EPFL

Thèse n° 8699

A dynamic scheduling tool and a methodology for creating digital twin of manufacturing systems for achieving Zero Defect Manufacturing

Présentée le 28 mai 2021

Faculté des sciences et techniques de l'ingénieur
Groupe SCI STI DK
Programme doctoral en manufacturing

pour l'obtention du grade de Docteur ès Sciences

par

Foivos PSAROMMATIS GIANNAKOPOULOS

Acceptée sur proposition du jury

Prof. Y. Bellouard, président du jury
Prof. D. Kyritsis, directeur de thèse
Prof. F. Ameri, rapporteur
Prof. B. Lalic, rapporteur
Prof. T. Weber, rapporteur

École
polytechnique
fédérale
de Lausanne

2021

To my father…

Acknowledgments

The work presented in this thesis constitutes the outcome of a long journey that goes beyond the pages of this manuscript. During this journey, I was fortunate enough to receive endless support and feedback from numerous people. I deeply acknowledge that without them, nothing would be the same. The least thing I can do for them is to open this thesis by thanking them.

First and foremost, I would like to thank my advisor, Prof. Dimitrios Kyritsis, for his guidance and advice. Dimitris has been an admirable supervisor, creative, honest and supportive. His flexibility and kind personality were two important ingredients for making the ICT4SM group such as a warm and pleasant working environment. I was really fortunate enough to have Dimitris as my thesis advisor and I deeply appreciate his belief in me and in my research.

I would also like to thank the members of my thesis committee for putting in the efforts to evaluate the merit of my research work as well as to provide insightful and constructive comments and feedback that shaped this thesis. Specifically, I would like to thank Prof. Thomas Weber, Prof. Bojan Lalic and Prof. Farhad Ameri, who acted as my thesis examiners, as well as Prof. Yves Bellouard, who accepted to act as the thesis jury president.

At this point, I would like to share my deepest gratitude to all the people who made Lausanne feel like home. First of all, I would like to thank all the people affiliated or shared connections with the group, i.e., Damiano, Sangje, Gökan, Giannis, Prodromos, Christos, Marlène, PA, Sylvia, Xiaochen and Jinzhi and many more, for everything. Life at the lab would not be the same without them.

This research has been supported by grants from the European Union Horizon 2020 research and innovation program Z-Fact0r under grant agreement No 723906, the EU Horizon 2020 research and innovation program QU4LITY under grant agreement No 825030

Summary

Over the years, the manufacturing industry has seen constant growth and change. From one side, it has been affected by the fourth industrial revolution (Industry 4.0). From the other side, it has had to enhance its ability to meet higher customer expectations, such as more customized products in a shorter time. Those factors have led many manufacturing companies to produce new products faster than ever for two main reasons: achieving higher profits and meeting increasing demand from their customers. This phenomenon has imposed new rules on the manufacturing of products, such us producing in a shorter time and smaller batches, making strategies that had been successful in the past useless or not as efficient as required. In the contemporary competitive market of manufacturing, quality is a criterion of primary importance for winning market share. Quality improvement must be coupled with a performance point of view. One of the most promising concepts for quality control and improvement is called zero defect manufacturing (ZDM), which utilizes the benefits from Industry 4.0 technologies. ZDM imposes the rule that any event in the production should have a counter-action to mitigate it. Specifically, in this thesis, a systematic literature review was performed on the ZDM concept from 1987 to 2017 to summarize the state of the art and highlight shortcomings and further directions in research. Accordingly, the ZDM implementation methods were investigated and evaluated identifying the main research patterns in the sample by analyzing key factors. Based on the extensive review of the ZDM literature, we identified and highlighted four distinct strategies based on overarching themes for ZDM, namely detection, repair, prediction, and prevention.

The goal of this research was twofold: first, it aimed to provide to manufacturers with a dynamic scheduling tool that embraces the principles of ZDM, which would grant the opportunity to implement ZDM strategies in their production lines and simultaneously maintain the performance of the production system at an acceptable level. The integration of ZDM into the scheduling tool was achieved by creating a separate component for each one of the four ZDM strategies. The second goal was focused on creating a methodology for the manufacturer to correctly select the appropriate ZDM strategies to implement at each manufacturing stage. This methodology consists of several steps. The first step is to conduct several simulations using the developed scheduling tool with specific data sets. The data sets are created using the design of experiments methodology. Using the results of the experiments, a digital twin model is created for predicting the results of the developed scheduling tool without using said tool. Using the digital twin model, multiple ZDM parameter-combination sets are created and plugged into the model. The outcome of this process will generate a set of maps that show the performance of each ZDM strategy at each manufacturing stage. These maps are intended to provide information for comparing different ZDM-oriented equipment to reach a final decision for correct and efficient ZDM implementation.

Keywords: Zero defect manufacturing (ZDM), Quality control, Quality improvement, Dynamic scheduling, Design, Digital twin, Design of experiments, Decision support system, Production mapping

Résumé

Au fil des années, l'industrie manufacturière a connu une croissance et des changements constants. D'une part, elle a été affectée par la quatrième révolution industrielle (Industrie 4.0). D'autre part, elle a dû améliorer sa capacité à répondre aux attentes plus élevées des clients, comme des produits plus personnalisés dans un délai plus court. Ces facteurs ont conduit de nombreuses entreprises manufacturières à fabriquer de nouveaux produits plus rapidement que jamais, pour deux raisons principales : réaliser des bénéfices plus élevés et répondre à la demande croissante de leurs clients. Ce phénomène a imposé de nouvelles règles à la fabrication des produits, comme le fait de produire dans un délai plus court et en plus petits lots, rendant ainsi inutiles ou moins efficaces des stratégies qui avaient réussi dans le passé. Dans le marché concurrentiel actuel de la fabrication, la qualité est un critère de première importance pour gagner des parts de marché. L'amélioration de la qualité doit être associée à un point de vue de performance. L'un des concepts les plus prometteurs pour le contrôle et l'amélioration de la qualité est appelé fabrication zéro défaut (ZDM), qui utilise les avantages des technologies de l'industrie 4.0. La ZDM impose la règle selon laquelle tout événement de la production doit faire l'objet d'une contre-action pour l'atténuer. Dans cette thèse, nous réaliserons plus précisément une revue systématique de la littérature a été réalisée sur le concept ZDM de 1987 à 2017 pour résumer les dernières avancées et mettre en évidence les lacunes et les nouvelles orientations de la recherche. En conséquence, les méthodes de mise en œuvre de la ZDM ont été étudiées et évaluées en identifiant les principaux modèles de recherche dans l'échantillon en analysant les facteurs clés. Sur la base de l'examen approfondi de la littérature de la ZDM, nous avons identifié et mis en évidence quatre stratégies distinctes basées sur les thèmes principaux de la ZDM, à savoir la détection, la réparation, la prédiction et la prévention.

L'objectif de cette recherche était double : elle visait tout d’abord à fournir aux fabricants un outil d'ordonnancement dynamique qui adopte les principes de la ZDM, ce qui leur donnerait la possibilité de mettre en œuvre des stratégies ZDM dans leurs lignes de production et de maintenir simultanément les performances du système de production à un niveau acceptable. L'intégration de la ZDM dans l'outil de planification a été réalisée en créant un composant séparé pour chacune des quatre stratégies ZDM. Le second objectif était de créer une méthodologie permettant au fabricant de sélectionner correctement les stratégies ZDM appropriées à mettre en œuvre à chaque étape de la fabrication. Cette méthodologie comprend plusieurs étapes. La première étape consiste à effectuer plusieurs simulations en utilisant l'outil de planification développé avec des ensembles de données spécifiques. Les ensembles de données sont créés à l'aide de la méthodologie du plan d'expériences. En utilisant les résultats des expériences, un modèle numérique double est créé pour prédire les résultats de l'outil de planification développé sans utiliser ledit outil. En utilisant le modèle numérique double, plusieurs ensembles de combinaisons de paramètres ZDM sont créés et connectés au modèle. Le résultat de ce processus génère un ensemble de cartes qui montrent la performance de chaque stratégie ZDM à chaque étape de la fabrication. Ces cartes sont destinées à fournir des informations permettant de comparer différents équipements orientés ZDM afin de prendre une décision finale pour une mise en œuvre correcte et efficace de la ZDM.

Mots-clés : Fabrication zéro défaut (ZDM), Contrôle de la qualité, Amélioration de la qualité, Planification dynamique, Conception, Jumelage numérique (Digital Twin), Plans d'expériences, Système d'aide à la décision, Cartographie de la production

Table of Contents

List of Tables

List of Figures

Abbreviation List

Abbreviation	**Description**
ANOM	Analysis of Means
ANOVA	Analysis of Variance
BOML	Beginning of Manufacturing Life
BoP	Bill of Processes
CBMMS	Condition Based Midterm Maintenance Scheduling
CI	Continuous Improvement
CT	Cognitive Twin
DFFS	Dynamic Flexible Flow Shop
DM	Digital Model
DS	Digital Shadow
DSS	Decision Support System
DoE	Design of Experiments
DT	Digital Twin
EOML	End of Manufacturing Life
FAP	Failure Avoidance Probability
FFS	Flexible Flow Shop
FMS	Flexible Manufacturing Systems
GRM	Green Rescheduling Method
HVNS	Hybrid Variable Neighborhood Search
IIoT	Industrial internet of things
KPI	Key Performance Indicators
MIS	Machine Improvement State
MFG	Manufacturing Stage
MSP	Mission Success Probability
MOML	Middle of Manufacturing Life
MTBT	Mid Time Between Tuning
NP	Non Polynomial
NSGA	Non-Dominated Sorting Genetic Algorithm
OC	Order Criticality
QM	Quality Management
ROIRP	Rush Order Insertion Rescheduling Problem
RSR	Right Shift Rescheduling
TABC	Two-Stage Artificial Bee Colony
TFSORP	Two-machine Flow Shop Outsourcing and Rescheduling Problem
WAAM	Wire Arc Additive Manufacturing
ZDM	Zero Defect Manufacturing

1 Introduction

Over the years, the manufacturing industry has seen constant growth and change. On one level, it has been affected by the fourth industrial revolution (Industry 4.0) [1]. On another, it has had to enhance its ability to meet higher customer expectations, such as more customized products at a faster rate [2]. These factors have led many manufacturing companies to produce new products quicker than ever for two main reasons: to achieve higher profits and to meet the increasing demand from their customers. This phenomenon has imposed new rules on product manufacture, such as shorter production time and smaller batch output, making strategies, which had been successfully used in the past, obsolete or less efficient [3]. For example, during the previous century, most of the large industrial sectors, such as the automotive sector, essentially relied on the mass production paradigm. However today, with the rise of product customization, they have shifted to manufacturing methods based on lean [4][5] and customer demand [6]. Due to these changes in the industrial environment, most of the production systems have to perform jobs in highly dynamic and stochastic scenarios [7]. Under these circumstances, unexpected events may occur causing the initial schedule to be changed because it may not fit in this new scenario [8]. Therefore, it requires more adaptability from firms to match their clients' increasing expectations. It also becomes much more challenging to apply systematic methodologies for monitoring and preventing defect occurrence within the manufacturing shop floors due to the increasing complexity of both products and production systems [9]. In addition, the time to optimize the production process lines has been significantly reduced, because companies are generally no longer mass producing but making smaller batches of customized products [10], and as a consequence, the rate of defected products has increased. With these factors taken into consideration, newer and more sophisticated strategies and tools are needed [11][12]. More specifically, better techniques of quality management are required to cope with the current needs [13][14][15][16].

To match the real-life industrial settings and respect today's highly competitive environment, numerous realistic constraints have been incorporated into scheduling problems. Since production is not a standalone activity, this scheduling must take into account equipment unavailability due to breakdowns or maintenance operations. In this field, interactions between production and maintenance have been a challenge to researchers to balance resource availability and avoid conflictual situations. This problem is known in the literature as ”production scheduling with availability constraints” [17].

So far, there has been a considerable amount of literature on the scheduling problem integrating deterministic preventive maintenance, also known as time-based maintenance, where periods of unavailability are known and fixed in advance (deterministic unavailability). Y. Ma and C. Zuo 2010, provided a detailed review on this topic where works were featured with different production configurations and under various criteria using different optimization methods [18]. Furthermore, the current scheduling models work only under ideal conditions since they do not take into account external events [19]. Indeed, in real life, different unpredictable events could happen at shop floor level and bring inconsistencies into the ongoing schedule. Therefore, it is important to broaden the research area by analyzing existing rescheduling models and creating new ones to mitigate the consequences of unpredictable events [19].

A proactive approach may not foresee all possible disruptive events, even if the original schedule is robust [8]. Thus, rescheduling has a central role for the robustness of production processes under uncertain conditions, since the process has to be the most reactive and flexible as possible in order not to interrupt the production flow [20]. According to J. Lindström *et al.* 2019, the generic strategy to apply is the following: firstly, the scheduling solution has to be produced (predictive), then, when an unexpected event occurs, the rescheduling is done to generate a new feasible solution (reactive) [21].

In the context of this globalized ultra-connected world, benchmarking leads to a large number of competitive solutions to address a need [22] [23]. For a company, increasing and even keeping its market share is tougher than ever. One of the main factors that drives a product's commercial success is its quality [24]. The companies are paying particular attention to the product quality to assure that all of their customers are satisfied. Nevertheless, a need is not defined in a fixed manner. It evolves and so does the manufacturing to produce the items. This evolution places the organizations in a permanent state of questioning the quality of their products and processes, and binds them into a continuous improvement (CI) initiative to stay competitive [25][26].

CI is done using Quality Management Systems (QMS) which traditionally rely on methodologies such as Lean Manufacturing (LM), Six Sigma (SS), Theory of Constraints (TOC), Total Quality Management (TQM), and Lean Six Sigma (L6S) (Hutchins, 2016), which are well established in the production systems with the goal to improve product quality. These methods can be characterized as "corrective", which means that they act after the creation of a problem abd they do not take advantage of modern data-driven technologies that offers predive capabilities. Furthermore, the traditional QMS methods do not learn from defects, they just remove them. These methodologies analyze the past to improve in the future. Therefore, there is a loss of potentially important information from the present. Not analyzing the present creates an inertia between the occurrence of an event and the identification of an improvement linked to this event [27].

Modern technological advancements provided capabilities that were not possible at the past. These technological advancements initiated the emergence of another QMS method named Zero Defect Manufacturing (ZDM). One major change in ZDM is about the flow of information. Indeed, ZDM uses both historical and real-time data to prevent product from defect. Doing this, ZDM combines several quality control applications concerning production lines, machinery, automation applications, and supply chain processes [27]. This is possible thanks to the development of IT systems and Industry 4.0. The core concept of ZDM is "Make it right at first attempt".

ZDM can offer higher efficiency and quality in the process by eliminating the defected components, but implementing ZDM into a production system is not a straight forward process. Scheduling tools and rescheduling technics should be updated because scheduling is a critical component that can realize ZDM into production systems [21]. However, to apply ZDM in the rescheduling process, a new category of real-time events has to be added: the product-oriented unexpected events [28]. Therefore, it is imperative to integrate the ZDM concept and in general the product quality aspect to the regular scheduling problem, and solve the problem in unison and not separately in a series as it is performed now. This will allow to balance the productivity and quality of the production achieving higher efficiency. Therefore, the current research focuses on two main points:

(i) The development of a dynamic scheduling tool that integrates ZDM principles and extends the product quality aspects.

(ii) The development of a methodology for assisting manufacturers during the design phase of a quality assurance policy for achieving ZDM.

2 State of the Art

2.1 Zero Defect Manufacturing

ZDM is considered by many researchers and industries to be a viable replacement of the traditional QMS methods [29][30][28][31]. This strategy's goal is to decrease and mitigate failures within manufacturing processes and "to do things right the first time" [3], or in other words, to eliminate defective parts during production. However, the idea of ZDM is not new; it was first mentioned during the Cold War in the U.S. Army regarding their defective weapon system [32]. ZDM is a disruptive concept that is able to entirely reshape current manufacturing paradigms.

The core concept of ZDM is that 100% of the parts are inspected to assure that no defective product will be sent to the customers [33]. ZDM can be implemented through two different approaches: product-oriented ZDM and process-oriented ZDM [33]. The difference is that product-oriented ZDM studies defects in the actual parts and attempts to find a solution, whereas process-oriented ZDM studies defects in the manufacturing equipment, and based on this it can evaluate whether the manufactured products are good. Process-oriented ZDM lies within the predictive maintenance concept. In Figure 1, we illustrate these two approaches as one concept, namely the ZDM concept, which comprises two start points, one for each approach.

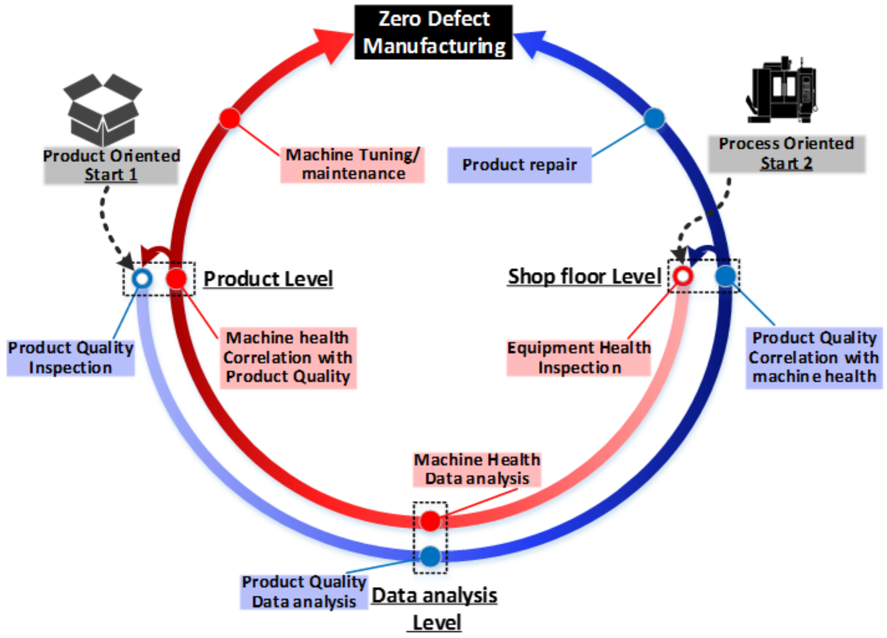

Zero Defect Manufacturing
Product Oriented Start 1
Machine Tuning/ maintenance
Product repair
Process Oriented Start 2
Product Level
Shop floor Level
Product Quality Inspection
Machine health Correlation with Product Quality
Equipment Health Inspection
Product Quality Correlation with machine health
Machine Health Data analysis
Product Quality Data analysis
Data analysis Level

Figure 1: Zero defect manufacturing concept [33]

The reasons that ZDM thinking is attractive to companies are manifold. First, it can considerably reduce costs of the company's resources related to the treatment of defective products [34]. The ZDM process essentially relies on the fact that no useless element is present within a process. Useless elements refer to anything that does not bring any added value to the product, such as defective machines and tools and inefficient employees. Significant reductions in scrap production and therefore money savings can be realized with ZDM [35]. Beyond that, the overall production chain should be continuously improved. Any possibility of system enhancement must also be meticulously and extensively assessed. In this way, product manufacturing is getting increasingly closer to perfection [36]. This approach can also be motivated by increasing safety and customer satisfaction, which might strengthen customer loyalty and cause financial benefits of the company to soar [37].

This concept had been implemented only partially so far due to numerous technological limitations prohibiting its implementation. Currently, with the evolution of Industry 4.0, the ZDM concept is easier to implement due to the availability of the required amount of data for techniques such as machine learning to work properly [38][39][40][41]; however, much effort is still required to ensure more effective integration and coordination of the capabilities of each technology. Furthermore, the equipment required for such data recording used to be very expensive and companies did not invest in it [42]. However, the landscape has changed with increasing computing power and data storage and significantly dropping sensor prices, together with new technologies that make the implementation of the concept of ZDM possible.

ZDM will be the new standard for companies toward eco-friendlier and more efficient production lines with zero defects. In this regard, Figure 2 illustrates how the ZDM concept can be implemented and also how ZDM strategies are interconnected among them. ZDM consists of four strategies: detection, repair, prediction, and prevention [33][43].

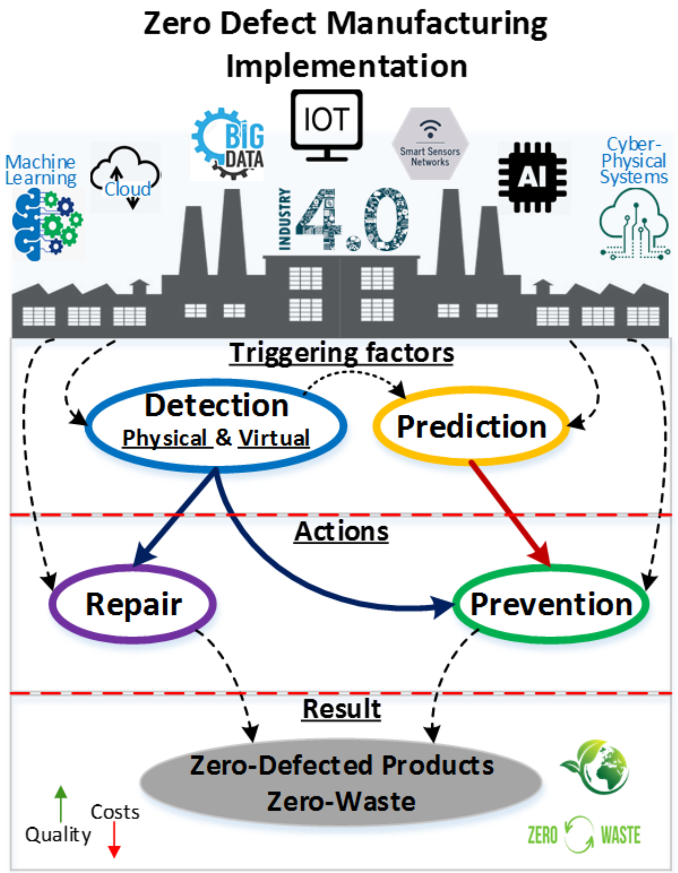


Figure 2: Zero defect manufacturing implementation [33]

These strategies are interconnected as follows (which applies to both product- and process-oriented approaches): If a defect is detected then it can be repaired, and the data gathered by the defect-detection module are populated to specifically designed algorithms for predicting when a defect may occur, and can thus be prevented (Figure 2). Here, the phrase mentioned earlier fits, namely “to do things right the first time.”

Since ZDM is an emerging approach there is no significant literature to refer to. The literature that exists is focused mainly on developing technologies and methodologies which are related to individual ZDM strategies [33]. This is a step forward but there is a significant lack of tools such as scheduling tools or design methodologies for achieving ZDM.

2.2 Scheduling

Today, companies have several factories around the world. In this context of a globalized economy, all these factories must have their production as optimized as possible [44]. Companies that have multiple production sites have superior production quality, reduced manufacturing costs, and shorter delivery times [45][46]. The optimization of schedules is performed through the use of computer science because complexity and calculation times are critical.

Many manufacturers use a flexible job shop architecture for their productions sites. In a job shop, each job has its own route. Flexible job shops are a generalization of the job shop where instead of "***m*** *machines in series* there are ***c*** *work centers with at each work center a number of identical machines in parallel. Each job has its own route to follow through the shop*" [47]. The job shop problem is considered to be NP-hard [48], and therefore, nonanalytical methods are required for acquiring a solution. The most effective method for solving this type of problem is to combine heuristics and metaheuristics. Heuristics are used for generating a high-quality initial solution, and then metaheuristics are used to optimize that initial solution to reach a local optimum [49].

Some heuristic algorithms in the literature use the earliest competition time rule to create a solution that assigns a job to the schedule at each step; job j is put in each factory and in it is kept in the factory with the smallest makespan [50]. These rules have been adapted through the work of Nawaz, Enscore, and Ham [51], known as NEH, the abbreviation of their names. In recent years, these rules based on the makespan have often been used to feed metaheuristics. The NEH heuristic is known as the most effective heuristic for the permutation flow-shop scheduling problem with makespan minimization. The job set is sorted by descending order according to processing times. When one picks the two first jobs, the sequence that minimizes the makespan is given. Subsequently, one does not change the relative position of these two jobs. Then, each job is inserted into the sorted list, and the makespan is calculated each time for each possible sequence, and finally, the sequence with the smallest makespan is selected.

In the literature, different metaheuristic algorithms have been described in the context of optimizing a scheduling problem. Some of the most commonly used metaheuristic algorithms are as follows: the hybrid genetic algorithm [52] [53], tabu-search algorithm [54] [55], electromagnetism-like mechanism algorithm [56], immune algorithm developed [57], chemical reaction optimization algorithm [58], iterated greedy algorithm [59] , simulated annealing [60], and Bat algorithm [61].

Several metaheuristic methods that aimed to solve the job-shop rescheduling problem were investigated and compared by M. T. Jensen, such as the right-shifting, neighborhood-based, and hill climbing methods. The findings indicated that the scheduling performance is heavily dependent on the rescheduling method, breakdown duration, and other details [62]. Other authors proposed the integration of a genetic algorithm into a simulation tool to solve flexible job-shop scheduling problems, aiming to minimize the expected mean makespan and expected mean tardiness [63].

In the literature, several cases remain general and take features of simple machines and simple jobs without constraints between them. In our case, we attempt to get as close as possible to a real case with machines fully defined in cost, properties, and jobs that have processing and setup times depending on the machine. In this paper, we first discuss how the problem is defined, then describe the different heuristics used to generate initial schedules for the tabu search, and finally analyze the results of simulations.

The scheduling/rescheduling of manufacturing systems has been a key focus in both research and industry for a long time. Studies on quantitative scheduling started back in the 1950s [64][65] and abundant research efforts have been spent on this topic over the decades that followed. Although it has been commonly understood that scheduling/rescheduling is crucial for manufacturing systems and has been highlighted in many studies [66][67][68], a gap still exists between the theory and practice of manufacturing scheduling/rescheduling [65]. In practical applications, it is very difficult to follow the original schedule exactly due to unforeseen disturbances, such as the constant arrival of new orders, the importance of jobs' priorities, and the possibility of disruptions related to manufacturing resources, such as breakdowns [65]. Rush orders were identified as one of the four production disturbance sources beside incorrect work, machine breakdowns, and rework due to quality problems [69]. The authors of [70] investigated the general reasons and corresponding solutions for rush orders. According to the study, the main causes of rush orders include implicit customer priorities; concern over extra returns, outlays, or usuries; special orders authorized by higher-ups; and production disturbances such as machinery breakdowns and lack of materials.

Many studies have been conducted on the rescheduling problem caused by rush orders. A production rescheduling expert simulation system was developed in [71], which was enabled by a simulation technique, artificial neural network, expert knowledge, and dispatching rules. The impact of rush orders was analyzed. A rescheduling method was proposed in [72][73], which was based on the ant colony optimization algorithm enhanced by mutation operations, aiming to solve the multi-shop rescheduling problem under the condition of rush orders. The weighted mean flow time of both original jobs and rush orders were used as the objective function, in which the weight for rush orders is much higher than that of the original jobs, in such a way that the rush orders would be produced as early as possible in the new schedule.

With the development of artificial intelligence, advanced data analysis techniques such as machine learning and multiagent-based systems have been widely used to optimize production schedules [74][75][76][77][78]. For example, a self-organized integration mechanism module based on supervised learning techniques was proposed in [79] with the aim of enhancing the scheduling of manufacturing orders in dynamic environments. This module enables scheduling systems to decide autonomously which integration mechanism will be used to incorporate new orders in the current scheduling plan.

As a crucial topic of scheduling/rescheduling, the rush order insertion rescheduling problem (ROIRP) has also attracted much research attention, especially for enterprises that apply the make-to-order production mode [80]. For example, [80] analyzed the ROIRP with preventive maintenance in a two-machine flow shop, and [81] used an improved elitist non-dominated sorting genetic algorithm (NSGA-II) to solve the multi-objective ROIRP just in a single device system. The authors of [80] investigated the ROIRP in a hybrid flow shop with multiple stages and machines. They proposed a mathematical model that simultaneously considers constraints such as lots, sequence-dependent setup times, and transportation times.

Despite abundant studies in past decades, the scheduling/rescheduling problem of manufacturing systems caused by real-time events such as rush orders remains a challenge for most make-to-order enterprises.

2.3 Rescheduling for Zero-Defect Manufacturing

In the literature, most models developed to face rescheduling problems are machine-oriented [82]. However, to adopt the ZDM philosophy, the rescheduling model must be product-oriented. The first model that links rescheduling problems caused by unexpected product-oriented events to ZDM was proposed by F. Psarommatis and D. Kiritsis [43]. The model integrates a decision support system (DSS) into a dynamic scheduling tool to comply with ZDM principles as well as prevent and predict defects. When a disruptive event occurs, the DSS and dynamic scheduling tool interact together to produce a new schedule. The new solution is evaluated based on the product quality and other key performance indicators (KPIs). Through adopting this model, manufacturing companies are able to face new production challenges and produce highly customized products in small batches. H. Zörrer et al. [83] also applied a DSS linked to ZDM. Their proposed model was applied to a multi-stage manufacturing environment for the production of carbon fiber components for aircraft. The aim was to generate a tool for quality control that helps operators decide which parts need reworking. To achieve this result, an extensible hybrid DSS was applied that combined a software application for 3D visualization and a business analytics dashboard for supporting the rework decision.

P. A. Dreyfus and D. Kyritsis [30] proposed a more generic model that links scheduling with ZDM. Their proposed model was based on the combination of three different strategies: ZDM, predictive maintenance, and scheduling algorithms. The aim of the model is to increase production capabilities without large investments. Automatic scheduling is considered the brain of the tool, which takes the uncertainty into account and decides whether it is required to launch and schedule a maintenance operation. To make this decision, the model calculates the probability of failures and the time required to repair them.

2.4 Tackling Unexpected Product-Oriented Events

Moreover, many researchers have addressed the issue of unexpected product-oriented events without linking to the ZDM philosophy. E. Kucharska et al. [84] proposed a model to solve unexpected defects in a flow-shop system with stochastic uncertainties. The model was developed around a hybrid algorithm based on the algebraic-logical meta-model, which is able to remove manufacturing defects detected during quality control. The approach distinguishes itself by adding the possibility of modeling the decision-making process. The results of the experiments were evaluated considering three factors: the cost of algebraic-logical models switching operation, the impact of the number of defect repairs for execution time, and the switching number between models. B. J. Joo et al. [85] proposed a model to solve scheduling problems in a three-stage dynamic flexible flow shop (DFFS). The model was mainly based on quality feedback linked to the defect rate; indeed, if the defect rate is over the limit, quality feedback would be generated. A dispatching rule-based scheduling algorithm was adopted to solve quality problems by maximizing the quality rate and minimizing job tardiness. G. Levitin et al. [86] proposed a tool based on the Poisson process of shocks, which may generate defects in a random environment. If defects are detected, an optimized reschedule is generated with the aim of maximizing and optimizing two performance indicators, namely mission success probability (MSP) and failure avoidance probability (FAP), respectively. Y. Xu et al. [87] presented a condition-based midterm maintenance scheduling (CBMMS) model that was able to reschedule preventive maintenance activities thanks to the use of a time-varying threshold decisions variable.

2.5 Rescheduling Models

More generally, rescheduling solutions have also been developed to address all types of unexpected scenarios. R. Barták and M. Vlk [88] proposed a back jumping heuristic algorithm to respond to the occurrence of unexpected events, such as resource breakdowns or hot orders. The aim was to reschedule the production system with the least amount of task change possible. The solution adopted was to replace activities present in the process with other new activities to react in the quickest way to disruptive events. Battaïa et al. [89] presented a rescheduling algorithm based on a constraint programming approach to handle the remaining tasks when unexpected events disrupt a low-volume assembly line. A fast rescheduling decision support tool was implemented to reschedule all the uncompleted tasks when unexpected events occur. The performances of the tool were evaluated through numerical experiments to check the sustainability of the model. C. Pascal and D. Panescu [66] studied a holonic coordination mechanism to deal with rescheduling problems in response to unexpected events in manufacturing systems that appear at the resource level. G. Mejía and D. Lefebvre [90] proposed a model that uses timed Petri nets to address operation interruptions and unreliable resources of flexible manufacturing systems (FMSs) in an uncertain environment. M. T. Jensen [62] compared various rescheduling methods (such as right-shifting, neighborhood-based rescheduling, and hillclimbing rescheduling) to solve job shop problems, and concluded that the performance of robust scheduling is very much dependent on the rescheduling method, breakdown duration, and other details. M. Gholami and M. Zandieh [63] proposed the approach to integrate simulation with a genetic algorithm to solve flexible job-shop scheduling problems. They applied the method to minimize the expected mean makespan and expected mean tardiness, concluding that the breakdown level (Ag) and mean time-to-repair (MTTR) are highly impactful on minimizations of both. One critical reason for scheduling tools to struggle create optimal schedules is the fact of the presence of unexpected events, which are disrupting the normal production[91][92]. Unexpected events are a very common situation in manufacturing systems. Unexpected event is characterized an event that is unplanned, in other words its occurrence in time is not known and cannot be predicted [93][94].

2.5.1 Unexpected job and order arrival

In the literature, many other researchers have addressed rescheduling problems that result from unexpected events from different points of view, namely various machine environments, processing characteristics, and objectives [47]. One of the main unexpected events studied in rescheduling is new job arrivals/new order events. D. Rahmani and R. Ramezanian [95] proposed a model that addresses the unexpected arrival of a new job in a dynamic flexible flow shop (FFS). This reactive model is based on total weighted tardiness, stability, and resistance to change. It aims to generate a stable reschedule against any unexpected job arrival, since it considers it more valuable to generate a stable solution rather than an optimal solution that neglects the disruptions. The rescheduling solutions are evaluated based on two parameters: systematic performance and deviation from the initial schedule. K. Z. Gao et al. [96] proposed a model to solve rescheduling problems caused by uncertainty of job arrivals in a flexible job shop. The model is based on a two-stage artificial bee colony (TABC) with the aim of minimizing the makespan. L. Liu [97] presented a model to solve two-machine flow-shop outsourcing and rescheduling problems (TFSORPs) upon the arrival of a new unexpected job. The goal was to optimize the makespan and outsourcing cost variables using a hybrid variable neighborhood search (HVNS) algorithm. In addition, an experimental design was implemented for optimizing and calibrating the settings of HVNS. N. C. O. Silva et al. [98] proposed a rescheduling model based on a mixed-integer formulation to address the arrival of new orders. The goal was to optimize both the makespan and tardiness. L. Liu and H. Zhou [99] addressed the parallel machine rescheduling problem caused by unexpected job rework. The model was based on two dependent factors: the number of disrupted jobs and the completion time. The problem was treated as a three bi-criteria scheduling problem through both lexicographical and simultaneous optimization approaches. S. K. Moghaddam and K. Saitou [100] proposed another rescheduling model for job arrivals/unplanned order arrivals. The tool developed was based on the concept of dynamic pegging in multi-level production and mixed-integer programming model, which link dynamic pegging with rescheduling. When unplanned orders arrive, the dynamic pegging reassigns the work-in-progress to the newly arrived orders by optimizing the rescheduling costs.

2.5.2 *Unexpected machine breakdown*

Other researchers have proposed rescheduling solutions for machine breakdown events. R. Buddala and S.S. Mahapatra [101] proposed a model to solve flexible job-shop scheduling problems caused by machine failure, applying two-stage teaching-learning-based optimization (2S-TLBO). The target of this approach was to minimize the makespan to generate robust and sustainable schedules, which could mitigate the costs of unexpected machine breakdowns. The results obtained with 2S-TLBO were analyzed using a one-way analysis of variance (ANOVA) test. F. Qiao et al. [102] proposed a rescheduling model for machine failure in a dynamic semiconductor manufacturing system. The model was based on a novel machine group-oriented match-up rescheduling (NMUR) approach, which achieved better results compared with right-shift rescheduling (RSR) in terms of rescheduling stability and efficiency. N. Al-Hinai and T.Y. ElMekkawy [103] considered four different types of machine breakdown in a flexible job shop. Their model implemented a two-stage hybrid genetic algorithm to minimize the makespan. Y. Yin et al. [104] considered a failure of two identical parallel machines. The aim of their model was to reschedule jobs by considering deviation costs and total completion time to not cause excessive schedule disruption. The tool was designed to generate a set of Pareto-optimal solutions based on the optimization of both the completion time of rescheduling and scheduling disruption factors.

Moreover, S. Ferrer et al. [105], M. Nouiri et al. [106], and Z. Li [107] have proposed models that take environmental objectives into account without overlooking the production objectives. S. Ferrer et al. [105] proposed a model to solve unrelated parallel machine rescheduling problems in a dynamic environment through applying two different approaches: greedy-heuristic and metaheuristic. The aim was to improve production management, in terms of rescheduling quality and computational time, by limiting energy consumption (the energy-aware scheduling problem). M. Nouiri et al. [106] proposed a green rescheduling method (GRM), which generates a rescheduling solution for dynamic flexible job shops under machine breakdowns. The aim of the model is to optimize the makespan and energy consumption. Z. Li [107] proposed a model that addresses the rescheduling problem for unexpected machine breakdowns. The aim was to minimize rescheduling by considering the consumption of energy and lead time.

As presented in this chapter, several frameworks and models for rescheduling solutions were analyzed and categorized. This investigation highlighted three main aspects. First, a huge gap between the theoretical and practical application of the models is evident. Second, very few papers have proposed a rescheduling model based on a product-oriented approach. Finally, the most important aspect is that none of the works have focused on the quality of the solution according to the different parameters and number of re-scheduling events. Therefore, in this study, a new rescheduling model was developed with the aim of paying more attention to the practical application and quality of the solution.

2.6 Decision Support Systems

This chapter addresses the industrial challenge and defines the key terminology associated with the current research work. The Industry 4.0 paradigm imposes the need for collaborative manufacturing systems. The Industrial Internet of things (IIoT) is used as the protocol for making heterogeneous distributed systems interact efficiently using an event-driven framework [108]. These types of collaborative systems heavily depend on data sharing, but the key components are autonomous or semiautonomous tools capable of taking decisions based on collected data [109] [110].

2.6.1 Types of DSS systems

DSSs can be clustered into six distinct categories: *model-driven*, *data-driven*, *communication-driven*, *document-driven*, *knowledge-driven*, and *relative* DSSs [111]. Model-driven DSSs focus on the simplification of a particular activity and evaluate alternative actions that can be taken. While comparing different possible outcomes, each scenario is assigned a probability and an output performance [111]. Data-driven DSSs exploit external and companywide historical information and retrieve relevant data for a specific decision-making process [112]. Communication-driven DSSs leverage communication and information technologies to gather and share information [113]. This enables more efficient collaborations among different groups, both inside and outside the organization [114]. Document-driven DSSs are able to quickly and efficiently retrieve information from available documents such as text files, images, videos, and sound recordings and leverage these documents to support decision actions [115]. Knowledge-driven DSSs englobe data-mining systems and empower computer-based decisions. Two main sub-categories are identified: the first type is rule-based systems, which are built according to the knowledge of experts and replicate their decisions process [116]. The second category implements machine-learning algorithms, neural networks, and artificial intelligence technologies [117]. These systems can conduct various experiments and compare results using evaluation functions. This allows autonomous and self-supported decision taking.

2.6.2 Supplier–client agreements

Supplier agreements play an essential role in defining the expectations of the client. When entering into an agreement, three essential objective functions are identified: the product price, quality, and lead time [118]. Depending on the nature of the supplier–client agreement, quality and delay penalties as well as rewards can be predefined. Delay costs are caused by the offset between promised and attained delivery time. Two cost models are identified: the first is defined as a fixed cost independent of the delay time [119] and the second is defined as a delay cost per unit time of delay resulting in a linear cost behavior [120]. Similarly, quality costs are caused by the offset between promised and attained quality of the order. The associated penalty is defined as the cost per unit of nonconforming product [121]. In addition, three types of defects can be defined individually in customer contracts, namely critical, main, and minor [122].

2.6.3 Actions types

Under the goal of reducing defects, manufacturing companies can implement direct and indirect actions. The latter are a set of actions and decisions that do not directly impact the performance of the process. Their effects are either delayed or measured in auxiliary processes. The indirect actions can be preventive or predictive, namely the maintenance of equipment [123], training of operators [124], and inspection of raw materials [125]. On the other hand, direct actions have unambiguous effects on processes and can be in-process or post-process actions. In-process actions include process and wear monitoring with sensor systems [126], in-line control, and the monitoring of complementary devices [127]. Post-process actions include post-compliance analysis and post-process tuning of equipment and production parameters.

2.6.4 Relevant research works

Quality control is an integrative aspect of manufacturing [128]. Hence, much research has been conducted to address the minimization of defects. The scientific investigation in this scope has revealed different decision and quality control systems. For the purposes of this research, the 14 most relevant research works were selected to be presented and are summarized in Table 1 according to their scope.

Firstly, M. Farooq et al. investigated the cost of quality trade-offs by modeling the costs of inspection, scrap, warranty, rework, and loss of goodwill. Their results on the selected scenario had proven overall costs savings [129]. Moreover, Sarkar and Saren considered production inspection policies and the resulting costs by taking into account the warranty costs [130]. Similarly, G. Levitin et al. studied the cost-effective scheduling of inspections and developed probabilistic models for evaluating the performance of their system [131]. V. Hirsch et al. investigated a DSS based on data analytics that effectively helps operators in fault diagnosis and quality control [132]. Their research was specifically oriented toward a system in the end of line testing. More specifically, the system listed the faulty products and ranked them according to the likelihood of being defective. This helped the operators in their final inspection and quality control tasks. Moreover, D. Soban et al. explored the use of visual analytics for manufacturing process decision making [133]. This research was targeted for process optimization in high-pressure die casting. In the proposed configuration, operators and decision makers can work with large datasets through visual analytics and find the optimum set of parameters to minimize defects. Similarly, M. Gewohn et al. proposed a simple visualization of product quality to enable quality benchmarking [134]. Moreover, A. U. Haq and D. Djurdjanovic focused their research on predicting the defect level in semiconductor manufacturing using virtual metrology concepts [135]. Their concept leverages production and sensor data for feature extraction and predicts the defects in semiconductors. Furthermore, J. Lindström et al. investigated an intelligent system combining predictive maintenance and continuous quality control [136]. The integration of both systems allows not only the assessment of defect likelihood but also the evaluation of the equipment state of wear for evaluating the root cause of defects and prioritizing maintenance tasks. Furthermore, T. Vafeiadis et al. proposed an early stage DSS that facilitates the inspection and condition monitoring processes [109]. This enables defect likelihood classification and allows quick diagnostics. Teti published research on methods of signal processing and decision making in terms of suggesting the necessary corrective actions to optimize the process for ZDM in machining [137]. Another study was conducted to model a supply chain by taking into account the cost of quality [138]. This approach considered the error rate of inspection and the fraction of defects and computed the prevention, appraisal, and failure costs. Similar to the cost-effective decision thinking applied in the present paper, the authors justified preventive and appraisal activities only if they generated financial benefits. In a similar manner, R. Lopes studied the integration of quality inspection with preventive maintenance and buffer stocks [139]. The objective was to minimize the total cost per item given the quality constraints and costs. This model considered that a fraction of the manufactured products were inspected and rework was performed if defects were identified. In addition, a buffer stock was established to respond to the demand during preventive maintenance. Jafari-Marandi et al. developed a cost-based DSS to describe the cost of microstructural defects and suggested the necessary actions during the manufacturing process in additive manufacturing parts [140]. Furthermore, a cloud-based, knowledge-enriched framework was proposed by Mourtzis et al. for improving machining efficiency using data from the monitoring of the corresponding machine tool [110].

Based on the investigation of relevant works, one can observe that previous research has mainly focused on quality control for verifying that the produced productions meet specifications; however, no attention has been paid to the defective products. Furthermore, only one research work identified combining the scheduling problem with cost models to have more cost-effective inspection strategies [131]. Additionally, the research work presented by Jafari-Marandi et al. focused on repairing defective parts during the additive manufacturing process [140]. This approach is very close to the one followed in the current work; the difference is that the present work focused on the post-process repair of defective parts.

Table 1: Relevant decision and quality control systems

Categories References	Dss	Quality Improvement	Quality Control	Inspection Strategies	Product Defect Prediction	Predictive Maintenance	Preventing Maintenance	Process Optimization	Diagnostics	Cost Based	Scheduling	Action On Defected Parts
[129]		X		X						X		
[130]				X						X		
[131]				X						X	X	
[132]	X		X	X					X			
[133]	X	X						X				
[134]			X	X					X			
[135]			X		X							
[136]			X		X	X			X			
[109]	X		X	X					X			
[137]	X							X				
[138]			X				X			X		
[139]			X	X			X			X		
[140]	X		X					X	X	X		X
[110]	X		X					X	X			

2.7 Digital Twins (DTs)

The precursor of DTs was NASA's Apollo 13, where a mirrored physical system of the spacecraft on Earth was simulated to provide a solution to oxygen tank explosion [141]. NASA's later work on DTs has been considered highly influential in the field of aerospace, where a DT is "an integrated multi-physics, multiscale, probabilistic simulation of an as-built vehicle or system that uses the best available physical models, sensor updates, fleet history, etc., to mirror the life of its corresponding flying twin" [142][143]. In manufacturing, the DT concept was first presented by Grieves on the topic of product lifecycle management, which defines DT as an informational construct about a physical system [144]. The concepts of DT were initially more advanced in the aerospace industry in providing and ensuring safer flights through prognostics and diagnostics. However, with the emergence of Industry 4.0, DTs gained great traction in the field of manufacturing driven by advances in related technologies [145].

Despite the numerous research efforts on DTs from both academia and industry, many scholars have expressed the view that no consensus exists on their definition and characteristics [145][146][147][148]. Therefore, differences exist in the understanding of the DT concept [149][150][151]. Kritzinger el al. proposed an interesting categorization of DTs based on their level of integration, namely digital models (DMs), digital shadows (DSs), and DTs. The main difference between DMs, DSs, and DTs lies in the data flow; a DT must satisfy the condition of bi-directional automatic data flow between the physical and digital systems. Under this categorization, despite the majority of the papers claiming to present DTs, only 18% of them are truly DTs [148]. Jones et al. conducted a thematic analysis to reach a common understanding of DTs by consolidating the common themes and key concepts. The foundational 12 themes are presented in Table 2, and each is mapped to the papers supporting the themes [146].

Table 2: Twelve digital twin foundational themes [146]

No.	Theme	Description
1	**Physical Entity**	A “real-world” artefact; e.g., a vehicle, component, product, system, or model.
2	**Virtual Entity**	A computer-generated representation of the physical artefact; e.g., a vehicle, component, product, system, or model.
3	**Physical Environment**	The measurable “real-world” environment within which the physical entity exists.
4	**Virtual Environment**	Any number of virtual “worlds” or simulations that replicate the state of the physical environment and designed for specific use case(s); e.g., health monitoring and production schedule optimization.
5	**Fidelity**	The number of parameters transferred between the physical and virtual entities, their accuracy, and their level of abstraction. Examples found in literature include fully comprehensive, ultra-realistic, high fidelity data from multiple sources, from the micro-atomic level to the macro-geometrical level.
6	**State**	The current value of all parameters of either the physical or virtual entity/environment.
7	**Parameters**	The types of data, information, and process transferred between entities; e.g., temperature, production scores, and processes.
8	**Physical to Virtual Connection**	The connection from the physical to the virtual environment. Comprises physical metrology and virtual realization stages.
9	**Virtual to Physical Connection**	The connection from the virtual to the physical environment. Comprises virtual metrology and physical realization stages.
10	**Twinning and Twinning Rate**	The act of synchronization between the two entities and the rate at which synchronization occurs.
11	**Physical Processes**	The physical purposes and process within which the physical entity engages; e.g., a manufacturing production line.
12	**Virtual Processes**	The computational techniques employed within the virtual world; e.g., optimization, prediction, simulation, analysis, integrated multi-physics, multi-scale, and probabilistic simulation.

