendix

7.1 Abbreviations

Abbreviation	Meaning
ASSISTANT	LeArning and robuSt decision SupporT systems for agile mANu-facTuring environments
DT	Digital Twin
ML	Machine Learning
AI	Artificial Intelligence
APS	Advanced Planning System
MES	Manufacturing Execution System
ERP	Enterprise Resources Planning
MRP	Material Resources Planning
FIFO	First In First Out
IoT	Internet of Things
RFID	Radio-frequency identification
WP	Work Package
TX	Task X
DX	Deliverable X
MX	Month X
RSim	Requirement for simulation
RD	Requirement for data
RP	Requirement for production planner
RSc	Requirement for scheduling
API	Application Programming Interface
KPI	Key Performance indicator
UI	User Interface

Table 10: Abbreviations

7.2 Validated Data sets

The use cases provided several data sets:

- Small size data set that will be use to ensure all partners have a precised understanding of the use cases, and of the tools developed in the project. The small size data sets that will help for the development of the methodology and the implementation. To ensure a proper implementation of the communications between the tools, we provide input and output data sets.
- Realistic size data sets with full complexity that will be used to benchmark the performance of the tools.

At this early stage in the project, only the small size data sets are available. Larger data set will be provided when the tool are running, because at this point in time we will be sure of all the data we need and its format. Table 11 lists the provided data set. Note that the data may require some change once in used by the tool. Therefore, they will be published only once validated by a use in the softwares.

Data set name	Description
SE_smallsize_Plan_Input	This data set contains small size input data for production planning in the Siemens Energy use case
SE_small_Simulation_Input	This data set contains small size input data for the material flow simulation of Siemens Energy , the model acquisition, and the scheduler's optimization
AC_smallsize_Plan_Input	This data set contains small size input data for production planning in the Atlas Copco use case
AC_smallsize_Plan_Output	This data set contains an example of output that must be created automatically of the planning tool if AC_smallsize_Planning_Input is given as input.
AC_small_Simulation_Input	This data set contains small size input data for the material flow simulation, the model acquisition, and the scheduler's optimization of Atlas Copco
AC_Historical_Data	This data set contains historical data regarding the demand, and the lead time in Atlas Copco. This data set is the input for the AI-based probability learner.
Planning_Output_Model	Give the data model for the output of the planner
Simulation_Output_Model	Give the data model for the output of the simulation

Table 11: Data sets

7.3 Requirement elicitation methodology

Requirements Engineering Procedure
(Short document)

The confrontation with requirements happens almost daily in work and private life. Requirements are distinguished from wishes and goals by documenting them in writing or storing them electronically. Requirements are divided into different types. Commonly, they are categorized as functional and non-functional. In the context of identifying and dealing with requirements, existing publications identify a procedure for requirements engineering as shown in Figure 1. Procedural steps, which appear as main activities in existing literature, are marked in blue.

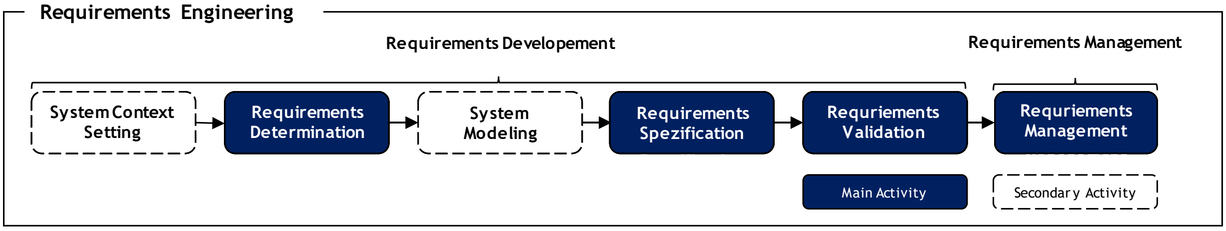


Figure 1: Procedure for Requirements Engineering

Requirements engineering describes the systematic procedure to design and administer requirements and aims at the creation of an efficient and error-free system. This paper focuses on requirements development because requirement management represents an ongoing activity during the implementation of the system. In the following, we explain the procedural steps for requirements development roughly. For some ASSISTANT tools, the system modeling is usually part of the first deliverable, whereas the system model of the data fabric is presented in a separate deliverable due to its complexity. Nevertheless, the procedure is done iterative and in parallel.

1. **System Context** is the part of the environment that is responsible for the definition as well as the understanding of the requirements. The system boundary is determined first, which determines, on the one hand, the system and determines on the other hand, which aspects are a component of the system environment. Subsequently, the context delimitation explains how the system environment is differentiated from the irrelevant environment, by describing relations to the system, which can be developed.
   In ASSISTANT, here we expect to see the scope of the system that should be developed, the link with other work-packages and its KPIs if possible.

2. **Requirements Determination** of the system requirements takes place in the second step. Techniques like brainstorming and document analysis help to identify and detail the requirements from stakeholders and other sources. For this purpose, preliminary requirements are first developed as notes. Traceability is of great importance for determining the requirements. The sources from which the requirements are determined must be evident.
   In ASSITANT, here we expect use cases and together with requirements specification all possible requirements that can cover the scope of the system and their acceptance criteria.

3. **Modeling** is the process of developing abstract models of a system. The architecture of the system is developed during requirements engineering to iteratively describe the requirements of a system during the system design phase.
   In ASSISTANT, here we expect diagrams that provide the components and the functionality of the system under study.

4. **Requirements Specification** is used to formulate the preliminary requirements that were identified during the requirements elicitation process. This is an important step in creating

transparency for all stakeholders. Formulation rules and blueprints can help document the requirements. In the course of requirements documentation, so-called bidirectional traceability must be ensured. This includes identifying which sources are responsible for defining the respective requirement and which development artifacts arose from which requirements.
In ASSITANT, here we expect clean requirements represented using table formalism as shown in the document Requirement_Engineering_Procedure (See Partage web platform).