Simulation is another divisive topic among researchers; some believe that DT should place emphasis on simulations [152][153], whereas some argue that DT contains physical, virtual, and connection parts, and virtual space is mapped to physical space through connection parts [154][155]. Tao et al. further proposed a five-dimensional DT model, comprising physical parts, virtual parts, connection, data, and service [156]. Under this model, the theoretical foundations of DTs include the following [157]:

- DT modeling, simulation, verification, validation, and accreditation (VV&A);
- Data fusion;
- Interaction and collaboration;
- Service.

Under this framework, a number of DT-driven applications have been developed, such as smart product design [158], job-shop scheduling [159], and virtual commissioning [160]. Negri et al. reviewed DT papers by categorizing them into possible uses, which are presented in Table 3 [147].

Table 3: Digital twin categories [147]

Support of health analyses for improved maintenance and planning
• Monitoring anomalies, fatigue, and crack paths in the physical systems. • Monitoring geometric and plastic deformation on the material of the physical systems and reliability of the physical systems. • Modeling reliability of the physical systems.
Digital mirroring of the life of a physical entity
• Studying and predicting the behavior and performance by accounting for environmental conditions. • Providing information continuity across different stages of lifecycle c. Virtual commissioning of the system. • Managing the lifecycle of IoT devices.
Decision support through engineering and statistical analyses
• Optimizing system behaviors during the design phase. • Optimizing product lifecycles, knowing the past and present states to predict future performances.

With rapidly developing technologies to support DTs, semantic technologies have been playing increasingly important roles in ensuring the interoperability of DT systems and extracting full values of DTs across the entire lifecycle [161]. Semantic modeling is a promising method for integrating different technologies with different formats, protocols, and standards, which is challenging to address in DT modeling. Therefore, cognitive twins (CTs) emerged as an enhancement of DTs with the capability of managing model versions across lifecycles [162]. Lu et al. developed a knowledge graph-centric framework to support CT development [163]. As an emerging topic, CTs have much to offer for enhancing DT applications.

2.8 Identified research gaps

The implementation of ZDM is critically affecting the production scheduling. This is caused because the number of events that require the scheduling of a mitigation action is increased significantly. As said in chapter 2.1 in ZDM there are two strategies for identifying a quality issue: detection and prediction. The core concept of ZDM implies that a mitigation action should be scheduled for each of the events. Contemporary literature on scheduling tools is not considering the prediction of product defects. The ZDM events are considered as time sensitive which means that the counteractions need to be scheduled at a given point of time otherwise there is no point of implementing ZDM approach. Considering these facts enhanced rescheduling methods are required, designed specifically for ZDM. Those rescheduling methods should be flexible enough in order to be able to address also traditional events such as new orders, maintenance actions etc., as presented in [91]. Additionally, existing scheduling tools are rarely considering product quality as KPI. Furthermore, the introduction of product quality to production management systems has first happened in 2011 [164]. To add on this, a recent literature review on scheduling tools revealed that product or process quality is the least used KPI among the available in literature scheduling tools [165]. To this extent there is the need for scheduling tools that incorporate product and process quality as suggested in [166].

On the other hand, setting up a production system based on ZDM approach is a complex task. Because of the scheduling-oriented problems that are arising the current methods and because of the fact ZDM is an emerging approach, there are no methods for assisting manufacturers on selecting the proper quality-oriented equipment and method for achieving efficient ZDM. Currently, manufacturers are struggling to cope with market needs. The product life cycle is shortened significantly and manufacturers are forced to produce new products faster than ever. This created the need for reconfiguring the production and re-designing quality assurance plans more frequently. There is an urgent need for automated methods for assisting on the design or reconfiguring process in order to achieve efficient ZDM.

3 Research Questions & Research Plan

In this chapter, the research questions are presented. These questions are the drivers for the developments that follow in the upcoming sud-chapters. The research questions are not independent; each relies on the results of the previous questions. In other words, the results of Q1 are used in Q2, and those of Q2 are required for answering Q3. Research Question 1 and 3 are the outcome from the identified research gaps presented in chapter 2.8

3.1 Research Question 1

Q1: How can a scheduling tool satisfy the ZDM objectives?

Production scheduling and production quality are two critical and challenging problems [167]. The rules and goals that scheduling systems must satisfy are constantly changing. The contemporary paradigm of ZDM, which implies that there must be no defective part during production, means that there will be less waste material and less energy consumed, and therefore, the production is eco-friendlier but simultaneously more efficient with higher production quality. The outcome of this research question would be a software scheduling tool that will have incorporated the ZDM objectives. The answer to the research question can be found in Chapter 4, and particularly in chapters 4.2, 4.3, 4.4, 4.5, and 4.8.

3.2 Research Question 2

Research Question 2 was not part of the initial plan of the current research work. It come up as a necessity to move forward and be able to answer Research Question 3. Once Research Question 1 was answered and the proposed ZDM dynamic scheduling tool was developed, initial experiments revealed that although the results were very promising in terms of simulation KPIs the computation time was significant, on average 54 minutes per simulation on a modern 6 core computer. This computation time was prohibiting for the methodology conceptualized for answering Research Question 3. Therefore, the idea of creating a DT for the developed scheduling tool was a viable solution.

Q2: How can a digital twin of the developed scheduling tool be created?

Time, both in general and in particular for the manufacturing environment, is a critical aspect that may lead to success if properly managed or failure if not well managed. On the other hand, simulations are critical for designing a new manufacturing system or for redesigning and optimizing an existing one. In other words, the more simulation data that are available, the better the decisions that can be taken. Simulations, however, can be very time consuming and as well as very costly. Therefore, alternative solutions are required for the acquisition of simulation data without running simulations or by running only a small portion and then creating models for estimating–predicting the simulation results. This research question is focused on developing a method for creating a DT of the developed scheduling tool to assist in designing for ZDM and selecting the most appropriate ZDM strategy for each of the manufacturing processes. The answer to this research question can be found in Chapter 4.12.

3.3 Research Question 3

Q3: How should the developed digital twin model be used to investigate the ZDM alternatives for a specific use case?

Quality control in a manufacturing environment is another critical aspect and, depending on the design of the quality control strategy, can either be very efficient and competitive or not. The process of designing new production systems or reconfiguring existing ones can be highly complex and relies on the expertise of senior employees. To assist in this process, namely the design of a quality control strategy, the current research question was formulated to provide manufacturers with a systematic method for a DT to analyze their future production facility and conclude which quality control configuration is more suited to their case (using the results from Q2) quickly without the need for simulations. This will significantly reduce the time required for designing the quality control strategy and simultaneously improve the accuracy of the results. The answer to this research question can be found in Chapter 5.

3.4 Research plan

To answer the defined research questions, we followed the plan presented in Figure 3. The first step was to define the scheduling problem and clarify the types of objects that will be included into the proposed scheduling tool. Once the object types were defined the relations among the objects should be defined. Using the defined objects and relations the architecture of the proposed ZDM oriented scheduling tool was created to visualize all the modules that need to be developed and define their interactions. The architecture was used as roadmap for the development of the scheduling tool. The development process started by the core engine of a scheduling tool which was the tasks allocation to the available resources taking into consideration the capabilities of each resource and the preemptions that might exist, which was very important to make certain that the produced schedule was feasible. The rest of the identified components were built around the core component, adding a specific feature at each time. At each step the newly developed component was tested to verify that the results produced were correct.

Furthermore, the performance of each individual component and the entire solution were examined to proceed to the simulations with the knowledge that the acquired results were of a certain specified quality. If not, redesign of the scheduling tool, or of a component would be required in order to assure high quality schedule solutions. When this procedure was finished successfully, the next steps were the collection of the industrial data to be used for demonstration purposes and the formulation of the data into the data model of the developed scheduling tool. In parallel, the design and development of the methodology for creating the DT model could take place. When the method was ready, the next step was the design of the experiments to create the DT model for a specific real industrial case. Using the collected data and the developed methodology, the use case of the DT model was created. Before the use of the created DT model its performance must be validated. The validation method used was to create a set of random sets of DT control parameters and plug each set to both the scheduling tool to perform an actual simulation and to the DT model and compare the results. The DT is a digital model of the scheduling tool. This means that the DT can predict what the result of the actual simulation would be. If the accuracy calculated was at acceptable level then the DT model could be used and move forward to the next and final step of the current research. This step corresponds to the mapping of the ZDM strategies for the specific industrial use case. Once this is done, the manufacturer can use those mappings to select the most suitable ZDM strategy and ZDM parameters.

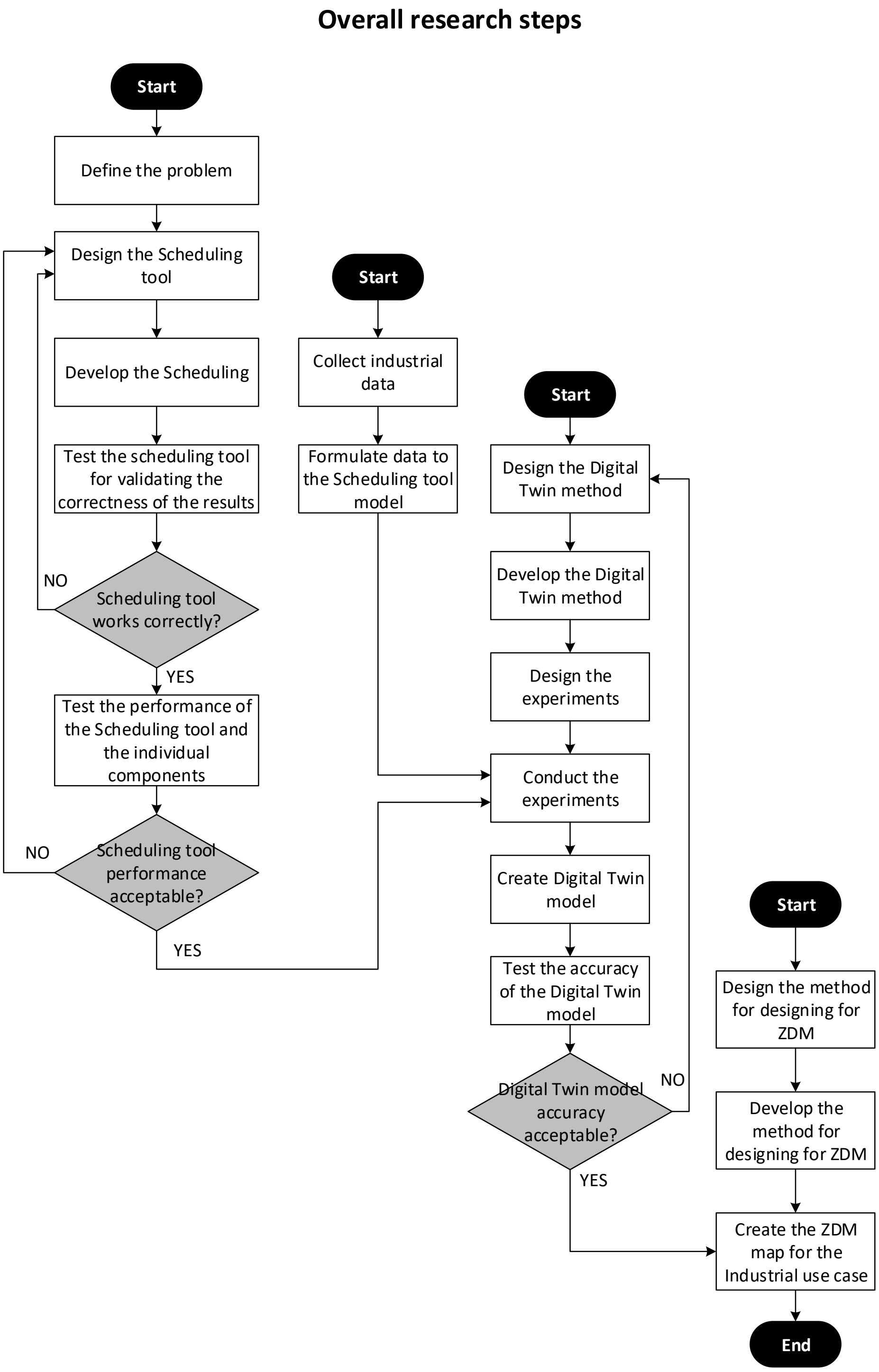

Overall research steps
Start
Define the problem
Design the Scheduling tool
Develop the Scheduling
Test the scheduling tool for validating the correctness of the results
NO
Scheduling tool works correctly?
YES
Test the performance of the Scheduling tool and the individual components
NO
Scheduling tool performance acceptable?
YES
Start
Collect industrial data
Formulate data to the Scheduling tool model
Start
Design the Digital Twin method
Develop the Digital Twin method
Design the experiments
Conduct the experiments
Create Digital Twin model
Test the accuracy of the Digital Twin model
Digital Twin model accuracy acceptable?
NO
YES
Start
Design the method for designing for ZDM
Develop the method for designing for ZDM
Create the ZDM map for the Industrial use case
End

Figure 3: Research plan and steps for conducting this study

An important element of the current thesis is that the approach followed for all the developments was the data-driven approach and not an analytical one. Manufacturing environments have become very complex, volatile and require frequent changes to cope with market need, fact that makes analytical solutions impractical and difficult to use [168][169]. From predicting the future events to designing new products, computational models have played a crucial role in describing the behaviour of those complex natural processes and yielding valuable insights to guide decision-making. In many cases, computational models appear as a set of mathematical equations (e.g., partial differential equations). For quite a long time, however, those equations can only be simulated for a handful of simple academic problems, thus possessing limited values to help people quantitatively understand reality. Starting from the mid-20th century, this situation has improved significantly, thanks to the rapid development of numerical algorithms and powerful computers. For that reason, the data-driven approach was used throughout the entire thesis. Currently, in the manufacturing domain data are generated from various sources. On the other hand, in the context of Industry4.0 concept many technologies have been developed for analyzing the manufacturing data and offering solutions that were not possible with analytical models [170]. Another reason that contributed to the decision of data-driven approaches is that industries have shown significant interest for data-driven approaches because of the capabilities that can offer compared to analytical approaches [171].

4 Scheduling Tool Development

This chapter is devoted to describing in detail the architecture, individual components, and functionalities of the developed scheduling tool. Initially, a standard scheduling tool was developed and tested, as described in [47], to be able to schedule simple tasks. Once this had been done, the individual components regarding the ZDM were developed and integrated into the scheduling tool to add functionalities step by step. At the end of each integration round, experiments were conducted to validate the results and performance.

The design of the scheduling started from the definition of the specification that the desired scheduling should have. The specifications were divided into two categories: those related to the operation of the scheduling tool and those related to the implementation of the ZDM to the scheduling tool [43]. Regarding the operation of the scheduling tool, the main decision that was taken was what types of manufacturing systems the scheduling tool would be able to schedule. The decision was that discrete event manufacturing systems fit the ZDM concept better; therefore, the developed scheduling tool should be able to simulate any type of discrete event manufacturing system.

The problem to solve is formulated as follows (notations at Table 4): a set of Order $R=\{R_k|k=[1,r]\}$, each Order being composed of different operations $O_k=\{O_{j,k}|j=[1,n]\}$, must be manufactured on a set of machines $M=\{M_i|i=[1,m]\}$. $O_{j,k}$ is the jth operations of the order R_k. P_{ij} is the processing time of operation j on machine i. In the same way, the setup time of all operations on each machine is S_{ij}. The machine setup time of each operation depends on the machine's previous operation. Several assumptions were made, such as that once the operation has started the machine cannot be interrupted; that each machine can only handle one job at a time; and that all orders and machines are available at zero time. Each customer order has a due date $D_{k,}$ which should be respected as much as possible. When an Order R_k is selected, all these operations are applied to the machines to complete R_k. Let $\pi=\{\pi_1,...,\pi_m\}$ represent a schedule and $\pi_i=\{O_{i(1)},\ldots,O_{i(n)}\}$ be the sequence of the operations on the machine M_i. Thus, $O_{i(n)}$ represents the nth job assigned to the ith machine. Operations have precedence constraints, which means that some operations should be done before others for them to be feasible. Based on the Bill of Process (BoP), for each operation a number (task level) is assigned to designate the manufacturing sequence. Table 20 illustrates the task levels for the specific product analyzed.

Table 4. Notation of problem definition

Symbol	Description
$R=\{R_k\|k = [1,r]\}$	Set of Order
$O_k=\{O_{j,k}\|j=[1, n]\}$	Set of Operation of order R_k
$M=\{M_i \|i=[1,m])\}$	Set of Machines
$\pi=\{\pi_1, ..., \pi_m\}$	Schedule
$\pi_i=\{O_{i(1)},\ldots,O_{i(n)}\}$	Schedule of machine i
P_{ij}	Processing time of operation j on machine i
S_{ij}	Setup time of operation j on machine i
m_p	Number of possible machines for one operation
Length	Variable used to balance production
M	Machine

The current scheduling tool is formulated using the following objects: task, machine, order and customer. Each of the object holds its own attributes. The main and most important object is the task object. The tool should be able to support multiple tasks to schedule and to offer the possibility to add precedence constraints to be able to define any type of product structure. Each task is a job that must be fulfilled and is characterized by a number of attributes, such as taskID, order number, type, in which machines can be performed, the processing time required etc. Figure 4 presents the state diagram of the task object. Once a new order comes the corresponding task instances are created and waiting to be release to the shopfloor. When the precedence constrains of a task are fulfilled the task is unlocked and it is added to the tasks that can be released. Once the scheduling method decides that the task should be assigned and there is an available machine, the task is assigned. Once the task has finished and the quality is inspected, if the quality is acceptable the task is characterized as finished, if not there are two options, in case the task is repairable the task is repaired and then is characterized as finished or it is sent back as not released because it must be performed from the begging in order to mitigate the defect.

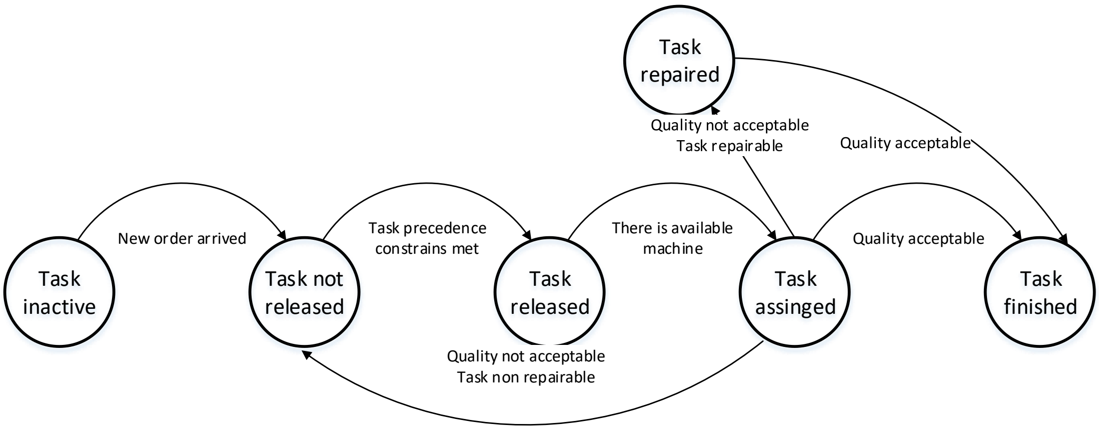
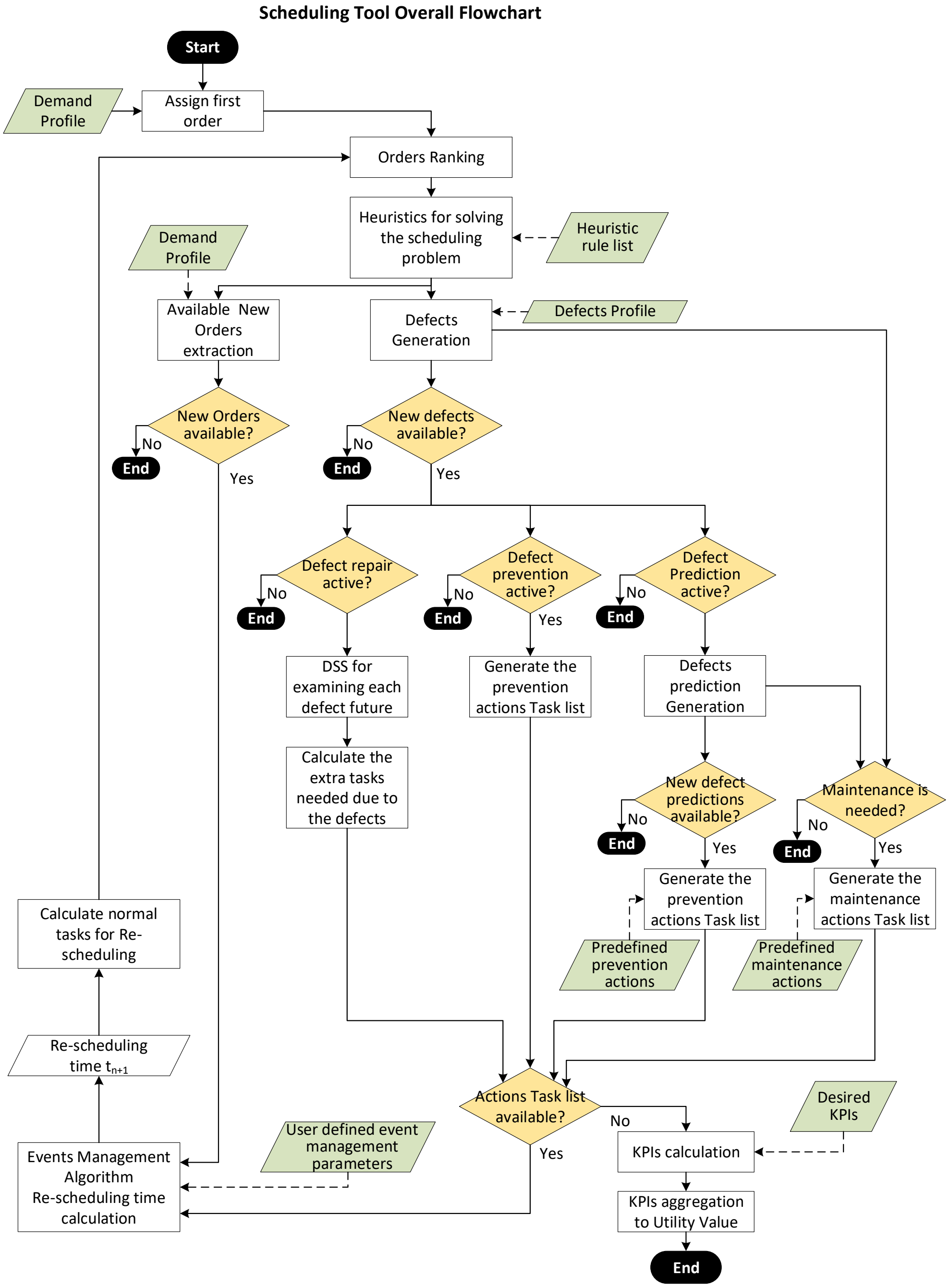


Figure 4: Task object state transition diagram

There are three types of task objects: normal, repairing and maintenance tasks. All the rest of the objects are surrounding the task objects. Other types of objects are machine objects where we have normal, inspection and repairing machines. The tool should be able to handle different orders with different order characteristics, such as order placement day, due date, and quantity. Each order is placed by a "customer" who must have some parameters to be able to rank them and create a system for serving good, loyal, or customers with high volume orders faster. The tool should be able to handle multiple machines and each machine should be able to perform one or more of the defined tasks. This way it would be possible to simulate any discrete event manufacturing system. The tool should be able to calculate multiple criteria and KPIs and have the flexibility of adding or eliminating criteria and KPIs to fulfill the needs of any manufacturer. Another important functionality that the tool should have is a system for deciding when to reschedule and which events should be included in each rescheduling round. The tool should be able to save the scheduling results for later use and also to load data from the predefined data structure file. Additionally, each simulation should require an acceptable amount of computation time, or otherwise the tool would be unusable. Finally, the tool should be able to solve the scheduling problem using different types of heuristic rules, and the operator should be able to select which heuristic rule he or she wants to use.

The first ZDM-related specification that the scheduling tool should have is a function that generates defects according to a defined profile or randomly. This is the basis for the implementation of the ZDM into the simulation engine, because in ZDM, everything starts and ends with product defects. The tool should be able to generate defect predictions using the generated defect list from the previous component. With these implemented, the ZDM triggering factors next in row are the actions that need to take place. Therefore, the tool should be able to decide whether to repair or discard a defective part. Furthermore, the tool should be able to generate preventive actions, which are either machine tuning or machine small maintenance to prevent future defects. The tool should also be able to suggest complete maintenance actions based on the generated defects and product quality levels.

Based on those specifications, the scheduling tool was developed. Figure 5 illustrates the complete overall flowchart showing how the individual components are connected. The basic principle of the developed tool is that it acts as a simulation tool for a period defined by the demand profile the user has defined. This means that one simulation run consists of several scheduling iterations to complete all the orders and also until there are no more ZDM-triggering factors such as new product defects or new defect predictions.



Scheduling Tool Overall Flowchart
Start
Demand Profile
Assign first order
Orders Ranking
Heuristics for solving the scheduling problem
Heuristic rule list
Demand Profile
Available New Orders extraction
Defects Generation
Defects Profile
New Orders available?
No
End
Yes
New defects available?
No
End
Yes
Defect repair active?
No
End
Defect prevention active?
No
End
Yes
Defect Prediction active?
No
End
DSS for examining each defect future
Generate the prevention actions Task list
Defects prediction Generation
Calculate the extra tasks needed due to the defects
New defect predictions available?
No
End
Yes
Maintenance is needed?
No
End
Yes
Generate the prevention actions Task list
Generate the maintenance actions Task list
Predefined prevention actions
Predefined maintenance actions
Calculate normal tasks for Re-scheduling
Re-scheduling time t_{n+1}
Actions Task list available?
No
Yes
Desired KPIs
KPIs calculation
KPIs aggregation to Utility Value
End
User defined event management parameters
Events Management Algorithm Re-scheduling time calculation

Figure 5: Scheduling tool – simulation engine overall flowchart

The whole process starts with the assignment of the first order in time t=0, and since there is only one order, no order ranking occurs. Next, the selected heuristic rule solves the scheduling problem and provides a specific sequence for the tasks that compose the first order. Based on this task sequence and the machine characteristics, the defect generation module calculates whether it will generate defects for the time period defined by the produced schedule. If no defects are generated, this branch is ended and no actions are required; if yes, the defects list is forwarded to the next modules for further analysis. At the same level, the tool checks if available orders are left for assignment. Again, if there are no remaining orders, the orders branch is ended. On the other hand, if yes, the list with the remaining orders list is forwarded to the events management module. In the event of defects, there are three options available that act in parallel: (1) the DSS tool for deciding whether to repair or discard the defective parts; (2) the generation of prevention actions because of the detected defects; and (3) the prediction of future defects based on the generated defects. The outcome of all three is a list with tasks that are required to counteract the effects of the detected defects or the effects that might have future defects.

Furthermore, based on the generated defects and the generated predicted defects, maintenance might be required for the specific machine to regulate defect generation. If any of the four modules regarding orders, defect repair, prevention from detected defects, or defect prediction has generated action tasks, the process continues with all the action tasks to be forwarded to the events management module, where it is calculated which events are going to be considered in the upcoming scheduling iteration and which are not. Moreover, the events management module calculates the time the rescheduling should occur, and based on this time, all the tasks from the current schedule that start after the rescheduling time are added to the task list for rescheduling alongside the action tasks from the previous modules. At this point, once complete iterations of the simulation have been finished, the process starts again but with updated data to consider. The simulation is ended when there are no more new orders or action tasks. In that case, the final KPIs are calculated for each order separately. For the current simulation tool, many different KPIs were developed as the tool was evolving. At each step that simulations are performed, the used KPIs are explained in detail. Moreover, at the final simulation runs for the creation of the DT of the scheduling tool, the KPIs set are presented separately in chapter 4.10. The proposed scheduling tool was developed using MATLAB software.

4.1 Uncertainty consideration

Uncertainties in any manufacturing process lead to deviations between nominal result and their actual counterpart result. Differences from nominal values may cause variations to the final outcome of a process leading to undesired results [172]. Therefore, it is important to take uncertainty under consideration to the current problem because is a key factor for accurate simulations. In manufacturing environments uncertainty is everywhere and influences many different systems. In the current study uncertainty was introduced to many different components of the developed scheduling tool. The approach that was followed was using uncertain random variables. An uncertain random variable is a measurable function from a probability space to the set of uncertain variables. In other words, an uncertain random variable is a random element taking ‘‘uncertain variable’’ values [173]. This approach might be simple but at the same time is the worst-case scenario for a manufacturing system, because of the randomness. Usually in some cases patterns can be found that can reduce the uncertainty and be able to estimate or predict an event, but is not always the case.

In the scope of the current study uncertainty was introduced to the following modules of the scheduling tool: defects generation module, defects prediction module, tasks processing time assignment, quality inspection successfulness, defected part repairability and prevention action success. All the above-mentioned modules of the scheduling tool have the uncertainty incorporated in the form of random numbers. Further details about each component will be given at the corresponding chapters.

4.2 Defect Generation Module

A defect generation module is one of the most crucial components for the current simulation tool. This module is related to the ZDM implementation and has the goal of approximating the real defect generation in an industrial environment as closely as possible. Figure 6 illustrates the flowchart regarding the defect generation procedure.

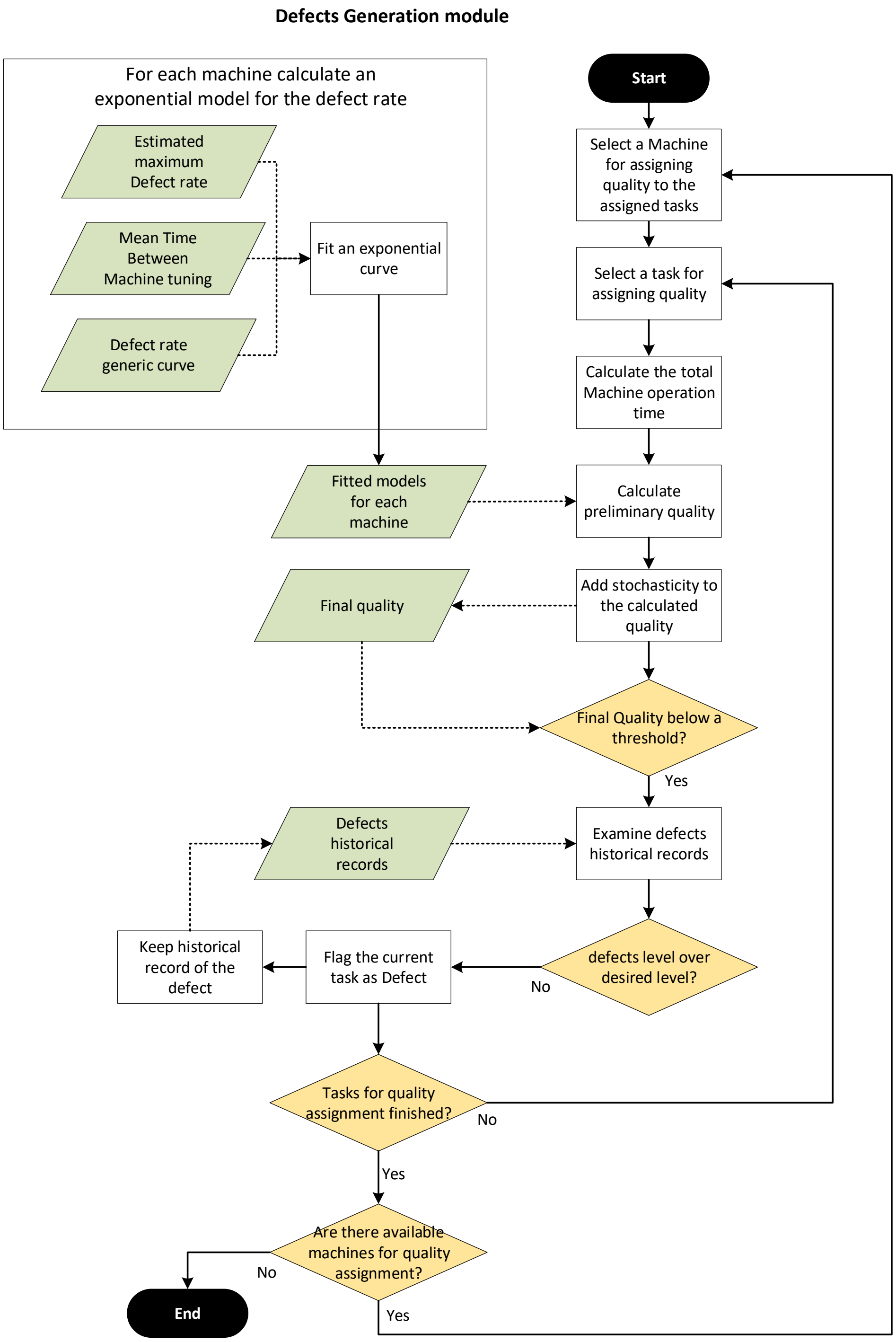

Defects Generation module
For each machine calculate an exponential model for the defect rate
Estimated maximum Defect rate
Mean Time Between Machine tuning
Defect rate generic curve
Fit an exponential curve
Fitted models for each machine
Final quality
Defects historical records
Keep historical record of the defect
Start
Select a Machine for assigning quality to the assigned tasks
Select a task for assigning quality
Calculate the total Machine operation time
Calculate preliminary quality
Add stochasticity to the calculated quality
Final Quality below a threshold?
Yes
Examine defects historical records
defects level over desired level?
No
Flag the current task as Defect
Tasks for quality assignment finished?
No
Yes
Are there available machines for quality assignment?
No
Yes
End

Figure 6: Defect generation module flowchart

The defect generation module relies on a generic curve regarding the defect probability versus the machine’s total operational time, which was extracted from the industrial case under investigation. For each machine, there is a mean time that the machine requires tuning (MTBT) or small maintenance to avoid a defect rate beyond the desired levels. Before each simulation run has started, the tool based on the generic defect generation curve calculates the machine-specific defect generation curves based on the corresponding data of each machine. This procedure is performed only once at the beginning of the simulation, and then the models are saved for use throughout the entire simulation period. Those models are a function of the current total machine operation time, and the output is the preliminary task quality. The quality is measured in the form of a percentage, with 100% being the ultimate quality, and at lower thresholds below that the tasks is considered defective.

In every rescheduling iteration, the defect generation module is run to generate the estimated defects, which will trigger the ZDM components and counteract the problems caused by the defective parts. For each task assigned to each machine, the defect generation module calculates the current machine operation time including the processing time for the current task, and this value and the defect generation fitted model for the current machine are used to calculate the preliminary quality. Those models can be created using specific data regarding the defect rate of a specific equipment. In Figure 7 are illustrated some examples of how those fitted to the real data curves can look like. On the x-axis there is the total operation time of the machine (TOT) counting from the last maintenance operation. Each time maintenance is performed the TOT is zeroed. If assumed that 100% quality is 1 then the y-axis shows how much quality is lost related to the TOT increase. The first and second plots in Figure 7 are illustrating an exponential and a linear behaviour to the quality loss as the TOT increases, which have a constant upwards trend. In the manufacturing environment it is not always possible to derive to such graphs and the situation looks like the third plot which illustrates a random behaviour of the quality loss compared to the TOT. In order to address this situation, the approach followed in the current thesis is a data-driven approach and use case specific, each machine has its own characteristics and, as a consequence, it is impossible to generalize or describe with analytical models [168][174]. Therefore, empirical formulas were used to describe the loss of quality compared to the TOT.

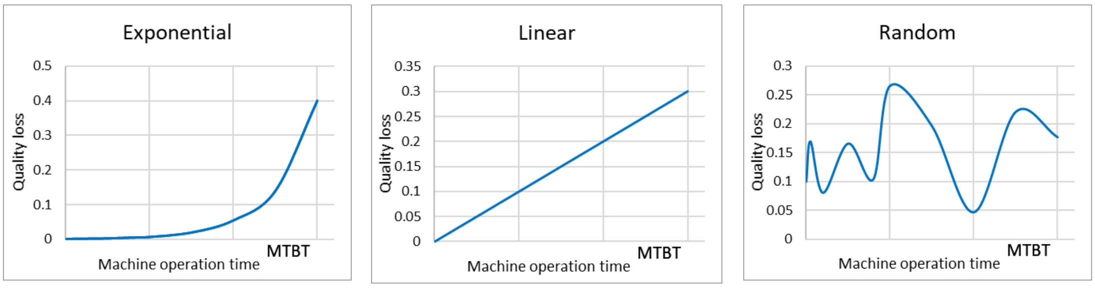


Figure 7: Examples of data driven quality deterioration curves

Because in the real-world defects do not occur in a deterministic way but in a stochastic way, the need exists in this case to add uncertainty to the calculated preliminary quality to refine it and approximate the reality more accurately. This is achieved using equation (2), which is an empirical formula, where from the preliminary quality (calculated by equation (1)) is subtracted the product of a random number $Q \in [0,1]$ with a weight W ∈ ▯. The weight W is responsible for fitting the result closer to a specific real case. The outcome is the final task quality.

$$PreliminaryQual = 1 - f(TOT) \tag{1}$$

$$FinalQuality = PreliminaryQual - W * Q \tag{2}$$

If the calculated final quality is below a threshold defined by the user, the task under investigation is considered a defect. At this point the algorithm examines the historical records of defects for that particular machine to verify that the total number of defects at the current stage are not exceeding the desired–defined defect level by +10%. If this condition is met, then the task under investigation is flagged as a defect for later modules to consider. If the total number of defects exceeds the defined threshold, then the defect is not considered and the tasks remain flagged as healthy. When all the tasks of a machine have been assigned quality, the tool moves to the next machine until all the tasks of all machines have been assigned qualities.

4.3 Defect Prediction Generation Module

Defect prediction is another important module for the success of the current tool. This module is also related to the ZDM concept and represents one of the most promising ZDM strategies. Defect prediction can be implemented with a variety of methods such as machine learning, artificial intelligence, or similar approaches. In the current research, the defect prediction module simulated only the outcome of the actual defect prediction algorithm and did not perform any predictions. In other words, the defect prediction module is considered a black box and the only concern is the input and output. The method for simulating the outcome of defect prediction is rather simple but efficient and successfully achieves the assigned task. The developed tool has the possibility of assigning defect prediction instances after each machine and with different prediction characteristics. Each defect prediction instance has two key control parameters: prediction accuracy, which denotes how accurate the implemented prediction algorithm is and is measured in the form of a percentage, and the prediction horizon, which denotes how far ahead in terms of time the theoretical prediction algorithm can predict with the defined accuracy.

The defect prediction module process is illustrated in Figure 8, based on which the defect prediction module relies on the defect generation module described in chapter 4.2. Once the defect generation module has created the defects list, then this list is passed to the defect prediction module and each defect is examined separately. The generated defect list concerns all the defects that will occur until the end of the current schedule if no actions are taken. For each of the defects on the list, a random number is generated between [0,1], which denotes the temporary accuracy of the actual defect prediction algorithm when predicting the current defect. This number is then compared with the defined actual accuracy of the defect prediction module, and if the random number is lower than or equal to the actual accuracy, then the algorithm flags that defect that the prediction will be successful. If the prediction is flagged as non-successful, then the defect will occur and no measures will be taken to avoid it. The process ends when all of the defects on the provided list are flagged with the outcome of the hypothetical prediction algorithm. In the real world, these processes occur in real time and a few each time, whereas in the current work, the defect generation and defect prediction occur in batches until the end of the calculated schedule and based on the outcome of the event management algorithm (chapter 4.8). Some of those events are considered for the next rescheduling iteration and all the others are deleted because from that moment the schedule is going to change from that point, and after the generated defects and predictions will not be aligned with the new schedule, and therefore they are deleted. When the new schedule is created, the process of defect generation and by extension that of defect prediction is performed all over again with new defects assigned without being influenced by the previous results.

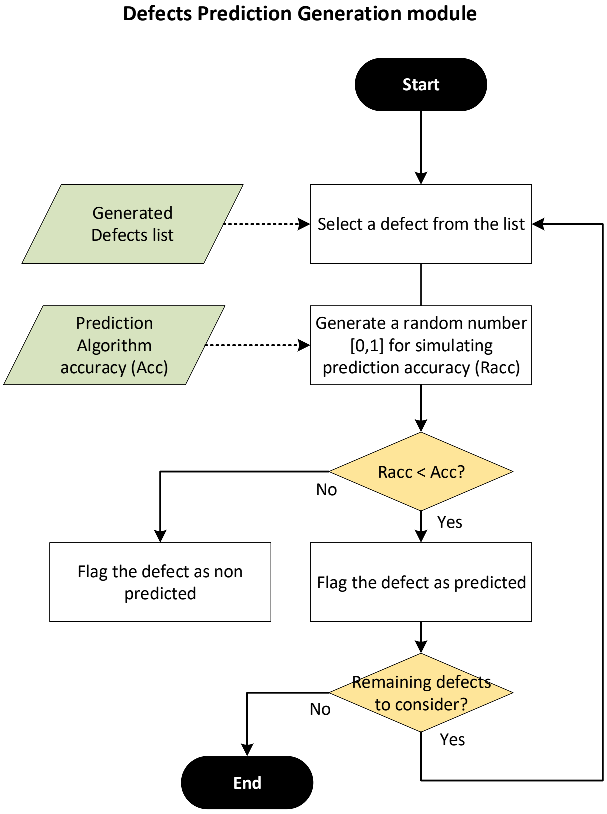

Defects Prediction Generation module
Start
Generated Defects list
Select a defect from the list
Prediction Algorithm accuracy (Acc)
Generate a random number [0,1] for simulating prediction accuracy (Racc)
Racc < Acc?
No
Yes
Flag the defect as non predicted
Flag the defect as predicted
Remaining defects to consider?
No
Yes
End

Figure 8: Defect prediction flowchart

4.4 Detect Repair DSS Cost Model

The DSS is designed to automate the decision process when a defect has been detected in the production line, and its aim is to optimize costs, quality, and lead time. Figure 9 illustrates the flowchart of the proposed DSS tool, which describes the decision-making process for selecting one of the three potential solutions. The DSS tool is integrated into a dynamic scheduling tool [43], which is responsible for generating the production schedule. When the production schedule is realized, numerous defects are detected via an in-line inspection system. Those defects are listed from the earliest to the latest, and the developed DSS tool calculates the three alternative solutions costs (deferral, disposal, and repair). These costs are calculated using equations (3), (4), and (5), respectively. Then, all three calculated costs are compared and the alternative with the minimum cost is selected for the defect under investigation. If there are more defects in the defects list, the same procedure is applied until all the defects in the list have a decision. This means that the DSS will work in series for suggesting decisions, which is due to the fact that the decision of the next defect is affected by the decision of the previous one. The next step is to dispatch the actions needed, if any back to the scheduling tool to be scheduled into the next rescheduling procedure. It should be reminded that our model is built upon the consideration that all parts are inspected with automated systems. These cost functions are based on the three costs pillars: cost of deferral (A), disposal (B), and repair (C). All the parameter names and descriptions are summarized in Table 5.

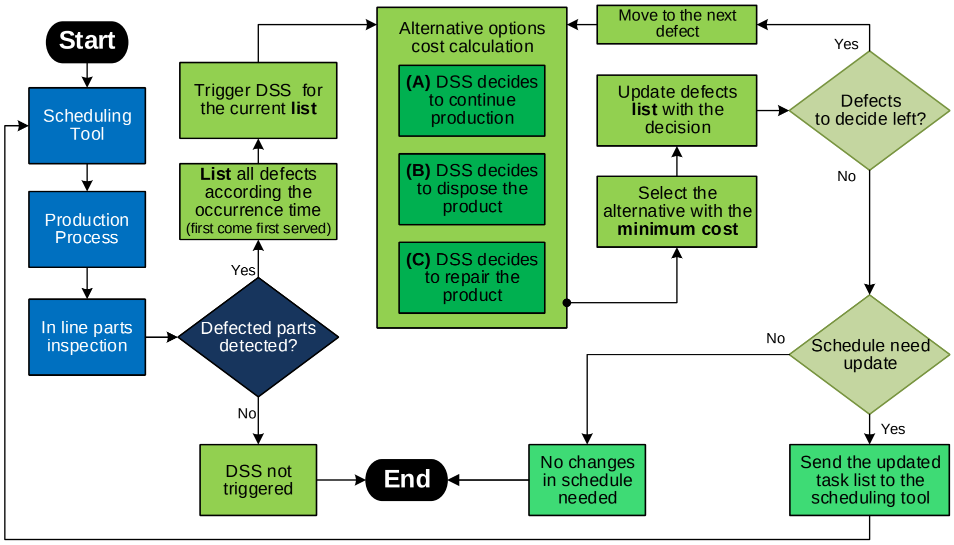

Start
Scheduling Tool
Production Process
In line parts inspection
Defected parts detected?
Yes
No
DSS not triggered
End
List all defects according the occurrence time (first come first served)
Trigger DSS for the current list
Alternative options cost calculation
(A) DSS decides to continue production
(B) DSS decides to dispose the product
(C) DSS decides to repair the product
Move to the next defect
Update defects list with the decision
Select the alternative with the minimum cost
Defects to decide left?
Yes
No
Schedule need update
No
Yes
No changes in schedule needed
Send the updated task list to the scheduling tool

Figure 9: Flowchart of the defect decision support system

Table 5: Abbreviations list

AD	:	Accepted Delay For Customer Order	**M**	:	Order Number (Id)
AIT	:	Average Inspection Time	**M**	:	Total Number Of Orders
AQL	:	Attained Quality Level	**MC**	:	Material Cost
AQT	:	Accepted Quality Threshold	**MOC**	:	Machine Operational Cost
ART	:	Average Repair Time	**N**	:	Total Number Production Stages
BS	:	Batch Size	**N**	:	Number Of Production Stages
C	:	Job Completion Time	**ND**	:	Number Of Defective parts
CODI	:	Cost Of Delay Due To Inspection	**OC**	:	Order Criticality
CODR	:	Cost Of Delay Due To Repair	**OV**	:	Order Value
CODS	:	Cost Of Delay Due To Scrap	**PD**	:	Penalty Of Delay (Customer)
COI	:	Cost Of Inspection	**PQ**	:	Penalty Of Quality (Customer
COQ	:	Cost Of Quality	**PC**	:	Product Cost
COR	:	Cost Of Repair	**PFIX**	:	Fixed Penalty Sum
COS	:	Cost Of Scrap	**PT**	:	Process Time
D	:	Disposal Cost	**Q**	:	Order Size
DAVG	:	Buyer’s Average Demand Rate	**R**	:	Runtime Of Process
DC	:	Delay Cost Per Unit Time	**RP**	:	Cost Of Replacement Part
DD	:	Due Date For Customer Order	**RT**	:	Repair Time
DR	:	Disposal Cost Of Sub-part	**RTD**	:	Delay Caused By Repair Time
FIP	:	Forecasted Inspected Parts (%/Year)	**SC**	:	Buyer’s Critical Stock
HR	:	Hourly Rate For Production Process	**SD**	:	Delay Caused By Scraping
I	:	Production Step	**SS**	:	Buyer’s Safety Stock
IEC	:	Inspection Equipment Cost	**SUC**	:	Setup Cost Per Unit Time
IV	:	Annual Inspection Volume	**SUT**	:	Setup Time Required
J	:	Product Type	**W**	:	Weight Factor
K	:	K-Th Item In The Batch	**Tʏ**	:	Current Year
LC	:	Labor Cost Per Unit Time	**T***	:	Critical Delay Time
LGW	:	Client's Loss Of Goodwill	**TD**	:	Delay Time
LOR	:	Loss Of Reputation			

4.4.1 Decision A: Cost of deferral

This decision, the so-called “Do nothing” approach, decides to proceed to the next step of production. In such a case, the product might still carry an uncertainty around its quality. This is because in automated quality assessment methods, the certainty about the quality of the product is not absolute and lies inside a confidence interval. Consequently, the cost that the product can carry is the cost of quality.

$$A = \mathbf{COQ}(\mathrm{i}, \mathrm{j}, \mathrm{k}, \mathrm{m}) \tag{3}$$

There are two scenarios under this decision option: (a) after the quality assessment of the product, the estimated quality level of the customer order is above the promised quality level. The cost of quality is then null. As such, the cost of deferral is the cost-effective choice and the product can proceed to the next production step. (b) After the quality assessment of the product, the estimated quality level of the customer order is below the promised quality level. Consequently, the cost of quality is nonzero. However, after the comparison with the cost of disposal and cost of repair, the deferral option is chosen as the cost-effective scenario.

4.4.2 Decision B: Disposing the product

This model evaluates the cost of scrapping the product. This function includes the cost of product (*PC*), cost of scrap (*COS*), and cost of delay due to scrap (*COD$_s$*). This delay is caused by remanufacturing the same product after scrapping.

$$\mathrm{B} = \mathbf{PC}(\mathrm{i,j}) + \mathbf{COS}(\mathrm{i,j}) + \mathbf{COD_s}(\mathrm{i,j}) \tag{4}$$

4.4.3 Decision C: Inspecting and repairing the product

The decision to repair is placed inside the same decision path as inspection. This is because the automated inspection system's quality assessment carries a certain uncertainty. As such, prior to repairing the product the operator has to inspect whether the product is repairable. Thus, the common costs associated are the cost of product (PC), the cost of inspection (*COI*), and the cost of delay due to inspection (*COD$_i$*). In addition, in the inspection process we obtain the following sub-decision trees: (a) after the inspection of the operator, the product appears to not have any defects. It is therefore reintroduced back to the production process. (b) After the inspection by the operator, the product is repaired and is reintroduced to the production process. Inside decision C, the ratio of products that are repaired is expressed by the ratio of repair. This value is obtained according to the previous production data. The associated costs in this case are the cost of repair (*COR*) and the cost of delay due to repair (*COD$_r$*). (c) After the inspection by the operator, the product is deemed unrepairable and is therefore scrapped. A new production order is given to compensate the disposal. Inside decision C, the ratio of products which are scrapped is expressed by RS. This value is obtained according to the previous production data. The associated costs in this case are the product cost (*PC*), the *COS*, and the cost of delay due to scrap (*COD$_s$*).

$$\mathrm{C} = \begin{cases} \mathbf{COI}(\mathrm{i,j,t_y}) + \mathbf{COD}_\mathrm{i} & \text{, if not a defect} \\ \mathbf{COI}(\mathrm{i,j,t_y}) + \mathbf{COR}(\mathrm{i,j,t_y}) + \mathbf{COD}_\mathrm{i} + \mathbf{COD_r}(\mathrm{i,j,m}) & \text{, if repairable defect} \\ \mathbf{COI}(\mathrm{i,j,t_y}) + \mathbf{PC}(\mathrm{i,j}) + \mathbf{COS}(\mathrm{i,j}) + \mathbf{COD}_\mathrm{i} + \mathbf{COD_s}(\mathrm{i,j}) & \text{, if non repairable defect} \end{cases} \tag{5}$$

4.4.4 Product cost (PC)

The cost of producing a part in a given stage of production will include different cost functions. The cost of sub-products or cost of materials is given by the necessary raw material price, production equipment setup costs, labor cost, equipment investment cost, and overhead cost. The cost function is given in the formula below and adapted from the formula proposed by [175]. Additionally, equipment costs and overhead costs were introduced into the product cost function.

$$\mathbf{PC}(\mathrm{i,j}) = \left\{ \mathrm{MC(i,j)} + \sum_{\mathrm{i=1}}^{\mathrm{n}} \left[\frac{\mathrm{SUT(i,j)} * \mathrm{SUC(i,j)}}{\mathrm{BS(i,j)}} + [\mathrm{R(i,j)} * \mathrm{MOC(i,j)}] \right] \right\} \tag{6}$$

4.4.5 Cost of quality (COQ)

Penalties related to the cost of quality are given if a mismatch exists between the promised product quality and delivered quality. These penalties include a financial sanction with a decrease in the agreed payment [176]; a reduction in price along with any additional compensation that can occur; a loss of goodwill resulting with a reduction in the future purchases by the customer; and a loss of reputation of the producer and propagation to other clients [177].

$$\mathbf{COQ}(\mathrm{i,j,k,m}) = \mathrm{P_q(i,j,k,m) + LGW + LOR} \tag{7}$$

Furthermore, a certain penalty limit that the supplier is willing to accept is defined. This threshold represents the break-even point of the supplier. Beyond this limit, the supplier suffers financial losses and is therefore unlikely to engage in a transaction. Further explanations regarding the penalty behaviors are given in the following sub-chapters. In all three quality agreements, there is an adjustment factor of 1+OC/10, which modifies the cost according to the order criticality (OC); in other words, the more important the order, the higher the cost penalty would be in the case of order quality lower than that agreed. The OC takes values between 0 and 10, where 10 means a very critical order and 0 means an order of low importance.

Fixed Quality Agreement

$$\mathbf{P_q} = \begin{cases} 0 & \text{if } \mathrm{AQL > AQT} \\ \mathrm{P_{fix}} * \left(1 + \frac{\mathrm{OC}}{10}\right) & \text{if } \mathrm{AQL < AQT} \end{cases} \tag{8}$$

Standard Quality Agreement

$$\boldsymbol{P_q} = \begin{cases} 0 & \text{if } \mathrm{AQL > AQT} \\ \mathrm{P_{fix} + W_1(AQL - AQT) * OV} * \left(1 + \frac{\mathrm{OC}}{10}\right) & \text{if } \mathrm{AQL < AQT} \end{cases} \tag{9}$$

Quality Sensitive Agreement

$$\mathbf{P_q} = \begin{cases} 0 & \text{if } \mathrm{AQL > AQT} \\ \mathrm{P_{fix} + W_2(e^{W_9(AQT-AQL)} - 1) * OV} * \left(1 + \frac{\mathrm{OC}}{10}\right) & \text{if } \mathrm{AQL < AQT} \end{cases} \tag{10}$$

Defining the financial loss suffered by the customer due to lower product quality is a complex and difficult process. Therefore, a suggestion is to include a fixed penalty per unit of defective product in the quality level agreement [177]. The fixed Quality Agreement consists only of the fixed penalty applied to the seller once a certain quality threshold is violated in a specific order. Standard Quality Agreements contain the fixed penalty but also have a linear behavior, which means that the penalty for not meeting the agreed quality increases proportionally to the quality of the order. Quality Sensitive Agreements have a nonlinear behavior in terms of sanctions that are applied. This penalty behavior was adapted from the insights of [178]. Their supply chain model under penalty policies exhibits an exponential sensitivity to the number of defective parts. Consequently, for the quality sensitive agreement model, an exponential penalty behavior was used for the description of the quality incompliance penalty cost.

The penalties applied by the customers are based on the quality level of an order. Given that the DSS quality results are product-based, an order level quality is defined according to the number of defects of a certain order and product type. AQL is the Attained Quality Level and represents the percentage of healthy parts in that specific order. This is calculated from the ratio between the total number of defective parts and the order total parts.

$$\mathbf{AQL}(\mathrm{j,m}) = 1 - \frac{1}{\mathrm{Q(m)}} \sum_{\mathrm{k=1}}^{\mathrm{Q(m)}} \mathrm{ND(j,k)} \tag{11}$$

4.4.6 Cost of scrap (COS)

The COS is determined by the cost of disposing of a product. It includes handling, storage, transportation, and recycling costs [179]. In our model we define a single parameter D for the disposal cost. As such, the COS is given by the cost of product and disposal costs.

$$\mathbf{COS}(\mathrm{i,j}) = \mathrm{PC}(\mathrm{i,j}) + \mathrm{D}(\mathrm{n,j}) \tag{12}$$

4.4.7 Cost of inspection (COI)

The cost of inspection will depend on the equipment cost and variable costs [129] [180]. To define the inspection equipment, cost per product, we first must forecast the number of expected inspections for the current business year based on the inspection ratio of the previous activity year.

$$\mathbf{IV}(\mathrm{i,t}) = \mathrm{FV}\left(\mathrm{i,t_y}\right) * \mathrm{FIP} \tag{13}$$

$$\mathbf{COI}(\mathrm{i,j,t}) = \frac{\mathrm{IEC(i)}}{\mathrm{IV}\left(\mathrm{i,t_y}\right)} + (\mathrm{AIT(i,j)} * \mathrm{LC}) \tag{14}$$

4.4.8 Cost of repair/rework (COR)

The COR consists of fixed and variable costs. The unit COR will depend on the repair time, equipment cost, replacement part cost, and disposal costs [181].

$$\mathbf{COR}(\mathrm{i,j},t_y) = \mathrm{ART(i,j)} * \mathrm{LC} + \mathrm{RP(i,j)} + \mathrm{DR(i,j)} \tag{15}$$

4.4.9 Cost of delay (COD)

The COD depends on different variables such as delivery date, estimated manufacturing time, and accepted delay. The estimated manufacturing time itself can be divided into three distinct categories [182]: no additional COD for conforming products, COD due to repair work, and COD due to product scrapping. As the delay time differs for repairing and scrapping decisions, the penalty of delay P_d is dependent on the consequent actions. In the following sub-chapters, the penalty types and time of delay are discussed.

$$\mathbf{COD}(\mathrm{i,j,k,m}) = \mathrm{P_d}(\mathrm{i,j,k,m,t_d}) + \mathrm{LGW} + \mathrm{LOR} \tag{16}$$

In our analysis, the COD is defined as the rate per unit time per customer order. It is given by the penalty that the customer applies to the producers for the given order. Therefore, the delay of a single product affects the total order. For instance, if one product is delayed in an order of 1000 units, the total order is delayed. The order can only be shipped once the missing product is manufactured. In the scope of this paper, three different types of orders are defined: delay insensitive order, standard order, and delay-sensitive order.

Delay insensitive orders are orders that are less sensitive to delays. Therefore, the agreement is based on a fixed penalty per delay unit time and the behavior of the cost with respect to time is chosen as a linear model. Standard orders are the most common policies used. From the customer perspective, the customers keep a certain safety stock to deal with demand variations and logistical perturbations. This safety stock sits above the minimum required inventory level, referred to as the critical stock level [183]. From the supplier's side, this critical delay time also denotes the time after which the supplier suffers a more significant penalty. As such, an S-function is derived for standard orders for the purpose of substantially penalizing the seller once a critical stock level is reached from the perspective of the buyer.

Delay-sensitive orders are events that are urgent and therefore start carrying significant delay costs right from the beginning of the delay. As the delay time increases, the gradient of the slope starts to decrease. This is because the supplier is unlikely to engage in a transaction resulting in financial loss, meaning that penalties will hit a cap. Therefore, as delay time increases, the delay cost curve converges toward the zero-profit zone of the supplier. W_8 determines the curve of the function, meaning that for a small W_8, the curve will have a more linear behavior, whereas for a high W_8, the steepness at t=0 will be accentuated. This concave delay cost function was derived from the pricing strategy according to the promised delivery lead time conducted by Yina Li, Qiang Lin, and Fei Ye [184]. Although their model focuses on the pricing strategy instead of delay costs, similar results can be interpolated. This assumption can be justified by the results of Baoshan Liu, Xu Guan, Haijun Wang, and Shihua M, in their comparison of the sensitivity function according to the delivery lead time [185].

Delay Insensitive order

$$P_d(t_d) = P_{fix} \tag{17}$$

Standard order

$$P_d(t_d) = \begin{cases} W_4\left\{(T^*)^{\frac{1}{W_5}} - (-t_d + T^*)^{\frac{1}{W_5}}\right\} OV * OC & , t_d < T^* = \dfrac{S_s - S_c}{D_{avg}} \\ W_6 + W_7(t_d - T^*)^{\frac{1}{W_5}} * OV * OC & , t_d > T^* = \dfrac{S_s - S_c}{D_{avg}} \end{cases} \tag{18}$$

Delay-sensitive order

$$P_d(t_d) = W_3 \ln(1 + W_8 t_d) * OV * OC \tag{19}$$

4.4.9.1 *Repair Time Delay (RTD)*

Deciding to repair a product will cause an increase in its production time. This increase can result in the violation of the promised delivery date, causing a COD. More specifically, the repair time delay (RTD) is estimated by summing the following terms: the repair time (RT) required, the total processing time for the product completion, if the defects are not detected at the final production stage, and the accepted order delay (AD). AD is a parameter that is set by the customer and defines the delivery tolerance where no delay penalty is charged. Therefore, the penalty applied by the customer will be obtained by plugging the repair delay time into the corresponding penalty function P_d(RTD*w).

$$\mathbf{RTD}(i, j, m) = \max\left\{\left[RT(i, j) + \sum_{t=n}^{N} PT(i, j) + AD(m) - DD(m)\right]; 0\right\} \tag{20}$$

4.4.9.2 *Scraping Delay (SD):*

To determine the scrapping cost, we apply a similar logic to the cost of repair. If the product has to be discarded, the production time will be increased by the process time. Furthermore, due to the production schedule, the new product cannot immediately be introduced as the next production order scheduled. As such, the penalty applied will be obtained by plugging the scrapping delay (equation (21) into the penalty function P_d(SD*w).

$$\mathbf{SD}(i, j, m) = \max\left\{\left[\sum_{i=1}^{N} PT(i, j) + AD(m) - DD(m)\right]; 0\right\} \tag{21}$$

4.4.10 *Validation of the DSS*

The goal of using the proposed DSS tool is to improve production quality and minimize waste in terms of time and raw materials as well as move one step ahead toward ZDM. In regard to this goal, the developed DSS model was tested against two other scenarios. The first was the current manufacturing practice and the second was a scenario under ideal manufacturing conditions. As current manufacturing practice is defined a scenario that is as close as possible to the manufacturing policy that is followed to the selected industrial case. This means that when a defect is occurred the part is discarded automatically and there is no attempt of repairing the defected part. The ideal scenario denotes the ideal manufacturing conditions without defects or any other type of interruptions to the manufacturing process. In this way, it is feasible to study how much better the proposed tool possibly behaves and simultaneously monitor how far from the ideal scenario the produced results are.

Furthermore, it is important to study the developed DSS tool for different time periods. This is because currently the demand volume and frequency fluctuate a lot and are uncertain. Therefore, companies' competitiveness and profitability rely on their ability to adapt quickly to customers' needs [110]. Manufacturers try to mitigate this problem by producing demand forecasts and projections to take the right decisions at the right time [186]. Moreover, manufacturers are forced to develop tools and procedures to be able to react efficiently and optimize their production systems under this uncertain demand profile [187]. Investigating different time periods would validate that the decisions made by the proposed DSS have positive effects on the production both in the short and medium term. Therefore, the three scenarios (i.e., DSS, current situation, and ideal) were simulated in two different time periods, namely one short-term and one mid-term period.

The simulation results were evaluated based on three KPIs. Each KPI was calculated for each individual customer order. Makespan was the first measured KPI and depicts the completion time of an order, equation (22). The next KPI was the maximum order tardiness, which shows the amount of time an order takes to be finished after the due date, equation (23); in cases where an order is finished before the due date, the tardiness is 0 [47]. This value is a relative value compared with the makespan, which is an absolute value. The final measured KPI was the production cost, which is a sum of the machines' operational cost, the setup cost, raw material costs, and the cost due to delay penalties presented in chapter 4.4.9 in equations (16), (17), (18), and (19), equation (24). All the abbreviations used in equations (22), (23), and (24) can be found in Table 5.

$$\boldsymbol{Makespan}_m = \max\,(C_{m1}, \dots, C_{mM}) \tag{22}$$

$$\boldsymbol{Tardiness}_m = \max\,(Makespan_m - DueDate_m, 0) \tag{23}$$

$$\boldsymbol{ProductionCost}_m = \sum_{n=1}^{N} MC_n + MOC_n * PT_n + SUC_n * SUT_n + COD_n \tag{24}$$

To validate the performance of the developed DSS system, a real-life industrial scenario was used, which came from the semiconductor domain and concerned the production of a specific printed circuit board (PCB) for use in a medical device. The manufacturing processes examined in the current scenario covered the final production stage of the product and were mostly "pick & place" and assembly operations. The selected production stage was configured in a flexible job shop layout [47], and more specifically it is composed of four work centers (WC). WC1 and WC2 were responsible for the "pick & place" and assembly operations accordingly, with three identical machines in parallel. WC3 was the quality inspection station where two identical inspections were installed. WC4 was the repairing center, with three repairing stations. Each repairing station is composed out of multiple machines and equipment, but in the current study are grouped in one machine. Figure 10 and Figure 11 illustrates the shop-floor layout and connections and the bill of processes (BoP) of the product under investigation. This part of the production was selected because the highest defect ratio was mainly observed there of between 5% and 6%. This defect ratio is relatively low, but the PCB under investigation is expensive at around €550 each.

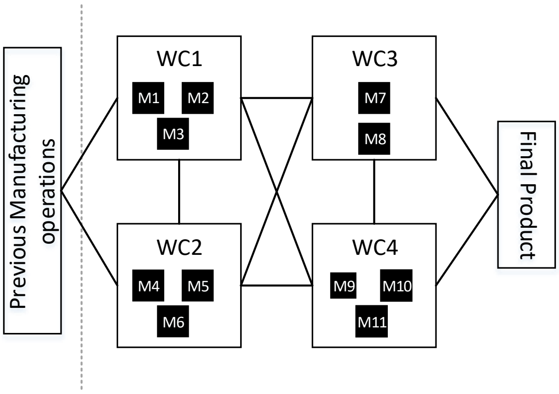


Figure 10: Industrial use case shop-floor layout

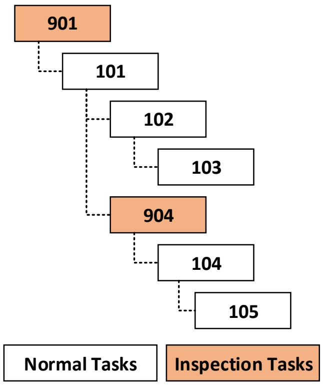


Figure 11: Bill of Processes (BoP)

The current policy is to discard all defective PCBs spotted during manual inspection by an operator, which can be very costly for the organization. In the context of improving and automating the production, an automated optical system was installed to inspect the PCBs at certain points and have more accurate and repeatable results. Furthermore, repair protocols were developed to repair some defective PCBs, but the need existed for an automated DSS system that automatically decides the future of each individual defect based on the cost. Additionally, the orders regarding this particular component are considered delay-sensitive since they are meant to be inside a medical device with high and strict demands, meaning that the COD was increasing exponentially with the delay time. Three different simulation runs were performed for each of the defined scenarios:

- The current situation where there was no DSS system to decide what will happen to each defect and all the defective parts were discarded and a new one was made for compensating the defective.
- The proposed solution with the utilization of the developed DSS system. In this scenario the DSS decides whether to repair or discard the defective part
- The ideal production scenario with no defects and no interruptions to the manufacturing process. This scenario is simulated in order to be used as a benchmark for comparing the other two scenarios.

Table 6: Simulation scenarios' demand profiles and order criticality

Scenarios	S1, S2	S1, S2	S1, S2	S1, S2	S1, S2	S1, S2	S1, S2	S1, S2
Orders ID	9101	9102	9103	9104	9105	9106	9107	9108
Date placed (days)	0	3	6	6	9	9	12	15
Due date (days)	9	24	24	27	27	39	33	51
Quantity	120	110	150	110	100	150	110	200
Order Criticality	9	2	8	4	7	6	8	9
Scenarios	S2	S2	S2	S2	S2	S2	S2	
Orders ID	9109	9110	9111	9112	9113	9114	9115	
Date placed (days)	18	18	21	24	24	27	27	
Due date (days)	45	39	39	45	66	45	48	
Quantity	150	100	100	125	220	100	100	
Order Criticality	10	5	3	8	7	3	3	

In the upcoming part with the results demonstration, those scenarios are named "*Discard or DRD*," "*DSS*," and "*Benchmark or B*" accordingly. For the three scenarios, two simulation periods were defined, namely a short-term and a mid-term period as described in Table 6. The short-term (S1) period considered all the orders received for the product under investigation for a period of 15 days (S1) and the mid-term period covered orders for a period of 27 days (S2). The mid-term period was an extension of the short-term period, which means that the first eight orders (9101 – 9108) of S2 were identical to S1, but S2 orders were from 9101-9115 because it covers a longer period of time. Table 6 contains some of the key information regarding the two simulation periods, such as the day that each order was placed along with the agreed due date. Additionally, the quantity and the criticality of each order can be found.

Table 7 and Figure 12 present the average (avg.) and detailed results of the measured KPIs for the *S1* scenario, respectively. In terms of tardiness (Figure 12, a), the DSS performed on average 50.40% and 697.98% better compared with the Benchmark and Discard scenarios, respectively. The same trend was observed in the makespan (Figure 12, b) results but with lower relative differences of 8.11% and 118.05%, respectively.

Table 7: S1 scenario average KPIs results equations (22), (23) and (24)

Solutions	Avg. Tardiness (mins)	Avg. Makespan (mins)	Avg. Materials cost (€)	Avg. Operational cost (€)	Avg. Delay penalty cost (€)	Avg. Machine utilization (%)
DSS	2,056.58	37,148.65	18,277.44	48,268.74	1,869.05	55.83%
Discard (DRD)	16,411.14	81,003.44	20,081.04	52,804.86	6,386.67	33.57%
Benchmark (B)	3,093.14	40,161.67	17,511.20	45,794.37	1,179.63	46.72%

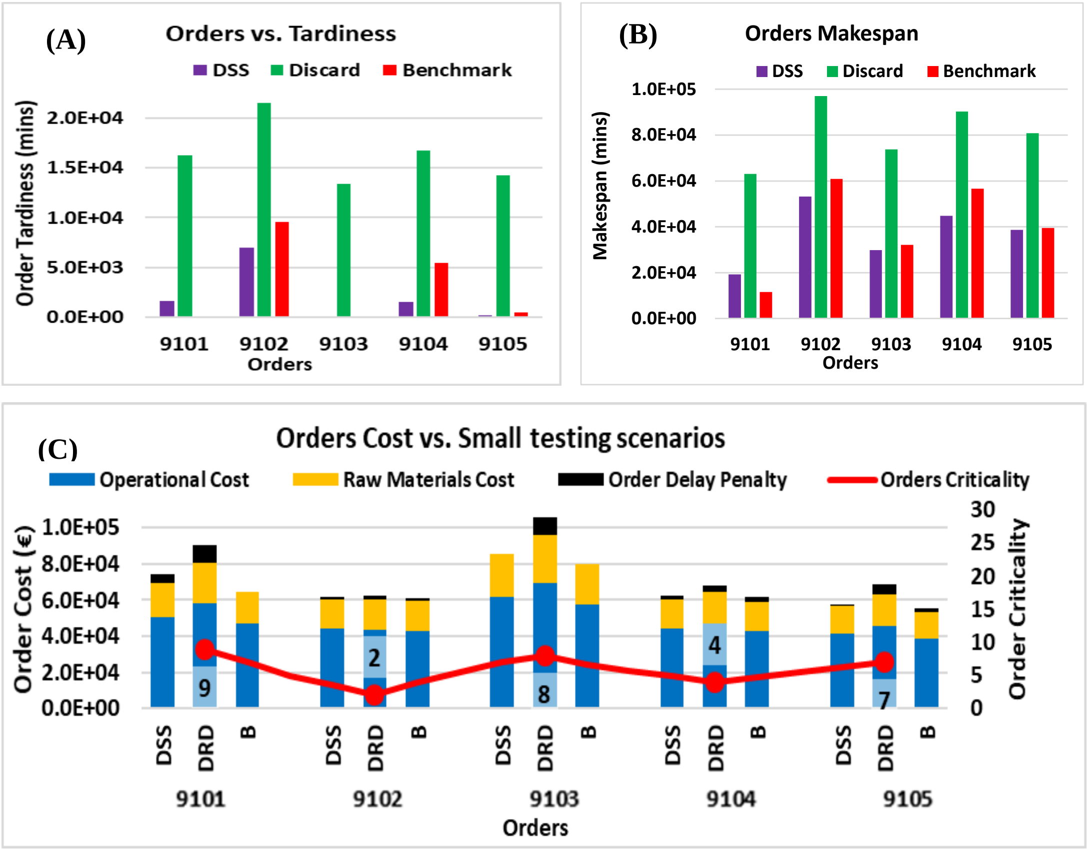


Figure 12: S1 scenario KPI results: (a) Tardiness, (b) Makespan, and (c) Total Cost

Figure 12 (C) presents the total order cost broken down into three categories: the operational cost, the raw materials cost, and the cost due to delay penalties. It is clear that the DSS performed better than the Discard scenario, which is the current production situation, by 15.87%. In addition, the DSS was on average 5.74% worse than the Benchmark scenario, showing the potential of this approach. Furthermore, the machine utilization was calculated for each of the sub-scenarios with 55.83% for the case of the DSS, whereas in the Discard and Benchmark scenarios it was 33.57% and 46.72%, respectively.

Table 8 and Figure 13 present the average and detailed results from the medium scenario (S2), respectively. The results followed the same trend as in the small scenario (S1). In terms of Tardiness, the DSS performed at 15.95% and 17.08% and in terms of makespan it was 4.60% and 7.75% compared with the Benchmark and Discard scenarios, respectively.

Table 8: S2 scenario average KPIs results

Solutions	Avg. Tardiness (mins)	Avg. Makespan (mins)	Avg. Materials cost (€)	Avg. Operational cost (€)	Avg. Delay penalty cost (€)	Avg. Machine utilization (%)
DSS	14,903.90	98,541.71	20,177.97	53,709.09	5,392.21	61.25%
Discard	17,449.31	106,177.93	21,797.62	57,499.90	6,211.84	56.62%
Benchmark	17,281.47	103,070.21	19,242.53	50,472.02	2,905.40	49.24%

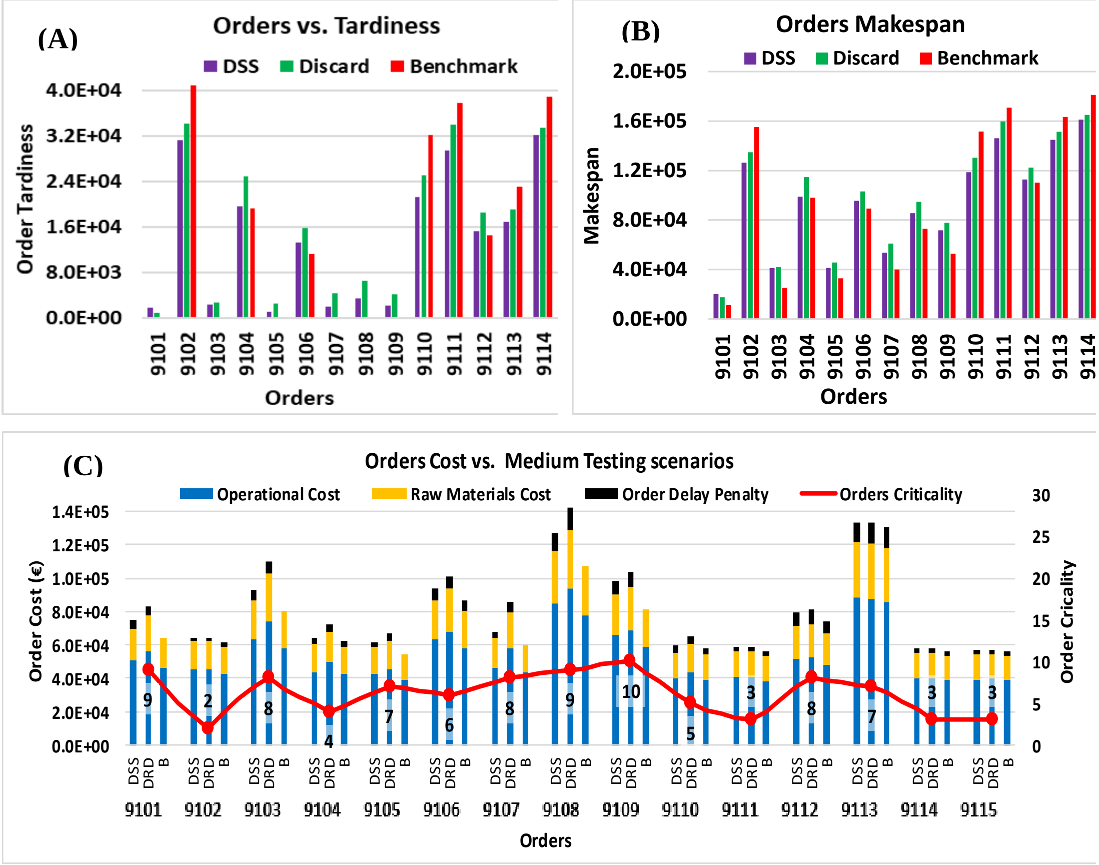


Figure 13: S2 scenario KPIs results: (a)Tardiness, (b) Makespan, and (c) Total Cost

In S2, the overall resource utilization of the DSS solution was 61.25%, whereas in the Benchmark and Discard cases it was 49.24% and 56.62%, respectively. Finally, comparing the overall performance of the DSS with the Benchmark and Disposal scenarios, taking into consideration all the measured KPIs, the DSS performed 4.05% and 10.90% better, respectively. Finally, the proposed DSS required 0.1458 seconds and with standard deviation of 0.0153 seconds for each decision cycle.

The overall outcome from the conducted experiments was that the proposed DSS tool was on average 7.47% better compared with the current production policy for the manufacturing process of the specific PCB. Furthermore, the DSS produced significantly better results in the events of important orders compared with the less important ones in both scenarios (S1 and S2).

In some cases, in both S1 and S2 the DSS produced results better than or equal to the Benchmark scenario. The reason behind this behavior was the frequency of rescheduling the production. In the case of the Benchmark scenario, there were no defects; therefore, the only events happening in the production were the new orders coming in. In this regard, the production had to be rescheduled only 7 and 14 times in scenarios S1 and S2, respectively. On the other hand, the DSS scenario, besides the new order events, also had to deal with the defects as events. This created the need/opportunity to reschedule the production more times to consider the actions required for the defective products. The higher number of rescheduling times provided the ability to achieve more optimized schedules in terms of the measured KPIs. More specifically, the S1 scenario was rescheduled 26 times and the S2 scenario 41 times. Another fact that verifies this is the machine utilization rates. In the cases of the Benchmark and Discard scenarios, the machine utilization was consistently lower than the machine utilization using the DSS.

The proposed DSS system is meant to be triggered in real time according to the events that occur during production. The simulations showed that the DSS tool requires on average only 0.1458 seconds to make a decision for each defect, which is acceptable for in-line use.

The simulation results revealed that the proposed DSS has positive effects on both simulation periods (short- and mid-term). Although the effect was positive, a huge difference between the two effects was observed. The overall performance of the proposed DSS was 147.45% better in the short-term scenario, whereas in the mid-term scenario the DSS was 7.47% better. The reason behind this significant difference is that in the short-term scenario there were only a few orders, and therefore, any performance difference was amplified because of the small number of orders. On the other hand, in the mid-term scenario, the results were more smoothed with no such huge differences.

The proposed DSS tool was developed to assist the decision-making process when a defective part is detected to decide whether to repair it, discard it, or do nothing. The simulation results revealed that in both simulation periods, the waste in terms of raw material costs was reduced by 4.702% and 3.858% for short- and mid-term periods, respectively. The implementation of the described system will contribute to moving one step closer toward ZDM.

4.5 Prevention Action Generation & Maintenance Actions

In ZDM, the most advanced strategy for avoiding defects is to perform some actions for preventing future defects. As the diagram in Figure 2 illustrates, the prevention strategy can work with both the detection and prediction strategies. Although the prevention works with both triggering strategies of ZDM, the prevention actions that are implemented in each case are different. This is because the triggering factors are different, as are the conditions for the prevention actions. The cause behind this is the information available at each stage. In the detection prevention strategy, significantly more information exists regarding the past defects, which can lead to more targeted and correct prevention actions. On the other hand, in the prediction prevention strategy, the produced defects had not yet been produced; therefore, the causes of the defect might not be clear and the prevention actions might not be as accurate as in the detection prevention strategy. Another difference between the two prevention implementations was the fact that in the detection prevention strategy, the operator has time to perform a short root cause analysis and consider alternative prevention actions; furthermore, the prevention actions can be combined with maintenance for even better results. Performing maintenance is possible because there is time for planning since there is no urge to act immediately. In the prediction prevention strategy, there is not much time for considering alternative actions because the time for acting is dependent on how far the prediction algorithm can look ahead, and from the moment the prediction algorithm has generated that in the near future a defect is going to occur, the only time for acting is from that moment until the time of the predicted defect. The point in this case is to try to avoid the defect happening; therefore, the prevention action should be quick and efficient. Maintenance is difficult alongside those prevention actions since there is not much time for planning.

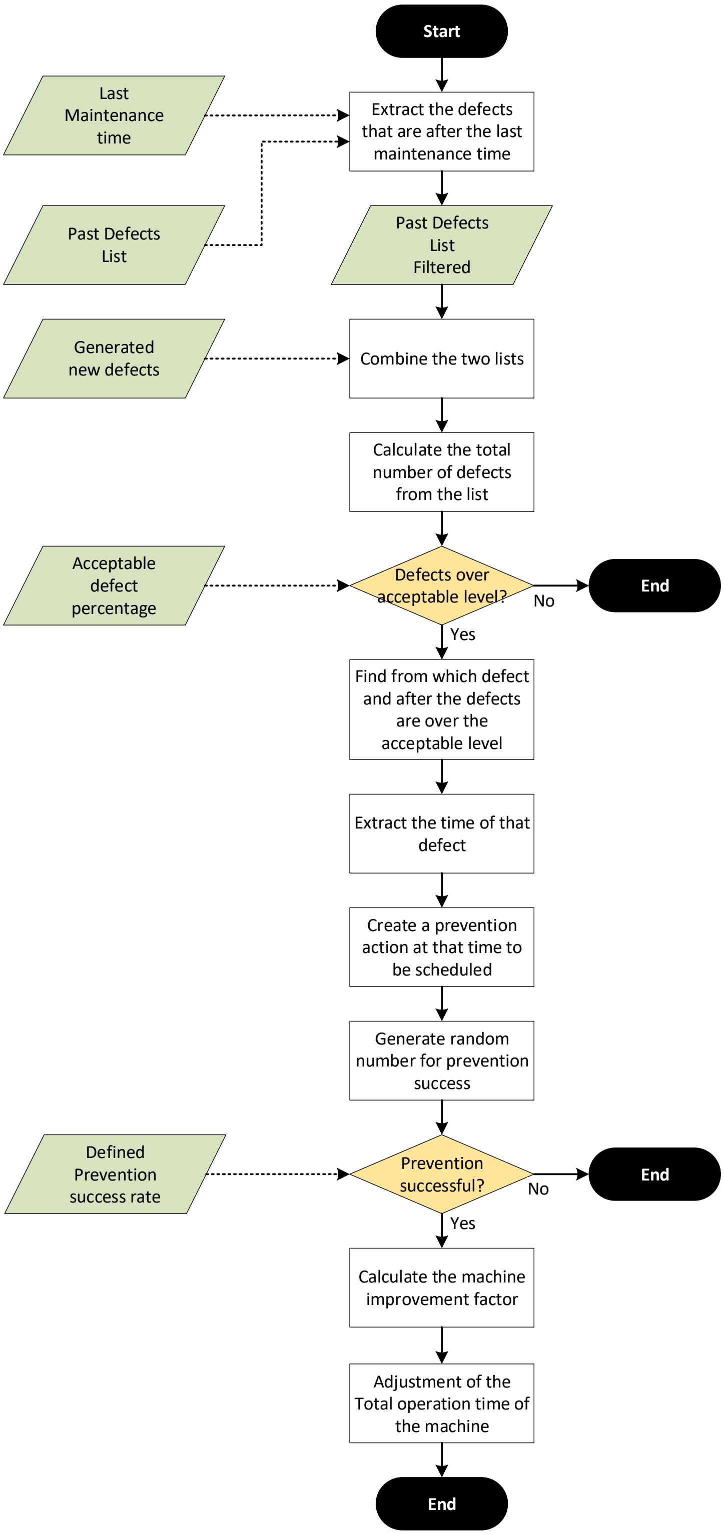
The prevention actions for both cases have high priority compared with the normal production tasks. This means that the scheduling algorithm will be forced to assign those tasks first and then all the others. By doing this, it is assured that the results of the prevention actions will be the desired one and the future defects will be avoided.

In the current research, the effect of the prevention actions or maintenance actions were implemented using a control parameter named "Machine Improvement State" (MIS) and its calculation formula is presented by equation (25). This parameter was used only in Detection – Prevention and Prediction – Prevention. In other words, it was used when prevention needed to take place. This parameter was created to quantify the benefits of the prevention actions, and it was also crucial for the accuracy and realism of the simulation tool. When prevention actions occur, the state of the machine is improved by a percentage. This improvement is translated into numbers by adjusting the total operation time of each machine and by extension the defect generation module. The total machine operation time is measured from the time the last maintenance action occurred at the particular machine. Furthermore, for the simulation tool to work and approximate the reality as close as possible, the assumption was made that after machine maintenance the machine returns to its original state. This improvement will move the main maintenance of the machine to later than normal. The amount of improvement is dependent on the amount of time for implementing the prevention actions and also on the cost of the prevention. Below, equation (25) was used for the calculation of the machine improvement state related to the prevention cost and time. This formula was based on experimental data and the assumption that a more time consuming and expensive prevention action would have more significant results than would shorter and less expensive prevention actions. The weights are defined in chapter 5.1.4 where the industrial case is presented.

$$MIS = \frac{\left(\frac{PC_{max} - PC}{PC_{max} - PC_{min}} * 0.5 + \frac{PT_{max} - PT}{PT_{max} - PT_{min}} * 0.5\right) * W_1 + W_2}{100} \tag{25}$$

4.5.1 Detect – Prevent, prevention process

The prevention actions in the detect – prevent ZDM strategy have a role not to diminish the defects but to keep the defect levels up to an acceptable level. The prevention actions are triggered by the detected defects, and the role of prevention is not to repair the defective parts but to ensure that the manufacturing process will be defect free in the upcoming future. Figure 14 illustrates the process for creating and scheduling the prevention actions. The prevention assignment algorithm runs for each machine separately, starting with the retrieval of the last maintenance time and the past defects list. Then, this list is filtered using the last maintenance time to keep only the defects that have occurred since the last maintenance time. Subsequently, the filtered past defects list is combined with the current defects list, which contains the defects that will occur in the current schedule. The combined defects list is used for counting the total number of defects after the maintenance and checking whether the defect percentage is over the acceptable level. If defects do not exceed the defined threshold then no prevention action is created. In the event that the defects are over the threshold, the algorithm searches to find the defect that after that the defect level is over the threshold. This defect points to the limit, and after that point a prevention action is required. The time of the defect marks the time after which a prevention action is required to be scheduled.



Start
Last Maintenance time
Extract the defects that are after the last maintenance time
Past Defects List
Past Defects List Filtered
Generated new defects
Combine the two lists
Calculate the total number of defects from the list
Acceptable defect percentage
Defects over acceptable level?
No
End
Yes
Find from which defect and after the defects are over the acceptable level
Extract the time of that defect
Create a prevention action at that time to be scheduled
Generate random number for prevention success
Defined Prevention success rate
Prevention successful?
No
End
Yes
Calculate the machine improvement factor
Adjustment of the Total operation time of the machine
End

Figure 14: Detect – Prevent, prevention process flowchart

Finally, this algorithm also incorporates some uncertainty to be more realistic. The incorporated uncertainty concerns the successfulness of the prevention action. It is incorporated as a random number within the interval [0,1] which is used as a comparison value with the prevention successful rate. For example, if the random number is 0.754 and the prevention successful rate is set at 0.85 this would mean that the prevention was successful, if the random number was higher than the 0.8,5 then the prevention event would be characterized as non-successful, with all the implications this might mean. In reality this means that when the operator observes an increased number of defects, then he or she proposes a prevention action based on data available. However, there is a possibility that the cause of the defect is different and the prevention action that is suggested will not have an effect on the process. Additionally, another alternative possibility exists that the operator identifies the problem correctly but the implementation of the prevention action is not successful. In both described cases, there is a possibility that the prevention action is unsuccessful with the corresponding results. Therefore, to simulate this possibility, a random number is generated [0,1] for each prevention action generated to be compared with estimated prevention success rate (in the form of a percentage) that was defined. If the check is performed and the outcome is that the prevention was successful, the next step is to calculate the machine improvement factor and adjust the total operation time of that particular machine accordingly (the total operation time is reduced by the amount defined by the machine improvement factor). If the prevention is unsuccessful, then the prevention action is also assigned for scheduling but no improvement on the machine is applied.

4.5.2 Predict – Prevent, prevention process

The assignment of prevention actions in the event of defect prediction is simpler than in the case of the detect – prevent strategy, because most of the work is done in the defect prediction module. Here, the algorithm acquires the predicted defects list, based on which a prevention action is assigned for each of the predictions. The prevention action assigned due to defect predictions can be considerably more than that in the detect – prevent case. This means that the prevention actions should be shorter in terms of time, otherwise their incorporation into the schedule would be inefficient. As in the previous case, the prevention action might be successful or it might not, and the uncertainty is incorporated with the same approach described in chapter 4.5.1. The process for this is the same as in the detect – prevent case.

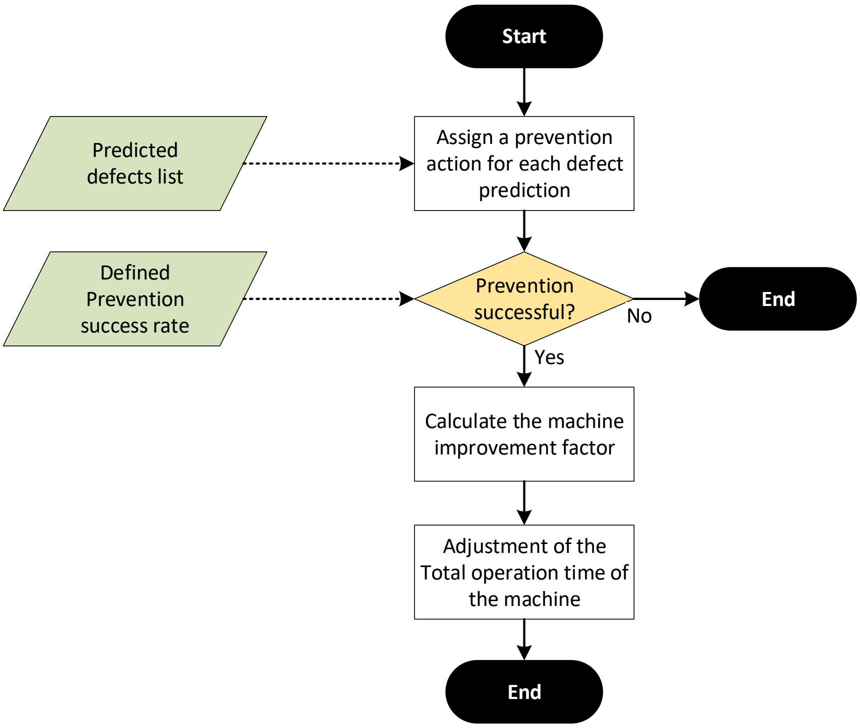

Start
Predicted defects list
Assign a prevention action for each defect prediction
Defined Prevention success rate
Prevention successful?
No
End
Yes
Calculate the machine improvement factor
Adjustment of the Total operation time of the machine
End

Figure 15: Predict – Prevent, prevention process flowchart

4.5.3 Maintenance actions

Maintenance actions are similar to the detect – prevent actions but they are not directly related to the ZDM implementation. Their presence is required to depict reality because in a real environment, maintenance is a crucial part of the production, and also because of the implementation of the prevention actions described in chapters 4.5.1 and 4.5.2. This is because the event of maintenance is used for “restarting” the health of each machine. Every time maintenance is performed, the total operation time used in the prevention actions sets to zero and counting starts from that point on. This is done based on the assumption that after maintenance the health of the machine returns to its initial state. In the current implementation, condition-based maintenance was the selected type of maintenance because it is widely used and is an efficient way to maintain manufacturing systems. Additionally, one could say that preventive maintenance is also implemented in the current tool (chapters 4.5.1 and 4.5.2), which is triggered by different factors. In the developed condition-based maintenance, there were two conditions developed.

4.6 Heuristics Rules

In ZDM, every abnormality requires a mitigation action [188]. This creates the need for more frequent re-scheduling, and therefore enhanced methods for producing high-quality schedules quickly. This is because the economic impact from a low-quality schedule is proportional to the rescheduling frequency. Therefore, in the current research, four different heuristic algorithms were developed and compared to find one that provides the best initial solution. Then, they were passed to the tabu search for further optimization. The heuristic algorithms presented in this chapter are tweaked versions of the algorithms described in [189][190][191][192].

Table 9. Notation of variables

Symbol	Description
$R=\{R_k \| k = [1,r]\}$	Set of Order
$O_k=\{O_{j,k} \| j=[1, n]\}$	Set of Operation of order R_k
$M=\{M_i \|i=[1,m])\}$	Set of Machines
$\pi=\{\pi_1, ..., \pi_m\}$	Schedule
$\pi_i=\{O_{i(1)},\dots,O_{i(n)}\}$	Schedule of machine i
P_{ij}	Processing time of operation j on machine i
S_{ij}	Setup time of operation j on machine i
D_k	Due date of order k
level(j)	Precedence constraints
m_p	Number of possible machines for one operation
Length	Variable used to balance production
M_{min}	Machine with the least operation during one algorithm step
M^*	Best machine during one algorithm step
π_c	Schedule close to π_k
π^*	Best solution optimized
L	The tabu list
$G(\pi_h)$	Optimization function to minimize

The proposed heuristics algorithms are composed of two parts: order prioritization and operation allocation to the available machines. Two methods are used for the prioritization of the orders: the “First Come First Served” (FCFS) method and one that considers the importance of the orders based on the due date, order volume, and customer profile, namely “OrderSequence” (OS). The heuristic algorithms used for allocating the operations to the available machines are the Machine Cost (MC), Shortest Processing Time (SPT), Sum of Shortest Processing Time (SumSPT), and Earliest Completion Machine (ECM). Eight heuristic algorithms are produced using the two order prioritization and four operation allocation algorithms. The notation of those heuristics is composed of three parts: the rule used for order prioritization, the heuristic algorithm, and in some cases the value of the parameter "Length," which is defined in upcoming chapters. In addition, Table 9 contains all the notations used in the upcoming chapters.