5. **Requirements Validation**, during this activity, all requirements must be reviewed and agreed upon on time. In requirements validation, developers check whether the requirements are formulated as desired by the stakeholders.
   In ASSITANT, here we expect the validation of requirements based on acceptance criteria. Each validated requirement should be linked to corresponding tasks and deliverables in work package. In addition the model should be verified using the requirements developed.

6. **Requirements Management**
   Requirements management consists of different activities related to the management of the developed requirements. Traceability, impact analysis, cost estimation, risk management, requirements change, and variations are handled with these activities.
   In ASSITANT, here we expect continuous activity during project lifecycle for requirements management.

A detailed description of the requirements engineering procedure used is in Partage called Requirements_Engineering_Procedure.doc.

7.4 Interview Questions

Table 1: Machines related questions.

Nr	Question	Related WPs
Q2.1.1	What kind of machines do exist? (CAD available?)	WP5, WP3
Q2.1.2	Do machines located in the same machine area have the same capabilities?	WP3
Q2.1.3	Are you using a combination of old and new machines of the same type? if so, are there any deviations in the processing times?	all
Q2.1.4	What kind of capabilities do they have?	WP5, WP3
Q2.1.5	Are there any machines that need auxiliary resources?	WP3
Q2.1.6	What are the specific set-up times for the machines?	WP5
Q2.1.7	Are CAD files of the production system available? (Required input for process planning)	WP3

Table 2: Products related questions.

Nr	Question	Related WPs
Q2.2.1	What products do you produce?	WP5, WP3
Q2.2.2	What are the product variants / parameters?	WP5, WP3
Q2.2.3	What are the production steps (operations) to produce an item? How are they linked to machines and resources (BOP, BOM, mapping of steps to capabilities and machines, processing times, required worker skills ...)?	WP5, WP3
Q2.2.4	What is the number of production steps/operations for each product?	WP5, WP3
Q2.2.5	What are the characteristics of the steps? (duration depend on the resource/ duration depend on the hour/ precedence relationships, etc.)?	WP5, WP3
Q2.2.6	Are CAD files of the products, mockups, and components available? (Required input for process planning)	WP3

Table 3: Resource related questions.

Nr	Question	Related WPs
Q2.3.1.1	What kind of storage do you use? How is the storage managed? Any rules available (Based on allocation to orders/ production steps, transportation)? What is the availability,	
Q2.3.2.1	What means of transport do exist? Transport is manual, semi-automated or automated?	
Q2.3.2.2	What is the transport capacity?	
Q2.3.2.3	How does transport between certain steps / machines look like?	
Q2.3.2.4	How is the transport managed?	

	Any rules available? Are any workers required for transportation?	
Q2.3.3.1	How many individual operators are working under the same workstation on daily basis?	WP5, WP3
Q2.3.3.2	Are there different levels of responsibilities for operators that require a different level of information to be provided?	WP5, WP3
Q2.3.3.3	Preferable devices for human operators' interface: a) Smartphone, b) Tablet / Big touch screens, c) Smart watches/ bands, d) AR glasses (answer by placing in order from the most preferable to the less preferable)	WP5
Q2.3.3.4	Do you have currently a process for tracking human operators' errors? If yes, please could you provide more details?	WP5
Q2.3.3.5	Please indicate the daily work plan of an operator. (e.g., number of shifts, number of breaks in a shift, etc.)	WP5
Q2.3.3.6	How often new working procedures are introduced to an operator? (every X week, every X months)	WP5
Q2.3.3.7	Would be possible to conduct interviews with some operators of the factory?	WP5
Q2.3.3.8	Would be possible, later in the project, to perform some validation tests of the developments with some operators of the factory?	WP5
Q2.3.3.9	What is the current process for informing/training a worker about his/her activities on a workstation?	WP5
Q2.3.3.10	Do you believe an interface that uses voice commands would be useful given the noise in a shopfloor?	WP5
Q2.3.3.11	Which different skills exist between workers? Who is able to perform which process step (which categories)?	WP3
Q2.3.4.1	Do you have currently a process for tracking robots' failures? If yes, please could you provide more details?	WP5
Q2.3.4.2	How long does it take to program a robot?	WP5
Q2.3.5.1	How many workers/robots do you have, or you can have?	WP5, WP3
Q2.3.5.2	How can workers/Robots be characterized (availability, skills, allocation to machines / transportation)?	WP5, WP3
Q2.3.5.3	How dynamically can workers/Robots be allocated to machines? Can workers/Robots move among stations? If yes when they can move? Do you consider tasks re-assignment among stations? Are there any other types of workers/Robots, i.e., temporary workers? How do you react to workers/Robots unavailability? Does it impact on process time or other parameters?	WP5
Q2.3.6.1	How many sensors do you have? What kind of data they are collecting? Are there any sensors for quality control? Do they monitor both manual and automated operations? Do you have online or offline monitoring?	WP5

Table 4: Factory and process related questions.