The evaluation of the produced schedules was performed using four KPIs: two cost-related and two time-related. The "production cost" includes the machine operational, setup, labor, and raw material costs, and the "COD" is a penalty fee that manufacturers pay in case of delaying the delivery of an order [73]. The next two time-oriented KPIs are tardiness and makespan [193][47]. To aggregate all KPIs into one value and compare the alternative schedules, the methodology presented in chapter 4.9 was utilized, using equation (31) for the normalization of the KPIs because all of them have cost behavior. Then, equation (33) was used for the calculation of the utility value.

4.6.1 Initial solution generation – Machine cost

This algorithm aims to identify the machine that costs the least to perform the operation O_j. For this, we multiply the processing time by the hourly cost of the machine, which gives a production cost. For each machine available for an operation, one calculates this cost and chooses the machine that minimizes the cost. In this algorithm, a limit is introduced named "Length." Indeed, without this limit the algorithm would always have the same operations on the same machines, the ones with the lowest operational cost (Figure 16).

```
Input:  M, R, $P_{ij}$, the cost/hour for each machine.
For k=1 to r
    Classify the operations of $R_k$ by level
    For j=1 to size(level)
        While level(j) is not empty
            Select one operation $O_j$ from level(j)
            Get $m_p$ possible machines  for operations level(j)
            For i=1 to $m_p$
                Calculate cost/machine = $P_{ij}$*cost/hour.
            Find the machine $M^*$ which has the smallest cost.
            Find the machine $M_{min}$ with the least operation
            While size($M^*$)-size($M_{min}$)>Length :
                $M^*$ change for the next cheapest machine
        $M^*$(size($M^*$)+1) = $O_j$
        Remove $O_j$ from level(j)
Output : $\pi$  (All the set of operations on each machine)
```

Figure 16: Machine cost heuristic algorithm

For example, for Length = 10, we grant a 10-step advance between the machine performing the most and the machine doing the least. Exceeding this limit, the operation will be put on the second cheapest machine, and if the second also has 10 operations in advance, then the operation is put on the third cheapest and so on. This limit makes it possible to both reduce the imbalance between the machines and to use them all.

4.6.2 Initial solution generation – Shortest processing time

Similarly, the algorithm described in this chapter takes all operations and sorts them from the smallest to largest processing time on the machine that makes it the fastest, respecting the precedence constraints. The variable Length is used in the same way as in the previous algorithm for reducing the imbalance between machines (Figure 17).

```
Input:  M, R, $P_{ij}$.
For k=1 to r
    Classify the operations of $R_k$ by level
    For j=1 to size(level)
    While level(j) is not empty
        Select all operation $O_j$ from level(j)
        Get $m_p$ possible machines for all operations level(j)
        For each operation $O_j$
            Find $M_j^*$ which has the smallest
             $P_{ij}$ to do the operation $O_j$.
        $M^*$ is the machine $M_j^*$ with the smallest $P_{ij}$
        Find the machine $M_{min}$ with the least operation
        While size($M^*$)-size($M_{min}$)>Length
            $M^*$ change for the next fastest machine
        $M^*$(size($M^*$)+1) = $O_j$
        Remove $O_j$ from level(j)
Output : $\pi$  (All the set of operations on each machine)
```

Figure 17: Shortest processing time heuristic algorithm

4.6.3 Initial solution generation – Sum of the shortest processing time

In this chapter, we describe an adapted version of a heuristic algorithm developed originally by Naderi and Ruiz [192]. It calculates the sum of the machine processing times when a new operation is assigned. At each stage, the machine with the smallest sum receives the operation to be done (Figure 18).

```
Input: M, R, $P_{ij}$
For k=1 to r
Classify the operations of $R_k$ by level
For j=1 to size(level)
    While level(j) is not empty
    Select one operation $O_j$ from level(j)
    Get the $m_p$ possible machines for operations level(j)
    For i=1 to $m_p$
        Calculate all sum of $P_{ij}$ on the possible machine.
    Find $M^*$ which has the smallest sum of  $P_{ij}$
        $M^*$(size($M^*$)+1) = Oj
    Remove Oj from level(j)
Output : $\pi$ (All the set of operations on each machine)
```

Figure 18: Sum of the shortest processing time heuristic algorithm

4.6.4 Initial solution generation – Earliest completion machine

The ECM algorithm is an adapted version of the earliest completion factory developed by Naderi and Ruiz [192] and taken over by Zhang et al. [190]. For the current problem, we adapt this algorithm to our case study: an order R that has several operations. Each operation (O_j) is tested for all possible machines (M_i) and the option with the smallest Makespan is selected (Figure 19).

```
Input: Set Machine M, set Order R with its operations, processing time $P_{ij}$
For k=1 to r
Classify the operations of $R_k$ by level
For j=1 to size(level)
    While level(j) is not empty
        Select one operation $O_j$ from level(j)
        Get the $m_p$ possible machines for operations $O_j$
        For i=1 to $m_p$
            Calculate all the sum of $P_{ij}$ for the possible
            Machine as if they had the operation $O_j$ to do.
        Find $M^*$ which has the smallest sum of
        $P_{ij}$ once it made the job $O_j$.
        $M^*(size(M^*)+1) = O_j$
        Remove $O_j$ from level(j)
Output: $\pi$  (All the set of operations on each machine)
```

Figure 19: Earliest completion machine heuristic algorithm

4.6.5 Optimization method

In the current research work, the tabu search optimization algorithm was selected (Figure 20). Where π_h is the planned schedule at iteration h, π^* is the best schedule we have found, π_c is the schedule in the neighborhood of π_h, and N the maximum number of iterations. G() is the function to be optimized, and L is the tabu list where all the recently considered schedules are stored. If the solution π_c is not in the list L, then π_c will systematically become $\pi_h + 1$ even if π_c is a worse schedule than π_h (i.e., even if G (π_c)> G (π_h)).

```
Input: Initial solution $\pi_1$, N, G
Set $\pi^* = \pi_1$
For h = 1 to N
Select a schedule $\pi_c$ from the neighbourhood of $\pi_h$.
If the move $\pi_h \rightarrow \pi_c$ is prohibited by L
    Set $\pi_{h+1} = \pi_h$
If the move $\pi_h \rightarrow \pi_c$ is not prohibited by L,
    Set $\pi_{h+1}=\pi_c$
    Enter reverse mutation at the top of  L.
Push all other entries in L one position down
Delete the entry at the bottom of L.
If $G(\pi_c) < G(\pi^*)$
    Set $\pi^* = \pi_c$
Output: $\pi^*$
```

Figure 20: Tabu search algorithm

4.6.6 Validation of heuristic rules

For the testing of the developed heuristic algorithm, a real-life industrial data set was used that came from the semiconductor domain. The conducted simulations considered the assembly stage of a customized PCB, which is composed of seven operations. In addition, the simulation period was 1 week; in that period, 10 orders were received. In total, the 10 orders were composed of 5180 operations to be scheduled. For the initial solution, each algorithm was executed 15 times, and in the end, the average KPIs were calculated for both the initial and optimized solutions. This was performed to improve the accuracy of the results and overcome the inherent randomness of creating alternatives. Furthermore, the parameters used for the tabu search were as follows: tabu-list length = 20, number of tweaks = 12, and number of runs =

1000. Additionally, in the process of calculating the overall quality of the solution, each KPI had the same importance – weight factor = 0.25.

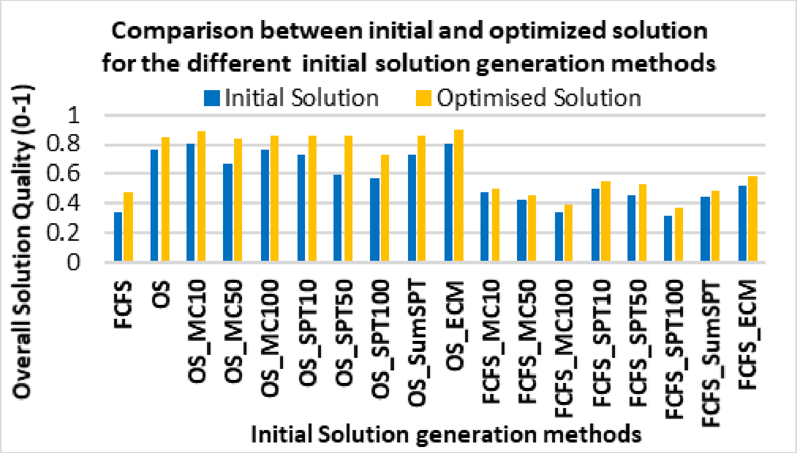


Figure 21: Overall solution quality diagram

Figure 21 illustrates the aggregated KPIs into a single value; the higher it is, the better the solution. This diagram compares all the developed algorithms before and after optimization. As was expected, the tabu search method succeeded in improving the generated initial solutions by 7.481% on average. The maximum solution improvement observed was 21.463% in the OS_SPT50 method and the minimum was 2.08% in the FCFS_MC10 method. Another immediate observation was that the initial solutions generated using the OS rule for prioritizing the received orders produced better results than did the initial solutions produced using the FCFS. In particular, the pure FCFS rule, where everything is done optimally on the order arrival time, gave the worst solution. The same trend was observed for the optimized solutions, where the initial solutions produced by the OS were further optimized by 20.718%, whereas those produced by the FCFS were only optimized by 11.341%. Furthermore, the standard deviation of the optimization level for OS and FCFS were 11.103% and 3.915%, respectively.

Moving forward, the Length parameter that was added to the heuristic algorithms for the initial solution generation (MC and SPT) had a significant impact on solution quality. In most cases, the higher the value of Length, the worse the produced solution. More specifically, all the solutions generated using Length = 10 produced results that were better by 7.267% and 13.171% than those using Length = 50 and Length = 100, respectively. Only in the OS_MC algorithm were the results generated using Length = 100 better by 6.32% compared with those using Length = 50 and worse than those using Length = 10 by 2.46%.

The algorithm SumSPT created solutions of high quality and in some cases better than the simple SPT method. In the simulation runs using the OS method, the SumSPT produced a solution almost equal to the OS_SPT10, only a 0.94% inferior solution. On the other hand, using the FCFS approach for order prioritization, the FCFS_SumSPT was better than FCFS_SPT100 by only 34.13%. More specifically, FCFS_SumSPT produced an inferior solution to FCFS_SPT10 and FCFS_SPT50 by 10.331% and 1.561%, respectively.

The results presented in Table 10 are sorted from best to the worst for both the initial and optimized solutions. In the top two places and with a marginal difference of 0.369% are the OS_ECM and the OS_MC10 for both the initial and optimized solutions. As seen below second place, there is no such an alignment for the initial and optimized solutions. Furthermore, only the two best initial solution algorithms succeeded in generating a solution with quality above 0.8, whereas for optimized solutions almost half were above that value. More specifically, the eight best-optimized solutions were very close to each other by 4.528% and with a standard deviation of 1.899%.

Finally, another important result was the computation time required to run each algorithm. As one would expect, the optimization algorithm, tabu search, required the most time of 4.638 hours on average with a standard deviation of 0.16 hours. On the other hand, the initial solution algorithms required 35.47 seconds on average with a standard deviation of 11.48 seconds.

Table 10. Initial and optimized solution ranking

	Initial Solution		Optimized Solutions	
Rank	Method	Sol. Quality	Method	Sol. Quality
1	OS_ECM	0.804847	OS_ECM	0.89952
2	OS_MC10	0.801892	OS_MC10	0.892662
3	OS	0.766712	OS_SPT10	0.856925
4	OS_MC100	0.763384	OS_SPT50	0.856893
5	OS_SPT10	0.735322	OS_SumSPT	0.856502
6	OS_SumSPT	0.728442	OS_MC100	0.854648
7	OS_MC50	0.672561	OS	0.847155
8	OS_SPT50	0.597566	OS_MC50	0.838956
9	OS_SPT100	0.573064	OS_SPT100	0.736095
10	FCFS_ECM	0.524242	FCFS_ECM	0.580186
11	FCFS_SPT10	0.493547	FCFS_SPT10	0.555431
12	FCFS_MC10	0.475711	FCFS_SPT50	0.528928
13	FCFS_SPT50	0.454315	FCFS_MC10	0.500837
14	FCFS_SumSPT	0.447332	FCFS_SumSPT	0.489369
15	FCFS_MC50	0.423365	FCFS_MC50	0.454732
16	FCFS_MC100	0.344279	FCFS_MC100	0.389603
17	FCFS_SPT100	0.316916	FCFS_SPT100	0.367129
18	FCFS	0.137178	FCFS	0.18221

The simulation results revealed a clear relation between the initial and optimized solutions. More specifically, the better the initial solution, the higher the quality of the optimized solution. The algorithms SumSPT and ECM were developed to take the results from the previous assignments into account during the operation assignment, and therefore achieve better results. OS_ECM was found to be the best of the tested algorithms because produced schedules were more balanced than the others. Moreover, OS allows prioritization of the most important order made first, which increases the quality of the solution. The addition of Length allowed the balancing of the production schedule, thus achieving better initial solutions. The results showed that the best schedules were those for Length = 10, and as the Length parameter moved increasingly closer to the result of the corresponding algorithm without that parameter, the schedules became worse and more unbalanced.

The optimization of the schedules required significantly more computation time than the initial solution algorithm. This was expected, but in many cases not much time was available in real production environments, and the need for a fast, high-quality solution arose. The developed algorithms showed that we could produce fast schedules of high quality taking into account the required time, especially OS_ECM.

In the literature, the NEH algorithm demonstrated promising results [51]. In the context of the current research, work there was a development of an adapted version of the NEH. The results are not presented due to the computation time required by the NEH algorithm to generate an initial solution, which was 17 hours (due to the high number of operations to schedule 5180), something unrealistic for real production environments.

4.7 Multiple Order Evaluation

The proposed methodology relies on two key points, namely the creation of a method for the automated ranking of different orders according to their criticality to give priority to those they require. Furthermore, a crucial step is the development of a method in the scheduling tool used for the evaluation of the criteria for each individual order combined with the OC factor and not for the total production.

4.7.1 Order criticality ranking method

The OC is a simple measure for ranking the different orders to schedule them accordingly. This will allow rush orders to be evaluated and when the rescheduling of the production takes place the scheduling of the remaining and new orders according to that value. This measure is a combination of four different factors: (1) the order volume (OV), in other words how many units the current customer has ordered; and (2) the available timeframe for completing the order, which is given by subtracting the order due date (DD) and the time of rescheduling (RST). The next two factors are related to the specific customer: (3) how frequently the customer orders (orders/year, OF); and (4) the importance of this customer, which takes a value between 0 and 10 (CI). Equation (26) shows how the OC is calculated. The summation of all the weights (W_j) is equal to 1. Furthermore, the symbol "^" denotes a normalized value. When rescheduling is required, the OV, DD, RST, OF, and CI are known for all the involved orders, and with the usage of equation (32) those values are normalized [26], where i: represents the order and j:1. In other words, the initial values are converted to a [0,1] range and without units. This procedure is performed for each parameter separately. Furthermore, the higher the value, the more critical the order is, and it will be scheduled accordingly.

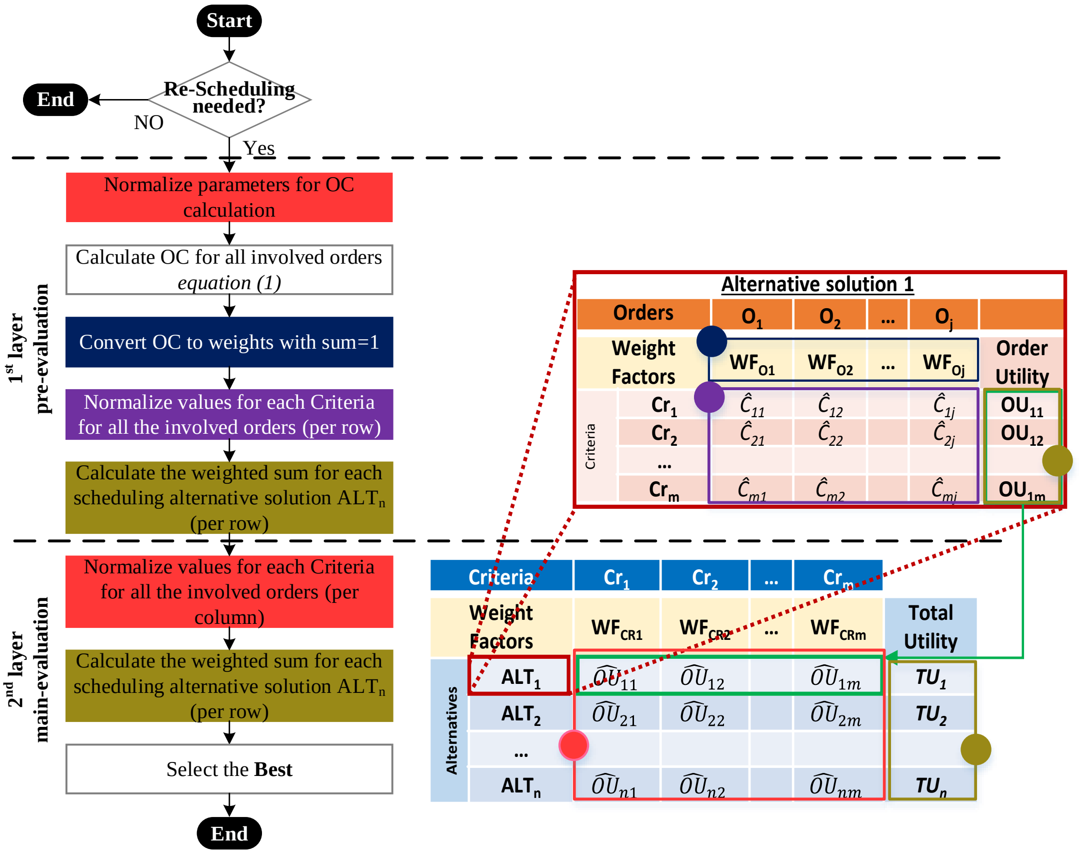
$$OC = W_1^{OV} * \widehat{OV} + W_2^{DD} * (\widehat{DD - RST}) + W_3^{OF} * \widehat{OF} + W_4^{CI} * \widehat{CI} \qquad (26)$$

At this point it should be mentioned that the OC value is a relative value and cannot be compared with previous ones. Every time rescheduling is required, OC is calculated as described. This happens because the involved orders might be different in each rescheduling the minimum and maximum values of each parameter (e.g., OV, OF) might be different, and therefore give a different normalized value for the same value.

4.7.2 Two-layer criteria evaluation method

The optimization of a production schedule is performed by producing a number of alternative schedule solutions, and then those alternatives are evaluated based on certain criteria to select the best among them. The contribution of the current research is the addition of one level prior to the main evaluation of the alternative solutions. This is added to capture the importance of each order and incorporated into the solution evaluation process. Figure 22 illustrates the proposed approach. This procedure is activated every time rescheduling is required, such as when a new order comes in and needs to be incorporated into the current schedule.

If rescheduling is required, then for all the unfinished and new orders the OC is calculated as described in chapter 4.7.1, equation (26). The next step is to convert the calculated OC into weights with sums equal to 1 using equation (52), simply by dividing each OC value by the total sum of all the OC. This is required for the final weighted summation procedure, equation (33). After the calculation of the "order" weight factors, the criteria values are normalized. This is performed using equation (31) because the criteria that will be used have "cost" behavior (the lower the better). The formula is used for each criteria separately and for all the involved orders. More specifically, in equation (31), i: represents the order and j: the corresponding criterion. At this point it should be mentioned that the minimum and maximum values for each criterion are the global max and min calculated from all alternative solutions. The final step of the pre-evaluation method is to multiply each normalized value by the corresponding weight factor and sum all the elements per row together (equation (33)). This will result in *m* values (m: number of criteria), each of which corresponds to a specific criterion and also contains all the information regarding the order's criticality. This will allow a more precise and accurate evaluation of the different alternative solutions.



Start
Re-Scheduling needed?
NO
End
Yes
1st layer pre-evaluation
Normalize parameters for OC calculation
Calculate OC for all involved orders equation (1)
Convert OC to weights with sum=1
Normalize values for each Criteria for all the involved orders (per row)
Calculate the weighted sum for each scheduling alternative solution ALTn (per row)
2nd layer main-evaluation
Normalize values for each Criteria for all the involved orders (per column)
Calculate the weighted sum for each scheduling alternative solution ALTn (per row)
Select the Best
End
Alternative solution 1
Orders
O1
O2
...
Oj
Weight Factors
WFO1
WFO2
WFOj
Order Utility
Criteria
Cr1
Cr2
Crm
Ĉ11
Ĉ12
Ĉ1j
Ĉ21
Ĉ22
Ĉ2j
Ĉm1
Ĉm2
Ĉmj
OU11
OU12
OU1m
Criteria
Cr1
Cr2
Crm
Weight Factors
WFCR1
WFCR2
WFCRm
Total Utility
Alternatives
ALT1
ALT2
ALTn
ÔU11
ÔU12
ÔU1m
ÔU21
ÔU22
ÔU2m
ÔUn1
ÔUn2
ÔUnm
TU1
TU2
TUn

Figure 22: Two layer evaluation methodology workflow

In the second evaluation layer, the final evaluation is performed. First, for each column the criteria values are normalized once more using equation (32), since the values have "benefit" behavior (the higher the better) because of the pre-evaluation step. Then, equation (33) is used for each row, summing all the normalized values multiplied by the corresponding weight factor (criteria weight factor, user defined and sum equal to 1). The best alternative is the one with the highest utility value (U_{max}).

4.7.3 Simulation tool and optimization criteria

The simulations were performed using a dynamic scheduling tool [27], with the optimization criteria of makespan and tardiness as described in [28]. Furthermore, the total production cost was used for the optimization, and this value contains the operational cost of the machines, raw materials used, labor cost, setup cost, quality inspection cost, and penalty to pay in case the order is delayed. The COD is given by equation (27); [29] and estimates the cost incurred due to delay for sensitive-to-delay orders and is also linked to the OC factor presented in chapter 4.7.1, where "d" is the delay time, w_{d1}, w_{d2} are weights for adjusting the formula, and "OV" is order volume.

$$COD(d) = \mathrm{w}_{d1} \ln(1 + \mathrm{w}_{d2} * d) * \mathrm{OV} * \mathrm{OC} \tag{27}$$

4.7.4 Industrial use case

The proposed approach was validated using data from a European semiconductor manufacturer. More specifically, the infrastructure that was simulated was a semiautomated assembly line for producing complete PCBs for the medical domain. The production line is characterized as semi-automated because quality inspection and the transportation of trays with parts from one machine to the next is performed manually. In addition, this case was used because they have a high number of rush orders that disturb normal production and negatively affect the performance of the production. Furthermore, the orders received are sensitive to delay and financial penalties are incurred according to the delay time.

The studied assembly process concerns a product that requires 27 tasks in the final assembly process. The final cost for this particular product is €1094.46. The assembly line is configured as a flexible job shop, where there are six work centers with two or three parallel identical machines. Figure 23 represents the demand profile for the specific period and the criticality for the 30 orders that are simulated, which cover a period of 74 days. The current demand profile was selected to be used because the volume of orders was fluctuating between 20 and 75 pieces per order. Furthermore, there are many rush orders in the selected demand profile, which is ideal for validating the proposed methodology.

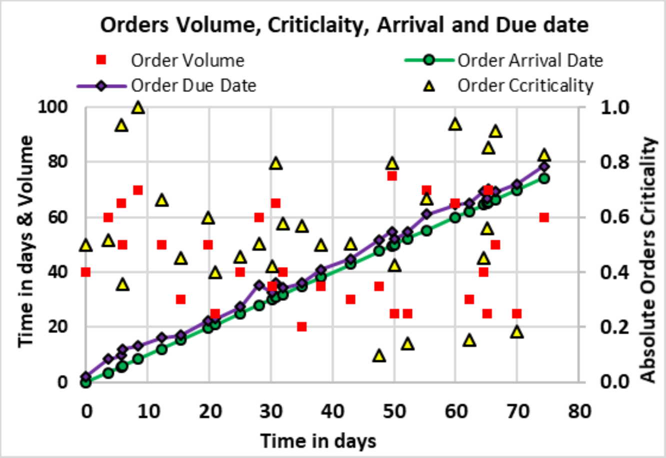


Figure 23: Demand profile and characteristics

In normal production conditions, a new order is expected on average every 3.47 days and constitutes 56.66% of the total orders received. By contrast, rush orders arrive unexpectedly on average every 1.64 days and correspond to 46.69% of the total products ordered and with importance usually higher than normal orders. Because 46.69% of total orders arrive unexpectedly and in shorter periods of time compared with normal orders, the current demand profile is characterized by high rush orders. This also means that 43.33% of the total orders are not expected and create problems that the proposed methodology aims to solve.

In all the simulation results, the OC was plotted as well to observe the behavior of the proposed methodology and also to compare the importance of all orders. Furthermore, the depicted OC was the absolute value for this specific experiment, which means that the values were calculated taking into account all orders considered in this experiment. This was mentioned because the OC values were dynamic values and depend on the current order to be rescheduled.

For the simulations, a dynamic scheduling tool was employed using the proposed methodology and optimization criteria described in chapter 4.7.3, [26]. Two simulations were performed, one using the proposed methodology and one without the second evaluation layer, to compare it with the performance of the proposed methodology. Figure 24 illustrates the overall quality of the solutions (utility value) for the two simulation scenarios. The higher the utility value, the better the quality of the solution. The simulation results showed that the proposed methodology in 70% of the simulated orders produced higher-quality solutions than the single level approach.

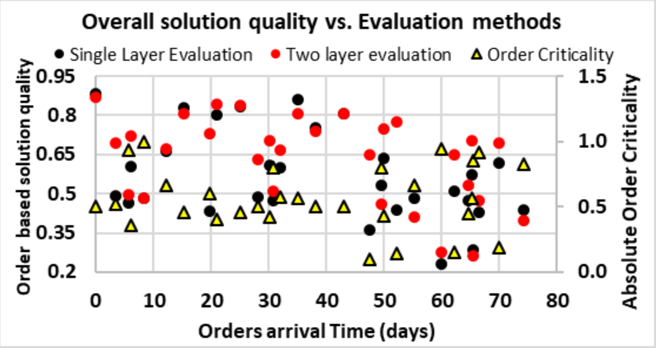


Figure 24: Overall solution quality

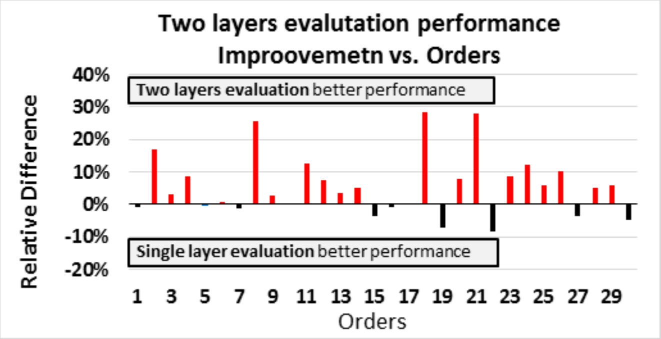


Figure 25: Performance improvement with the proposed method vs. orders

Moving forward, the relative difference of each individual order's utility value was calculated between the proposed approach and single level criteria evaluation approach. These relative values are plotted in Figure 25. The upper half (positive) of Figure 25 shows the orders in which the proposed two-level criteria evaluation method behaved better than the single level approach, and the lower half (negative) shows the orders for which the single level approach behaved better. The two-level evaluation method produced better results for 21 out of 30 orders. The solution quality improvement varied among the different orders from 0.114% up to 28.57%.

On the other hand, the observed fluctuation for the orders in the lower part of Figure 25 was lower compared with those in the upper half, from 0.19% up to 8.26%. On average, the proposed methodology achieved a 9.497% quality improvement for the 70% of orders in the current simulation period. Regarding the remaining 30%, the quality loss was on average 3.32% compared with the quality gain on the other 70% of orders. Furthermore, the proposed approach produced overall 5.615% on average better compared with the single solution approach.

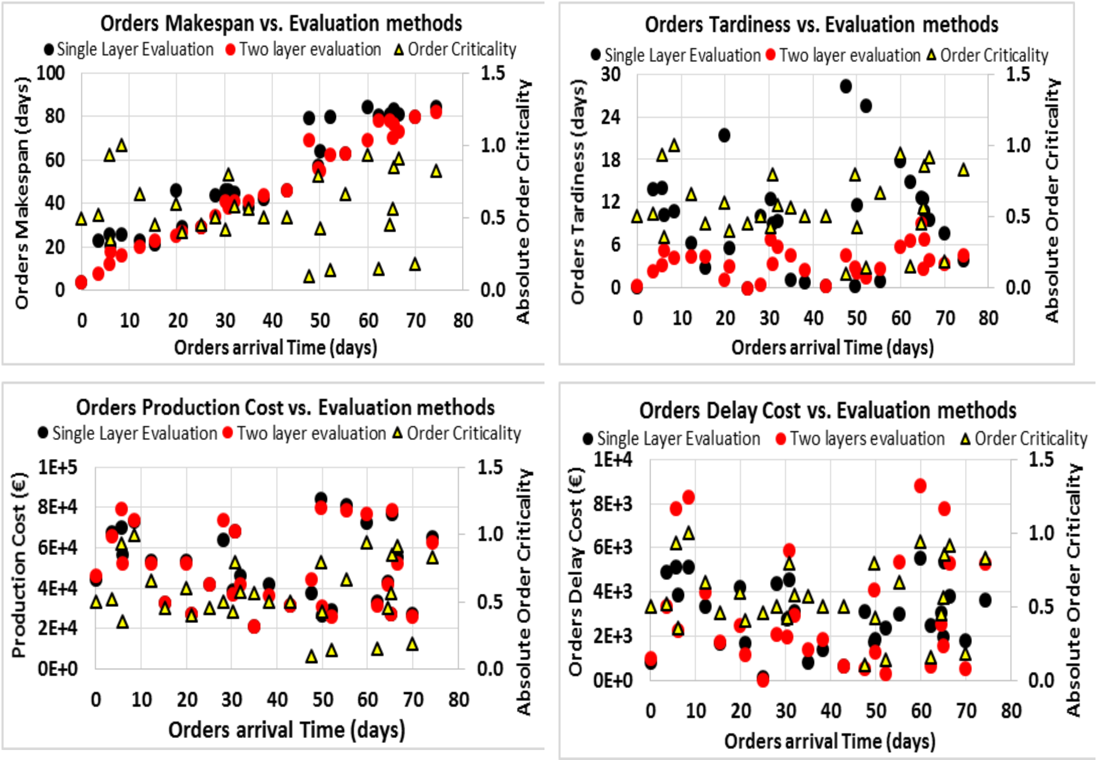


Figure 26: Simulation results for the individual optimization criteria

All the above considered the overall behavior of the proposed approach, which combined the four measured criteria described in chapter 4.7.3. Figure 26 presents the simulation results for each individual criterion. Regarding the makespan, 80% of the orders were finished earlier in the scenario using the proposed approach. In addition, the same behavior was observed for the tardiness of the orders, where 76.66% had less of a delay from the agreed order due date. The production cost and order delay cost for the simulated orders showed that 56.66% and 53.33% had lower cost values using the proposed approach.

Overall, the simulation results showed that in most of the cases, the orders that had high OC had better results than did those produced by the single level method. Some of the orders with high OC, however, had worse results, which is because it is not always possible when scheduling to optimize the criteria for all the involved orders. This is because the proposed approach is based on the ranking of orders, and therefore, when optimizing the most important order, there will be a loss. Furthermore, the proposed method produced significantly better results for the time criteria rather than for the cost criteria. On average, the order tardiness was 3.63 days and 9.48 days for the two and single layer methods, respectively.

Production cost had a small fluctuation because each product requires almost the same raw materials and processing time. The COD was more complicated because it is a function of many factors. It relies on the tardiness, volume, and criticality of the order, and therefore, it is difficult to draw a conclusion. Furthermore, it was observed that orders with a high OC factor tended to have higher COD, which is expected and shows that the order ranking approach produces the desired results.

The simulation results showed that the proposed method was capable of efficiently scheduling rush orders, which was the goal of the present research. Notably, the proposed methodology produced slightly worse results for orders 7, 15, and 16 with a maximum loss of performance of 3.42% for order 15. This is for the reason explained at the beginning of the Discussion chapter, namely that priority had been given to other normal and rush orders that arrived earlier, and therefore, the optimization of those orders rendered the optimization of orders 7, 15, and 16 impossible.

Finally, the reason behind the achievement of more efficient schedules was derived from the proposed dynamic order ranking method. More specifically, every time that rescheduling was performed, the order ranking changed according to the orders involved as well as the time the rescheduling was performed, optimizing the importance of each order and to that extent the measured performance indicators.

4.8 Event Management Methodology

The aim of this research was to close the gap between the theory and practice of production rescheduling by proposing a real rescheduling solution for a semiconductor manufacturing company in the ZDM context. This company produces PCBs for the healthcare sector. A new model was developed to optimize rescheduling in a flexible job shop by analyzing the quality of the solution according to the Analysis of Mean (ANOM) methodology. The goal of the model is to provide a tool that guides manufacturers to obtain a more resilient and flexible production system to mitigate the risks of unexpected events. In this chapter, the methodology used to develop the framework was accurately described in four sub-chapters: “Description of factors,” “Simulation model,” “Performance indicators,” and “Design of experiments,” The abbreviations used in this chapter are listed in Table 11 and Table 12.

Table 11: *Factor Abbreviations*

NORDT(F1)	New Order Response Time
NDRT(F3)	New Defect Response Time
NPRT(F5)	New Prediction Response Time
NORDT(F2)	New Order Delay Response Time
NDDRT(F4)	New Defect Delay Response Time
NPDRT(F6)	New Prediction Delay Response Time
PH(F7)	Prediction Horizon

Table 12: Equation Parameter Abbreviations

Ev	Current Events
EvRT	Current Events Releasing Time
EvDRT	Current Events Delay Realising Time
RSC	Production and Rescheduling Cost
w_c	Specific Weight
RMSCF	Raw Materials Set Up Cost
NT	Number of Tasks
$Effect_{k,z}$	Effect of Factor Level
k	Corresponding Factor
z	Level of k^{th} Factor
$m_{k,z}$	Number of Experiments for that Facto Level

4.8.1 Description of factors

In this research, different unexpected events were analyzed to mitigate their impact on the manufacturing process. Indeed, the aim was to reduce their negative effect so as not to disturb the normal flow of production and maintain an efficient system. The unexpected events that were analyzed are as follows:

- New orders
- Defective parts
- Defect prediction

New orders are the most common unexpected event that can disrupt the production at shop level. Indeed, if the production system is not flexible enough, a new order could generate high manufacturing costs and increase lead time. The other two events come from the ZDM concept and they are also of primary importance. Defective parts are the most critical events in a production system in terms of costs. Indeed, if a defective part is detected too late in the process, the part has to be produced again. Moreover, a defective part or a low-quality one that is already launched on the market could result in loss of business for the company since it could damage its image and lead to a reduction in market share.

Therefore, some actions must be pursued to mitigate the risks of those unpredictable events. First of all, in the case of a new order, the order itself is released to the production to satisfy the customer's demand in the short term. In the case of a defective part, the action depends on the level of damage of the part; if it is worth it in terms of costs and time, the part is repaired, but otherwise the part is completely produced again. Finally, in the case of defect prediction, the action required is machine tuning or maintenance to prevent future disruption in the production flow.

However, all those actions require a specific amount of time to be prepared before they can be released on the shop floor; this time is called the "response time." It could be also defined as the amount of time needed to react to a new request. In the case of a new order, it would be the time needed to contact the supplier, wait for the raw material to arrive, control the specifications of the order, and check the capacity of the production system. Moreover, the action can be postponed for a certain time to reduce the number of times rescheduling is required; this time is called the "delay response time." Theoretically, a shorter delay response time means more efficient production. However, it is important to link the delay response time to the number of rescheduling actions since the aim is to include most of the actions in one rescheduling to generate the lowest number of problems possible, such as a higher lead time, higher costs, and highly unbalanced station workloads. Therefore, the delay response time is fundamental to combine the different actions and reduce disruption in the production line.

The prediction horizon factor was also considered in the analysis. It refers to how far machine learning or artificial intelligence looks for a defect accurately.

To sum up, seven factors were defined to solve the problem in a statistical way and create an added-value tool that identifies the optimal settings of tuning parameters. Table 13 shows all the factor values per each factor level used in the simulation model.

Table 13: Factor Levels

Factors	LvL1	LvL2
NORT(hours)	5	100
NODRT(hours)	2	30
NDRT(hours)	3	13
NDDRT(hours)	1	20
NPRT(hours)	0.2	1.2
NPDRT(hours)	0.1	0.9
PH(minutes)	5	50

4.8.2 Events Management Algorithm

Re-scheduling policy is a critical for the success of a manufacturing system. Figure 27 presents an example of the proposed methodology, which is explained in details in Figure 28, and two examples of other re-scheduling policies, i.e. re-scheduling after every event and re-scheduling at fixed time intervals. Re-scheduling after every event is not efficient at present and at the same time, it is impossible because it will cause great confusion within the production line. If it was possible to coordinate something like this then it would be the ultimate solution but now it is impossible to be implemented. A widely used re-scheduling policy is to re-schedule at fixed time intervals, which is not as effective as the proposed events management algorithm, because it does not capture the dynamic nature of the production and the schedule ends up to be inferior to the one produced by the proposed approach. In Figure 27, several unexpected events that occur at different times are represented. The first event (O1) is the new order. It occurs at time t_o and could be released to the shop floor at time t_1. However, while event O1 is prepared to be released, the new defect event (D1) occurs with a releasing time t_3. The main problem is that $t_3>t_1$; thus, the production has to be rescheduled twice in a short period, both at t_3 and t_1. This could cause inefficiency in the production flow and increase operation costs. Therefore, the release of the O1 event is delayed to manage both the O1 and D1 events in the same rescheduling and reduce the waste of time and resources.

Before the rescheduling time S1, event P1 also occurs at t_4. However, it cannot be released at S1 since $t_4 >$ S1. Therefore, both P1 and O2 are released together in the second rescheduling of the production at S2. The last event D2 cannot be released at S2 since $t_{12} >$ S2; therefore, production has to be rescheduled again at S3.

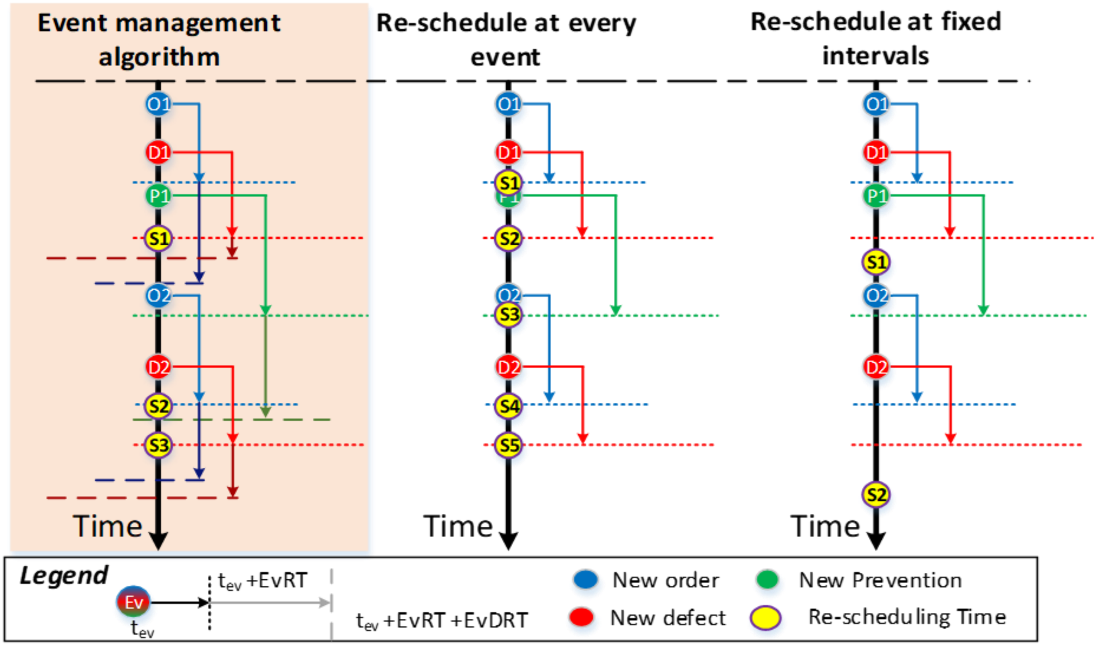
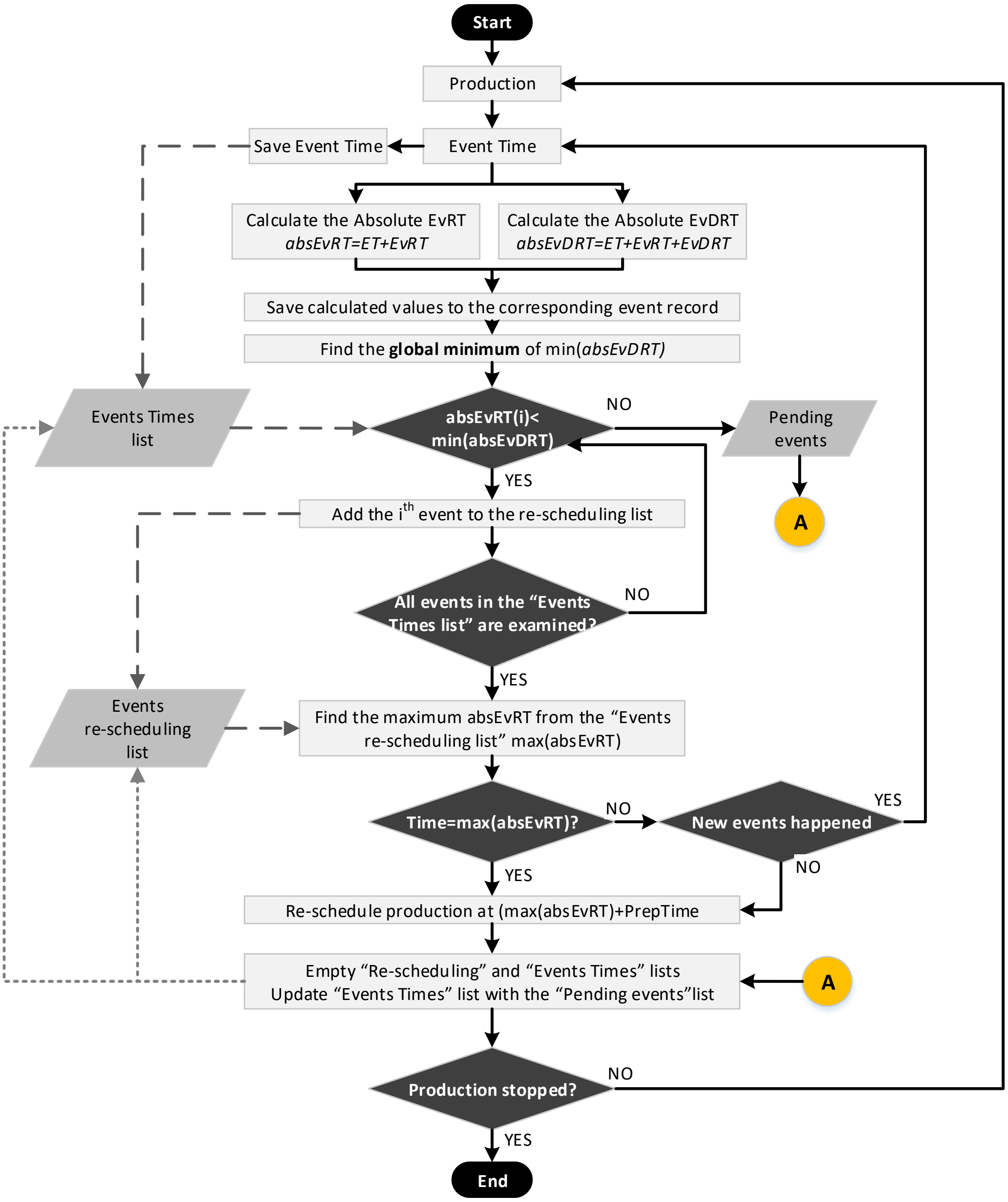


Figure 27: Re-scheduling policies

Using the defined parameters EvRT and EvDRT an event management algorithm is designed to identify the optimal time to perform the re-scheduling in the most efficient way by reducing the number of rescheduling actions. Figure 28 illustrates in details the flowchart of the developed algorithm. The algorithm runs continuously in order to serve all the incoming events. Each event is accompanied by the time that occurred (ET), which is saved to a list with all the current events under investigation. For each event two times are calculated, i.e. the absolute EvRT and the absolute EvDRT. Then the algorithm finds the minimum EvRT from the current event list and use it as a comparison point. For each event saved in the event list its absEvRT is compared with the minimum absEvDRT and if it is smaller, then the current event will be included to the next re-scheduling and it is saved to the re-scheduling event list; otherwise the event is saved for future re-scheduling. This process continues until there are no other events in the event list. Once this procedure is finished then the maximum absEvRT from the re-scheduling events list is found and if the current time is equal to the maximum absEvRT the production is rescheduled at max(absEvRT) + PrepTime. PrepTime is the time that the production needs for preparing for the re-scheduling. The records of the event times and re-scheduling lists are deleted, and the event time list is filled with the events at the pending event list.



Start
Production
Save Event Time
Event Time
Calculate the Absolute EvRT
absEvRT=ET+EvRT
Calculate the Absolute EvDRT
absEvDRT=ET+EvRT+EvDRT
Save calculated values to the corresponding event record
Find the global minimum of min(absEvDRT)
Events Times list
absEvRT(i)< min(absEvDRT)
NO
Pending events
YES
A
Add the ith event to the re-scheduling list
All events in the "Events Times list" are examined?
NO
YES
Events re-scheduling list
Find the maximum absEvRT from the "Events re-scheduling list" max(absEvRT)
Time=max(absEvRT)?
NO
New events happened
YES
NO
YES
Re-schedule production at (max(absEvRT)+PrepTime
Empty "Re-scheduling" and "Events Times" lists
Update "Events Times" list with the "Pending events"list
A
Production stopped?
NO
YES
End

Figure 28: Event management algorithm

4.8.3 Performance indicators

The quality of the solution of each simulation is measured based on five KPIs, which are considered the main ones for evaluating the production system, as already implemented in [63] [194] and [97]. More specifically, the solutions are measured based on (i) makespan, (ii) tardiness, (iii) number of production rescheduling actions, (iv) production cost, and (v) rescheduling cost. Makespan and tardiness are measures of schedule efficiency based on time and the equation reported [47]. The number of production rescheduling actions points out the number of times the production has to be rescheduled in a precise interval due to the occurrence of unexpected events, as mentioned in chapter 2.5.

However, time-based performance measures do not reflect the economic aspect of the production system. Therefore, it is important to evaluate scheduling decisions and strategy based on economic KPIs. In this research, the two economic performance indicators of production and re-scheduling cost (RSC) are implemented in equation (28) as follows:

$$RSC = w_c * NT * RMSCF \tag{28}$$

However, the five performance indicators have different units. Therefore, they must be normalized to combine them in one value, using the methodology presented in chapter 4.9. The performance indicators are normalized according to equation (31) since they are all cost behaving indicators, which means they need to be minimized. As shown in Figure 32, the last step is to calculate the utility value according to equation (34). The normalized values ($\hat{C}_{ij}$) are multiplied by a certain weight factor (WF_j) and added together into one single value, namely the utility value U_i [195]. The utility value can vary between 0 and 1, where 1 indicates the best result, while 0 indicates the worst. This framework enables the comparison among the different rescheduling solutions.

4.8.4 Design of experiments

At this stage, the solutions of the simulations and factors are analyzed through subparts of the Design of Experiments methodology to optimize the process [196]. The Design of Experiments (or Matrix Experiment) is a method developed by Taguchi. It is a statistical methodology that performs experiments in a structured way and not randomly. It is based on a matrix containing a set of experiments, where the settings of the parameters are changed to analyze the effect of those parameters. More specifically, the matrix comprises orthogonal arrays that allow the simultaneous analysis of the process parameters. It is considered one of the most important methods in robust design since it provides more accurate results of the effect of parameters compared with other traditional techniques. Indeed, it is possible to solve complex problems with a smaller number of experiments to be conducted. Therefore, it is possible to conduct this method in a faster and more cost-effective way [196].

In this chapter, the aim of conducting orthogonal experiments is twofold:

- To identify the optimal factor combinations
- To determine the impact of each factor on the solution

The matrix selected for this analysis was the orthogonal array L32, which means that 32 experiments had to be performed. The L_{32} orthogonal array can be found in Annex 4, chapter B and Table 50. As discussed previously in chapter 4.8.1, seven factors were analyzed and each had two factor levels. To achieve our aim, the results were analyzed using the ANOM and ANOVA methods. ANOM was used to determine the optimal level of each factor. The effect of a factor level is the deviation it generated from the overall mean and it is calculated according to equation (34) [196]. The different factor levels for each parameter are represented with linear graphs. Moreover, to dig deeper into the analysis, the relative difference was also calculated using equation (30).

$$Effect_{k,z} = \frac{1}{m_{k,z}} * \sum\nolimits_{z=1}^{m_{k,z}} Q_{k,z} \tag{29}$$

$$\text{Relative Difference} = \frac{[\mathrm{m}_{max} - \mathrm{m}_{\mathrm{min}}]}{[\mathrm{m}_{\mathrm{min}}]} \times 100 \tag{30}$$

ANOVA was used to establish the contribution of each factor to the objective function and to estimate the error variance. The larger the value, the higher the factor’s effect on the production system. The ANOVA table was generated through the support of MATLAB and the contribution percentage was calculated with the sum of squares. Moreover, the ANOVA table helped us to identify the most statistically significant interactions based on the probability F. Furthermore, linear graphs were used to properly assign the individual interactions to the L_{32} matrix to avoid confounding results and be able to extract the interactions. The linear graph used is presented in Figure 76 in Annex 4, chapter B. Furthermore, Table 52 (Annex 4, chapter B) shows the interactions studied as well as the columns of L_{32} orthogonal arrays that have been assigned. Therefore, ANOVA analyses only show much a factor impacts, while ANOM gives us the direction of the impact.

4.8.5 Results and discussion

The methodology described above was applied in an industry environment, and more specifically in a semiconductor manufacturing production system, to gain a deeper knowledge on the behavior of the defined factors. In this chapter, the results of the simulation model are presented and analyzed through the ANOM and ANOVA methodologies.

Table 51 in Annex 4, chapter B shows the experimental matrix, where the seven factors with two levels each are combined through the use of the L32 orthogonal array. The solution quality values represent the results of each experiment and they range between 0 and 1. The solution quality values were calculated by combining equations (32) and (33). Overall in the experiments, the result average was 0.375 and the standard deviation was 0.151. The best result was obtained in experiment n° 10 where the solution quality value was 0.680. To increase the solution quality of the model, ANOM and ANOVA were performed with the aim of finding the optimal factor combination and observing the effect of each.

Thanks to the orthogonal array, it was possible to apply the ANOM methodology, which is responsible for performing the experiment in a structured way and not randomly. It enables studying each factor separately in how they affect the results and estimate the impact of interactions between factors.

The results are plotted in Figure 29, which shows how each factor (the direction) affects the quality of the solution. The aim was to find the right setting for those parameters to tune them in the most efficient way and develop the most effective production system. Therefore, the optimal factor level for each factor had to be selected to obtain the highest quality solution.

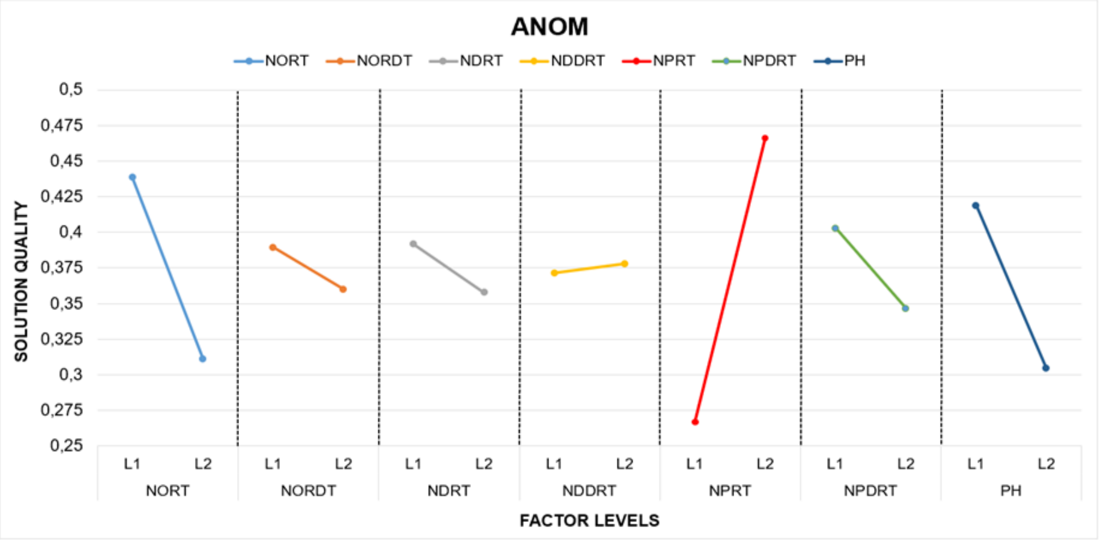


Figure 29: Solution quality per each factor (ANOM)

Overall, the main observation was based on the slope of the linear lines. As can be observed, the slope of most of the factors was negative, which indicates that the lowest are the timing variables and the best would be the solution. However, the new prevention response time (NPRT) and NDRT factors had an opposite trend, which highlights an opposite impact on the solution quality. Moreover, it is possible to observe that the change in the values of the delay response time variables had an almost negligible effect on the solution quality; this type of variable contributes to making the model more robust.

Digging deeper into the analysis, the effects of the main factors are described as follows. As shown in Figure 29, the factor with the highest relative difference is the NPRT factor at almost 75%. From the ANOM graph, it can be observed that the higher the prediction time, the better the quality of the solution. Therefore, a slower response time to prevention action means better rescheduling. This result is expected because it could be linked to the number of rescheduling actions of production. Indeed, a higher reaction time means that it is possible to manage the rescheduling in a more efficient manner since the numbers of production rescheduling actions are reduced. Moreover, a slower response time could have a positive impact on the economic and production flow aspects. Indeed, using a low response time value means doing maintenance or machine tuning too many times in a short period, which would cause many interruptions in the production flow and increase the production costs per item. The optimal value refers to a response time of 1.2 hours, which is a reasonable value because it allows the prevention of most defects and ensures a more continuous flow.

Another factor with a high relative difference is NORT. As the NPRT factor, it is a very reactive factor; changing its value would change the solution quality significantly. On the other hand, its trend (negative) is opposite to NPRT since the scheduling solution becomes worse through increasing the reaction time. New order events are one of the main causes of disorder at the shop floor level and could create huge costs if not managed correctly. Indeed, it is better to act immediately when new orders arrive so as not to accumulate them and be able to satisfy demand without incrementing supply chain costs (inventory cost for raw materials). Moreover, a quick reaction to new orders arriving indicates that the production is very flexible and has the capability to efficiently handle this type of unexpected event. It also has a positive impact on customers' experience and satisfaction since they are able to have their demands and needs met in a short time. This feature would help the company to stand out in the market and become more resilient. For instance, during the COVID-19 pandemic, the demand for medical instruments has received an incredible boost, which has consequently led to the company receiving many new orders every day. Thanks to their resiliency, they would have been able to meet the demand and increase their market share.

However, the huge gap between the two quality solutions could also be explained by the high difference between the factor levels (5 and 100). Indeed, considering a shift of 8 hours per day, a production line with factor level 1 could manage a new order per day. By contrast, by adopting factor level 2, it is possible to manage a new order in 12.5 days.

The last factor with a relevant effect is PH since it had a relative difference of 46.1%. As the NORT factor, it had a negative trend, which means that the higher the reaction time, the better the solution quality. A lower response time to PH means a highly digitalized system, which can forecast accurately and the possible defects in a short time.

The factors NORDT and NDRT had small effects on the solution since they had a relative difference of 8.2% and 9.4%, respectively. The NDDRT had a negligible effect on the solution quality since its relative difference was 1.7%.

One of the main goals in conducting a matrix experiment is to optimize the process design. To achieve this goal, the optimum level per each factor has to be identified. In this case, the optimum level for a factor was the level that gave the highest value for solution quality. Overall, the quality of the solution increases if the factor level is 1, except for NPRT and NDDRT, which achieved higher solution quality with factor level 2. Therefore, it is possible to conclude that the best setting is $NORT_1$ $NORDT_1$ $NDRT_1$ $NDDRT_2$ $NPRT_2$ $NPDRT_1$ PH_1.

It is important to precisely state that the predicted best setting could not be one of the rows in the orthogonal array, as in this case. In the second step, the impact of each factor on the solution was analyzed using the ANOVA table. The table provides important information that would help us to build a more robust model. Indeed, from the sum square values, it is possible to calculate the contribution of each factor on the solution. The results of the contributions are shown in Figure 30.

Table 14: ANOVA Table

Variable	Coef.	Sum Sq.	DoF	Mean Sq.	F	Prob>F
Intercept	0.37463	0.00706	-	-	-	-
NORT	-0.06382	0.13035	1	0.13035	81.63	0
NORDT	-0.01479	0.007	1	0.007	4.39	0.0492
NDRT	-0.01682	0.00905	1	0.00905	5.67	0.0273
NDDRT	0.0032	0.00033	1	0.00033	0.2	0.6559
NPRT	0.09117	0.26601	1	0.26601	166.59	0
NPDRT	-0.02805	0.02517	1	0.02517	15.77	0.0008
Prediction horizon	-0.07013	0.15737	1	0.15737	98.55	0
NORT*NPRT	-0.02052	0.01348	1	0.01348	8.44	0.0087
NORDT*NPDRT	0.03423	0.03749	1	0.03749	23.48	0.0001
NDRT*NPDRT	-0.02295	0.01685	1	0.01685	10.55	0.004
NDDRT*NPRT	0.01698	0.00923	1	0.00923	5.78	0.026
Error	-	0.03194	20	0.0016		
Total	-	0.70427	31			

The highest contribution was given by the NPRT factor, which had a contribution of almost 38%. Therefore, it is of primary importance for the company to initiate the rescheduling of the production examining the NPRT factor, since the success of the model mainly depends on it. The high impact of NPRT could derive from the fact that the production of PCBs is highly automized; thus, a defect in the system could cause high damage in terms of production rates. Therefore, the maintenance of the production system and defect prediction is one of the main goals for this kind of company. In addition, PH and NORT factors had a contribution of almost 23% and 18%, respectively, while the impact of the remaining factors (NORDT, NDRT, NDDRT, and NPDRT) was almost negligible. In total, they only accounted for 5.90%.

Furthermore, Design of Experiments is a powerful method for estimating any interaction effect. The interaction is the relation among two factors, where the change in value of one factor could affect the output of the other factor. In this research, all possible interactions were analyzed. However, the confounding effect limited the number of combinations to be studied since it was not possible to distinguish between the effect of the interactions and the factors [196]. Moreover, only the most statistically significant interactions have been discussed. The importance of interactions is based on the probability F of each factor, which is shown in Table 4.1 (ANOVA Table); only the factors with a probability F (PF) lower than 5% can have statistically important results.

Four main interactions were identified:

- NORT*NPRT
- NORDT*NPDRT
- NDRT*NPDRT
- NDDRT*NPRT

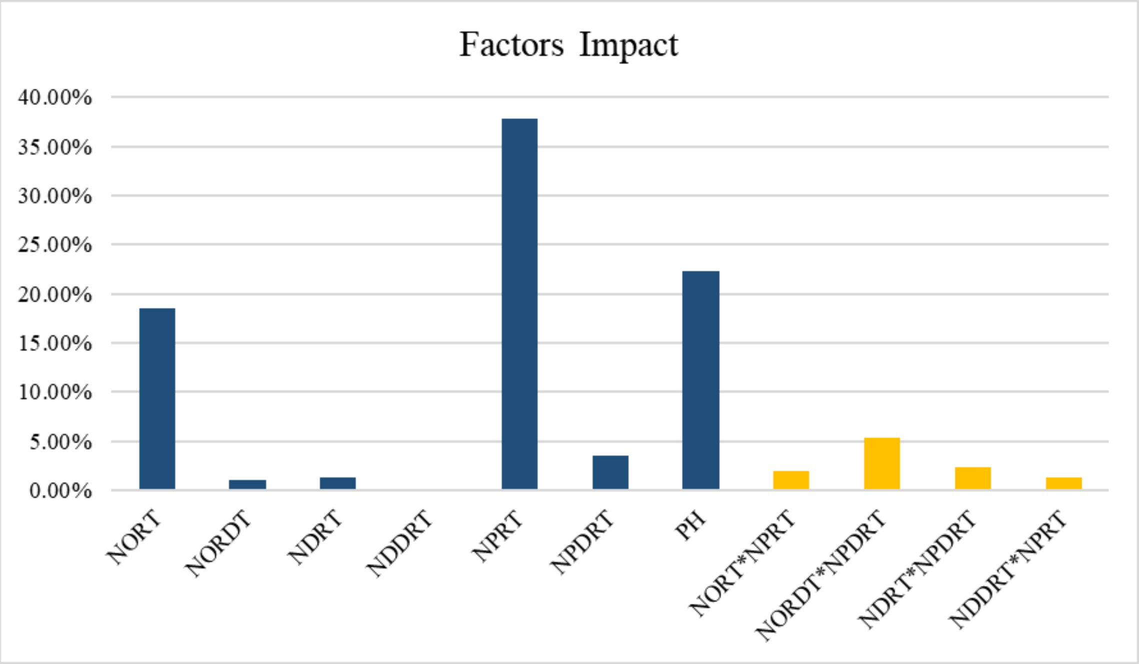


Figure 30: Impact of each factor on the solution

As shown in Figure 30, the total contribution of the interactions accounted for 10.94%. This can be considered quite high since they are interactions and higher than the sum of four factors. The interaction with the highest contribution was NORDT*NPDRT, which accounted for 5.32%. By contrast, the interaction with the lowest contribution was NDDRT*NPRT, which accounted for 1.31%.

The four interactions are represented by linear graphs, which provide the possibility of illustrating the interactions in the most direct way. Each dot and line represent a specific column of the orthogonal array. Figure 31 illustrates the significant interactions that were revealed from the ANOVA analysis.

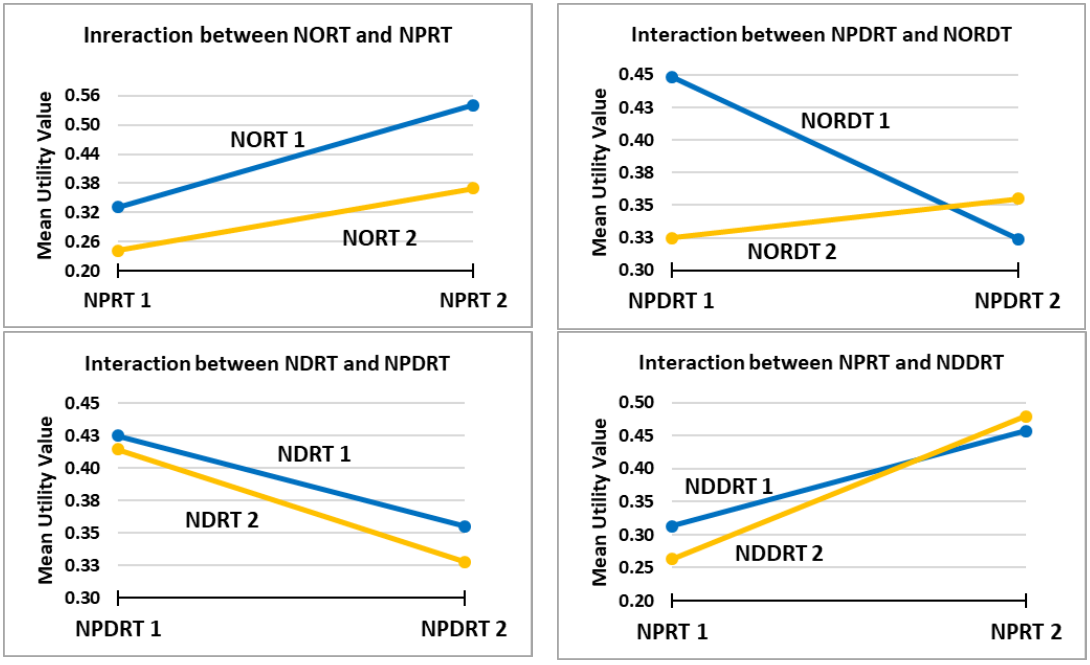


Figure 31: Factor interaction diagrams

NORT *NPRT, NDRT*NPDRT, and NDDRT*NPRT were synergistic interactions since the lines were not parallel and the direction of the solution quality did not change. By contrast, NORDT*NPDRT was an antisynergistic interaction since the lines were not parallel and the solution quality direction changed.

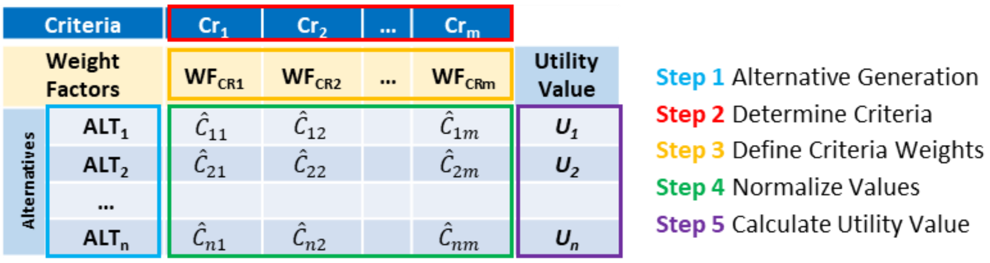
4.9 Aggregation of Multiple Values to Utility Value

The quality of the produced scheduling is evaluated based on a set of KPIs. A detailed description of the final KPIs set developed in the current scheduling tool is presented in chapter 4.10. But because the KPIs set is composed by more than one KPI it is almost impossible to compare alternative solutions comparing each KPI. For this reason, a method was selected to aggregate all the KPIs onto one single value and using that value be able to compare the alternative schedules. This value is named “utility value” taking values within the interval [0,1], where 0 is the worst solution and 1 the best. The selected method is not new, it is defined as multi-attribute decision-making (MADM) problem [197], this method might be old but it is used by many researchers [198][199][200]. Figure 32 presents the method at a glance for aggregating multiple values into one to reach a decision based on all the desired criteria.

At this point it should be mentioned that each KPI might have different units of measure from each other. Furthermore, some KPIs need to be minimized while others need to be maximized. Thus, a normalization of the obtained criteria values is required to overcome their conflicting nature and their different units of measure [198][199][200]. There are two types of criteria, “cost” criteria which are those that must be minimized and on the other hand the “benefit” criteria which must be maximized. Each type of criteria must be normalized by a different formula, more specifically equation (31) and equation (32) are responsible for normalizing “cost” and “benefit” criteria accordingly and the normalization process is performed relatively to the minimum and maximum KPI values as described by the equations (31) and (32). In those formulas C_{ij} corresponds to the j^th KPI value of alternative solution i, whereas $\hat{C}_{ij}$ represents the normalized value of C_{ij}. C_j^{max} and C_j^{min} is the global maximum and minimum values of j^th KPI.

The next step is to define the weight that each KPI will have. The weights illustrate the contribution to the utility value of each KPI. The KPI weight factors are user defined values according to the desired outcome. The cardinal rule of the selected method is that the following statement is satisfied $\sum_{j=1}^{m} W_j = 1$, $W_j \in [0,1]$ where m the number of KPIs and W_c the weight factor for KPI_c. The user can use different scale for the weights but before they are used, they must be converted to values between [0,1] and have a sum of 1, using equation (34).

The final step for calculating the utility value of each alternative solution is to multiply each KPI normalized value with the corresponding KPI weight factors and then sum up all the products together for each alternative as denoted by equation (33) [197].



	Criteria	Cr_1	Cr_2	...	Cr_m	
	Weight Factors	WF_{CR1}	WF_{CR2}	...	WF_{CRm}	Utility Value
Alternatives	ALT_1	$\hat{C}_{11}$	$\hat{C}_{12}$		$\hat{C}_{1m}$	U_1
	ALT_2	$\hat{C}_{21}$	$\hat{C}_{22}$		$\hat{C}_{2m}$	U_2
	...					
	ALT_n	$\hat{C}_{n1}$	$\hat{C}_{n2}$		$\hat{C}_{nm}$	U_n

Step 1 Alternative Generation
Step 2 Determine Criteria
Step 3 Define Criteria Weights
Step 4 Normalize Values
Step 5 Calculate Utility Value

Figure 32: Aggregation of multiple values to one method

Cost
$$\hat{\boldsymbol{C}}_{\boldsymbol{ij}} = \frac{C_j^{max} - C_{ij}}{C_j^{max} - C_j^{min}} \quad (31)$$

Benefit

$$\widehat{\boldsymbol{C}}_{\boldsymbol{ij}} = \frac{C_{ij} - C_j^{min}}{C_j^{max} - C_j^{min}} \tag{32}$$

Weighted Sum

$$\boldsymbol{U_i} = \sum\nolimits_{j=1}^{m} W_j \hat{C}_{ij} \tag{33}$$

Weight Factors
conversion to [0,1]
range with sum of 1

$$W_j = \frac{\widehat{W}_j}{\sum_{j=1}^{m} \widehat{W}_j} \ , \widehat{W}_j > 0 \tag{34}$$

4.10 Key Performance Indicators (KPIs)

The simulation performance (utility value) was evaluated based on seven main KPIs, which are shown in Table 15. Those KPIs were selected carefully to capture all the aspects that the ZDM concept might require. The implementation of ZDM requires the frequent rescheduling of the shop floor to incorporate all mitigation actions into the new schedule. Therefore, the first measured KPI was the rescheduling time, and more specifically the average rescheduling frequency. This KPI represents the average interval time for each rescheduling round. This measure simply shows on average how much time mediates between two rescheduling events. Equation (35) shows how it is calculated, the difference from every pair of successive rescheduling events is calculated, and then all the differences are summed and divided by the total number of rescheduling times in the given period of the simulation. This measure was selected as one of the main KPIs because it critical to study how the ZDM strategies affect it.

The second KPI is heavily and directly related to ZDM, namely the defect ratio of a specific manufacturing stage. It was measured for each manufacturing stage separately using equation (36). The calculation is simple: the quotient of the number of defective parts divided by the total number of parts produced by the specific manufacturing stage. In other words this KPI measures the percentage of defects at each manufacturing stage. In the implementation of the ZDM concept, this measure is crucial since it shows the effect of the implementation of the ZDM concept.

ZDM is implemented to improve the production quality and by extent reduce the amount of materials and energy required for manufacturing a specific quantity of products. Therefore, the third KPI was the energy consumption required for the manufacturing of the specific products. Equation (37) shows how it is calculated and what terms it includes. More specifically, this measure considers the energy required for the normal manufacturing of the products, the energy required for the repair of the repairable parts, and the energy required by the inspection equipment.

Meeting due dates is a crucial aspect of staying competitive and reliable. Therefore, the next KPI was the total weighted tardiness. In the current research work, each individual order was treated as a separate event. This means that the tardiness is calculated for each of the orders, but some orders are more important than others. For that reason, the weighted tardiness was calculated and not the simple tardiness. The weight in equation (38) depicts the importance of the order, which in chapter 4.7.1 was referred as OC.

The average order's makespan is the next KPI used for the evaluation of the performance of different simulation runs. Equation (39) shows how it is calculated: first, all the individual order makespans are summed together, and then this sum is divided by the number of orders.

Another important KPI that shows the efficiency of the production and quality of the schedule is the average machine utilization, equation (40). First, the individual machine utilization is calculated for each of the MFGs and then the average value of all MFGs is calculated. It is important that the machines are constantly occupied but simultaneously that time is left for maintenance and other actions for the improvement of the production.

The final—and probably the most important—KPI in the current study was the final PC (equation (41)). The KPI is not an outcome of one equation; it consists of many different factors. Table 16 contains all the terms that compose the final unit cost. This term is composed of 10 individual terms. Briefly, the terms that compose the final unit cost are the rescheduling cost, raw materials cost, operational cost, losses due to poor quality, detection prevention cost, prediction prevention cost, repair cost, maintenance cost, and delay penalty cost.

Table 15: Main KPIs

$\boldsymbol{AverageReSchedulingFrequency = \frac{\sum_{r=1}^{TotalReschedulingTimes}(RTime_r - RTime_{r-1})}{TotalReschedulingTimes}}$	(35)
$\boldsymbol{DefRatio_{MFG}} = \frac{NumberOfDefectedParts_{MFG}}{TotalNumberOfPartsProduced_{MFG}} * 100\%$	(36)
$\boldsymbol{EnergyConsumption} = \sum_{i=1}^{MFG}(TotalOperationTime_i * EnergyConsumption_i) + \sum_{q=1}^{nRepairTask} TotalRepairOperationTime_q * EnergyConsumption_q + \sum_{f=1}^{inspM}(TotalInspectionTime_f * EnergyConsumption_f$	(37)
$\boldsymbol{WeightedTotalTardiness} = \sum_{o=1}^{nOrders}(OrderFinishTime_o - DueDate_o) * W_o$	(38)
$\boldsymbol{AverageMakespan} = \frac{\sum_{0=1}^{nOrders} OrderFinishTime_o}{nOrders}$	(39)
$\boldsymbol{AverageMachineUtilization} = \frac{\sum_{i=1}^{MFG}\frac{MachineOperationTime_i}{TotalMachineTime_i}}{MFG} * 100\%$	(40)
$\boldsymbol{FinalUnitCost = \frac{RSC + IC + RepC + PPC + DPC + PQL + OC + MC + MaintC + DPenC}{OrderSize}}$	(41)

The first KPI that the final unit cost is composed of is the rescheduling cost (RSC), which was explained in chapter 4.8.3. Next is the operational cost, which includes the cost for operating the machines and the labor cost of the operators (equation (44)). The machine operational cost (OpC) includes the cost for the operation of the machine, setup cost, and machine degradation. The terms totalProcessingTime and LabourTime have incorporated some uncertainty to simulate the real production environment, where unpredictable events might happen and disrupt the normal production. This is achieved by varying the processing time of each task by ±5% of the predetermined value. Moving forward, defects are an unavoidable event; therefore, the cost that arises from poor quality (PQL) should be included in the final PC in order to be closer to reality. The next three KPIs refer directly to the ZDM concept, and more specifically to the three ZDM strategies. Starting with the detection – prevention strategy, the cost that arises from the implementation of the prevention actions of that ZDM strategy are calculated and compose the fourth KPI (DPC, equation (47)). In the same concept for the prediction – prevention strategy, the cost that arises from the prevention actions required is the fifth KPI (PPC, equation (48)). The last ZDM strategy is detect – repair, and this KPI is composed of the cost for manually inspecting the defective part to establish a correct procedure for repairing it, followed by the raw materials required for performing the repair and the operational cost for the machine to be used for repair and labor costs for the operator that will operate the machine (RepC, equation (49)). Moving forward, the maintenance cost is also included in the final PC (MaintC, equation (50)). Maintenance is a critical factor for achieving ZDM. Finally, when the due dates are not met, there is a penalty cost the manufacturers pay to their customers to compensate the delay from the agreed delivery time (DPenC, equation (51)).

Table 16: FinalUnitCost sub-KPIs

$\boldsymbol{RSC = ReschedulingCost} = 2 * RMSCF * NT - \frac{RMSCF}{NT_{tot}} * NT^2$	(42)
$\boldsymbol{MC = TotalMaterialCost} = \sum_{r=1}^{NumberOfTasks} RawMatCost_r$	(43)
$\boldsymbol{OpC = TotalOperationalCost} = \sum_{i=1}^{MFG} (TotalProcessingTime_i * MachineOpCost_i) + \sum_{w=1}^{nOperators} (LabourTime_w * LabourCost_w)$	(44)
$\boldsymbol{PQL = PourQualityLosses} = NumberofDefectedProducts * productTotalCost$	(45)
$\boldsymbol{IC = InspectionCost} = \sum_{i=1}^{MFG} \sum_{f=1}^{nInpsTasks} \boldsymbol{InspMachineOpCost_i * InspTime_f}$	(46)

$\boldsymbol{DPC = DetectionPreventionCost} = \sum_{i=1}^{MFG} \sum_{e=1}^{nPrevActions} \{SparePartsCost_{ie} + PrevTime_{ie} * LabourCost_{ie}\}$	(47)
$\boldsymbol{PPC = PredictionPreventionCost} = \sum_{i=1}^{MFG} \sum_{e=1}^{nPrevActions} \{SparePartsCost_{ie} + PrevTime_{ie} * (LabourCost_{ie} + ProdLosses_{ie})\}$	(48)
$\boldsymbol{RepC = RepairCost} = \sum_{q=1}^{nRepairTask} \left(ManualInspTime_q * labourCost + RawMaterialsCost + ProcessingTime_q * MachineOperationCost + labourTime_q * labourCost\right)$	(49)
$\boldsymbol{MaintC = MaintenanceCost} = \sum_{i=1}^{MFG} \sum_{j=1}^{nMaint} \left\{SparePartsCost_{ij} + MaintTime_{ij} * (LabourCost_{ij} + ProdLosses_{ij})\right\}$	(50)
$\boldsymbol{DPenC = DelayPenaltyCost} = \sum_{\boldsymbol{o=1}}^{nOrders} W_3\, ln\left(1 + W_8(OrderFinishTime_o - DueDate_o)\right) * OV_o * OC_o$	(51)

4.11 ZDM control Parameters definition

To answer Research Question 3 it is required to define which are the critical ZDM parameters that describe the equipment or the software that implements each of the four ZDM strategies Figure 2. The generic category types for the ZDM parameters are cost and time required for the implementation of the ZDM strategy as well as the accuracy and effectiveness. For each ZDM strategy three parameters are defined, Table 17, Table 18 and Table 19 present the selected parameters. These parameters were selected because they are the most important when selecting equipment for ZDM and most importantly they are generic and common across all the types of equipment and software. The parameters marked with "R" are relative values. The current methodology aims to assist manufacturers in the process of designing, re-designing, or adjusting manufacturing systems for new products to determine the optimal specifications for the equipment in need for quality improvement. The method is quite simple but the outcome has a significant impact on the achievement of ZDM.

Each product has some nominal characteristics, which are considered those that are calculated under ideal manufacturing conditions, which means that there are no defects, no delays and in general there is no interruptions to the manufacturing process. In other words, they are the theoretical characteristic values. In this case, the total production cost and time were calculated and used for converting the absolute use-case specific values to relative values. This was achieved using equation (52), which is simply the absolute value of the ZDM parameter divided by the corresponding total estimated product value. The ratio approach selected provided a relative indicator that shows how much extra time or cost is required for the inspection having the nominal values as a reference. The reason behind this simple idea is to unlink the results from a specific case and be able to reuse them for other cases where the product is different but the ratios remain the same. For example, there are currently numerous different inspection technologies, each with different characteristics, and the implementation of inspection points in every manufacturing stage is not possible due to cost and time constraints. Therefore, manufacturers require a tool that can indicate the acceptable combinations between inspection time and inspection cost based on the total processing time and total PC [193].

$$Relative\ factor\ Value = \frac{Absolute\ ZDM\ Value}{Estimated\ Total\ Value} \quad (52)$$

Table 17: Detection – Repair control parameters

Parameter Name	Parameter short Name	Parameter Description
Inspection Cost (R)	F1/IC	The cost related to the operation of the inspection machine per item inspected
Inspection Time(R)	F2/IT	The time that the inspection equipment requires in order to inspect one part
Detection Accuracy	F3/DA	The accuracy that the inspection equipment has. Measured in percentage.
Repairing Cost (R)	F4/RC	The average repairing cost. This cost includes the extra raw materials needed for the repair and the labor and machine operational cost for performing the repair
Repairing Time (R)	F5/RT	The time that is required in order to perform the repair
Reparability	F6/Rep	Reparability represents a percentage that shows how many parts are reparable out of the total.

Table 18: Detection – Prevention control parameters

Parameter Name	Parameter short Name	Parameter Description
Inspection Cost (R)	F1/IC	The cost related to the operation of the inspection machine per item inspected
Inspection Time (R)	F2/IT	The time that the inspection equipment requires in order to inspect one part
Detection Accuracy	F3/DA	The accuracy that the inspection equipment has. Measured in percentage.
Prevention Cost (R)	F4/PvC	The related cost for the raw materials and operator time cost that are required for the implementation of the prevention actions.
Prevention Time (R)	F5/PvT	The time that is required in order for the operator to implement the prevention actions. Those prevention actions could be either small maintenance or machine tuning
Prevention success Rate	F6/PvSR	It is a percentage that indicates the probability of the prevention actions to have real effect to the production line. In

		other words, if the prevention actions are successful or there was a miss-diagnose.

Table 19: Prediction – Prevention control parameters

Parameter Name	Parameter short Name	Parameter Description
Prediction Horizon	F1/PdH	Is the timeframe that the prediction algorithm looks ahead
Prediction Accuracy	F2/PdA	Is the probability of successfully predicting a defect in the given prediction horizon
Prevention Reaction Time	F3/PdReaT	Is the time that is required for implementing the prevention actions.
Prevention Cost (R)	F4/PvC	The related cost for the raw materials and operator time cost that are required for the implementation of the prevention actions.
Prevention Time (R)	F5/PvT	The time that is required in order for the operator to implement the prevention actions. Those prevention actions could be either small maintenance or machine tuning
Prevention success Rate	F6/PvSR	It is a percentage that indicates the probability of the prevention actions to have real effect to the production line. In other words if the prevention actions are successful or there was a miss-diagnose.

4.12 Digital Twin methodology

As explained briefly in the definition of Research Question 2 the creation of a DT for describing the developed scheduling tool was not part of the initial research plan. Once the development and the validation of the scheduling tool were finished it was observed that the simulation takes a significant amount of time, which is a prohibiting factor for running a significant number of simulations. To solve this problem the use of DT approach was introduced in order to create a DT for the scheduling tool developed to avoid the actual simulation will give the same results as if the simulation was run. The methodology for the creation of the DT that will be presented will be capable to make a model of the scheduling for a specific use case. If the case changes the same methodology should followed again. The goal of Research Question 3 is to be able to map the performance of each ZDM pair of strategies under different parameter sets. Therefore, the control parameters of the DT model would be the ZDM control parameters defined in chapter 4.11. Therefore, for each ZDM pair strategy a different DT model will be created because each ZDM pair strategy has different control parameters.

The creation of the DT model is based on a statistical method called Design of Experiments (DoE). More specifically the Taguchi approach was used as the basis of the developed DT method [196][201]. The DoE approach was selected because it provides a methodological approach to capture the individual effects of each of the control parameters. Furthermore, the Taguchi method also provides the methodology for performing the minimum number of experiments that can produce statistically significant results and more importantly defines the experiments that must be performed.

To capture the effect of each control parameter – factor, a high resolution is required for the factor values. To this extend the L_{25} orthogonal array was selected as it fits exactly to the current problem. More specifically L_{25} can host up to six factors with five levels each. Furthermore, L_{25} does not consider interactions between the factors and therefore the results will contain the factors main effects unconfounded. L_{25} orthogonal array imposes that 25 experiments should be performed using the factor levels that are denoted by the experiment line in L_{25}. The selected orthogonal array can be found in Annex 4A.

Once the experiments denoted by L_{25} are performed, they are analyzed using the Analysis of Means (ANOM) method. This method captures the effect of each factor level has to the observed value, in this case the utility value (U). Using equation (53) the average S/N ratio of each factor level is calculated. Where R and z are the number of levels and the actual level of factor k. The effect of a factor level is defined as the deviation it causes from the overall mean, equation (54). Equation (55) calculates the overall mean of the results, where Ne is the total number of experiments imposed by L_{25}.

$$M_{k,z} = \frac{1}{R_k} * \sum_{z=1}^{R_k} U_{k,z} \tag{53}$$

$$effect_{k,z} = (M_{k,z} - \mu) \tag{54}$$

$$\mu = \frac{1}{Ne} * \sum_{z=1}^{Ne} U_z \tag{55}$$

At this point it should be reminded that the goal is to create a mathematical model that when we enter a set of ZDM control parameters, it will give as an output the predicted utility value that would be the outcome of the scheduling tool. So far, the individual effects and average S/N ratios of each factor have been calculated. To integrate all those data together into one equation the method of the additive model was used as described in [196].

The additive model requires the ANOM results for calculating the corresponding factor coefficients for each level. For the current design of experiments (L_{25}, and six factors), the additive model has the following form (equation (56)). The letters A, B, C, D, E, and G represent each factor and the subscripts denote the level of each factor. The observed value is marked with "$\widehat{U}$" and it is the predicted utility value calculated by DT model. The lowercase letters of the factors represent the coefficients that correspond to each factor level. Furthermore, "μ" represents the overall mean and σ_e the error variance. In the current research, the error variance was considered near zero and therefore it was not taken into account. $M_{k,z}$ represents the ANOM results for each factor for each level. Using equation (57), the additive model coefficients could be calculated, considering that σ_e=0 [196], which simplifies the equation (57) to equation (54).

$$\widehat{U}(A_i, B_j, C_k, D_l, E_q, G_h) = \mu + a_i + b_j + c_k + d_l + e_q + g_h + \sigma_e \tag{56}$$

$$M_{k,z} = \mu + m_{k,z} + \frac{1}{3}\sigma_e^2 \tag{57}$$

The result from the additive model would be a set of five coefficients for each factor, which represent the coefficients for each level. In that way, the observed value can be estimated for all possible combinations of factor levels. This is very helpful because with only 25 experiments that occur from the L_{25} orthogonal array, we can calculate the results of $5^6 = 15625$ combinations without the need for extra simulations, saving valuable time. The prediction model created is capable only of estimating the result for the specific factor levels defined in the L_{25} orthogonal array. This is limiting since the prediction models are not flexible because they cannot estimate the observed value for any value of a factor within the defined range.

Up to this point the methods presented were not new, but taken from the literature. The addition that was made to this method was to convert the discrete coefficient values to a continuous model. This idea came up as an answer to the question that was raised, "what is happening to the internal points between two factor levels?”. The question was raised because the goal of Research Question 3 is to examine as many as possible combination of ZDM parameters to map the performance of each ZDM pair strategy. The result of the additive model is the matrix presented in equation (58), which contains the coefficients for each factor level ($m_{k,z}$, e.g. $m_{1,1}=\alpha_1$, $m_{2,2}=b_2$). Additionally, from the initial data the matrix on equation (59) is formed which contains the levels values for each factor.

$$Fcoef = \begin{bmatrix} a_1 & b_1 & c_1 & d_1 & e_1 & g_1 \\ a_2 & b_2 & c_2 & d_2 & e_2 & g_2 \\ a_3 & b_3 & c_3 & d_3 & e_3 & g_3 \\ a_4 & b_4 & c_4 & d_4 & e_4 & g_4 \\ a_5 & b_5 & c_5 & d_5 & e_5 & g_5 \end{bmatrix} \tag{58}$$

$$Flev = \begin{bmatrix} A_1 & B_1 & C_1 & D_1 & E & G_1 \\ A_2 & B_2 & C_2 & D_2 & E_2 & G_2 \\ A_3 & B_3 & C_3 & D_3 & E_3 & G_3 \\ A_4 & B_4 & C_4 & D_4 & E_4 & G_4 \\ A_5 & B_5 & C_5 & D_5 & E_5 & G_5 \end{bmatrix} \tag{59}$$

To achieve the conversion from discrete values to continue values the two matrices Fcoef and Flev were combined and formed cartesian points. As x-coordinate was set the Flev value and as y-coordinate the Fcoef value, leading to the set of cartesian points V, equation (60) which is one step before the desired result. As a last step, a piecewise quadratic interpolation is performed for each of the columns of V, leading to the creation of a set of quadratic equations that pass through the defined points. An example of that process is presented in equation (61) where the four quadratic equations for factor 1 are presented. Equation (62) illustrates the generic form of the quadratic equations for describing all the factors. Finally, equation (63) illustrates the final form of the DT which is responsible for estimating the utility value for a given set of parameter values within the specified range. Where K is the number of factors, which in the current case is 6.

$$V = \begin{bmatrix} (A_1, a_1) & (B_1, b_1) & (C_1, c_1) & (D_1, d_1) & (E_1, e_1) & (G_1, g_1) \\ (A_2, a_2) & (B_2, b_2) & (C_2, c_2) & (D_2, d_2) & (E_2, e_2) & (G_2, g_2) \\ (A_3, a_3) & (B_3, b_3) & (C_3, c_3) & (D_3, d_3) & (E_3, e_3) & (G_3, g_3) \\ (A_4, a_4) & (B_4, b_4) & (C_4, c_4) & (D_4, d_4) & (E_4, e_4) & (G_4, g_4) \\ (A_5, a_5) & (B_5, b_5) & (C_5, c_5) & (D_5, d_5) & (E_5, e_5) & (G_5, g_5) \end{bmatrix} \tag{60}$$

$$\begin{bmatrix} (A_1, a_1) \\ (A_2, a_2) \\ (A_3, a_3) \\ (A_4, a_4) \\ (A_5, a_5) \end{bmatrix} \quad => \quad \alpha(x) = \begin{bmatrix} q_{11} * x^2 + s_{11} * x + t_{11} & , A_1 \leq x < A_2 \\ q_{12} * x^2 + s_{12} * x + t_{12} & , A_2 \leq x < A_3 \\ q_{13} * x^2 + s_{13} * x + t_{13} & , A_3 \leq x < A_4 \\ q_{14} * x^2 + s_{14} * x + t_{14} & , A_4 \leq x \leq A_5 \end{bmatrix} \tag{61}$$

$$FcoefContinues(x)_k = \begin{bmatrix} q_{k1} * x^2 + s_{k1} * x + t_{k1} & , Flev_{1k} \leq x < Flev_{2k} \\ q_{k2} * x^2 + s_{k2} * x + t_{k2} & , Flev_{2k} \leq x < Flev_{3k} \\ q_{k3} * x^2 + s_{k3} * x + t_{k3} & , Flev_{3k} \leq x < Flev_{4k} \\ q_{k4} * x^2 + s_{k4} * x + t_{k4} & , Flev_{4k} \leq x < Flev_{5k} \end{bmatrix} \tag{62}$$

$$\widehat{U} = \mu + \sum_{r=1}^{K} FcoefContinues(x)_r \tag{63}$$

5 Industrial Application

For the testing and validation of the proposed tools and methodologies, a real-life industrial use case was used. The results that will be presented are created using the industrial data and the developed scheduling tool and the DT methodology. The chapter is organized as follows: chapter 5.1 will present the details and some analytics about the industrial use case, chapter 5.2 will present the DT models and the results for the validation of the accuracy of the DT models. Finally chapter 5.3 will present the results from the simulations regarding the current industrial case, whereas those results are used to answer Research Question 3.

5.1 Industrial use case definition

The industrial use case is in the semiconductor domain, and more specifically in the manufacturing of PCBs for medical equipment. Moreover, the product under investigation can be considered expensive since its manufacturing cost is within the range of €3800–6000. Therefore, it is essential for that production system to have a correct and balanced quality control system to minimize defective parts and stay competitive.

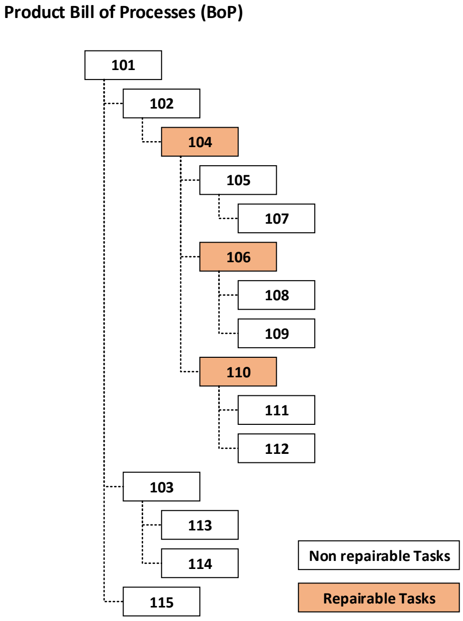


Figure 33: Product Bill of Processes (BoP)

Figure 33 presents the BoP of the product under investigation. In total, it is composed of 15 tasks to be manufactured. Those tasks represent only the manufacturing operations with no quality control tasks included. In other words, this is the BoP in an ideal world without defects. This BoP is modified at each simulation run to add all the required inspection tasks that are imposed by the corresponding ZDM strategy. In the detect – repair strategy, a small complication exists compared with the other two ZDM strategies in that it cannot be applied to all of the tasks. This is because after some processes, repair is not possible and the defective part must be discarded. The tasks that are repairable are marked in orange. The repairable tasks concern some assembly operations which are easy to repair. The rest of the tasks concern manufacturing operations which due to the size of the components are difficult and time consuming to repair. Furthermore, those repairable assembly operations have high defect rate and therefore repairing is crucial.

Each of the tasks has its own characteristics, such as the cost of the raw materials that are required for the completion of the task and the average processing time. Those tasks' characteristics are depicted in Table 20. Moving forward, the current industrial use case is composed of 15 main machines that are capable of performing the main tasks shown in Figure 33. The characteristics and capabilities of those machines are shown in Table 20 and Table 21. Those machines are configured as a flow-shop, where they are in series and with the layout as in Figure 33. Each machine can perform only one task, but the tasks that are assigned to each machine might be from a different order, and therefore they are considered as a different task by the machine. In the cases where there are repair tasks, the flow-shop configuration is changed to a hybrid layout between a flow-shop and an open-shop.

Depending on the ZDM strategy under investigation, extra machines are added to those 15 basic ones responsible for the inspection or repair of products. Those machines have variable characteristics that are controlled by the control parameters described in chapter 4.11. As described in chapter 4.11, the ZDM control parameters are relative to the products estimated – ideal manufacturing cost and time. Based on the data provided in Table 20 and Table 21, the product cost (PC) is €3760.57 and the manufacturing time is 267.04 minutes. Once again, those values represent the ideal scenario where there are no defects, no delays in the production, and no uncertainty. Those values will be the drivers for the conduction of the required experiments to develop the DT model and be able to properly design the production for ZDM.