Nr	Question	Related WPs
Q2.4.1	What is the factory layout? Are CAD Files available?	WP5, WP3
Q2.4.2	Would you characterize the workstation as a noisy place?	WP5
Q2.4.3	What is the available capacity? Can you adjust your production capacity each day/week/month depending on the load? Is the total amount of inventory limited? How many end-items are currently produced on your shop floor? How many levels are there in typical Bill of Material for the items produced in your factory?	WP5
Q2.4.4	How can production be characterized (job shop, flow shop, batch production, lot-size-1 production, …)? Assembly or manufacturing? High mix low volume or low volume high mix or both? Is the production a make to order or a make to stock?	WP3
Q2.4.5	What kind of control strategies/dispatching rules are implemented at the machines (FIFO, ...)?	
Q2.4.6	How do the order processes / order lists look like? What data do they contain (type, quantity, due date, …)?	
Q2.4.7	In what scheme product are released in the factory?	
Q2.4.8	Products are produced individually, or they are used in a mix? If yes are there any constraints to consider?	
Q2.4.9	Is there a specific bottleneck on your shop floor?	
Q2.4.10	For a given operation, is there a single machine that can perform this operation, or can the processing machine be selected? Are there different types of tasks, i.e., manual (need worker), automated (need only robot), hybrid (need both worker and robot)? How is the interaction between worker and robot for hybrid tasks/stations).	WP5
Q2.4.11	What are the characteristics of the tasks? (duration depend on the resource/ duration depend on the hour/ precedence relationships...)	WP5
Q2.4.12	What granularity are you using for planning your production (a day/week/ a month)?	
Q2.4.13	How are current process plans created?	
Q2.4.14	Are there alternative process plans currently in place for short-term adaptation?	
Q2.4.15	What do the current process plans include?	
Q2.4.16	On factory level what feedback data is currently being recorded? Quality, costs, time, etc.? How detailed are these recorded?	
Q2.4.17	On factory level what change data is currently being recorded? Product changes? Changes to the production system (personnel, resources, etc.)?	
Q2.4.18	What and where do buffers exist?	

Table 5: Use of Artificial Intelligence related questions.

Nr	Question	Related WPs
Q2.5.1	What are your expectations regarding the use of AI in process planning?	WP5, WP3
Q2.5.2	What are your fears regarding the use of AI in process planning?	WP5, WP3
Q2.5.3	What are your expectations regarding the use of AI in scheduling/production planning?	WP5
Q2.5.4	What are your fears regarding the use of AI in scheduling/production planning?	WP5

Table 6: Simulation related questions.

Nr	Question	Related WPs
Q3.1	What is the role of simulation in production planning and scheduling so far?	WP5
Q3.2	What is the role of simulation in process planning so far?	WP5, WP3
Q3.3	What kind of simulation are you doing so far (material flow, 3D simulation, process simulation, …)?	WP5, WP3
Q3.4	What type of simulation are you doing (multi-agents, discrete event? Dynamic system?)	
Q3.5	What is the level of detail?	WP5, WP3
Q3.6	What tools are you using?	WP5, WP3
Q3.7	If you are using simulation, how long does it take to run a simulation (e.g., to generate data for learning)?	WP5
Q3.8	What are the existing data interfaces that could be used for feeding /synchronizing the digital twin (e.g., SAP export)?	WP5, WP3
Q3.9	What are the inputs and outputs of the simulation today? Order lists, shift schedules, etc.? What are the most relevant KPIs?	WP5
Q3.10	What simulation experiments are executed? How are they generated? What are the degrees of freedom?	WP5
Q3.11	Who is doing the simulation today?	WP5

Table 7: Production planning related questions.

Nr	Question
Q5.1	What tools are currently used for production planning and scheduling?
Q5.2	What are the issues/ limitations with the current tools?
Q5.3	What are the main issues/ main area of improvement regarding operation management in the shop floor?
Q5.4	What is the expectation from a scheduling/production planning software (functionality/computation time/number of tasks /…)?

Q5.5	What are the main sources of uncertainty on your shop floor? How does it impact production planning and scheduling?
Q5.6	What is the main objective of mid-term planning horizon in your factory?
Q5.7	What KPIs do you use for production planning and scheduling?
Q5.8	How is production planning validated? How scheduling is validated?
Q5.9	Is there any integration between the production planning and scheduling? How consistent between production planning and scheduling is ensured? What kind of scheduling do you use? (Global or local scheduling?)
Q5.10	Do you make use or do you intend to make use of a domain model for production planning and scheduling? If so, 1. do you follow a standard (e.g., ISA-95)? 2. what is its format (ontology, entity-relation model, etc.)? 3. how do you link the data 4. do you have or do you see a need for a company specific domain model? 5. who are the users (people or IT processes) of your domain model?
Q5.11	What is the complexity of the information you currently need to solve a production planning or scheduling problem? Describe which data sources are involved (databases, PLM system, Excel files, etc.), and how many concepts (input parameters to the problem) are used.
Q5.12	How is the current process streamlined to acquire all necessary input data for production planning and scheduling (data curation, aggregation, etc.)?

References

H.-M. Afsar, O. Ben-Ammar, A. Dolgui, and F. Hnaien. Supplier replacement model in a one-level assembly system under lead-time uncertainty. *Applied Sciences*, 10(10):3366, 2020.

E.-H. Aghezzaf, C. Sitompul, and N. M. Najid. Models for robust tactical planning in multi-stage production systems with uncertain demands. *Computers & Operations Research*, 37(5):880–889, 2010.

Í. R. S. Agostino, E. Broda, E. M. Frazzon, and M. Freitag. Using a digital twin for production planning and control in industry 4.0. In *Scheduling in Industry 4.0 and Cloud Manufacturing*, pages 39–60. Springer, 2020.

A. Alenezi, S. A. Moses, and T. B. Trafalis. Real-time prediction of order flowtimes using support vector regression. *Computers Operations Research*, 35(11):3489–3503, 2008.

T. Aouam, K. Geryl, K. Kumar, and N. Brahimi. Production planning with order acceptance and demand uncertainty. *Computers & Operations Research*, 91:145–159, 2018.

M. Aouchiche, G. Caporossi, P. Hansen, and M. Laffay. Autographix: a survey. *Electron. Notes Discret. Math.*, 22:515–520, 2005. doi: 10.1016/j.endm.2005.06.090. URL https://doi.org/10.1016/j.endm.2005.06.090.