Table 20: Task characteristics

Task Name	Task Reparable (0:No,1:Yes)	Raw Materials Cost (€)	Average Processing Time (Mins)	Task level
101	**0**	**77.6261**	**17.5335**	**1**
102	**0**	**75.4928**	**15.5738**	**2**
103	**0**	**55.5688**	**17.2173**	**2**
104	**1**	**77.0744**	**25.6327**	**3**
105	**0**	**40.8960**	**15.9950**	**4**
106	**1**	**58.0152**	**15.3506**	**4**
107	**0**	**44.4479**	**28.8266**	**5**
108	**0**	**30.3497**	**6.5480**	**5**
109	**0**	**98.9972**	**23.3134**	**5**
110	**1**	**82.7139**	**27.1681**	**4**
111	**0**	**81.2628**	**3.1026**	**5**
112	**0**	**92.9253**	**12.8309**	**5**
113	**0**	**206.5667**	**26.2677**	**3**
114	**0**	**218.9517**	**20.6340**	**3**
115	**0**	**158.8546**	**11.0477**	**2**

Table 21: Machine capabilities and characteristics

Machine Name	Operational Cost Of Machine €/Minute	Estimated Healthy Parts Rate	Mean Time Between Tuning Need (Mins)	Energy Consumption Kw/Min	Taks Capabilities
201	**7.5338**	**0.97**	**10500.00**	**0.1815**	**101**
202	**7.9520**	**0.95**	**13500.00**	**0.3490**	**102**
203	**9.2543**	**0.94**	**19500.00**	**0.3044**	**103**
204	**14.2082**	**0.96**	**30000.00**	**0.3256**	**104**
205	**6.4285**	**0.96**	**14642.85**	**0.2081**	**105**
206	**8.0408**	**0.97**	**11388.88**	**0.3866**	**106**
207	**8.7004**	**0.99**	**7884.61**	**0.3754**	**107**
208	**12.3080**	**0.99**	**5125.00**	**0.3455**	**108**
209	**7.7119**	**0.99**	**7875.00**	**0.2118**	**109**
210	**8.1136**	**0.94**	**10125.00**	**0.2415**	**110**
211	**8.4941**	**0.99**	**14625.00**	**0.4226**	**111**
212	**12.7684**	**0.99**	**22500.00**	**0.2499**	**112**
213	**6.3794**	**0.99**	**10357.14**	**0.2387**	**113**
214	**7.8707**	**0.98**	**8055.55**	**0.3195**	**114**
215	**9.3534**	**0.95**	**5576.92**	**0.3897**	**115**

5.1.1 Product characteristics analysis

Based on the data from the industrial case presented in Table 20 and, the following analysis was performed and the results are shown in Figure 34, Figure 35, and Figure 36. This analysis had the goal of decoding the key characteristics of the product under investigation. The key characteristics are related to the theoretical – ideal cost at each manufacturing stage (top left of Figure 34). This cost includes all the costs required for each task to be manufactured. Those include the raw materials and the machine operational cost. Therefore, it is the cost for each task separately at a glance. Moving forward, the top right illustration of Figure 34 shows the manufacturing time that each task required on average to be made.

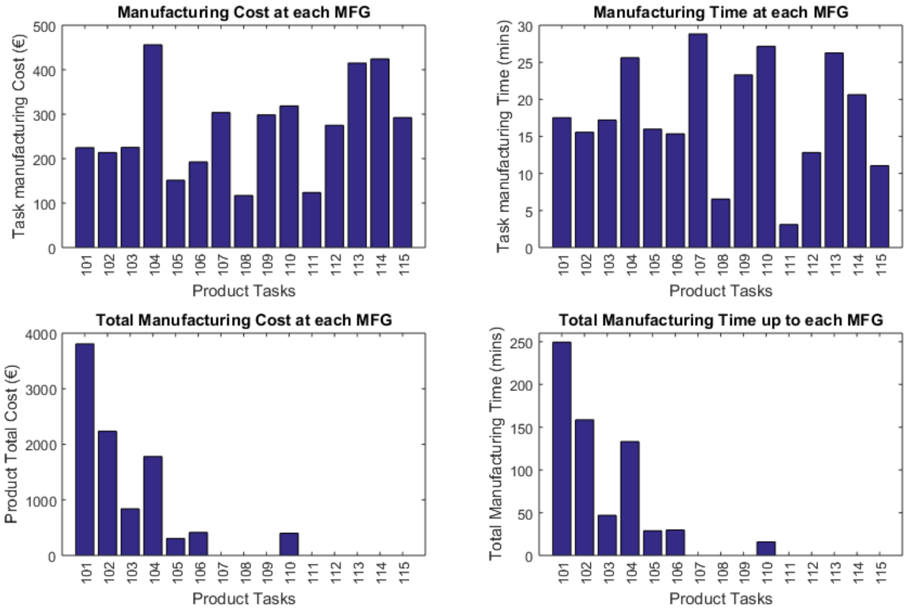


Figure 34: Product characteristics at each MFG

The bottom illustration of Figure 34 demonstrates the value of the product in terms of cost and time, taking into consideration the previous manufacturing steps required for the task under investigation to be possible. Figure 35 depicts the cost and time percentage out of the total. It is a result generated by dividing the results presented in top left and right Figure 34 with the total estimated PC and time accordingly. This was done to illustrate the requirements in terms of cost and time.

The last result from the product analysis is presented in Figure 36. It illustrates an indication factor that shows the magnitude of the potential losses from poor quality. The higher the value of the quality control index, the higher the losses due to poor quality. The quality control index is calculated as follows: both the cost and time percentages of each task (Figure 35) are multiplied by the estimated defect rate, and then those two values are summed together. All those preliminary product results will be used for reaching conclusions regarding the different ZDM strategies.

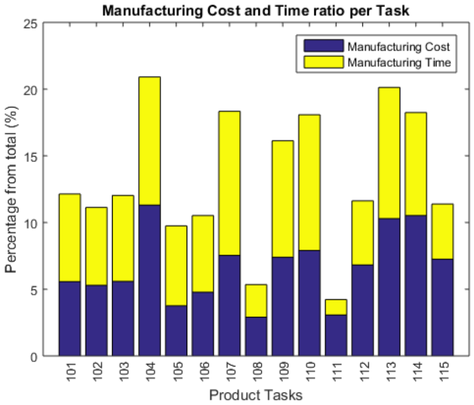


Figure 35: Cost and time ratio for each task from the total

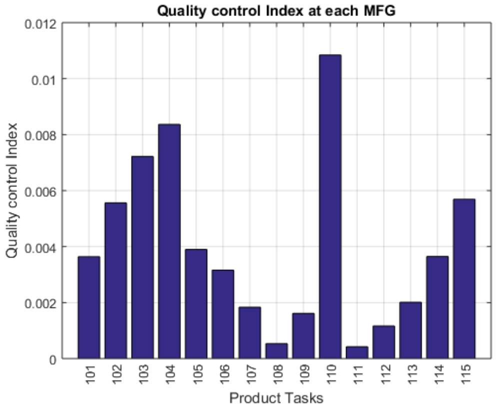


Figure 36: Quality control index for each task

Table 22: Product utility value

Task Name	Product Utility value	Product utility value with defect rate	Relative difference
101	0.7062	0.6850	-3.05%
102	0.6846	0.5877	-15.23%
103	0.5931	0.4745	-22.22%
104	0.2191	0.2553	15.26%
105	0.7635	0.6908	-10.00%
106	0.7283	0.7026	-3.59%
107	0.6295	0.7036	11.12%
108	0.7772	0.8218	5.58%
109	0.6424	0.7139	10.54%
110	0.5911	0.4729	-22.22%
111	0.7695	0.8156	5.82%
112	0.4439	0.5551	22.26%
113	0.4887	0.5910	18.95%
114	0.5025	0.5620	11.18%
115	0.6529	0.5623	-14.91%

Table 22 presents another analysis of the current product, namely the product utility value, which is a value based on the aggregation of the raw materials, machine operational cost, average processing time, and mean time between tuning. All four factors have the same weight

(0.25) for the weighted sum formula, which means that all four have equal importance to the final product utility value. Furthermore, in this utility value, the estimated defect rate of each MFG was incorporated and the (product utility value with defect rate) column of Table 22 was calculated. The last column of Table 22 shows the relative difference between the product utility value with and without the estimated defect rate. In this relative difference, the direction that the addition of the defect rate moves the product utility value is also shown. The relative differences that have a negative value mean the state for that task is deteriorating and potentially requires more attention. An exception is task 104, where there is an improvement of the product utility value by 15.26%, but by considering the absolute utility value in both cases (with and without defect rate), it is in the worst possible state, and potentially the use of any ZDM strategy is mandatory.

5.1.2 Demand profile

In this chapter, the demand profile used for the simulations is presented. As with the product and machine data, the demand profile also came from the same industry. This demand profile represents the orders per month (Figure 37) and the values are the average order size for a period of 5 years. Based on this demand profile, an average of 337 products are ordered per month with the minimum order size per month being 240 and the maximum being 400. More specifically, the current demand profile is composed of 56 orders for the period of 12 months. Table 23 shows the detailed demand profile with the order arrival time, due date, and quantity. Furthermore, the average order interval time is 6.49 days and the average time for completing an order is 5.54 days.

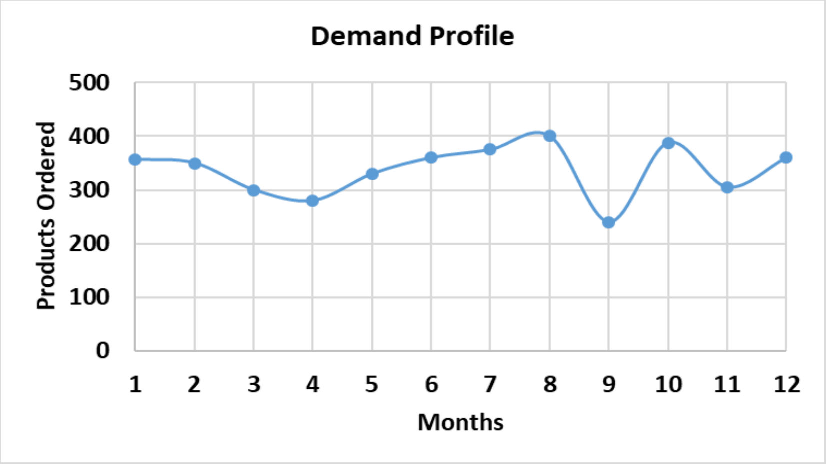


Figure 37: Overall demand profile for one year

Table 23: Demand profile details

Order Name	Order Arrival Time (days)	Order Due Date (days)	Order Quantity	Order Name	Order Arrival Time (days)	Order Due Date (days)	Order Quantity
9101	0.00	4.18	51	**9129**	158.17	163.22	72
9102	4.17	8.35	51	**9130**	164.25	169.31	72
9103	8.33	12.51	51	**9131**	170.33	175.39	72
9104	12.50	16.68	51	**9132**	176.42	181.47	72
9105	16.67	20.85	51	**9133**	182.50	191.74	125
9106	20.83	25.01	51	**9134**	192.64	201.88	125
9107	25.00	29.18	51	**9135**	202.78	212.01	125
9108	30.42	35.28	70	**9136**	212.92	221.96	125

9109	36.50	41.36	70	**9137**	220.52	229.56	100
9110	42.58	47.44	70	**9138**	228.13	237.17	100
9111	48.67	53.53	70	**9139**	235.73	244.77	100
9112	54.75	59.61	70	**9140**	243.33	250.24	60
9113	60.83	63.75	60	**9141**	250.94	257.84	60
9114	66.92	69.83	60	**9142**	258.54	265.44	60
9115	73.00	75.92	60	**9143**	266.15	273.05	60
9116	79.08	82.00	60	**9144**	273.75	280.26	97
9117	85.17	88.08	60	**9145**	281.35	287.87	97
9118	91.25	94.26	40	**9146**	288.96	295.47	97
9119	95.60	98.61	40	**9147**	296.56	303.08	97
9120	99.94	102.95	40	**9148**	304.17	308.15	61
9121	104.29	107.30	40	**9149**	310.25	314.24	61
9122	108.63	111.64	40	**9150**	316.33	320.32	61
9123	112.98	115.99	40	**9151**	322.42	326.40	61
9124	117.32	120.33	40	**9152**	328.50	332.49	61
9125	121.67	129.74	110	**9153**	334.58	340.42	90
9126	131.81	139.88	110	**9154**	342.19	348.02	90
9127	141.94	150.01	110	**9155**	349.79	355.63	90
9128	152.08	157.14	72	**9156**	357.40	363.23	90

5.1.3 Defect generation generic model

Based on the provided data regarding the occurrence of defects in the machines under investigation, Figure 38 was created for modeling the defect occurrence. This graph presents the generic defect occurrence mentioned in chapter 4.2. The created model shows that when the machine operation time reaches the mean time between tuning, which is calculated based on experience for each machine, the defects level reaches 80% of the acceptable defect rate for the specific machine. This occurs because of the stochasticity that was incorporated, and this randomness increases the defect level to reach 100% of the acceptable level. If the model below was used as it is, then the defect generation would be deterministic and therefore not be representative of the real word.

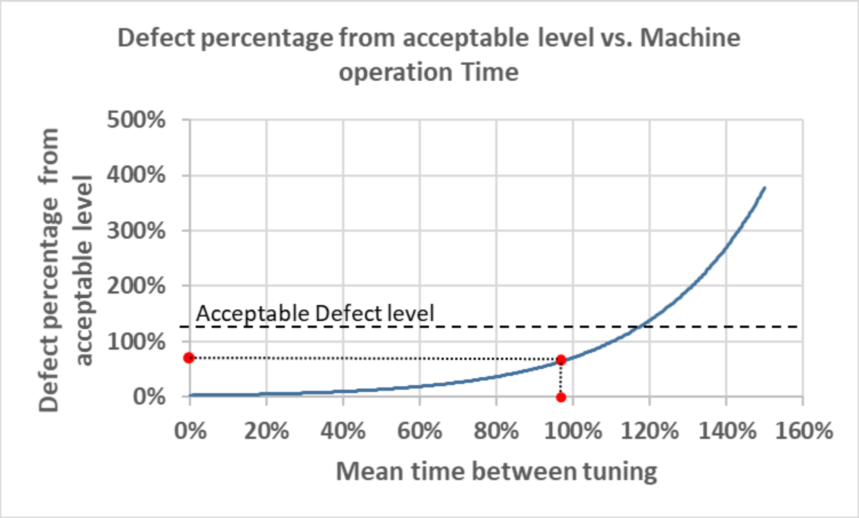


Figure 38: Defect generation generic curve

5.1.4 Machine improvement state

In equation (25), W_1 and W_2 represent the calibrating values of the formula, and in the current research, W_1=10 and W_2=3 in the case of detection – prevention and W_1=5 and W_2=2 in the case of prediction – prevention. The difference between the two ZDM strategies is that in detection – prevention, the machine improvement state is higher than in prediction – prevention. This is because when predicting a defect, there are less data for the root of the problem, simply because the problem has not yet occurred. Therefore, the machine improvement state can potentially be higher in detection – prevention because it has the time and date for better prevention actions.

5.2 Digital Twin Model Creation

In the current chapter the data from the industrial use case presented in chapter 5.1 are used for the creation of the DT model. The data will be the input to the developed scheduling tool presented in chapter 4. The experiments will be performed as described by the L_{25} orthogonal array (chapter 4.12). It is important to mention again that each DT model will describe the results of the scheduling tool for the current industrial case with ZDM applied to only one machine at each time. Therefore, for each of the 15 machines that the current scenario has three DTs will be created, one for each ZDM pair of strategy. This will lead to a set of 75 DT models (3xZDM*15xMFG). The overall procedure that was followed for the experiments and the elaboration of the results are summarized in Figure 39. Because the DT method proposed in chapter 4.12 is not in the literature the results acquired from the DT models must be validated. The plan to validate the results and measure the accuracy of the created DT models is to create sets of random ZDM parameters for each ZDM pair strategy and simulate those sets using the scheduling tool and, also, using the DT. This will lead to the actual and predicted values accordingly and by calculating the relative difference the error of each DT can be calculated.

This being said, there are two sets of experiments that should be simulated with the scheduling tool, the set that is generated by the L_{25} orthogonal array and the set with the random ZDM parameters. Those experiments must be simulated before any other calculation. This is due to the KPI aggregation method described in chapter 4.9. The aggregation method requires the global minimum and maximum KPI values in order to perform the normalization. Therefore, the minimum and maximum KPI values are common for all the utility values calculations illustrated in Figure 39. Once the utility values are calculated, the DT models can be created using the methodology described in chapter 4.12 and the accuracy of the DT models can be calculated. If the accuracy is at acceptable range, higher than 95%, then the DT will be used for answering of Research Question 3. Otherwise, the DT method should be revised alongside the ZDM parameters.

A crucial step for the conduction of the two aforementioned experiment sets is to define the factor levels, meaning to define the values that each factor should have at each level. The definition of the factor values will be performed in the chapter 5.2.1. Once this step is done the minimum and maximum values from each factor will be used as the limits for generating the random values, which must be within the min-max range defined.

Besides the creation of the DT models, the results from the experiments performed based on the L_{25} orthogonal array will be used also for extracting more insights regarding the current industrial used case and the implementation of the ZDM to the production. Those results will be presented in chapters 5.3.1 and 5.3.2.

At this point, it should be mentioned that the computer system used to run the experiments so that the reader has a reference for the computational times that will be presented at the results chapter (5.3). The specifications of the computer used were: CPU i7-8700K, 6 cores @3.70GHz, 32GB of DDR4 RAM @ 3400MHz, running Windows 10 professional and using Matlab 2019.

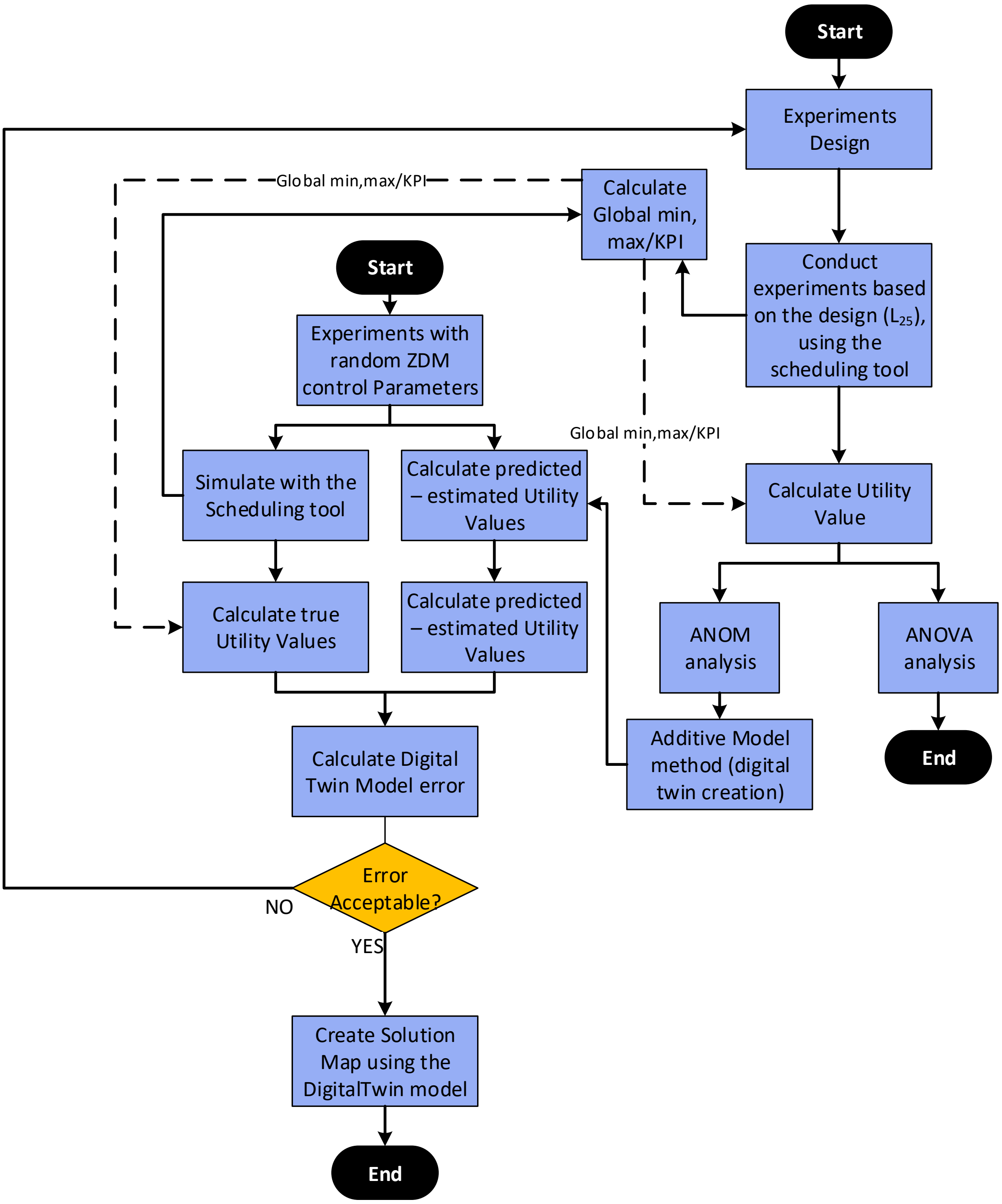

Start
Experiments Design
Global min,max/KPI
Calculate Global min, max/KPI
Start
Conduct experiments based on the design (L25), using the scheduling tool
Experiments with random ZDM control Parameters
Global min,max/KPI
Simulate with the Scheduling tool
Calculate predicted – estimated Utility Values
Calculate Utility Value
Calculate true Utility Values
Calculate predicted – estimated Utility Values
ANOM analysis
ANOVA analysis
Additive Model method (digital twin creation)
End
Calculate Digital Twin Model error
Error Acceptable?
NO
YES
Create Solution Map using the DigitalTwin model
End

Figure 39: Experiments and results of the overall procedure for each MFG and each ZDM

5.2.1 ZDM control parameters values definition

To create a DT model with a high level of accuracy, each parameter needs to take more than two values during the simulations for the creation of the DT model. This is required in order to avoid the DT model to be a linear model, but described by quadratic equations. In light of this, the L_{25} orthogonal array was selected. The L_{25} orthogonal array is presented in Annex 4, chapter A. This orthogonal array fits the current problem exactly because it can host up to six parameters and each parameter is a five-level parameter. Furthermore, L_{25} orthogonal array sets the number of experiments to 25, which for the selected number of parameters and levels is the minimum number of experiments for conducting the required analysis [196]. Table 24, Table 25, and Table 26 present the values for each level for each parameter. Those values were set, based on the results of a survey conducted to various manufacturing companies. These values represent the reality and include the extreme values both in upper and lower limits. The values in between were assigned in a way to capture the effect of each factor. Using the try and error method, many iterations for adjusting the factor levels were performed to end up to the values in the tables below. During those explorative simulations, the limits were defined from where there was no significant alteration of the results. The values of the parameters with “R” on the side represent the ratio that was defined by equation (52). This ratio represents how much from the nominal value the ZDM strategy is performed. This applies to cost and time parameters, with €3760.57 and 267.04 minutes used as nominal values, as defined in chapter 5.1. Using the ratios defined in the tables and with equation (52) the absolute values were calculated and fed into the scheduling tool.

Table 24: Detection – Repair factor levels

factors/levels	1	2	3	4	5
Inspection Cost R	0.01	0.1	0.25	0.35	0.5
Inspection Time R	0.01	0.04	0.11	0.18	0.25
Detection Accuracy %	0.7	0.85	0.9	0.93	0.99
Repairing Cost R	0.05	0.3	0.8	1.3	2.5
Repairing Time R	0.05	0.4	0.8	1.3	2
reparabilty %	0.1	0.45	0.55	0.75	0.95

Table 25: Detection – Prevention factor levels

factors/levels	1	2	3	4	5
Inspection Cost R	0.01	0.1	0.25	0.35	0.5
Inspection Time R	0.01	0.04	0.11	0.18	0.25
Detection Accuracy %	0.7	0.85	0.9	0.93	0.99
Prevention Cost R	0.05	0.4	1.5	3	5
Prevention Time R	0.3	0.8	3	5	7
Prevention success rate %	0.6	0.7	0.8	0.9	0.95

Table 26: Prediction – Prevention factor levels

factors/levels	1	2	3	4	5
Prediction Horizon (mins)	0.0187	0.0749	0.2247	0.3745	1.1234
Prediction Accuracy %	0.7	0.85	0.9	0.93	0.99
Prevention Reaction Time R (mins)	0.1123	0.2247	0.5617	1.3107	2.2468
Prevention Cost R	0.05	0.4	1.5	3	5
Prevention Time R	0.3	0.8	3	5	7
Prevention success rate %	0.6	0.7	0.8	0.9	0.95

5.2.2 DT models creation and validation

This chapter presents the creation of the 75 DT models, three for each MFG (one for each ZDM pair strategy). The simulations performed based on the L_{25} orthogonal array and the ZDM parameters values defined in chapter 5.2.1. In total 1125 simulation runs were performed using the developed scheduling tool, using the computer system presented earlier in this chapter. Each simulation required between 35-78 minutes to run and produce the results. The utility values were calculated for the KPIs defined in chapter 4.10 and for the current experiments all the KPIs had equal weight factors. Using the method described in chapter 4.12 the DT models were created, to show the behavior of the coefficients of each DT model the resulted equations from equation (62) were plotted and the detailed plots can be found in Annex 2, because they are lengthy. Figure 40 is one out of the 15 plots illustrating the DT coefficients for all three ZDM pair strategies, in order to show to the reader, the form of those coefficients and justify the reason for selecting a piecewise quadratic interpolation. On those graphs on the x-axis are the values of the ZDM control parameters and on the y-axis is the contribution of each factor to the predicted utility value. The results from the 1125 experiments can be found in Annex 1, where the utility values for each experiment set are presented.

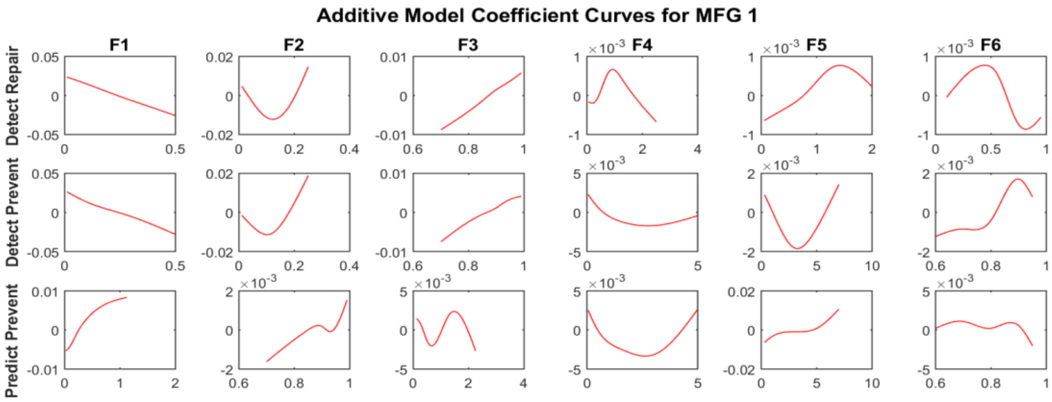


Figure 40: Example DT models MFG1 plots, equation (62)

5.2.2.1 Utility value digital twin model accuracy

Before using the developed DT model, the accuracy of the models had to be tested and validated, otherwise the results would not be trustworthy. For the calculation of the accuracy for the developed DT model, the following procedure was followed. For each manufacturing stage, 50 random scenarios were generated and simulated for each ZDM strategy. All the random scenarios had all the same parameters except the six control parameters under investigation. The values of the six parameters per ZDM strategy took completely random values within the designated range in which the DT models were created. In total, 2250 extra random simulations were performed to test the accuracy of the produced DT model. Using the results from those experiments, all the developed models were validated by comparing the estimated value with the actual value from the scheduling tool.

Using the actual results of the KPIs, taken from simulating the random set of experiments with the scheduling tool, the actual utility value was produced for each of the scenarios. At the same time, the exact same factor values for each of the random scenarios were fed into the developed DT model and the estimated utility value was produced. The next step was to calculate the relative difference between the theoretical and actual utility value. The average relative differences for each MFG and each ZDM are presented in Table 27. The global average relative difference for all the MFGs and ZDM strategies was 1.066% of the deviation from the actual utility value. In other words, the developed DT model regarding the utility value had an accuracy of 98.934% on average.

Moving in greater depth to the results regarding the error of the DT model, the maximum error noticed was 4.3108% at MFG3 and the predict – prevent ZDM strategy. Furthermore, the lowest error noticed was 0.2493% at MFG 9 and the predict – prevent ZDM strategy. Another observation was that the average error for each ZDM strategy was different to the others. The detect – repair strategy had the lowest average error with 0.5639% of error among the three ZDM strategies. This was followed by the detect – prevent strategy with 1.0810% and predict – prevent with 1.5533%.

Table 27: Utility value prediction model error

	Detect – Repair	Detect – Prevent	Predict – Prevent
MFG1	0.4940 %	0.7841 %	1.6245 %
MFG2	0.5481 %	1.4620 %	2.3254 %
MFG3	0.7320 %	2.0638 %	4.3108 %
MFG4	1.7189 %	3.5686 %	1.9362 %
MFG5	0.5590 %	0.9489 %	3.8798 %
MFG6	0.5675 %	0.9334 %	1.7139 %
MFG7	0.2978 %	0.4543 %	0.3299 %
MFG8	0.3217 %	0.4893 %	0.3746 %
MFG9	0.3883 %	0.3435 %	0.2493 %
MFG10	0.7282 %	2.6136 %	2.0202 %
MFG11	0.3731 %	0.3467 %	1.0939 %
MFG12	0.4077 %	0.6028 %	0.4782 %
MFG13	0.2766 %	0.4004 %	0.4118 %
MFG14	0.3523 %	0.4015 %	0.5683 %
MFG15	0.6931 %	0.8015 %	1.9828 %
Average	**0.5639%**	**1.0810%**	**1.5533%**

5.2.2.2 KPI prediction model error

To test the capabilities of the DT methodology the DT models for each individual KPI were created. The results were mixed but not discouraging. On average, all the KPIs DT models had an error of 6.223%. More specifically, 12 of the 18 KPIs had an error less than 5%, two had an error between 5% and 10%, and four had an error between 10% and 20%. The factors that had the highest error were weighted tardiness, detection prevention cost, prediction prevention cost, and rescheduling cost. Taking into account how the KPIs were defined, the cost KPIs with high levels of error were not much of a problem since all the cost KPIs are used for the calculation of the final unit cost, which the corresponding DT model achieved an error of 2.34%. Regarding the weighted tardiness, the model had the lowest accuracy of 81.54%.

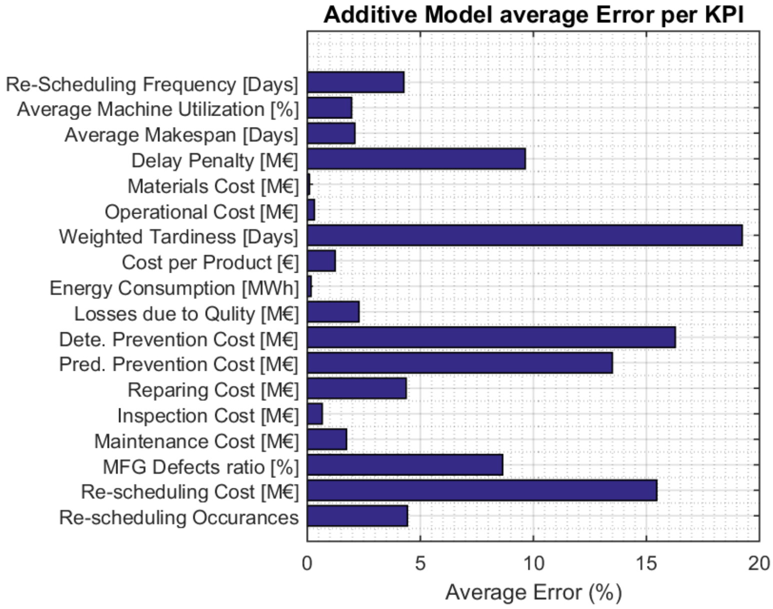


Figure 41: Digital twin KPIs error

5.3 Results

This chapter presents the results acquired from the experiments conducted for the creation of the DT models. Using those results the ANOM and ANOVA analysis can be performed and their results are presented in their corresponding chapters 5.3.1 and 5.3.2 respectively. Furthermore, in chapter 5.3.3 the ZDM performance maps are going to be presented, illustrating the performance of each ZDM pair strategy with different ZDM control parameters values. In those experiments all three ZDM strategies were simulated for each of the MFGs. This is mentioned because someone could ask, “Why simulate the detect – repair strategy for MFGs for which repair is not possible?” as indicated in Table 20. Only tasks 104, 106, and 110 have the possibility of repair. The detect – repair strategy can work in two different ways. In cases where repair is possible, the developed DSS (explained in chapter 4.4) decides whether to repair it or discard it and produce a completely new product. On the other hand, in the case that repair is not possible, the DSS algorithm (chapter 4.4) automatically discards the part and produces a completely new one. Therefore, there is meaning in simulating such MFGs for the detect – repair strategy.

To analyze the results from the experiments were used the KPIs defined in chapter 4.10, but those KPIs are studied all together using the aggregation method presented in chapter 4.9. This means that all the observations and conclusions are made based on the utility value and there is no analysis of each single KPI. This was decided because some of the KPIs are conflicting and in general the relations between each other as a whole are not known. Analyzing each KPI individually will produce wrong conclusions. The purpose of the KPIs aggregation method is to solve this problem of conflicting KPIs and incorporate the impact of each KPI to the final solution. The current problem is a multi-dimensional problem and therefore, it should be studied as multi-dimensional. The aggregation method with the KPIs weight factors can change the influence of each KPI to the final solution. In the current study all the KPIs had the same weight factors.

5.3.1 Utility value ANOM diagrams

Using the results of the simulations presented in the Annex, the utility value ANOM diagrams can be calculated using the L_{25} orthogonal array presented in Annex 4 A and the ANOM analysis [196], equation (29). The ANOM diagrams demonstrate the effect of each of the defined parameters on the final quality of the solution. Figure 42 to Figure 56 illustrate the ANOM diagrams for each MFG as well as for each ZDM. Each diagram is an outcome of a separate simulation set, but as explained in Figure 39, all utility values were calculated using the global minimum and maximum KPI values for the results to be comparable. Furthermore, all the ANOM diagrams have the same scale on both the x-axis and y-axis for easier comparison between the MFGs and ZDM strategies. Table 28 presents the calculated average effect that each factor at each ZDM has on the final quality of the solution. Another interesting observation is that each set of graphs for each MFG is different, demonstrating the unique characteristics of each MFG. Furthermore, a common observed behavior was that the higher the defect rate, the higher the effect of the factors on the quality of the solution. Based on those preliminary results, someone can have an understanding of which ZDM strategy is better suited for each MFG. This is analyzed in depth later in chapter 5.3.3.

Viewing the overall results, it is obvious that certain factors such as the inspection cost or inspection time had the most significant impact on the solution quality with average effects of 8.91% and 4.56% in detect – repair and 9.93% and 3.28% in detection – prevention accordingly. On average, it seems that the inspection cost had more effect in the detection – prevention strategy, whereas the inspection cost had more impact in the detection – repair strategy. Regarding the inspection cost in all MFGs, the increase of the inspection cost negatively affected the quality of the solution, something that was not the case with the inspection time.

Moreover, the prediction – prevention ZDM strategy seemed to be influenced less by the selected factors compared with the other two ZDM strategies. Furthermore, the detection – repair and detection – prevention strategies had similar results in the common parameters in most cases but not identical. The detection accuracy also had a common positive effect on the quality of the solution as the levels increased in all MFGs except MFG4 for the detection – repair strategy, where the optimal level for detection accuracy was level 4 and the quality subsequently dropped.

Table 28: Average (Avg.) effect of factors for all MFGs (ANOM)

Detect - Repair		**Detect - Prevent**		**Predict – Prevent**	
Factors	Avg. factor effect	Factors	Avg. factor effect	Factors	Avg. factor effect
Inspection Cost	8.91%	**Inspection Cost**	9.93%	**Prediction Horizon**	1.70%
Inspection Time	4.56%	**Inspection Time**	3.28%	**Prediction Accuracy**	2.04%
Detection accuracy	2.91%	**Detection accuracy**	2.05%	**Prediction reaction time**	2.05%
Repair Cost	1.08%	**Prevention Cost**	1.28%	**Prevention Cost**	1.65%
Repair Time	0.91%	**Prevention Time**	1.57%	**Prevention Time**	2.43%
Reparability	0.78%	**Prevention Success rate**	1.01%	**Prevention Success rate**	2.11%

Additionally, as was expected, the parameters regarding the repair of a defective part had almost no effect in the cases where the part was not repairable and a higher impact on the MFGs (104, 106, and 110) for which repair is possible. More specifically, the more the repair cost and time increased, the more the quality of the solution decreased except for MFG 104 again. In general, MFG 4 was the most complicated MFG based on the results with the highest variation in quality of the solution from all the factors, compared with the rest of the MFGs. Furthermore, in MFG 104, the ANOM results had completely different forms compared with all other MFGs.

In the prediction – prevention strategy, some common behaviors were also observed in all of the MFGs. As the prediction horizon increased, the quality of the solution increased, and as indicated before, the higher the defect rate, the more the solution quality improved as the prediction horizon value increased. Prediction accuracy exhibited different behavior; it had a bell shape in most of the MFGs expect MFG15. This means that the optimal solution qualities were observed in the middle levels, and at levels 1 and 5 the solution quality had the lowest values. Prediction reaction time had no uniform behavior across the different MFGs. In some MFGs the effect was constant, whereas in some others it had a bell or “U” shape. The prevention cost and time had almost uniform behavior across all the MFGs (except MFG15), namely that the higher they were, the higher the quality of the solution. In MFG15, the optimal solution quality was produced by level 1 for both prevention cost and time. The prevention success rate also had non-uniform behavior.

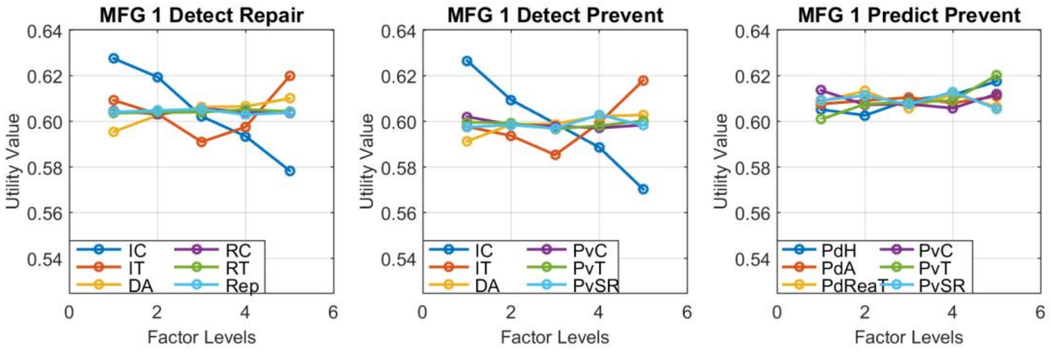


Figure 42: MFG 1 ANOM diagrams for each ZDM (defect rate 3%)

Table 29: ANOM minimum, maximum, & relative difference MFG 1

	Detect – Repair			Detect – Prevent			Predict – Prevent		
Factors	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**
F1	0.6159	0.6738	8.99%	0.6073	0.6718	10.07%	0.6471	0.6628	2.38%
F2	0.6314	0.6641	5.04%	0.6246	0.6623	5.86%	0.6521	0.6554	0.51%
F3	0.6371	0.6523	2.34%	0.6328	0.6444	1.81%	0.6501	0.6581	1.22%
F4	0.6453	0.6472	0.29%	0.6386	0.6436	0.78%	0.6497	0.6581	1.29%
F5	0.6455	0.6468	0.20%	0.6382	0.6417	0.54%	0.6449	0.6654	3.11%
F6	0.6449	0.6473	0.37%	0.6388	0.6446	0.88%	0.6492	0.6579	1.32%

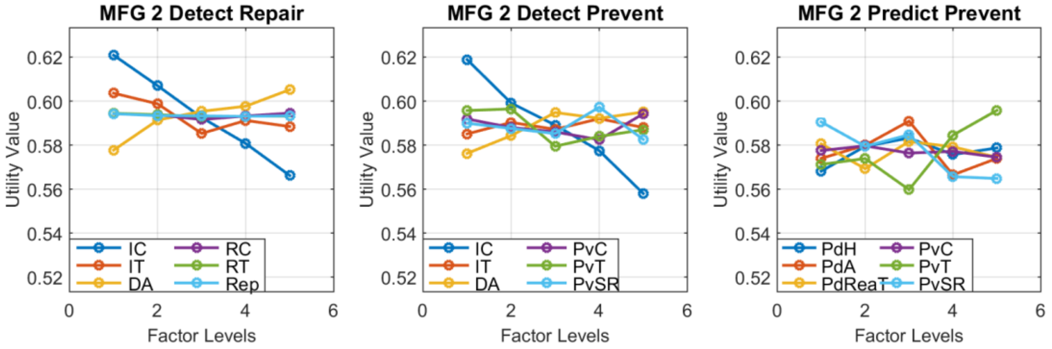


Figure 43: MFG 2 ANOM diagrams for each ZDM (defect rate 5%)

Table 30: ANOM minimum, maximum, & relative difference MFG 2

	Detect – Repair			Detect – Prevent			Predict – Prevent		
Factors	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**
F1	0.6000	0.6620	9.83%	0.5909	0.6598	11.02%	0.6077	0.6230	2.49%
F2	0.6214	0.6431	3.43%	0.6218	0.6285	1.07%	0.6060	0.6316	4.14%
F3	0.6148	0.6436	4.58%	0.6129	0.6320	3.08%	0.6086	0.6221	2.19%
F4	0.6302	0.6319	0.27%	0.6183	0.6316	2.12%	0.6131	0.6214	1.34%
F5	0.6299	0.6326	0.43%	0.6161	0.6336	2.79%	0.5993	0.6365	6.02%
F6	0.6306	0.6318	0.20%	0.6190	0.6350	2.55%	0.6032	0.6308	4.48%

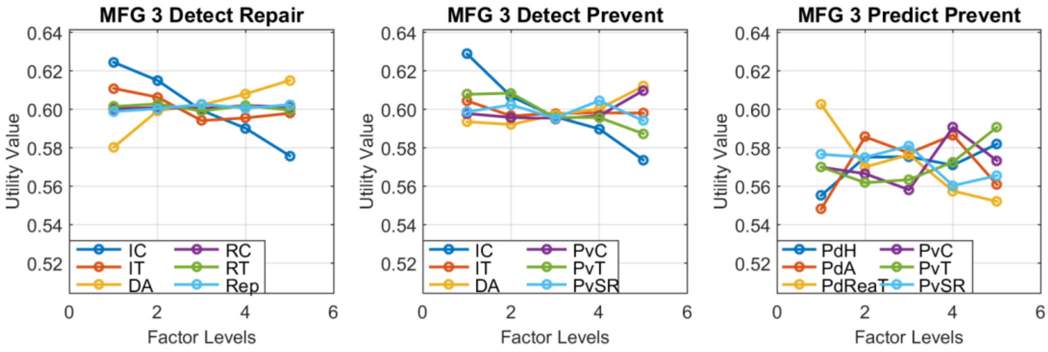


Figure 44: MFG 3 ANOM diagrams for each ZDM (defect rate 6%)

Table 31: ANOM minimum, maximum, & relative difference MFG 3

	Detect – Repair			Detect – Prevent			Predict – Prevent		
Factors	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**
F1	0.6083	0.6639	8.73%	0.6057	0.6684	9.84%	0.5930	0.6204	4.51%
F2	0.6291	0.6481	2.97%	0.6321	0.6404	1.30%	0.5852	0.6261	6.75%
F3	0.6146	0.6525	5.97%	0.6274	0.6480	3.24%	0.5888	0.6420	8.65%
F4	0.6362	0.6377	0.24%	0.6301	0.6451	2.37%	0.5958	0.6291	5.44%
F5	0.6356	0.6394	0.61%	0.6205	0.6452	3.90%	0.6000	0.6291	4.73%
F6	0.6350	0.6389	0.61%	0.6284	0.6409	1.98%	0.5981	0.6201	3.62%

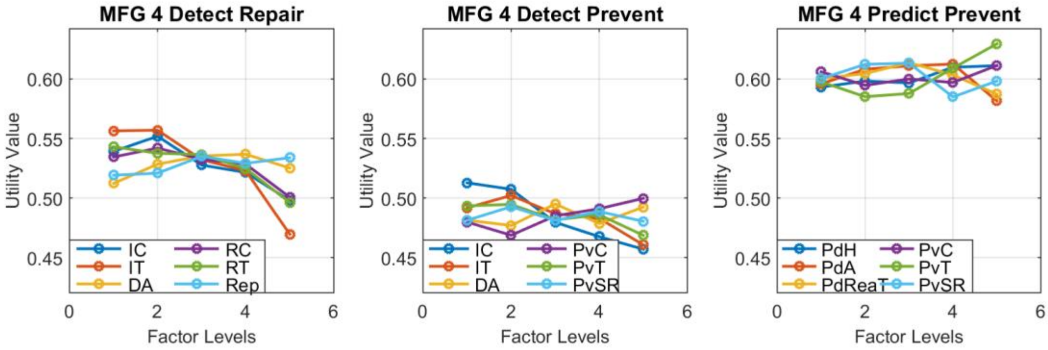


Figure 45: MFG 4 ANOM diagrams for each ZDM (defect rate 4%)

Table 32: ANOM minimum, maximum, & relative difference MFG 4

	Detect – Repair			Detect – Prevent			Predict – Prevent		
Factors	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**
F1	0.5347	0.5910	10.01%	0.4856	0.5465	11.82%	0.6351	0.6542	2.96%
F2	0.5198	0.5901	12.68%	0.4989	0.5291	5.88%	0.6231	0.6562	5.16%
F3	0.5513	0.5764	4.44%	0.5095	0.5254	3.07%	0.6300	0.6569	4.18%
F4	0.5367	0.5822	8.13%	0.5025	0.5306	5.44%	0.6372	0.6544	2.67%
F5	0.5437	0.5778	6.08%	0.5057	0.5245	3.66%	0.6270	0.6740	7.23%
F6	0.5531	0.5764	4.12%	0.5108	0.5224	2.24%	0.6261	0.6571	4.83%