K. Ashton et al. That 'internet of things' thing. *RFID journal*, 22(7):97–114, 2009.

L. Atzori, A. Iera, and G. Morabito. The internet of things: A survey. *Computer networks*, 54 (15):2787–2805, 2010.

R. Babukartik and P. Dhavachelvan. Hybrid algorithm using the advantage of aco and cuckoo search for job scheduling. *International Journal of Information Technology Convergence and Services*, 2(4):25, 2012.

P. Baptiste, C. Pape, and W. Nuijten. *Constraint-Based Scheduling: Applying Constraint Programming to Scheduling Problems*, volume 39. 01 2001. ISBN 978-1-4613-5574-8. doi: 10.1007/978-1-4615-1479-4.

F. Barahona, S. Bermon, O. Günlük, and S. Hood. Robust capacity planning in semiconductor manufacturing. *Naval Research Logistics (NRL)*, 52(5):459–468, 2005.

BARC. The planning survey 16, 2016. https://barc.de/uploads/static/files/Infografik THE Planning Survey 2016_v2.pdf.

J. Begnaud, S. Benjaafar, and L. A. Miller. The multi-level lot sizing problem with flexible production sequences. *IIE Transactions*, 41(8):702–715, 2009.

N. Beldiceanu and H. Simonis. Modelseeker: Extracting global constraint models from positive examples. In *Data Mining and Constraint Programming - Foundations of a Cross-Disciplinary Approach*, pages 77–95. 2016. doi: 10.1007/978-3-319-50137-6_4. URL https://doi.org/10.1007/978-3-319-50137-6_4.

N. Beldiceanu, A. Dolgui, C. Gonnermann, G. Castañé, N. Kousi, B. Meyers, J. Prud'homme, S. Thevenin, E. Vyhmeister, and P.-O. Östberg. Assistant: Learning and robust decision support system for agile manufacturing environments. In *17th IFAC Symposium on Information Control Problems in Manufacturing (INCOM)*. IFAC, 2021.

L. A. Bewoor, V. C. Prakash, and S. U. Sapkal. Production scheduling optimization in foundry using hybrid particle swarm optimization algorithm. *Procedia Manufacturing*, 22:57–64, 2018.

F. Biesinger, D. Meike, B. Kraß, and M. Weyrich. A digital twin for production planning based on cyber-physical systems: A case study for a cyber-physical system-based creation of a digital twin. *Procedia CIRP*, 79:355–360, 2019.

M. Biggs and R. Hariss. Optimizing objective functions determined from random forests. *Available at SSRN 2986630*, 2018.

M. Bonney. Reflections on production planning and control (ppc). *Gest ao produç ao*, 7(3): 181–207, 2000.

J. H. Bookbinder and J.-Y. Tan. Strategies for the probabilistic lot-sizing problem with service-level constraints. *Management Science*, 34(9):1096–1108, 1988.

J. Brence, L. Todorovski, and S. Dzeroski. Probabilistic grammars for equation discovery. *Knowl. Based Syst.*, 224:107077, 2021. doi: 10.1016/j.knosys.2021.107077. URL https://doi.org/10.1016/j.knosys.2021.107077.

A. F. Bueno, F. Godinho, M. Frank, and A. Germán. Smart production planning and control in the industry 4.0 context: A systematic literature review. *Computers Industrial Engineering*, 106774, 2020.

L. Buschkühl, F. Sahling, S. Helber, and H. Tempelmeier. Dynamic capacitated lot-sizing problems: a classification and review of solution approaches. *Or Spectrum*, 32(2):231–261, 2010.

J. P. U. Cadavid, S. Lamouri, B. Grabot, R. Pellerin, and A. Fortin. Machine learning applied in production planning and control: a state-of-the-art in the era of industry 4.0. *Journal of Intelligent Manufacturing*, 31:1–28, 2020.

M. Carvalho, J. P. Pedroso, C. Telha, and M. Van Vyve. Competitive uncapacitated lot-sizing game. *International Journal of Production Economics*, 204:148–159, 2018.

J. S. Correa, M. Sampaio, R. d. C. Barros, and W. d. C. Hilsdorf. Iot and bda in the brazilian future logistics 4.0 scenario. *Production*, 30, 2020.

A. L. Cunha, M. O. Santos, R. Morabito, and A. Barbosa-Póvoa. An integrated approach for production lot sizing and raw material purchasing. *European Journal of Operational Research*, 269(3):923–938, 2018.

K. Ding, F. T. S. Chan, X. Zhang, G. Zhou, and F. Zhang. Defining a digital twin-based cyber-physical production system for autonomous manufacturing in smart shop floors. *Journal of Ambient Intelligence and Humanized Computing*, 57(20):6315–6334, 2019.

A. Dolgui and C. Prodhon. Supply planning under uncertainties in mrp environments: A state of the art. *Annual reviews in control*, 31(2):269–279, 2007.

C. Fang, X. Liu, P. M. Pardalos, and J. Pei. Optimization for a three-stage production system in the internet of things: procurement, production and product recovery, and acquisition. *The International Journal of Advanced Manufacturing Technology*, 83(5-8):689–710, 2016.

G. Figueira and B. Almada-Lobo. Hybrid simulation–optimization methods: A taxonomy and discussion. *Simulation Modelling Practice and Theory*, 46:118–134, 2014.

J. Galbraith. Designing complex organizations. *Reading, Mass*, 1973.

M. Garetti and M. Taisch. Neural networks in production planning and control. *Production Planning Control*, 10(4):324–339, 1999.

E. Geisberger and M. Broy. *agendaCPS: Integrierte Forschungsagenda Cyber-Physical Systems*. Springer-Verlag, 1, 2012.

C. Giacomello, Z. Kapelan, and M. Nicolini. Fast hybrid optimization method for effective pump scheduling. *Journal of Water Resources Planning and Management*, 139(2):175–183, 2013.

A. Gonzalez-Vidal, F. Jimenez, and A. F. Gomez-Skarmeta. A methodology for energy multivariate time series forecasting in smart buildings based on feature selection. *Energy and Buildings*, 196:71–82, 2019.

D. Gyulai, A. Pfeiffer, G. Nick, V. Gallina, W. Sihn, and L. Monostori. Lead time prediction in a flow-shop environment with analytical and machine learning approaches. *IFAC-PapersOnLine*, 51(11):1029–1034, 2018.

Y. Hajjaji, W. Boulila, I. R. Farah, I. Romdhani, and A. Hussain. Big data and iot-based applications in smart environments: A systematic review. *Computer Science Review*, 39:100318, 2021.

S. Helber. Lot sizing in capacitated production planning and control systems. *Operations-Research-Spektrum*, 17(1):5–18, 1995.

S. Helber and F. Sahling. A fix-and-optimize approach for the multi-level capacitated lot sizing problem. *International Journal of Production Economics*, 123(2):247–256, 2010.

L. Hellemo, P. I. Barton, and A. Tomasgard. Decision-dependent probabilities in stochastic programs with recourse. *Computational Management Science*, 15(3):369–395, 2018.

F. Hnaien, A. Dolgui, and D. D. Wu. Single-period inventory model for one-level assembly system with stochastic lead times and demand. *International Journal of Production Research*, 54(1): 186–203, 2016.

Y.-M. Hwang, M. G. Kim, and J.-J. Rho. Understanding internet of things (iot) diffusion: Focusing on value configuration of rfid and sensors in business cases (2008–2012). *Information Development*, 32(4):969–985, 2016.

D. Ivanov and A. Dolgui. A digital supply chain twin for managing the disruption risks and resilience in the era of industry 4.0. *Production Planning Control*, pages 1–14, 2020.

D. Ivanov, A. Dolgui, A. Das, and B. Sokolov. Digital supply chain twins: Managing the ripple effect, resilience, and disruption risks by data-driven optimization, simulation, and visibility. In *Handbook of ripple effects in the supply chain*, pages 309–332. Springer, 2019.

D. Ivanov, B. Sokolov, and A. Dolgui. *Scheduling in Industry 4.0 and Cloud Manufacturing*. 1–9, Springer, 2020.

F. R. Jacobs, W. L. Berry, D. C. Whybark, and T. E. Vollmann. *Manufacturing Planning and Control for Supply Chain Management: APICS/CPIM Certification Edition*. McGraw-Hill, Education, 2011.

J. Y. Jung, G. Blau, J. F. Pekny, G. V. Reklaitis, and D. Eversdyk. A simulation based optimization approach to supply chain management under demand uncertainty. *Computers & chemical engineering*, 28(10):2087–2106, 2004.

S. H. Kim and Y. H. Lee. Synchronized production planning and scheduling in semiconductor fabrication. *Computers & Industrial Engineering*, 96:72–85, 2016.

P. Köchel and S. Thiem. Search for good policies in a single-warehouse, multi-retailer system by particle swarm optimisation. *International journal of production economics*, 133(1):319–325, 2011.

S. Kolb, S. Paramonov, T. Guns, and L. D. Raedt. Learning constraints in spreadsheets and tabular data. *Mach. Learn.*, 106(9-10):1441–1468, 2017. doi: 10.1007/s10994-017-5640-x. URL https://doi.org/10.1007/s10994-017-5640-x.

K. Korpela, J. Hallikas, and T. Dahlberg. Digital supply chain transformation toward blockchain integration. In *proceedings of the 50th Hawaii international conference on system sciences*, 2017.

N. Kourentzes. Intermittent demand forecasts with neural networks. *International Journal of Production Economics*, 143(1):198–206, 2013.

N. Kourentzes, D. K. Barrow, and S. F. Crone. Neural network ensemble operators for time series forecasting. *Expert Systems with Applications*, 41(9):4235–4244, 2014.

L. R. Kroer, K. Foverskov, C. Vilhelmsen, A. S. Hansen, and J. Larsen. Planning and scheduling operating rooms for elective and emergency surgeries with uncertain duration. *Operations research for health care*, 19:107–119, 2018.

M. Kück and M. Freitag. Forecasting of customer demands for production planning by local k-nearest neighbor models. *International Journal of Production Economics*, 231:107837, 2021.

M. Kumar, S. Teso, and L. D. Raedt. Acquiring integer programs from data. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 1130–1136, 2019a. doi: 10.24963/ijcai.2019/158. URL https://doi.org/10.24963/ijcai.2019/158.

V. R. S. Kumar, A. Khamis, S. Fiorini, J. L. Carbonera, A. O. Alarcos, M. Habib, P. Goncalves, H. Li, and J. I. Olszewska. Ontologies for industry 4.0. *The Knowledge Engineering Review*, 34, 2019b.

K. Lamba and S. P. Singh. Big data in operations and supply chain management: current trends and future perspectives. *Production Planning Control*, 28(11-12):877–890, 2017.

C. E. Larson and N. V. Cleemput. Automated conjecturing I: fajtlowicz's dalmatian heuristic revisited. *Artif. Intell.*, 231:17–38, 2016. doi: 10.1016/j.artint.2015.10.002. URL https://doi.org/10.1016/j.artint.2015.10.002.

L. L. Lim, G. Alpan, and B. Penz. A simulation-optimization approach for sales and operations planning in build-to-order industries with distant sourcing: Focus on the automotive industry. *Computers & Industrial Engineering*, 112:469–482, 2017.