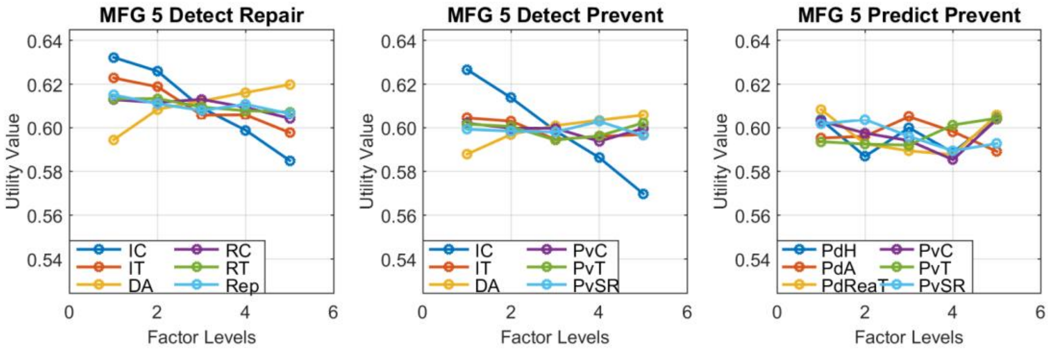


Figure 46: MFG 5 ANOM diagrams for each ZDM (defect rate 4%)

Table 33: ANOM minimum, maximum, & relative difference MFG 5

	Detect – Repair			Detect – Prevent			Predict – Prevent		
Factors	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**
F1	0.6213	0.6754	8.34%	0.6045	0.6685	10.06%	0.6294	0.6489	3.04%
F2	0.6365	0.6648	4.36%	0.6327	0.6439	1.75%	0.6319	0.6486	2.61%
F3	0.6346	0.6596	3.86%	0.6255	0.6443	2.96%	0.6310	0.6521	3.28%
F4	0.6441	0.6538	1.49%	0.6331	0.6409	1.21%	0.6282	0.6477	3.06%
F5	0.6472	0.6534	0.94%	0.6322	0.6423	1.59%	0.6342	0.6475	2.09%
F6	0.6459	0.6556	1.48%	0.6358	0.6428	1.10%	0.6327	0.6477	2.34%

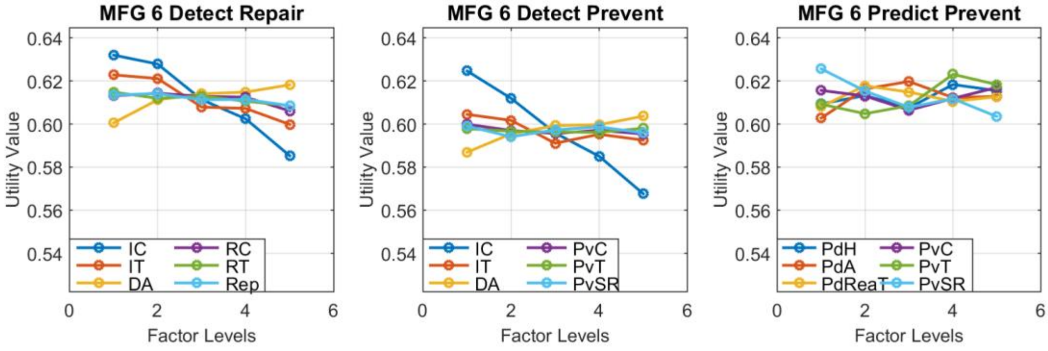


Figure 47: MFG 6 ANOM diagrams for each ZDM (defect rate 3%)

Table 34: ANOM minimum, maximum, & relative difference MFG 6

	Detect – Repair			Detect – Prevent			Predict – Prevent		
Factors	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**
F1	0.6235	0.6770	8.23%	0.6046	0.6679	9.94%	0.6511	0.6638	1.93%
F2	0.6415	0.6655	3.66%	0.6290	0.6454	2.57%	0.6480	0.6646	2.54%
F3	0.6420	0.6594	2.67%	0.6270	0.6440	2.69%	0.6516	0.6630	1.74%
F4	0.6478	0.6571	1.41%	0.6351	0.6400	0.77%	0.6508	0.6623	1.76%
F5	0.6495	0.6577	1.26%	0.6364	0.6387	0.37%	0.6493	0.6684	2.89%
F6	0.6506	0.6564	0.89%	0.6351	0.6390	0.62%	0.6485	0.6717	3.52%

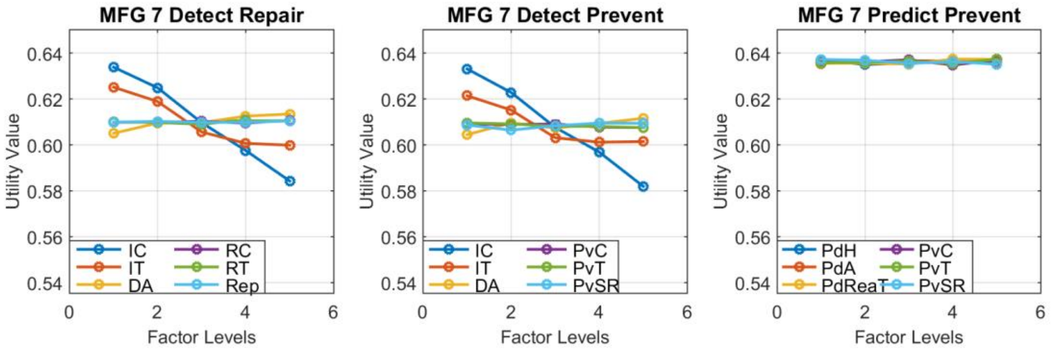


Figure 48: MFG 7 ANOM diagrams for each ZDM (defect rate 1%)

Table 35: ANOM minimum, maximum, & relative difference MFG 7

	Detect – Repair			Detect – Prevent			Predict – Prevent		
Factors	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**
F1	0.6259	0.6823	8.62%	0.6220	0.6817	9.16%	0.6847	0.6876	0.42%
F2	0.6432	0.6717	4.34%	0.6442	0.6670	3.48%	0.6852	0.6872	0.29%
F3	0.6498	0.6588	1.37%	0.6480	0.6559	1.22%	0.6847	0.6876	0.43%
F4	0.6548	0.6558	0.15%	0.6519	0.6535	0.24%	0.6846	0.6871	0.35%
F5	0.6545	0.6559	0.21%	0.6508	0.6541	0.51%	0.6850	0.6869	0.28%
F6	0.6546	0.6566	0.30%	0.6514	0.6537	0.35%	0.6848	0.6874	0.37%

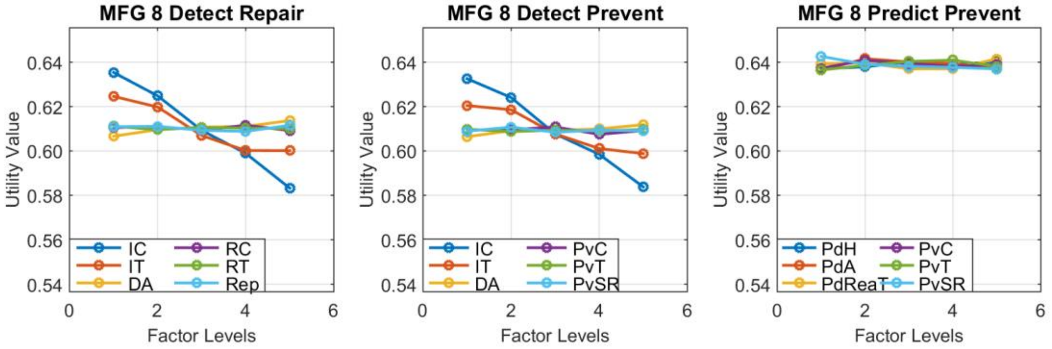


Figure 49: MFG 8 ANOM diagrams for each ZDM (defect rate 1%)

Table 36: ANOM minimum, maximum, & relative difference MFG 8

	Detect – Repair			Detect – Prevent			Predict – Prevent		
Factors	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**
F1	0.6252	0.6839	8.96%	0.6242	0.6816	8.80%	0.6873	0.6900	0.39%
F2	0.6452	0.6714	3.98%	0.6432	0.6668	3.61%	0.6858	0.6917	0.86%
F3	0.6519	0.6585	1.02%	0.6506	0.6570	0.98%	0.6861	0.6914	0.76%
F4	0.6553	0.6563	0.15%	0.6527	0.6554	0.41%	0.6874	0.6913	0.56%
F5	0.6556	0.6562	0.10%	0.6537	0.6547	0.16%	0.6858	0.6910	0.76%
F6	0.6548	0.6568	0.30%	0.6528	0.6550	0.34%	0.6873	0.6921	0.69%

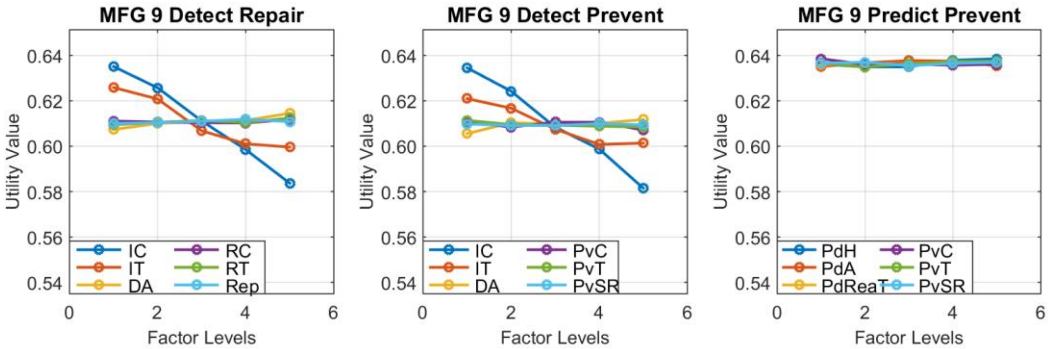


Figure 50: MFG 9 ANOM diagrams for each ZDM (defect rate 1%)

Table 37: ANOM minimum, maximum, & relative difference MFG 9

	Detect – Repair			Detect – Prevent			Predict – Prevent		
Factors	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**
F1	0.6255	0.6840	8.94%	0.6230	0.6826	9.13%	0.6854	0.6884	0.44%
F2	0.6446	0.6719	4.15%	0.6454	0.6678	3.42%	0.6847	0.6880	0.48%
F3	0.6530	0.6599	1.05%	0.6510	0.6568	0.90%	0.6857	0.6877	0.30%
F4	0.6556	0.6569	0.21%	0.6526	0.6553	0.42%	0.6856	0.6882	0.39%
F5	0.6553	0.6579	0.39%	0.6533	0.6561	0.43%	0.6862	0.6873	0.15%
F6	0.6553	0.6570	0.26%	0.6542	0.6546	0.07%	0.6850	0.6875	0.37%

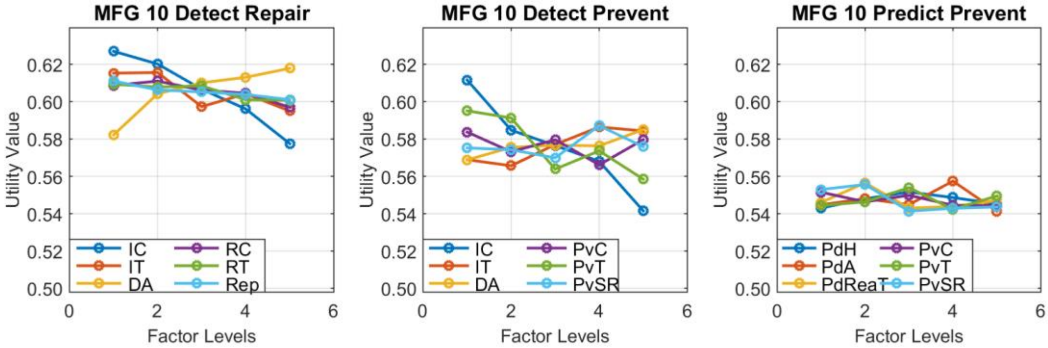


Figure 51: MFG 10 ANOM diagrams for each ZDM (defect rate 6%)

Table 38: ANOM minimum, maximum, & relative difference MFG 10

	Detect – Repair			Detect – Prevent			Predict – Prevent		
Factors	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**
F1	0.6100	0.6655	8.70%	0.5703	0.6477	12.71%	0.5786	0.5877	1.56%
F2	0.6310	0.6519	3.26%	0.5974	0.6209	3.87%	0.5759	0.5932	2.95%
F3	0.6172	0.6535	5.72%	0.6015	0.6180	2.71%	0.5782	0.5927	2.48%
F4	0.6318	0.6464	2.28%	0.5982	0.6171	3.10%	0.5790	0.5875	1.45%
F5	0.6365	0.6460	1.48%	0.5917	0.6287	6.07%	0.5774	0.5904	2.23%
F6	0.6357	0.6472	1.80%	0.6026	0.6204	2.91%	0.5773	0.5923	2.56%

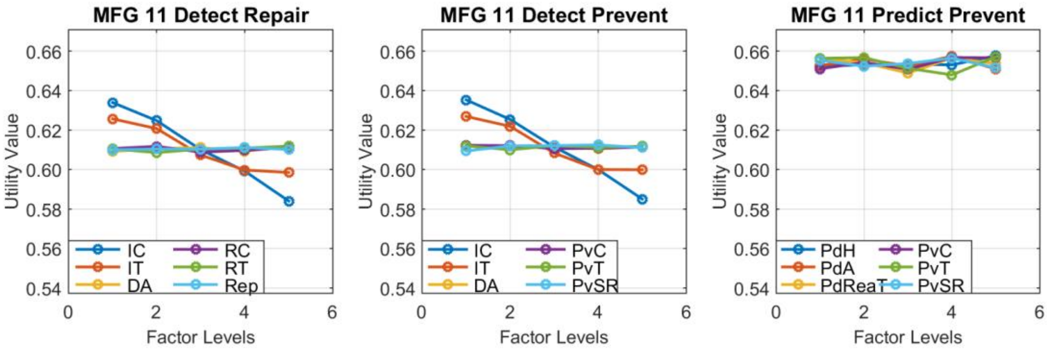


Figure 52: MFG 11 ANOM diagrams for each ZDM (defect rate 1%)

Table 39: ANOM minimum, maximum, & relative difference MFG 11

	Detect – Repair			Detect – Prevent			Predict – Prevent		
Factors	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**
F1	0.6262	0.6839	8.81%	0.6269	0.6852	8.88%	0.7051	0.7106	0.78%
F2	0.6440	0.6738	4.52%	0.6448	0.6746	4.52%	0.7039	0.7102	0.89%
F3	0.6553	0.6583	0.46%	0.6566	0.6584	0.27%	0.7014	0.7091	1.10%
F4	0.6554	0.6577	0.36%	0.6566	0.6584	0.27%	0.7027	0.7097	0.99%
F5	0.6552	0.6579	0.40%	0.6562	0.6585	0.36%	0.6996	0.7102	1.51%
F6	0.6560	0.6574	0.20%	0.6561	0.6587	0.40%	0.7035	0.7090	0.78%

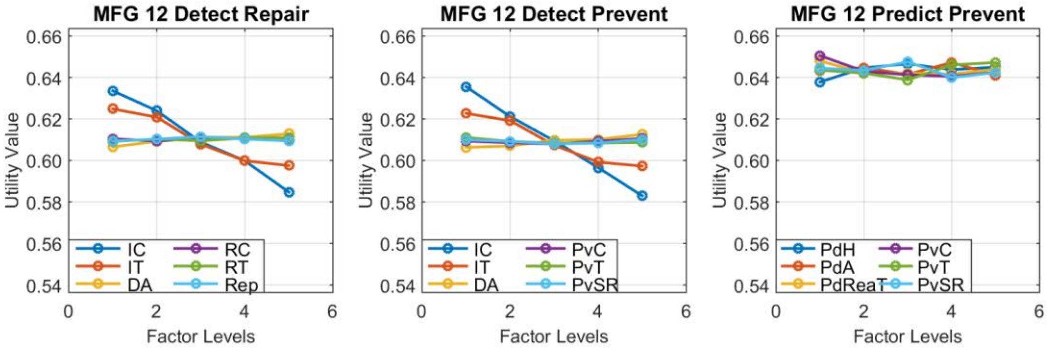


Figure 53: MFG 12 ANOM diagrams for each ZDM (defect rate 1%)

Table 40: ANOM minimum, maximum, & relative difference MFG 12

	Detect – Repair			Detect – Prevent			Predict – Prevent		
Factors	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**
F1	0.6256	0.6825	8.71%	0.6243	0.6838	9.10%	0.6878	0.6969	1.31%
F2	0.6423	0.6719	4.51%	0.6420	0.6691	4.13%	0.6916	0.6978	0.89%
F3	0.6515	0.6581	1.00%	0.6510	0.6580	1.07%	0.6913	0.6985	1.03%
F4	0.6549	0.6557	0.13%	0.6537	0.6555	0.28%	0.6914	0.7008	1.36%
F5	0.6549	0.6558	0.13%	0.6531	0.6553	0.34%	0.6888	0.6975	1.24%
F6	0.6548	0.6560	0.18%	0.6537	0.6547	0.15%	0.6905	0.6980	1.08%

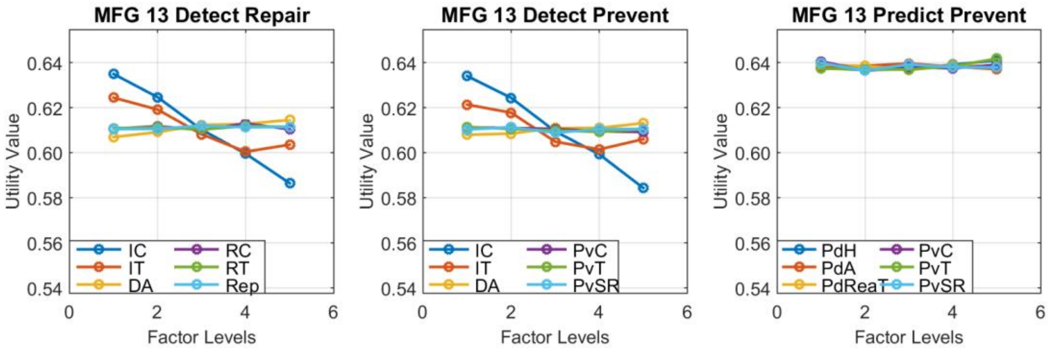


Figure 54: MFG 13 ANOM diagrams for each ZDM (defect rate 1%)

Table 41: ANOM minimum, maximum, & relative difference MFG 13

	Detect – Repair			Detect – Prevent			Predict – Prevent		
Factors	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**
F1	0.6269	0.6837	8.66%	0.6251	0.6823	8.74%	0.6866	0.6906	0.59%
F2	0.6448	0.6710	3.98%	0.6470	0.6677	3.14%	0.6862	0.6903	0.59%
F3	0.6516	0.6594	1.19%	0.6526	0.6579	0.80%	0.6873	0.6897	0.36%
F4	0.6553	0.6575	0.35%	0.6543	0.6562	0.28%	0.6861	0.6904	0.63%
F5	0.6552	0.6565	0.19%	0.6541	0.6563	0.34%	0.6870	0.6913	0.62%
F6	0.6551	0.6570	0.29%	0.6544	0.6558	0.22%	0.6868	0.6899	0.45%

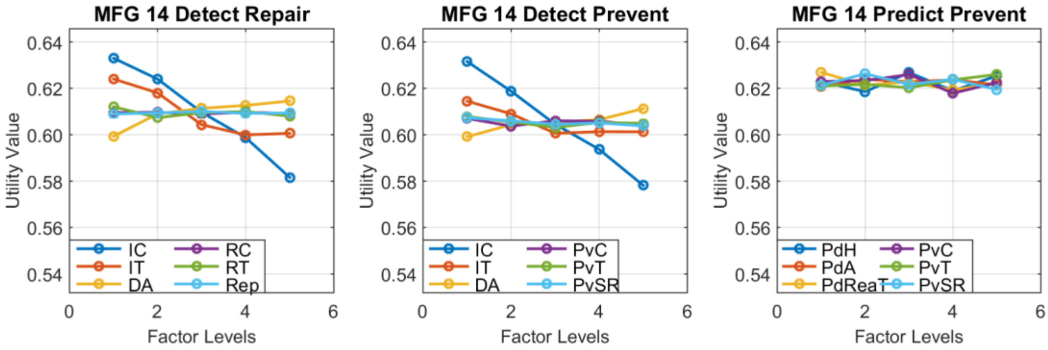


Figure 55: MFG 14 ANOM diagrams for each ZDM (defect rate 2%)

Table 42: ANOM minimum, maximum, & relative difference MFG 14

	Detect – Repair			Detect – Prevent			Predict – Prevent		
Factors	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**
F1	0.6215	0.6800	8.98%	0.6173	0.6785	9.45%	0.6662	0.6741	1.19%
F2	0.6431	0.6691	3.96%	0.6427	0.6584	2.41%	0.6681	0.6715	0.51%
F3	0.6432	0.6587	2.38%	0.6420	0.6546	1.94%	0.6667	0.6747	1.18%
F4	0.6525	0.6538	0.19%	0.6470	0.6504	0.51%	0.6648	0.6741	1.39%
F5	0.6517	0.6551	0.52%	0.6466	0.6512	0.71%	0.6671	0.6740	1.04%
F6	0.6524	0.6539	0.22%	0.6477	0.6502	0.38%	0.6669	0.6731	0.93%

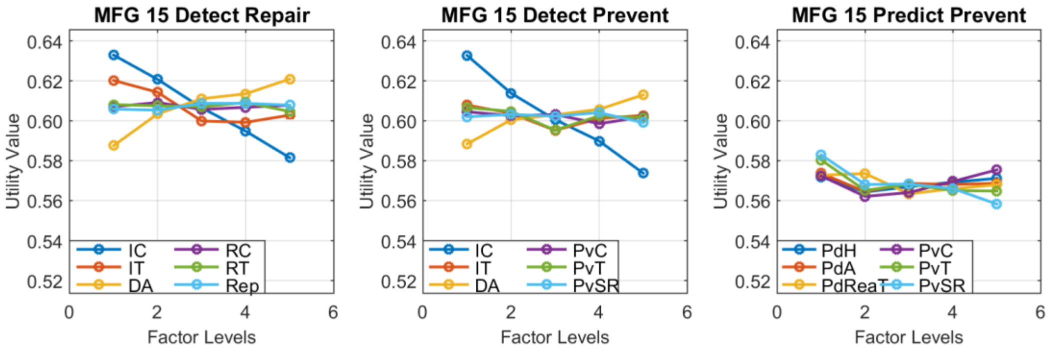


Figure 56: MFG 15 ANOM diagrams for each ZDM (defect rate 5%)

Table 43: ANOM minimum, maximum, & relative difference MFG 15

	Detect – Repair			Detect – Prevent			Predict – Prevent		
Factors	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**	**Min**	**Max**	**Rel. Dif.**
F1	0.6149	0.6740	9.17%	0.6069	0.6723	10.22%	0.6017	0.6112	1.56%
F2	0.6359	0.6590	3.56%	0.6302	0.6443	2.20%	0.6023	0.6109	1.42%
F3	0.6231	0.6586	5.53%	0.6240	0.6495	4.01%	0.6005	0.6115	1.83%
F4	0.6428	0.6463	0.54%	0.6344	0.6410	1.03%	0.6001	0.6124	2.03%
F5	0.6416	0.6458	0.65%	0.6313	0.6430	1.85%	0.6028	0.6184	2.54%
F6	0.6429	0.6457	0.44%	0.6344	0.6407	0.99%	0.5957	0.6217	4.27%

5.3.2 Analysis of variance (ANOVA)

The ANOVA was performed to study how much each factor contributes to the final solution quality. Table 44 contains the average values of the ANOVA for each factor and for each ZDM strategy. Immediately one can distinguish that the inspection cost has the higher contribution to the result by 68.44% and 77.95% for the detect –repair and detect – prevent strategies, respectively. To better explain the meaning of the ANOVA results, the detect – repair strategy was used. For this case, 68.44% of the result (100%) came from the inspection cost, and similarly 19.66% came from the inspection time and so on. In the detect –repair and detect – prevent strategies, it was clear which were the dominant factors contributing to the final result. This was not the case in the predict – prevent strategy where a near uniform influence of each factor existed around 15%. Only prevention time was at 22% but it was not a great deviation as it was in the other two ZDM strategies.

Table 44: ANOVA average (Avg.) factor influence

Detect - Repair		**Detect - Prevent**		**Predict – Prevent**	
Factors	Avg. % Factor influence	Factors	Avg. % factor influence	Factors	Avg. % factor influence
Inspection Cost	68.44%	**Inspection Cost**	77.95%	**Prediction Horizon**	15.01%
Inspection Time	19.66%	**Inspection Time**	11.57%	**Prediction Accuracy**	15.42%
Detection accuracy	8.45%	**Detection accuracy**	4.00%	**Prediction reaction time**	15.69%
Repair Cost	1.66%	**Prevention Cost**	1.79%	**Prevention Cost**	13.86%
Repair Time	1.07%	**Prevention Time**	3.55%	**Prevention Time**	22.00%

Reparability	0.71%	**Prevention Success rate**	1.14%	**Prevention Success rate**	18.02%
Total % sum	100%		100%		100%

Figure 57 illustrates the ANOVA analysis for the detect – repair strategy for each of the 15 MFGs. As was also noticed in the total average results, the dominant factor was inspection cost with 68.44% on average. This applied to all MFGs except MFG 4. In MFG4, the factor with the highest contribution to the final result was inspection time with 56%. Moreover, it was noticed that in the MFGs with higher defect rates, the detection accuracy factor had a significantly higher influence on the final result. Furthermore, the repair factors contributed more to the MFGs for which repair was possible.

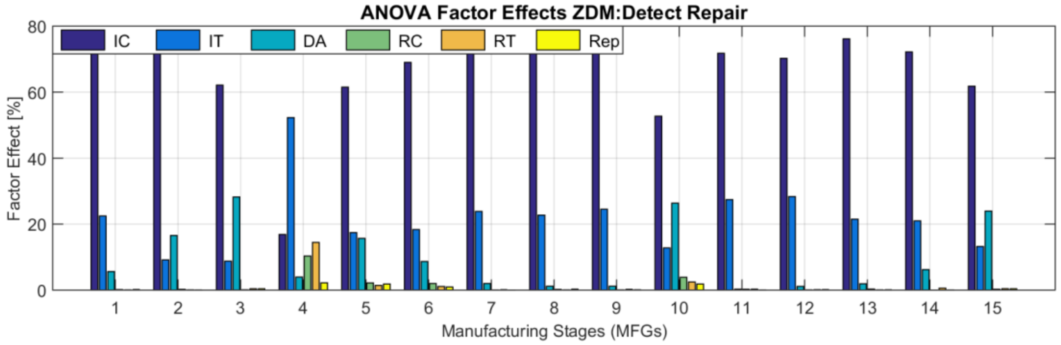


Figure 57: ANOVA diagram for detect – repair strategy

Similarly, Figure 58 presents the ANOVA results for the detect – prevent strategy. In this case, for all MFGs the dominant factor contributing to the final result was inspection cost with 77.95% on average. Furthermore, as noticed in the detect – repair strategy, the MFGs with the highest defect rate had the other factors that contributed more compared with the MFGs with a low defect rate. In the case of MFG10, prevention time was the second factor with the highest impact with 23%. This is noted because in most MFGs, the second factor was inspection time, with some exceptions (MFGs 2, 3, 5, and 15).

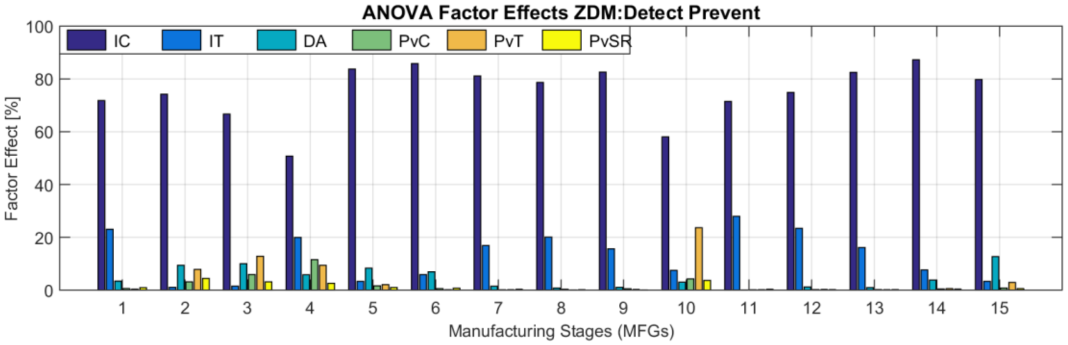


Figure 58: ANOVA diagram for detect – prevent strategy

In the presentation of the average ANOVA results in the predict – prevent ZDM strategy, a more uniform influence of each factor existed on the result. This can be seen in Figure 59. All factors contributed significantly to the final result. The greatest difference between this case and the two previous ones is the fact that no clear dominant factor influenced the final result. Furthermore, Figure 59 is a good example for demonstrating that each of the MFGs has its own almost unique characteristics and each factor affects each MFG differently. There are three MFGs (1, 2, and 15) for which one factor exceeded the 40% of influence. On MFGs 1 and 2, prevention time was the dominant factor, whereas on MFG 15 prevention success rate was dominant. Furthermore, on MFG 4 the prevention time had almost 40% of the influence, similar to MFGs 1 and 2.

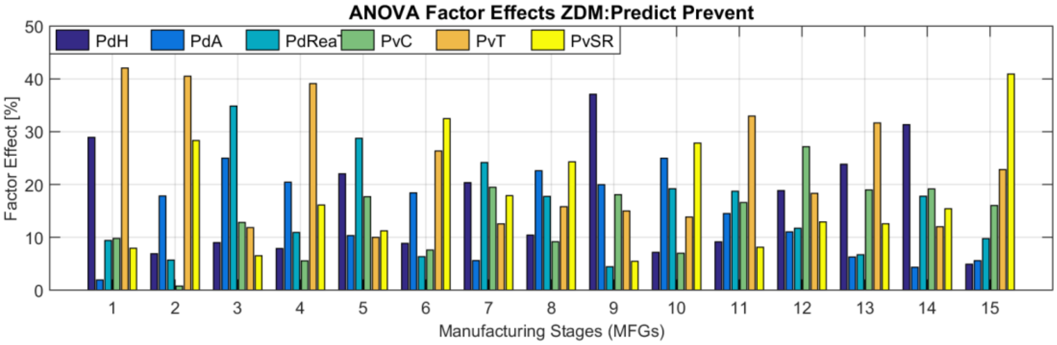


Figure 59: ANOVA diagram for predict – prevent strategy

5.3.3 MFGs' ZDM solution maps (combinations)

The final step of the present research work was to illustrate the performance of each ZDM strategy for each MFG for different factor combinations. To achieve this, a higher resolution of factor levels was required. In light of the tables, Table 45, Table 46, and

Table 47 were created with nine levels for each factor. Both the upper and lower limits were the same as in the experiment series for creating the DT model (chapter 5.2.1). Those tables refer to the three ZDM strategies. After the performance and accuracy of the utility value DT model were validated, the model could be used to estimate the utility values without running the simulator, allowing the "simulation" of a much higher number of experiments in a fraction of the actual simulation time.

Table 45: Expanded factor levels for detect – repair

Detect – Repair									
Levels	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
F1	0.0100	0.0713	0.1325	0.1938	0.2550	0.3163	0.3775	0.4388	0.5000
F2	0.0100	0.0400	0.0700	0.1000	0.1300	0.1600	0.1900	0.2200	0.2500
F3	0.7000	0.7363	0.7725	0.8088	0.8450	0.8813	0.9175	0.9538	0.9900
F4	0.0500	0.3563	0.6625	0.9688	1.2750	1.5813	1.8875	2.1938	2.5000
F5	0.0500	0.2938	0.5375	0.7813	1.0250	1.2688	1.5125	1.7563	2.0000
F6	0.1000	0.2063	0.3125	0.4188	0.5250	0.6313	0.7375	0.8438	0.9500

Table 46: Expanded factors for detect – prevent

Detect – Prevent									
Levels	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
F1	0.0100	0.0713	0.1325	0.1938	0.2550	0.3163	0.3775	0.4388	0.5000
F2	0.0100	0.0400	0.0700	0.1000	0.1300	0.1600	0.1900	0.2200	0.2500
F3	0.7000	0.7363	0.7725	0.8088	0.8450	0.8813	0.9175	0.9538	0.9900
F4	0.0500	0.6688	1.2875	1.9063	2.5250	3.1438	3.7625	4.3813	5.0000
F5	0.3000	1.1375	1.9750	2.8125	3.6500	4.4875	5.3250	6.1625	7.0000
F6	0.6000	0.6438	0.6875	0.7313	0.7750	0.8188	0.8625	0.9063	0.9500

Table 47: Expanded factors for predict – prevent

Predict – Prevent									
Levels	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
F1	0.0187	0.1568	0.2949	0.4330	0.5711	0.7092	0.8472	0.9853	1.1234
F2	0.7000	0.7363	0.7725	0.8088	0.8450	0.8813	0.9175	0.9538	0.9900
F3	0.1123	0.3792	0.6460	0.9128	1.1796	1.4464	1.7132	1.9800	2.2468
F4	0.0500	0.6688	1.2875	1.9063	2.5250	3.1438	3.7625	4.3813	5.0000
F5	0.3000	1.1375	1.9750	2.8125	3.6500	4.4875	5.3250	6.1625	7.0000
F6	0.6000	0.6438	0.6875	0.7313	0.7750	0.8188	0.8625	0.9063	0.9500

Therefore, for each ZDM table with the expanded factor levels, all possible combinations were considered and the utility values were calculated with the use of the developed DT model. In total, 531,441 (9^6) factor combinations were calculated for each ZDM strategy. To have a value to compare with the produced utility values, a benchmark scenario representing the ideal scenario without defects was simulated. The utility value for the ideal scenario was 0.6905, which means that the highest possible utility value for the current problem and KPIs is 0.6905. Using the predicted utility values and the acquired utility value from the ideal scenario, the relative difference was calculated between the optimal utility value and each estimated utility value. Those relative differences were sorted from lowest to highest (worst to best) and plotted, and those plots can be found in Figure 61 to Figure 75. The plots refer to each MFG, and each plot illustrates the performance of each of the ZDM strategies with the different ZDM factor sets.

Figure 60 illustrates an example of MFG 1 regarding detect repair with un-sorted values. The graph is difficult to read and if the graphs of other two ZDM pairs strategies are added it will be confusing. Furthermore, from the shape of Figure 60 it is difficult to make conclusions. For these reasons it was decided to sort the values. At this point it should be mentioned that on the x-axis the ZDM parameters sets are in the order that the algorithm created. In the sorted version of the graphs the order is changing because each utility value is accompanied by the corresponding set of ZDM parameters values.

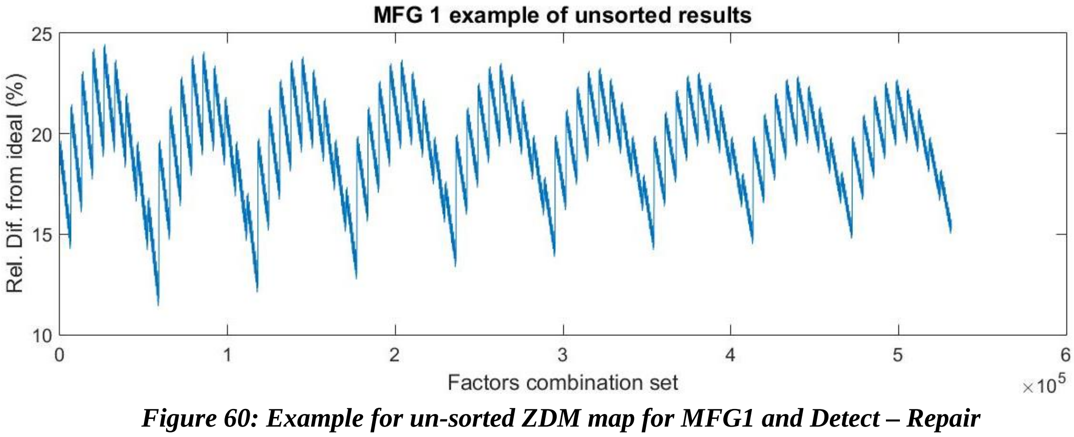


Figure 60: Example for un-sorted ZDM map for MFG1 and Detect – Repair

Table 48 presents a summary of the results produced using the DT model and the expanded factor levels. This table illustrates the minimum and maximum utility values from each of the MFGs and for each ZDM strategy. Furthermore, the average utility value is shown to demonstrate the average performance of each of the ZDM strategies. Once again, the results from MFG 4 draw attention immediately because MFG 4 holds both the highest and lowest utility value scores. The lowest utility value is for the detect – repair strategy and the highest is for the predict – prevent strategy. At this point it should be mentioned that the highest score is

0.61% worse than the ideal scenario. Furthermore, in most of the MFGs, the implementation of the ZDM strategies starts from an approximately 6% inferior total performance compared with the ideal scenario.

Table 48: Global minimum, maximum, and average utility values per MFG

	Detect Repair			**Detect Prevent**			**Predict Prevent**		
	Min	**Max**	**Avg.**	**Min**	**Max**	**Avg.**	**Min**	**Max**	**Avg.**
MFG 1	0.5554	0.6505	0.5992	0.5469	0.6535	0.5938	0.5878	0.6359	0.6125
MFG 2	0.5423	0.6447	0.5874	0.5281	0.6424	0.5808	0.5342	0.6241	0.5772
MFG 3	0.5446	0.6515	0.5947	0.5463	0.6648	0.5929	0.4896	0.6555	0.5697
MFG 4	0.3609	0.6159	0.5178	0.3974	0.5613	0.4827	0.5317	0.6863	0.6102
MFG 5	0.5447	0.6657	0.6037	0.5447	0.6458	0.5921	0.5489	0.6383	0.5889
MFG 6	0.5506	0.6576	0.6071	0.5494	0.6429	0.5917	0.5783	0.6556	0.6167
MFG 7	0.5675	0.6541	0.6056	0.5670	0.6512	0.6042	0.6314	0.6420	0.6366
MFG 8	0.5664	0.6552	0.6061	0.5680	0.6490	0.6056	0.6279	0.6518	0.6395
MFG 9	0.5659	0.6570	0.6065	0.5654	0.6508	0.6058	0.6308	0.6444	0.6377
MFG 10	0.5269	0.6643	0.5978	0.4959	0.6534	0.5686	0.5238	0.5715	0.5451
MFG 11	0.5680	0.6532	0.6067	0.5693	0.6536	0.6078	0.6378	0.6694	0.6544
MFG 12	0.5660	0.6531	0.6061	0.5644	0.6557	0.6041	0.6259	0.6601	0.6416
MFG 13	0.5704	0.6537	0.6067	0.5707	0.6504	0.6054	0.6315	0.6481	0.6391
MFG 14	0.5602	0.6550	0.6043	0.5628	0.6498	0.6009	0.6045	0.6372	0.6195
MFG 15	0.5478	0.6619	0.5995	0.5448	0.6547	0.5952	0.5391	0.6084	0.5670

The MFG's ZDM mapping plots reveal three different behaviors of the interaction of the ZDM strategies. First, the ZDM strategy performances are clearly distinguished and there is a clear overview of the most suitable ZDM for a specific ZDM. This means that the lines do not intersect at any point. In other MFGs, the ZDM lines do intersect, meaning that from that point on the most suitable ZDM strategy changes. Another behavior is that all three ZDM strategies converge to almost the same point, which applies to both the best and worst point. In the MFGs for which repair of the defective part is not possible, the ZDM strategies of detect – repair and detect – predict have almost equal performance with some variations. Furthermore, in many cases where the ZDM factors were becoming more resource-demanding, the dominant and most efficient ZDM strategy was prediction – prevention. This applies to the following MFGs: 1, 4, 7, 8, 9, 11, 12, 13, and 14. Additionally, in the MFGs with a small defect rate, the prediction – prevention strategy seemed to have constant performance with a very small variation. In the MFGs where repair was possible (104, 106, and 110), no clear strategy prevailed. In MFG 104, the best-performing ZDM strategy was prediction – prevention and the second best-performing was detect – repair. In MFG 106, the detect – repair and predict – prevent strategies had almost the same performance up to a 10% deviation from the ideal condition. After that point, the predict – prevent strategy behaved better as the factor values increased. In the case of MFG 110, the best-performing ZDM strategy was detect – repair followed by detect – prevent with an average margin of a 6% difference. Predict – prevent was constantly worse than the others except in the solutions with the highest ZDM factor values, where it became better than detect – prevent. Moreover, predict – prevent had very small performance variations.

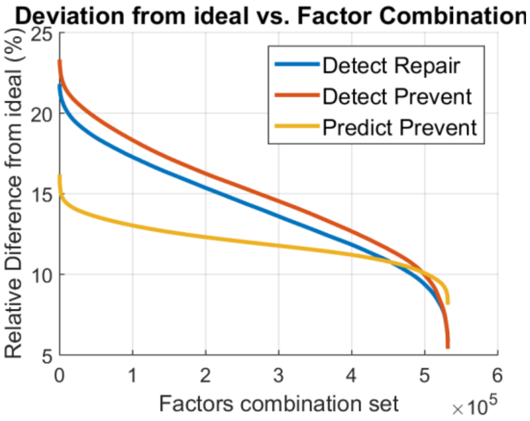


Figure 61: Final MFG 1 ZDM strategies mapping

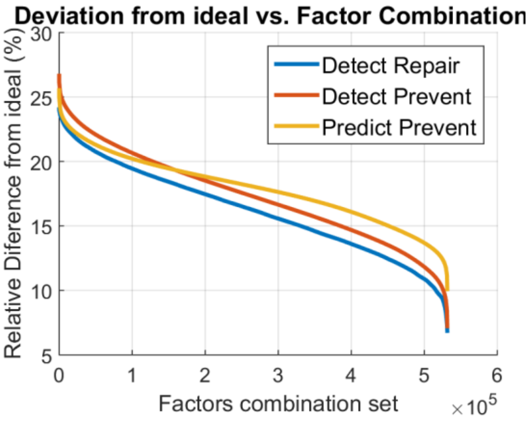


Figure 62: Final MFG 2 ZDM strategies mapping

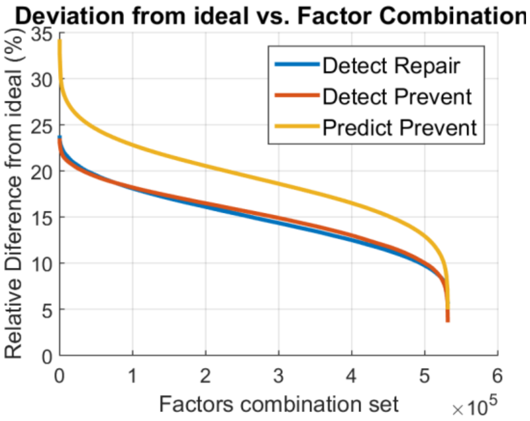


Figure 63: Final MFG 3 ZDM strategies mapping

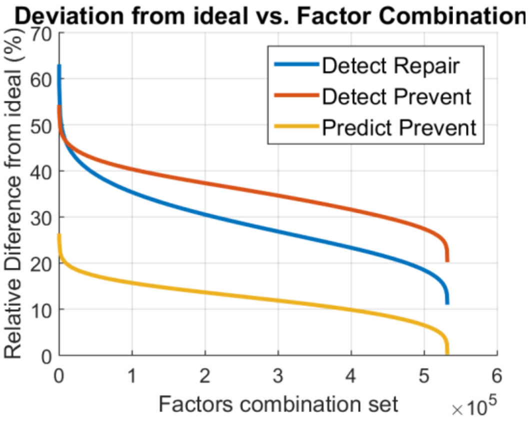


Figure 64: Final MFG 4 ZDM strategies mapping

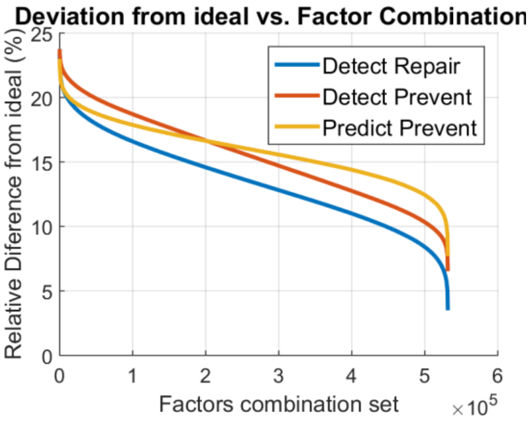


Figure 65: Final MFG 5 ZDM strategies mapping

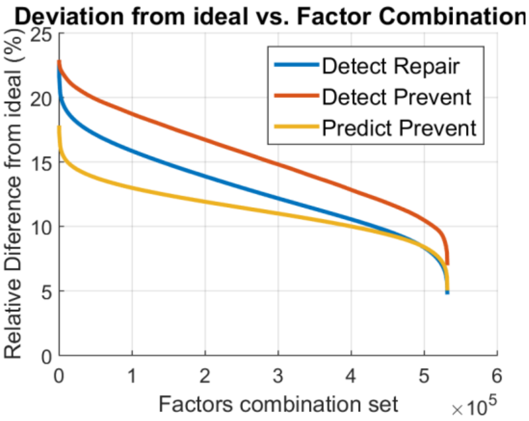


Figure 66: Final MFG 6 ZDM strategies mapping

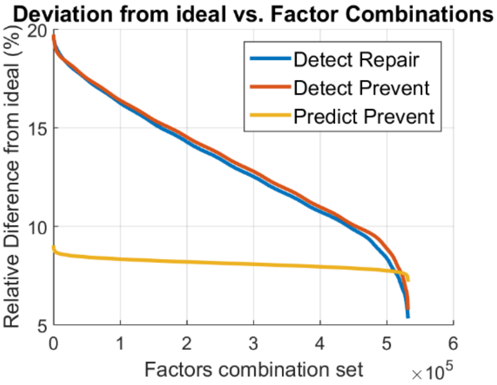


Figure 67: Final MFG 7 ZDM strategies mapping

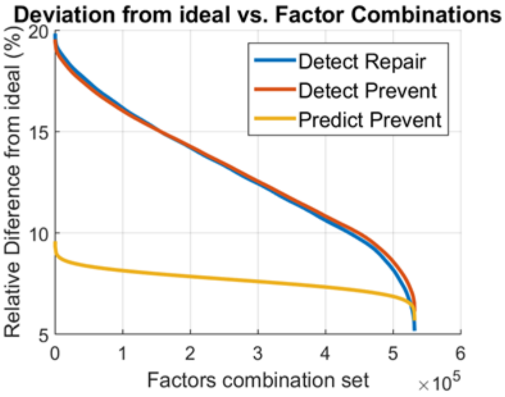


Figure 68: Final MFG 8 ZDM strategies mapping

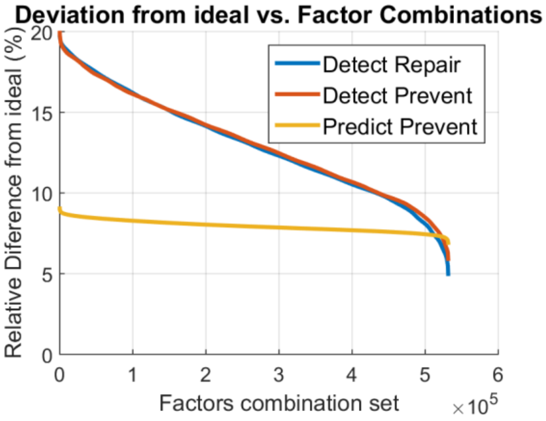


Figure 69: Final MFG 9 ZDM strategies mapping

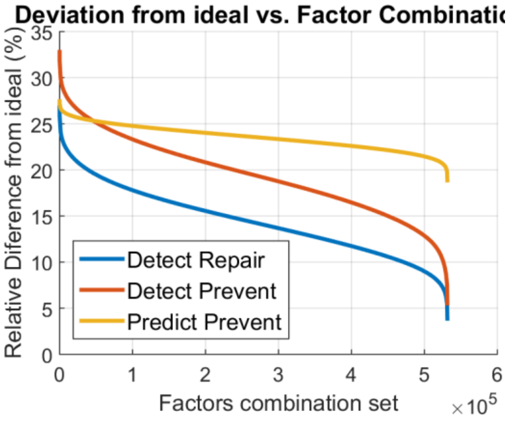


Figure 70: Final MFG 10 ZDM strategies mapping

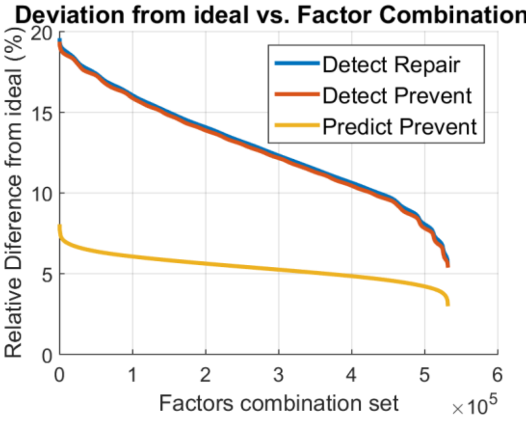


Figure 71: Final MFG 11 ZDM strategies mapping

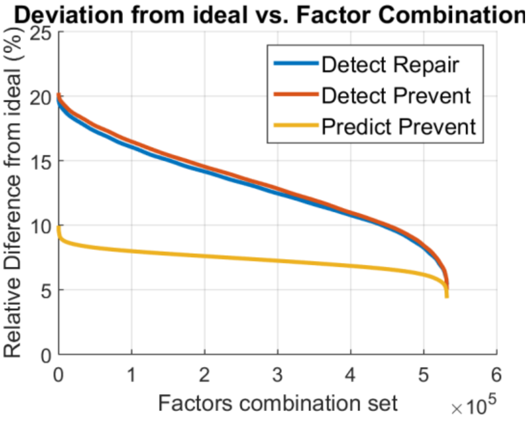


Figure 72: Final MFG 12 ZDM strategies mapping

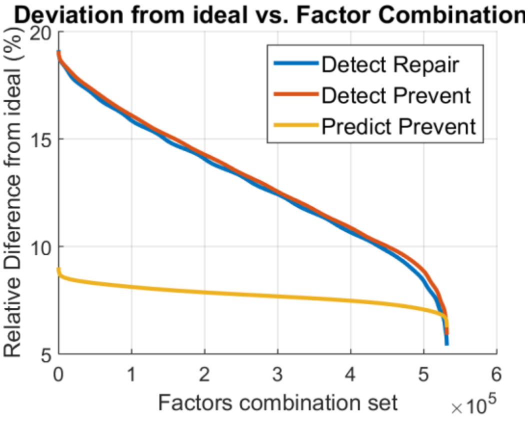


Figure 73: Final MFG 13 ZDM strategies mapping

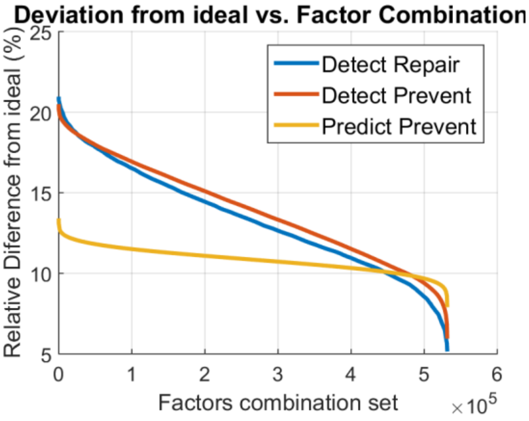


Figure 74: Final MFG 14 ZDM strategies mapping

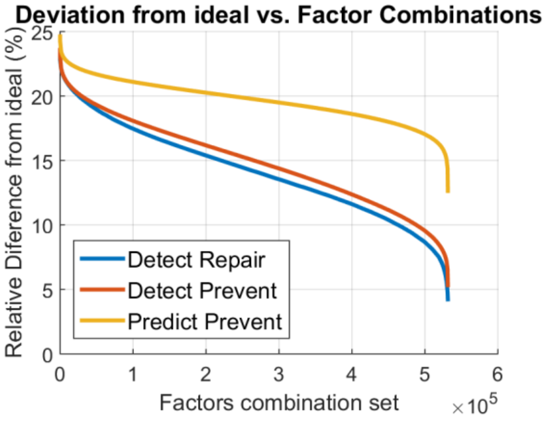


Figure 75: Final MFG 15 ZDM strategies mapping

6 Discussion

This chapter is devoted to the discussion of the results acquired from each component of the developed scheduling tool as well as the discussion of the results of the DT methodology and the DT model itself. The structure of this chapter follows a similar structure to previous chapters, explaining the individual components.