S. J. Lim, S. J. Jeong, K. S. Kim, and M. W. Park. A simulation approach for production-distribution planning with consideration given to replenishment policies. *The International Journal of Advanced Manufacturing Technology*, 27(5-6):593–603, 2006.

F. Lin, M. C. Wong, and M. Ge. Development of the digital model of the jewellery production process for resource optimisation and prediction. *HKIE Transactions*, 25(4):229–236, 2018.

L. Lingitz, V. Gallina, F. Ansari, D. Gyulai, A. Pfeiffer, W. Sihn, and L. Monostori. Lead time prediction using machine learning algorithms: A case study by a semiconductor manufacturer. *Procedia Cirp*, 72:1051–1056, 2018.

F. G. Listl, J. Fischer, D. Beyer, and M. Weyrich. Knowledge representation in modeling and simulation: A survey for the production and logistic domain. In *2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, volume 1, pages 1051-1056. IEEE, 2020.

J. Liu, C. Li, F. Yang, H. Wan, and R. Uzsoy. Production planning for semiconductor manufacturing via simulation optimization. In *Proceedings of the 2011 winter simulation conference (WSC)*, pages 3612–3622. IEEE, 2011.

W. K. Liu, G. Karniadakis, S. Tang, and J. Yvonnet. A computational mechanics special issue on: data-driven modeling and simulation—theory, methods, and applications, 2019.

F. Lolli, E. Balugani, A. Ishizaka, R. Gamberini, B. Rimini, and A. Regattieri. Machine learning for multi-criteria inventory classification applied to intermittent demand. *Production Planning Control*, 30(1):76–89, 2019.

L. L. Lorente-Leyva and M. M. E. a. Alemany. and peluffo-ordó nez. In H. Diego and I. D. Herrera-Granda, editors, *A Comparison of Machine Learning and Classical Demand Forecasting Methods: A Case Study of Ecuadorian Textile Industry*, pages 131–142, International Conference on Machine Learning, Optimization, and Data Science, 2020.

M.-A. Louly and A. Dolgui. Optimal mrp parameters for a single item inventory with random replenishment lead time, poq policy and service level constraint. *International Journal of Production Economics*, 143(1):35–40, 2013.

C.-J. Lu. Sales forecasting of computer products based on variable selection scheme and support vector regression. *Neurocomputing*, 128:491–499, 2014.

Y. Lu, C. Liu, I. Kevin, K. Wang, H. Huang, and X. Xu. Digital twin-driven smart manufacturing: Connotation, reference model, applications and research issues. *Robotics and Computer-Integrated Manufacturing*, 61(10183):7, 2020.

T. Luna, J. Ribau, D. Figueiredo, and R. Alves. Improving energy efficiency in water supply systems with pump scheduling optimization. *Journal of cleaner production*, 213:342–356, 2019.

Y. Meidan, B. Lerner, G. Rabinowitz, and M. Hassoun. Cycle-time key factor identification and prediction in semiconductor manufacturing using machine learning and data mining. *IEEE transactions on semiconductor manufacturing*, 24(2):237–248, 2011.

H. Mokhtari and A. Hasani. An energy-efficient multi-objective optimization for flexible job-shop scheduling problem. *Computers & Chemical Engineering*, 104:339–352, 2017.

J. Mori and V. Mahalec. Planning and scheduling of steel plates production. part i: Estimation of production times via hybrid Bayesian networks for large domain of discrete variables. *Computers Chemical Engineering*, 79:113–134, 2015.

N. Niknejad, W. Ismail, I. Ghani, B. Nazari, M. Bahari, et al. Understanding service-oriented architecture (soa): A systematic literature review and directions for further investigation. *Information Systems*, 91:101491, 2020.

S. Noroozi and J. Wikner. Sales and operations planning in the process industry: A literature review. *International Journal of Production Economics*, 188:139–155, 2017. ISSN 0925-5273. doi: https://doi.org/10.1016/j.ijpe.2017.03.006. URL https://www.sciencedirect.com/science/article/pii/S092552731730066X.

F. Oliveira, S. Hamacher, and M. R. Almeida. Process industry scheduling optimization using genetic algorithm and mathematical programming. *Journal of Intelligent Manufacturing*, 22 (5):801–813, 2011.

O. E. Oluyisola, F. Sgarbossa, and J. O. Strandhagen. Smart production planning and control: concept, use-cases and sustainability implications. *Sustainability*, 12:3791, 2020.

C. Osorio and M. Bierlaire. A simulation-based optimization framework for urban transportation problems. *Operations Research*, 61(6):1333–1345, 2013.

A. Öztürk, S. Kayalıgil, and N. E. Özdemirel. Manufacturing lead time estimation using data mining. *European Journal of Operational Research*, 173(2):683–700, 2006.

S. Paramonov, S. Kolb, T. Guns, and L. D. Raedt. Tacle: Learning constraints in tabular data. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM 2017, Singapore, November 06 - 10, 2017*, pages 2511–2514, 2017. doi: 10.1145/3132847.3133193. URL https://doi.org/10.1145/3132847.3133193.

T. P. Pawlak and K. Krawiec. Automatic synthesis of constraints from examples using mixed integer linear programming. *Eur. J. Oper. Res.*, 261(3):1141–1157, 2017. doi: 10.1016/j.ejor.2017.02.034. URL https://doi.org/10.1016/j.ejor.2017.02.034.

R. Pellerin, N. Perrier, and F. Berthaut. A survey of hybrid metaheuristics for the resource-constrained project scheduling problem. *European Journal of Operational Research*, 280(2):395–416, 2020.

D. F. Pereira, J. F. Oliveira, and M. A. Carravilla. Tactical sales and operations planning: A holistic framework and a literature review of decision-making models. *International Journal of Production Economics*, 228:107695, 2020. ISSN 0925-5273. doi: https://doi.org/10.1016/j.ijpe.2020.107695. URL https://www.sciencedirect.com/science/article/pii/S0925527320300852.

M. L. Pinedo. *Scheduling: theory, algorithms, and systems*. 2018. ISBN 978-3319265780.

Q. Qi, F. Tao, T. Hu, N. Anwer, A. Liu, Y. Wei, L. Wang, and A. Nee. *Enabling technologies and tools for digital twin*. Journal of Manufacturing Systems, 2019.

E. Rauch, P. Dallasega, and D. T. Matt. Complexity reduction in engineer-to-order industry through real-time capable production planning and control. *Production Engineering*, 12(3):341–352, 2018.

R. Rosen, J. Fischer, and S. Boschert. Next generation digital twin: an ecosystem for mechatronic systems? *IFAC-PapersOnLine*, 52(15):265–270, 2019. ISSN 2405-8963. doi: https://doi.org/10.1016/j.ifacol.2019.11.685. URL https://www.sciencedirect.com/science/article/pii/S2405896319316763. 8th IFAC Symposium on Mechatronic Systems MECHATRONICS 2019.

R. Rosen, D. Beyer, J. Fischer, W. Klein, V. Malik, and J. C. Wehrstedt. Flexiblere produktion durch digitale zwillinge in der automatisierungstechnik – methode zur automatischen generierung digitaler zwillinge für eine brownfield-produktion. In *Automation 2020, VDI Wissensforum GmbH*, pages 1039–1054. VDI Verlag, 2020.

T. Rossi, R. Pozzi, M. Pero, and R. Cigolini. Improving production planning through finite-capacity mrp. *International Journal of Production Research*, 55(2):377–391, 2017. doi: 10.1080/00207543.2016.1177235. URL https://doi.org/10.1080/00207543.2016.1177235.

D. A. Rossit, F. Tohme, and M. Frutos. Production planning and scheduling in cyber-physical production systems: a review. *International journal of computer integrated manufacturing*, 32(4-5):385–395, 2019.

K. Schemeleva, X. Delorme, and A. Dolgui. Evaluation of solution approaches for a stochastic lot-sizing and sequencing problem. *International Journal of Production Economics*, 199:179–192, 2018.

G. Schuh, J.-P. Prote, M. Molitor, F. Sauermann, and S. Schmitz. Databased learning of influencing factors in order specific transition times. *Procedia Manufacturing*, 31:356–362, 2019.

A. Senderovich, K. E. C. Booth, and J. C. Beck. Learning scheduling models from event data. In *Proceedings of the Twenty-Ninth International Conference on Automated Planning and Scheduling, ICAPS 2018, Berkeley, CA, USA, July 11-15, 2019*, pages 401–409, 2019. URL https://aaai.org/ojs/index.php/ICAPS/article/view/3504.

S. Sethi, H. Yan, H. Zhang, and Q. Zhang. Optimal and hierarchical controls in dynamic stochastic manufacturing systems: A survey. *Manufacturing & Service Operations Management*, 4(2):133–170, 2002.

M. Shafto, M. Conroy, R. Doyle, E. Glaessgen, C. Kemp, J. LeMoigne, and L. Wang. Modeling, simulation, information technology processing roadmap. *Technology Area*, 11, 2010.

G. Shao and M. Helu. Mitigating risks of perishable products in the cyber-physical systems based on the extended mrp model. *International Journal of Production Economics*, 193:51–62, 2017.

G. Shao and M. Helu. Framework for a digital twin in manufacturing: Scope and requirements. *Manufacturing Letters*, 24:105–107, 2020.

M. Sharp, R. Ak, and T. Hedberg Jr. A survey of the advancing use and development of machine learning in smart manufacturing. *Journal of manufacturing systems*, 48:170–179, 2018.

T. Simon, B.-A. Oussama, and B. Nadjib. Optimization approaches for the joint lot-sizing and supplier selection under lead time uncertainty. *Internal technical report*, 2021.

H. Simonis, P. Davern, J. Feldman, D. Mehta, L. Quesada, and M. Carlsson. A generic visualization platform for cp. In D. Cohen, editor, *Principles and Practice of Constraint Programming – CP 2010*, pages 460–474, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg. ISBN 978-3-642-15396-9.

D.-P. Song and J. Dinwoodie. Quantifying the effectiveness of vmi and integrated inventory management in a supply chain with uncertain lead-times and uncertain demands. *Production Planning and Control*, 19(6):590–600, 2008.

C. Soto, B. Dorronsoro, H. Fraire, L. Cruz-Reyes, C. Gomez-Santillan, and N. Rangel. Solving the multi-objective flexible job shop scheduling problem with a novel parallel branch and bound algorithm. *Swarm and Evolutionary Computation*, 53:100632, 2020.

J. C. Spall. *Introduction to stochastic search and optimization: estimation, simulation, and control*, volume 65. John Wiley & Sons, 2005.

T.-S. Su. A fuzzy multi-objective linear programming model for solving remanufacturing planning problems with multiple products and joint components. *Computers & Industrial Engineering*, 110:242–254, 2017.

D. Sun, R. Huang, Y. Chen, Y. Wang, J. Zeng, M. Yuan, T.-C. Pong, and H. Qu. Planningvis: A visual analytics approach to production planning in smart factories. *IEEE transactions on visualization and computer graphics*, 26(1):579–589, 2019.

M. Taal and J. C. Wortmann. Integrating mrp and finite capacity planning. *Production Planning & Control*, 8(3):245–254, 1997.

L. Tang, Y. Ma, J. Wang, L. Ouyang, and J.-H. Byun. Robust parameter design of supply chain inventory policy considering the uncertainty of demand and lead time. *Scientia Iranica*, 26 (5):2971–2987, 2019.

F. Tao, Y. Cheng, and D. Xu. Li and zhang, lin and li, bo hu. 2014. “cciot-cmfg: cloud computing and internet of things-based cloud manufacturing service system.”. *IEEE Transactions on industrial informatics*, 10(2):1435–1442.

F. Tao, T. Fan, K. K. Lai, and L. Li. Impact of rfid technology on inventory control policy. *Journal of the Operational Research Society*, 68(2):207–220, 2017.

F. Tao, J. Cheng, Q. Qi, M. Zhang, H. Zhang, and F. Sui. Digital twin-driven product design, manufacturing and service with big data. *The International Journal of Advanced Manufacturing Technology*, 94(9):3563–3576, 2018a.

F. Tao, Q. Qi, A. Liu, and A. Kusiak. Data-driven smart manufacturing. *Journal of manufacturing systems*, 48:157–169, 2018b.

F. Tao, F. Sui, A. Liu, Q. Qi, M. Zhang, B. Song, Z. Guo, S. C. Y. Lu, and A. Nee. Digital twin-driven product design framework. *International Journal of Production Research*, 57(12):3935–3953, 2019.

D. Tavaghof-Gigloo and S. Minner. Planning approaches for stochastic capacitated lot-sizing with service level constraints. *International Journal of Production Research*, pages 1–21, 2020.

H. Tempelmeier. *Material-Logistik: Modelle und Algorithmen für die Produktionsplanung und-steuerung in Advanced Planning-Systemen*, volume 6. Springer, 2006.

H. Tempelmeier and S. Helber. A heuristic for dynamic multi-item multi-level capacitated lotsizing for general product structures. *European Journal of Operational Research*, 75(2):296–311, 1994.

S. Thevenin, Y. Adulyasak, and J.-F. Cordeau. Material requirements planning under demand uncertainty using stochastic optimization. *Production and Operations Management*, a.

S. Thevenin, Y. Adulyasak, and J.-F. Cordeau. Stochastic dual dynamic programming for multi-echelon lot-sizing with component substitution. b.

S. Thevenin, N. Zufferey, and R. Glardon. Model and metaheuristics for a scheduling problem integrating procurement, sale and distribution decisions. *Annals of Operations Research*, 259 (1):437–460, 2017.

S. Thevenin, Y. Adulyasak, and J.-F. Cordeau. Material requirements planning under demand uncertainty using stochastic optimization. *Production and Operations Management*, 30(2): 475–493, 2021.

M. Vieira, S. Moniz, B. S. Gonçalves, T. Pinto-Varela, A. P. Barbosa-Póvoa, and P. Neto. A two-level optimisation-simulation method for production planning and scheduling: the industrial case of a human-robot collaborative assembly line. *International Journal of Production Research*, pages 1–21, 2021.

M. A. Villegas, D. J. Pedregal, and J. R. Trapero. A support vector machine for model selection in demand forecasting applications. *Computers industrial engineering*, 121:1–7, 2018.

J. Wang, J. Yang, J. Zhang, X. Wang, and W. Zhang. Big data driven cycle time parallel prediction for production planning in wafer manufacturing. *Enterprise information systems*, 12(6):714–732, 2018.

X. Wang, Y. Wang, F. Tao, and A. Liu. New paradigm of data-driven smart customisation through digital twin. *Journal of Manufacturing Systems*, 58:270–280, 2021.

M. Weyrich, M. Klein, J.-P. Schmidt, N. Jazdi, K. D. Bettenhausen, F. Buschmann, C. Rubner, M. Pirker, and K. Wurm. Evaluation model for assessment of cyber-physical production systems. In *Industrial internet of things*, pages 169–199. Springer, 2017.

J. Wy, S. Jeong, B.-I. Kim, J. Park, J. Shin, H. Yoon, and S. Lee. A data-driven generic simulation model for logistics-embedded assembly manufacturing lines. *Computers & Industrial Engineering*, 60(1):138–147, 2011.

C. A. Yano and H. L. Lee. Lot sizing with random yields: A review. *Operations research*, 43(2): 311–334, 1995.

C. Yu, W. Zhang, X. Xu, Y. Ji, and S. Yu. Data mining based multi-level aggregate service planning for cloud manufacturing. *Journal of Intelligent Manufacturing*, 29(6):1351–1361, 2018.

J. Zarei, M. Rasti-Barzoki, and S. R. Hejazi. A game theoretic approach for integrated pricing, lot-sizing and advertising decisions in a dual-channel supply chain. *International Journal of Operational Research*, 40(3):342–365, 2021.

C. Zhang, G. Zhou, J. He, Z. Li, and W. Cheng. A data-and knowledge-driven framework for digital twin manufacturing cell. *Procedia CIRP*, 83:345–350, 2019a.

H. Zhang, G. Zhang, and Q. Yan. Digital twin-driven cyber-physical production system towards smart shop-floor. *Journal of Ambient Intelligence and Humanized Computing*, 10(11):4439–4453, 2019b.

Y. Zhang, S. Shen, and J. L. Mathieu. Distributionally robust chance-constrained optimal power flow with uncertain renewables and uncertain reserves provided by loads. *IEEE Transactions on Power Systems*, 32(2):1378–1388, 2016.

Y. Zheng, S. Yang, and H. Cheng. An application framework of digital twin and its case study. *Journal of Ambient Intelligence and Humanized Computing*, 10(3):1141–1153, 2019.

C. Zikopoulos. Remanufacturing lotsizing with stochastic lead-time resulting from stochastic quality of returns. *International Journal of Production Research*, 55(6):1565–1587, 2017.

Y. Zuo, F. Tao, and A. Y. Nee. An internet of things and cloud-based approach for energy consumption evaluation and analysis for a product. *International Journal of Computer Integrated Manufacturing*, 31(4-5):337–348, 2018.