6.1 ZDM-Triggering Factor Modeling Discussion

As explained in chapter 2.1, there are two different types of triggering events in ZDM: the detection of a defect and the prediction of a defect that might occur in the near future. Chapter 4.2 presented the defect generation module, one of the most crucial components of the developed scheduling tool. This is because the entire ZDM implementation is based on the defects generated by this component. The developed method proved to be flexible and easily adaptable to the different machine characteristics. Furthermore, the generated defects were realistic and very close to the actual defect generation patterns of the real production. At this point it should be mentioned that the developed method can describe only defect generation patterns that follow exponential forms as the operation time of the machine increases. The added stochasticity brought the results generated by the defect generation module even closer to reality. The second ZDM-triggering factor is the prediction of defects, and chapter 4.3 presented the developed method. This component depends on the outcome of the defect generation to produce results. The selected approach for modeling the defect prediction module performed very well and simulated the concept of defect prediction very accurately. The additional stochasticity was also a crucial addition to accurately simulate the prediction of defects. Outcomes of chapter 4.2 and 4.3 contributed to answering Research Question 1.

6.2 DSS Component Results Discussion

The proposed DSS tool was developed to assist to the decision-making process when a defective part is detected to decide whether to repair it, discard it, or do nothing (chapter 4.4). The simulation results showed that in both simulation periods, the waste in terms of raw materials cost reduced by 4.702% and 3.858% for short- and mid-term periods, respectively. The implementation of the described system will contribute to moving one step closer toward ZDM. The developed DSS, presented in chapter 4.4, is a subpart of the answer to Research Question 1.

The overall outcome of the conducted experiments was that the proposed DSS tool was on average 7.47% better compared with the current production policy of the manufacturing process of the specific PCB. Furthermore, the DSS produced significantly superior results in the event of important orders compared with less important ones in both scenarios (S1 and S2).

In some cases, in both S1 and S2 the DSS produced better or equal results to the benchmark scenario. The reason behind this behavior is the frequency of rescheduling the production. In the case of the benchmark scenario, there were no defects; therefore, the only events happening in the production were new orders coming in. In this regard, the production had to be rescheduled only 7 and 14 times in scenarios S1 and S2, respectively. On the other hand, the DSS scenario, besides the new order evets, had to also deal with the defects as events. This created the need/opportunity to reschedule the production more times to consider the actions required for the defective products. More rescheduling actions enabled achieving a more optimized schedule in terms of the measured KPIs. More specifically, the S1 scenario was rescheduled 26 times and the S2 scenario 41 times. Another fact that verified this is the machine utilization rates. In the case of the benchmark and discard scenarios, the machine utilization was consistently lower than that using the DSS.

The proposed DSS system is meant to be triggered in real time according to the events that occur during production. The simulations showed that the DSS tool requires on average only 0.1458 seconds for making a decision for each defect, which is acceptable for in-line use.

The simulation results showed that the proposed DSS had a positive effect on both simulation periods (short- and mid-term). Although the effect was positive, a huge difference between the two effects was observed. The overall performance of the proposed DSS was 147.45% better in the short-term scenario, whereas in the mid-term scenario the DSS was 7.47% better. The reason behind this significant difference is that in the short term there were only a few orders, and therefore, any performance difference was amplified because of the small number of orders. On the other hand, in the mid-term scenario, the results were smoother with no such huge differences.

Despite the promising results, there were a few limitations. For this DSS to be able to improve the performance of a production system, it is compulsory that the production facility has a certain level of flexibility to adapt to more frequent shop floor rescheduling. Another dimension that was not taken into account in the current research is the estimation of the cost that arises due to higher numbers of rescheduling rounds. In addition, the increased frequency of rescheduling can create great confusion among workers on the production line, which can result in extra costs due to mistakes. The developed DSS was tested in a production system configured as a flexible job shop; therefore the measured performance of the DSS might not be the same for other production system configurations. Finally, the simulations considered only a part of the production and not the entire shop floor.

6.3 Heuristic Rule Results Discussion

Heuristics are the most widely used method for solving scheduling problems and in general NP-Hard problems. Chapter 4.6 presented a series of modified heuristic rules for identifying the best rule among the defined ones to be used for the simulations that were required. The simulation results showed a clear relation between the initial and optimized solution. More specifically, the better the initial solution, the higher the quality of the optimized solution. The algorithms SumSPT and ECM were developed to take into account the results from the previous assignments during the operation assignment, and therefore achieve better results. OS_ECM was found to be the best of the tested algorithms because produced schedules were more balanced than the others. Moreover, OS allowed the most important orders to be prioritized and made first, which increased the quality of the solution. The addition of the “Length” parameter allowed the balancing of the production schedule, thus achieving better initial solutions. The results showed that the best schedules were those for Length = 10 and the “Length” parameter moved increasingly closer to the result of the corresponding algorithm; without that parameter, the schedules were worse and more unbalanced.

The optimization of the schedules required significantly more computation time than the initial solution algorithm. This was expected but in many cases there is not much time available in real production environments and the need for a fast high-quality solution arises. The developed algorithms showed that fast schedules of high quality can be produced fast, especially OS_ECM.

The NEH algorithm has demonstrated promising results in the literature [51]. In the context of the current research work, an adapted version of the NEH was developed. The results were not presented due to the computation time required by the NEH algorithm to generate an initial solution, which was 17 hours (due to the high number of operations to schedule 5180), something not realistic for real production environments.

6.4 Multi-Order Evaluation Results Discussion

In the era of mass customization and personalization, manufacturers are forced to change their mass production standards and implement a more customized manufacturing plan. Those changes have had the effect of rapidly increasing the number of orders that manufacturers should deal with at the same time. Furthermore, ZDM requires more frequent rescheduling of production due to the need for implementation of mitigation actions required for avoiding defects. Therefore, the developed method for ranking the different orders is very useful and mandatory in such implementations (chapter 4.7). The results of this method contributed to answering Research Question 1. Overall, the simulation results showed that in most of the cases, the orders that had high OC had better results than those produced by the single-level method. Some of the orders with high OC, though, had worse results, and this is because it is not always possible when scheduling to optimize the criteria for all the involved orders. This is because the proposed approach is based on the ranking of orders, and therefore, when optimizing the most important order there will be a loss. Furthermore, the proposed method produced significantly better results for the time criteria compared with those for the cost criteria. On average, the order tardiness was 3.63 days and 9.48 days for the two- and single-layer methods, respectively.

Production cost had a small fluctuation because each product requires almost the same raw materials and processing time. The COD is more complicated because it is a function of many factors. It relies on the tardiness, volume, and criticality of the order, and therefore, it is difficult to draw a conclusion. Furthermore, it was observed that orders with a high OC factor tend to have higher COD, which was expected and shows that the order-ranking approach produced the desired results.

The simulation results showed that the proposed method was capable of efficiently scheduling rush orders, which was the goal of the present research. It is worth mentioning that the proposed methodology produced slightly worse results for orders 7, 15, and 16 with a maximum loss of performance of 3.42% in order 15. This was due to the reason explained at the beginning of the Discussion chapter, namely that priority was given to other normal and rush orders that arrived earlier, and therefore, the optimization of those orders rendered the optimization of orders 7, 15, and 16 impossible.

Finally, the reason behind the achievement of more efficient schedules is derived from the proposed dynamic order-ranking method. More specifically, every time that rescheduling was performed, the order ranking changed according to the orders involved as well as the time that the rescheduling was performed, optimizing the importance of each order and to that extent the measured performance indicators.

6.5 Events Management Methodology Results Discussion

The events management component is one of the most important modules of the developed scheduling tool. Because of it, it is possible to implement the ZDM concept efficiently with a balanced number of rescheduling iterations without losing performance. It has the ability to evaluate all the events that arise and suggest which are going to be included in the next rescheduling iteration. This was possible using the defined “time” parameters, which proved to be very helpful and close to reality. The results of this component also contributed to answering Research Question 1.

More specifically, the experimental results for this component indicated that the developed methodology can assist in balancing the number of rescheduling iterations versus the number of unexpected events. The ANOVA revealed that NORT*NPRT is one of the interactions with the highest effect. This interaction had a positive trend when both the values increased. More specifically, the most positive effect was when the NPRT was at factor level 2 and NORT was at factor level 1. In this way, the company is able to quickly react to defect predictions and have enough time to manage all orders. Similar to the NORT*NPRT interaction, the interaction between NDDRT and NPRT also had a positive trend when both values increased. More specifically, the most positive effect was achieved when both factors were at factor level 2. This means that the company prefers to delay the action for the defect and prevention to manage both in the same rescheduling action. In this way, they can reduce the number of rescheduling actions. Another significant interaction was that between NDRT and NPDRT. In contrast to the other two interactions, NDRT*NPDRT had a negative trend. More specifically, when the factors' values increased, the effect of the interaction became more negative. Therefore, the company has a short delay time to postpone the prevention action and must quickly react when a new defect occurs. Finally, NORDT*NPDRT was the only antisynergistic interaction. Indeed, the effect was positive when one factor increased and the other decreased. In this way, the company can have a shorter delay time to manage prevention actions, which is balanced with a longer delay time to manage new orders.

6.6 Digital Twin Methodology & Model Discussion

In chapter 4.12, a methodology for creating a DT model of the developed scheduling tool was presented. Furthermore, the creation and validation of the DT model was presented in chapter 0. Both chapters contributed to answering Research Question 2. The developed methodology for creating a DT model of the developed scheduling tool proved to be lean, efficient, and highly accurate. On average, the error of the developed DT model was 1.066%, which is very low, making the model highly accurate and reliable since the standard deviation was only 0.9929%. Furthermore, it was noticed that for the predict – prevent strategy, the corresponding errors for each MFG were slightly higher than those calculated for the other ZDM strategies. This is due to the fact that the predict – prevent strategy has more stochasticity implemented compared with the other ZDM strategies, and therefore, the DT model produced results with slightly higher errors but at acceptable levels. The maximum error observed for the utility value DT model was 4.3108%.

The same observation was also noticed for the KPI DT model error. The higher the stochasticity and variation of a KPI, the higher the error that the DT model produced. This was noticed in the weighted tardiness and rescheduling cost. Both KPIs had high errors, reaching almost 20% in the case of weighted tardiness. Rescheduling cost relied on the rescheduling frequency as well as the number of tasks to be rescheduled. This process is quite complex and characterized by high levels of unpredictability. This is because those KPIs are unpredictable and rely on other factors, and therefore, it is difficult to obtain accurate value estimations. This was also proved by the high accuracy of the DT model for the KPIs with smaller variations than the aforementioned KPIs.

Moreover, the developed DT methodology proved to be flexible and easy to use. The same methodology can be applied in a different case. The only part that must be altered is the definition of control parameters. In the current case, six factors were used. A limitation that arises with the method is that the experiments that must be conducted have to be performed based on an orthogonal array. This poses some limitations because standard orthogonal arrays are limited and might not fit other cases. For that reason, the construction of an orthogonal array is mandatory, which makes the process more complex.

The level of accuracy of the DT model is heavily dependent on the number of levels that each factor has. For that reason, five-level factors were selected to capture the behavior of each factor level. This was also proven by the DT coefficient graphs presented in Annex 2. The graphs show that the influence of each factor was not linear but a mode complex curve, requiring more than three points to be defined. The graphs in Annex 2 were produced by fitting a mathematical model between the calculated points from the DT methodology presented in chapters 4.12. The selected fitting method was a piecewise smoothed spline. After experimentation, this method produced the optimal results compared with other fitting methods.

Furthermore, the absolute values were converted to relative values based on the estimated total cost and time for a product to be able to use the same results for other cases. The absolute values of PC and time can vary greatly and it is impossible to run simulations for all the combinations. However, if two products have different absolute values but the ratios are the same, the performance results from one product can be used for the other as well.

6.7 ZDM Mapping Results Discussion

Using the results from Research Question 1 and Research Question 2, the ZDM maps for the presented industrial use case (chapter 5) were produced and presented in chapter 5.3.3, and the results answered Research Question 3.

The ZDM mapping results revealed that the smaller the defect rate in a station, the greater advantage of the predict – prevent strategy. This is because in cases where the defect rate is high, the frequent "small" prevention actions that the predict – prevent strategy imposes have a negative impact on the performance of the manufacturing system. Therefore, in cases with a defect rate lower than 3%, predict – prevent is the dominant ZDM strategy and the most efficient.

Furthermore, a pattern was observed regarding the manufacturing process and ZDM performance. In more detail, in the current industrial case there were three manufacturing process categories being used: assembly, manufacturing of the primary components, and processes for adding features to the existing components. MFGs 202 and 205 performed the process of adding features to the existing components and both showed a common ZDM behavior. In both cases, detect – repair was the best-performing ZDM strategy followed by detect – prevent up to a certain point where predict – prevent overlapped and became the second ZDM strategy. In addition, a crucial point is that all three ZDM strategies were very close together, something not observed in the other MFGs performing different manufacturing processes. Additionally, a common trend in the performance of the ZDM strategies was observed for MFGs performing the manufacturing of primary components (207, 208, 209, 211, 212, 213, and 214). In all those MFGs, the dominant ZDM strategy was predict – prevent, showing an almost-constant performance drop regardless of the values of the ZDM parameters. In both manufacturing processes, the ZDM behavior demonstrated similar trends in each case, which is due to the fact that those manufacturing processes were not dependent on other processes (manufacturing of primary components) or they only depended on the previous one (the addition of features to existing components). On the other hand, for the assembly operation, the results were not following a common trend as observed with the other two manufacturing processes. This was also due to the fact that the assembly operation was heavily dependent on the dynamics of the MFGs providing the components for the assembly. Therefore, the more uncertainty and complexity is introduced to an MFG, the more complex the ZDM strategies' behavior becomes, and tools such as the proposed one are critical for the correct design of the implementation of the ZDM concept.

Based on the results of the experiments, MFG 204 where task 104 was performed exhibited unique characteristics compared with the other MFGs. This is because at that stage an assembly operation is performed combining three subcomponents into one (BoP, Figure 33). This makes the MFG critical and susceptible to quality issues. Furthermore, the implementation or an unsuitable ZDM strategy may cause great losses (Figure 64) and up to 63% decreased performance.

Combining the ZDM mapping results presented in chapter 5.3.3 and the preliminary product analysis presented in chapter 5.1.1, the following can be concluded. In Table 22, the higher the negative value of the difference, the higher the impact of the incorporation of the estimated defect rate on the solution, and predict – prevent was not as efficient as detect – repair or detect – prevent. This is because in cases with more defects and higher impact, frequent prevention actions reduced the performance of the system. Furthermore, from the same product analysis it could be concluded that when the relative difference between the product utility value with and without the defect rate is positive, then the most effective ZDM strategies are detect – repair and detect – prevent. Moreover, as was mentioned before, if this relative difference is negative then predict – prevent is the most suitable ZDM approach to select.

6.7.1 ZDM mapping results utilization

The use of the developed DT model helped to map the performance of each ZDM strategy for each of the MFGs. This is extremely helpful for manufacturers because they can select the best ZDM configuration for their case. At this point it should be mentioned that some of the factor combination sets may have nonrealistic values and might be impossible to implement with current technologies. Therefore, the produced graphs should be used in a specific way and not simply by selecting the best-performing strategy with the factor set with the best result. Instead, there are two different ways that manufacturers can utilize those results. The first is when a manufacturer wants to establish a product quality improvement process at certain manufacturing stages and asks several third parties to provide a solution for the problem that the manufacturer faces or when the manufacturer estimates that will phase quality issues, if the production is at the design stage. When the manufacturer has all potential solutions, they can evaluate them using the graphs presented in chapter 5.3.3. Simply by giving the ZDM parameters of each of the provided solutions as an input, the manufacturer will be able to visualize where each solution is in the ZDM mapping graph, and that way, the best solution can be selected. The second way to use the results works the other way around. Based on the produced graphs, the manufacturer selects a range of ZDM parameter combinations, where the performance of the manufacturing systems is at an acceptable level, and then asks a third party to provide a solution with those specifications. This approach entails the danger that the manufacturer might select parameter ranges that are impossible to implement. By using those graphs, manufacturers can easily evaluate and rank alternative solutions for implementing ZDM into their manufacturing systems. Furthermore, the evaluation process is significantly faster and the results are repeatable and independent of the expertise of a single expert worker.

7 Conclusion

The current research work focused on an emerging concept in quality control and improvement domains named ZDM. The goal of this study was to provide a tool for operating and designing a manufacturing system taking into consideration the principles that ZDM imposes. The study was structured around three Research Questions, presented in chapter 3. All of the Research Questions have been successfully answered in various chapter of the current thesis. More specifically, answers for Research Question 1 can be found in chapters 4.2, 4.3, 4.4, 4.5, 4.7, and 4.8. Answers for Research Question 2 are found in chapters 4.12 and 5.2, and those for Research Question 3 are found in chapter 5.3.3.

7.1 Scheduling Tool: Concluding Remarks

The developed ZDM-oriented scheduling tool was described in chapter 4. All key components of the tool were presented separately, and furthermore, validation and performance evaluation were performed for each component before being used for the main experiments. In general, the results from chapter 4 answered Research Question 1 completely. More specifically, chapters 4.2, 4.3, 4.4, 4.5, 4.7, and 4.8 are the chapters related to ZDM implementation of the scheduling tool.

The cost-based DSS was developed to automate the decision-making process in the face of detected defects during the production process for minimizing costs due to poor quality. This DSS was integrated with a dynamic scheduling system to simulate the manufacturing process and measure the performance of the developed cost-based DSS. Upon the assessment of these principal cost functions, the system was able to make autonomous cost-effective decisions between deferral, repair, and scrapping the products when the autonomous quality assessment device detected a defective product. For the evaluation of the performance of the developed DSS tool, a real-life industrial use case was employed from the semiconductor domain. The results from the simulation showed improvement in the production system overall by 7.47%, which is a significant improvement. This performance improvement was measured with three main KPIs: overall tardiness, makespan, and total production cost, which included the machine operational cost, setup cost, raw material cost, and cost regarding delay penalties. In the current industrial case, the orders were delay-sensitive and the penalty was proportional to the OC and tardiness. The DSS solution was outperformed in terms tardiness and makespan in both the discard and benchmark cases, where in terms of total cost the DSS solution was very close to the ideal cost by 7.07%.

Four heuristic algorithms were developed and tweaked to enhance their performance and achieve high-quality schedules quickly. The alteration was the addition of an extra parameter, “Length,” for creating schedules that are more balanced. In addition, each customer’s order was prioritized for use during the optimization problem. The orders were sorted based on their due date, OV, and customer profile. The four developed heuristic algorithms were MC, which is based on the price of the operations on each machine; SPT, which places the operation on the machine that performs the fastest; sum of the SPT (SumSPT), which looks for the smallest sum of processing time at each operation assignment to a machine; and ECM, which places the operation on the machine with the smallest sum, but takes into account the processing time of the operation to be placed. The developed algorithms were tested under a real-life scenario with promising results. In general, the addition of the prioritization level to the existing heuristic showed that it can improve the resulting schedule. In addition, the best algorithm was the ECM, which considered the result of the previous assignments, and therefore produced better and more balanced schedules. Tabu search improved the initial solution on average by 7.481%. On the other hand, a significant difference existed between the computation time of the tabu search and the developed heuristic algorithms, with a relative difference of 199.152%. Therefore, the ECM algorithm can be used without optimization in cases where the rescheduling of production is required very often for counteracting unexpected events, thereby achieving ZDM.

The multi-order evaluation component had two parts: the first was the ranking of the individual orders for following that sequence in the scheduling process and not “first come first served.” This order ranking approach took the OV into consideration, the specific customer importance and ordering frequency, and the available timeframe for finishing the specific order. The second part of the proposed methodology was a two-level criteria evaluation approach during the rescheduling process. The additional extra level allowed the incorporation of the OC into the measured criteria used by the scheduling tool. The criteria used were the order makespan, tardiness, production cost, and delay cost. The proposed method was tested through a real-life industrial scenario from the semiconductor domain utilizing a semi-automated assembly line. Two scenarios were simulated, one using single-level criteria evaluation and one using the current proposal. The simulation results showed that the proposed method could efficiently handle rush orders and in general the rescheduling of the production under investigation. The current method created 5.615% better schedules for the given demand profile compared with the single-level approach.

The events management algorithm allows companies to cope with contemporary production while maintaining very high-quality standards. Indeed, most of the factors analyzed are from the ZDM concept, which has the aim of achieving higher quality by eliminating defective parts. The model developed is based on the Design of Experiments method, which allows the identification of the optimal setting for tuning parameters and the contribution of each parameter to the solution. The ANOM method has outlined that the changes in the values of the factors could significantly affect the solution quality. The factors with the highest effect were NORT, NPRT, and PH. The ANOVA method highlighted that the factors with the highest contribution are NORT, NPRT, and PH; they accounted for almost 79%. Moreover, thanks to the support of linear graphs, the most significant interactions were analyzed to understand the effect of combinations of factors on solution quality. In this way, it was possible to develop a tool that will guide companies to react to unexpected events in the most effective way by identifying the exact action time.

7.2 Digital Twin & Results: Concluding Remarks

The proposed methodology for creating a DT model for describing the results of the developed scheduling tool without running the simulation proved to be easy to use and very efficient, successfully answering Research Question 2. The DT model had six control parameters that were defined from the ZDM concept. The same methodology can be used with different numbers of parameters and levels according to the needs of the modeler. A series of 2250 random experiments were performed to calculate the accuracy of the created DT model. Those experiments were simulated using the developed scheduling tool and the utility values were calculated. Using the same random experiments and the DT model the predicted utility values were calculated. Therefore, the accuracy is the relative difference between the actual and the predicted utility values. The average accuracy of the utility value DT model was 98.934%, which was considered very accurate. Using the same results, from the random experiments, individual DT models were developed for each KPI. On average, the accuracy of all KPI models was 93.2%, which is also acceptable. More specifically, 12 out of 18 KPIs had very high accuracy, more than 95%, and the rest of the KPIs had an accuracy between 80% and 95%. Those KPIs were those with the highest fluctuation, and therefore, the model was unable to predict the value with high accuracy. Each simulation (a total of 1125 simulations were performed), for the creation of the DT model, required on average 2.5 hours of computation time to solve the given ZDM scenario, which in a real manufacturing environment is prohibitive. On the other hand, the produced model could estimate the output of the scheduling tool for different ZDM parameter values in less than a second, making the calculation of the results presented in chapter 5.3.3 possible.

Before the creation of the DT model, a preliminary analysis on the product under investigation was performed with the goal of identifying the criticality of each MFG in terms of implementing a ZDM strategy. The results from the preliminary product characteristics analysis indicated the findings are in line with the results of the use of the DT model, and some behavioral patterns could be identified.

The produced DT model was used to map the ZDM strategies' performance for each MFG, to provide manufacturers with the ability to select the best-performing ZDM strategy for their case (answering Research Question 3). Those maps can be very helpful to manufacturers in cases were a decision is required for the selection of equipment and strategies for implementing ZDM. Furthermore, they are meant to provide repeatable decisions and standardization for quality improvement design. One other advantage that the current methodology provides is user friendliness; the model is controlled only by six key ZDM parameters and therefore the only change that the manufacturers need to make is to plug in the ZDM parameters they desire to visualize where on the curve their selections lead, thus comparing and viewing the performance of their choices. The flexibility of the proposed methodology and tool can also be very useful and handy in cases where a quick reconfiguration of the production is required to adapt to market needs, as was required during the COVID-19 pandemic. A limitation of the current approach is that the presented methodology is use-case-specific, and therefore, if the use case changes, then the experiments and the analysis should be performed again. In chapter 7.3, some future steps are analyzed to solve this problem in the future.

7.3 Future Research Steps

Future research steps should focus mainly on the generalization of the developed tools and methodologies. More specifically, the current ZDM analysis was performed based on a specific use case. A detailed study is required to demonstrate if the concept presented in chapter 4.11, with the conversion of the absolute values to relative ones, can be generalized. This is required to be able to apply ZDM strategy trends from different industrial cases to others because the defined ratios are the same or similar. Additionally, a study is required to examine whether the ZDM trends for each MFG can be obtained by simulating only the desired MFG and not the entire production. This is considered for saving valuable time from simulating MFG nodes that are not necessary. Moreover, this will push toward the generalization of the method even further. Furthermore, the presented methodology regarding the creation of the DT method requires some enhancement to make it flexible enough to fit most cases. Currently the presented methodology is limited by the standard orthogonal arrays, which are themselves limited. The generalization of the method means that a method must be developed for the construction of orthogonal arrays to meet requirements and not be limited by the specifications of standard orthogonal arrays.

8 Publications

Publications in International Refereed Journals

1. F. Psarommatis, “A generic methodology and a digital twin for zero defect manufacturing (ZDM) performance mapping towards design for ZDM,” J. Manuf. Syst., vol. 59, pp. 507–521, Apr. 2021, doi: 10.1016/j.jmsy.2021.03.021 .
2. F. Psarommatis, G. Martiriggiano, X. Zheng, and D. Kiritsis, “A Generic Methodology for Calculating Rescheduling Time for Multiple Unexpected Events in the Era of Zero Defect Manufacturing,” Front. Mech. Eng., vol. 7, p. 646507, Apr. 2021, doi: 10.3389/fmech.2021.646507 .
3. F. Psarommatis, S. Prouvost, G. May, and D. Kiritsis, “Product Quality Improvement Policies in Industry 4.0 : Characteristics , Enabling Factors , Barriers , and Evolution Toward Zero Defect Manufacturing,” Front. Comput. Sci., vol. 2, no. August, pp. 1–15, 2020, doi: 10.3389/fcomp.2020.00026 .
4. F. Psarommatis, G. May, P.-A. Dreyfus, and D. Kiritsis, “Zero defect manufacturing: state-of-the-art review, shortcomings and future directions in research,” Int. J. Prod. Res., vol. 7543, pp. 1–17, 2019. DOI: https://doi.org/10.1080/00207543.2019.1605228
5. Grevenitis K, Psarommatis F, Reina A, Xu W, Tourkogiorgis I, Milenkovic J, et al. A hybrid framework for industrial data storage and exploitation. In: Procedia CIRP. Elsevier B.V.; 2019. p. 892–7, doi: 10.1016/j.procir.2019.03.221.
6. M. Doukas, F. Psarommatis, D. Mourtzis, Planning of manufacturing networks using an intelligent probabilistic approach for mass customised products, International Journal of Advanced Manufacturing Technology, 2014, DOI: http://link.springer.com/article/10.1007/s00170-014-6121-z
7. D. Mourtzis, M. Doukas, F. Psarommatis, “A Toolbox for the Design and Operation of Manufacturing Networks in a Mass Customisation Environment”, Journal of Manufacturing Systems (2014), DOI: 10.1016/j.jmsy.2014.06.004, http://www.sciencedirect.com/science/article/pii/S0278612514000703
8. D. Mourtzis, M. Doukas, F. Psarommatis, C. Giannoulis, G. Michalos, “A web-based platform for mass customisation and personalisation”, CIRP Journal of Manufacturing Science and Technology, (2014), DOI: http://dx.doi.org/10.1016/j.cirpj.2013.12.003
9. D. Mourtzis, M. Doukas, F. Psarommatis, “Design of manufacturing networks for mass customization using an intelligent search method”, International Journal of Computer Integrated Manufacturing, 2014, DOI: http://www.tandfonline.com/doi/full/10.1080/0951192X.2014.900867
10. D. Mourtzis, M. Doukas, F. Psarommatis, “Design and Operation of Manufacturing Networks for MassCustomisation”, CIRP Annals – Manufacturing Technology, Volume 63, Issue 1, 2013, pp. 467-470, DOI: http://dx.doi.org/10.1016/j.cirp.2013.03.126
11. D. Mourtzis, M. Doukas, F. Psarommatis, “A multi-criteria evaluation of centralized and decentralized production networks in a highly customer-driven environment”, CIRP Annals-Manufacturing Technology, (Vol.61, No.1, 2012), pp. 427-430, DOI: http://dx.doi.org/10.1016/j.cirp.2012.03.035

Publications in International Refereed Conferences

1. X. Zheng, F. Psarommatis, P. Petrali, C. Turrin, J. Lu, and D. Kiritsis, “A Quality-Oriented Digital Twin Modelling Method for Manufacturing Processes Based on A Multi-Agent

Architecture," Procedia Manuf., vol. 51, pp. 309–315, Jan. 2020. https://doi.org/10.1016/j.promfg.2020.10.044

2. F. Psarommatis, M. Vuichard, and D. Kiritsis, "Improved heuristics algorithms for re-scheduling flexible job shops in the era of Zero Defect manufacturing," Procedia Manuf., vol. 51, pp. 1485–1490, Jan. 2020. https://doi.org/10.1016/j.promfg.2020.10.206
3. F. Psarommatis, A. Boujemaoui, D Kiritsis, "A Computational Method for Identifying the Optimum Buffer Size in the Era of Zero Defect Manufacturing". In: IFIP Advances in Information and Communication Technology, Springer; 2020. p. 443–50. Available from: https://link.springer.com/chapter/10.1007/978-3-030-57997-5_51
4. F. Psarommatis and D. Kiritsis, "Identification of the Inspection Specifications for Achieving Zero Defect Manufacturing," in IFIP Advances in Information and Communication Technology, 2019, vol. 566, pp. 267–273, doi: 10.1007/978-3-030-30000-5_34
5. Hildebrand M, Tourkogiorgis I, Psarommatis F, Arena D, Kiritsis D. A Method for Converting Current Data to RDF in the Era of Industry 4.0. In: IFIP Advances in Information and Communication Technology, Springer New York LLC; 2019. p. 307–14. Available from: https://link.springer.com/chapter/10.1007/978-3-030-30000-5_39
6. Psarommatis, F., Kiritsis, D., 2018. A Scheduling Tool for Achieving Zero Defect Manufacturing (ZDM): A Conceptual Framework. Springer, Cham, pp. 271–278. https://doi.org/10.1007/978-3-319-99707-0_34
7. D. Mourtzis, M. Doukas, F. Psarommatis, "Manufacturing Network Design for Mass Customization using a Genetic Algorithm and an Intelligent Search Method", Procedia CIRP, 46th CIRP Conference on Manufacturing Systems 2013, Volume 7, pp. 37-42, DOI: http://dx.doi.org/10.1016/j.procir.2013.05.007
8. D. Mourtzis, M. Doukas, F. Psarommatis, N. Panopoulos "Design of multi-stage manufacturing networks for personalisation using metaheuristics", Advances in Sustainable and Competitive Manufacturing Systems, Lecture Notes in Mechanical Engineering 2013, FAIM 2013 Proceedings, Porto, Portugal, 26-28 June, DOI: 10.1007/978-3-319-00557-7_103, http://link.springer.com/chapter/10.1007/978-3-319-00557-7_103#page-1
9. D. Mourtzis, M. Doukas, F. Psarommatis, "Simulation-based design of production networks for manufacturing of personalised products", Proceeding of the IFIP WG 5.7 International Conference (IFIP AICT 397), APMS 2012, Advances in Production Management Systems, Part I, 24-26/09-2012, Rhodes, Greece (Awarded with the **Best Paper Award**), http://link.springer.com/chapter/10.1007%2F978-3-642-40352-1_38
10. D. Mourtzis, M. Doukas, F. Psarommatis, "Design and planning of decentralised production networks under high product variety demand", Procedia CIRP, 45th CIRP Conference on Manufacturing Systems 2012, Volume 3, 2012, pp. 293–298, DOI: http://dx.doi.org/10.1016/j.procir.2012.07.051
11. D. Mourtzis, M. Doukas, F. Psarommatis, "Environmental impact of centralised and decentralized production networks in the era of personalisation", Lecture Notes in Production Engineering 1, Springer, DOI: http://dx.doi.org/10.1007/978-3-642-30749-2, pp. 396-408 (2012)
12. D. Mourtzis, M. Doukas, G. Michalos, F. Psarommatis, "A Web-based Platform for Distributed Mass Product Customization: Conceptual Design", (DET 2011), ISBN 978-960-88104-2-6, 7th International Conference on Digital Enterprise Technology, Athens, Greece, pp. 604-613 (2011)

Chapters in Books

1. D. Mourtzis, M. Doukas, F. Psarommatis, "Environmental impact of centralised and decentralized production networks in the era of personalisation", K. Windt (ed.), Robust Manufacturing Control, Lecture Notes in Production Engineering, DOI: 10.1007/978-3-

642-30749-2_27, Springer-Verlag Berlin Heidelberg (2013), http://www.springer.com/engineering/production+engineering/book/978-3-642-30748-5

2. D. Mourtzis, M. Doukas, F. Psarommatis, N. Panopoulos, “Design of Multi-Stage Manufacturing Networks for Personalized Products Using Metaheuristics”, A. Azevedo (ed.), Advances in Sustainable and Competitive Manufacturing Systems, Lecture Notes in Mechanical Engineering 2013, Springer, pp. 1263-1276, DOI: http://dx.doi.org/10.1007/978-3-319-00557-7_103, http://www.springer.com/engineering/production+engineering/book/978-3-319-00556-0
3. D. Mourtzis, M. Doukas, F. Psarommatis, “Simulation-based design of production networks for manufacturing of personalised products”, Proceeding of the IFIP WG 5.7 International Conference (IFIP AICT 397), APMS 2012, Advances in Production Management Systems, Part I, 24-26/09-2012, Rhodes, Greece, ISSN 1868-4238, http://www.springer.com/computer/information+systems+and+applications/book/978-3-642-40351-4

9 References

[1] A. Khan and K. Turowski, “A survey of current challenges in manufacturing industry and preparation for industry 4.0,” 2016, doi: 10.1007/978-3-319-33609-1_2.

[2] L. Zhou, S. M. Gupta, Y. Kinoshita, and T. Yamada, “Pricing Decision Models for Remanufactured Short-Life Cycle Technology Products with Generation Consideration,” *Procedia CIRP*, vol. 61, pp. 195–200, 2017, doi: 10.1016/j.procir.2016.11.208.

[3] J. F. Halpin, “Zero defects: a new dimension in quality assurance,” *McGraw-Hill*, 1966.

[4] G. Marodin, A. G. Frank, G. L. Tortorella, and T. Netland, “Lean product development and lean manufacturing: Testing moderation effects,” *Int. J. Prod. Econ.*, vol. 203, pp. 301–310, Sep. 2018, doi: 10.1016/J.IJPE.2018.07.009.

[5] C. J. C. Jabbour, A. B. L. De Sousa Jabbour, K. Govindan, A. A. Teixeira, and W. R. De Souza Freitas, “Environmental management and operational performance in automotive companies in Brazil: The role of human resource management and lean manufacturing,” *J. Clean. Prod.*, vol. 47, pp. 129–140, May 2013, doi: 10.1016/j.jclepro.2012.07.010.

[6] Y. Lu, “Industry 4.0: A survey on technologies, applications and open research issues,” *J. Ind. Inf. Integr.*, vol. 6, pp. 1–10, Jun. 2017, doi: 10.1016/j.jii.2017.04.005.

[7] D. Ouelhadj and S. Petrovic, “A survey of dynamic scheduling in manufacturing systems,” *J. Sched.*, vol. 12, no. 4, pp. 417–431, 2009, doi: 10.1007/s10951-008-0090-8.

[8] R. Barták and M. Vlk, “Hierarchical task model for resource failure recovery in production scheduling,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2017, vol. 10061 LNAI, pp. 362–378, doi: 10.1007/978-3-319-62434-1_30.

[9] A. Jacob, K. Windhuber, D. Ranke, and G. Lanza, “Planning, Evaluation and Optimization of Product Design and Manufacturing Technology Chains for New Product and Production Technologies on the Example of Additive Manufacturing,” *Procedia CIRP*, vol. 70, pp. 108–113, Jan. 2018, doi: 10.1016/J.PROCIR.2018.02.049.

[10] J. Kletti and J. Schumacher, “Short Interval Technology (SIT),” in *Die perfekte Produktion*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 9–33.

[11] F. Eger, C. Reiff, B. Brantl, M. Colledani, and A. Verl, “Correlation analysis methods in multi-stage production systems for reaching zero-defect manufacturing,” *Procedia CIRP*, vol. 72, pp. 635–640, 2018, doi: 10.1016/j.procir.2018.03.163.

[12] R. Teti, “Advanced IT Methods of Signal Processing and Decision Making for Zero Defect Manufacturing in Machining,” *Procedia CIRP*, vol. 28, pp. 3–15, Jan. 2015, doi: 10.1016/J.PROCIR.2015.04.003.

[13] W. Utz and M. Lee, “Industrial Business Process Management Using Adonis Towards a Modular Business Process Modelling Method for Zero-Defect-Manufacturing,” *2017 Int. Conf. Ind. Eng. Manag. Sci. Appl.*, pp. 1–5, 2017, doi: 10.1109/ICIMSA.2017.7985590.

[14] A. Bufardi, O. Akten, M. Arif, P. Xirouchakis, and R. Perez, “Towards zero-defect manufacturing with a combined online - Offline fuzzy-nets approach in wire electrical discharge machining,” *WSEAS Trans. Environ. Dev.*, vol. 13, pp. 401–409, 2017.

[15] H. Raabe, O. Myklebust, and R. Eleftheriadis, “Vision Based Quality Control and Maintenance in High Volume Production by Use of Zero Defect Strategies,” in *Lecture Notes in Electrical Engineering*, 2018, vol. 451, pp. 405–412, doi: 10.1007/978-981-

10-5768-7_43.

[16] F. Ju, J. Li, G. Xiao, and J. Arinez, "Quality flow model in automotive paint shops," *Int. J. Prod. Res.*, vol. 51, no. 21, pp. 6470–6483, 2013, doi: 10.1080/00207543.2013.824629.

[17] A. Ladj, F. Benbouzid-Si Tayeb, C. Varnier, A. A. Dridi, and N. Selmane, "A Hybrid of Variable Neighbor Search and Fuzzy Logic for the permutation flowshop scheduling problem with predictive maintenance," *Procedia Comput. Sci.*, vol. 112, pp. 663–672, Jan. 2017, doi: 10.1016/J.PROCS.2017.08.120.

[18] Y. Ma and C. Zuo, "A survey of scheduling with deterministic machine availability constraints," *Comput. Ind. Eng.*, vol. 58, no. 2, pp. 199–211, Mar. 2010, doi: 10.1016/J.CIE.2009.04.014.

[19] I. R. Uhlmann and E. M. Frazzon, "Production rescheduling review: Opportunities for industrial integration and practical applications," *Journal of Manufacturing Systems*. 2018, doi: 10.1016/j.jmsy.2018.10.004.

[20] O. Cardin, D. Trentesaux, A. Thomas, P. Castagna, T. Berger, and H. Bril El-Haouzi, "Coupling predictive scheduling and reactive control in manufacturing hybrid control architectures: state of the art and future challenges," *J. Intell. Manuf.*, 2017, doi: 10.1007/s10845-015-1139-0.

[21] J. Lindström *et al.*, "Towards intelligent and sustainable production systems with a zero-defect manufacturing approach in an Industry4.0 context," in *Procedia CIRP*, Jan. 2019, vol. 81, pp. 880–885, doi: 10.1016/j.procir.2019.03.218.

[22] D. Gillen, "Benchmarking and Performance Measurement: The Role in Quality Management," in *Handbook of Logistics and Supply-Chain Management*, Emerald Group Publishing Limited, 2017, pp. 325–338.

[23] Anne Francine Martins, R. Costa Affonso, S. Tamayo, S. Lamouri, and C. Baldy Ngayo, "Relationships between national culture and Lean Management: A literature Review," in *2015 International Conference on Industrial Engineering and Systems Management (IESM)*, Oct. 2015, pp. 352–361, doi: 10.1109/IESM.2015.7380183.

[24] A. Wilson, V. Zeithaml, M. J. Bitner, and D. Gremler, *Services Marketing: Integrating Customer Focus Across the Firm*, 3rd Europe. UK, 2016.

[25] J. Singh and H. Singh, "Continuous improvement approach: State-of-art review and future implications," *International Journal of Lean Six Sigma*, vol. 3, no. 2. Emerald Group Publishing Limited, pp. 88–111, Jun. 01, 2012, doi: 10.1108/20401461211243694.

[26] P. Kumar, J. Maiti, and A. Gunasekaran, "Impact of quality management systems on firm performance," *Int. J. Qual. Reliab. Manag.*, vol. 35, no. 5, pp. 1034–1059, May 2018, doi: 10.1108/IJQRM-02-2017-0030.

[27] F. Psarommatis, S. Prouvost, G. May, and D. Kiritsis, "Product Quality Improvement Policies in Industry 4 . 0 : Characteristics , Enabling Factors , Barriers , and Evolution Toward Zero Defect Manufacturing," *Front. Comput. Sci.*, vol. 2, no. August, pp. 1–15, 2020, doi: 10.3389/fcomp.2020.00026.

[28] O. Myklebust, "Zero Defect Manufacturing: A Product and Plant Oriented Lifecycle Approach," *Procedia CIRP*, vol. 12, pp. 246–251, Jan. 2013, doi: 10.1016/J.PROCIR.2013.09.043.

[29] F. Eger *et al.*, "Zero Defect Manufacturing Strategies for Reduction of Scrap and Inspection Effort in Multi-stage Production Systems," *Procedia CIRP*, vol. 67, pp. 368–373, Jan. 2018, doi: 10.1016/J.PROCIR.2017.12.228.

[30] P. A. Dreyfus and D. Kyritsis, "A framework based on predictive maintenance, zero-defect manufacturing and scheduling under uncertainty tools, to optimize production capacities of high-end quality products," in *IFIP Advances in Information and Communication Technology*, 2018, vol. 536, pp. 296–303, doi: 10.1007/978-3-319-99707-0_37.

[31] J. Wang, L. Tan, and S. Lai, "Apply Neutron Radiation and X-Ray Images Fusion for Defect Detection in Manufacturing," *Adv. Mech. Electron. Eng. LNEE*, vol. 178, pp. 507–512, 2013.

[32] "Assistant Secretary of Defense (Manpower Installations and Logistics) Washington DC: Guide To Zero Defects," *Qual. Reliab. Assur. Handb.*, 1965.

[33] F. Psarommatis, G. May, P.-A. Dreyfus, and D. Kiritsis, "Zero defect manufacturing: state-of-the-art review, shortcomings and future directions in research," *Int. J. Prod. Res.*, vol. 7543, pp. 1–17, 2020, doi: 10.1080/00207543.2019.1605228.

[34] X. Bai, Y. Fang, W. Lin, L. Wang, and B. F. Ju, "Saliency-based defect detection in industrial images by using phase spectrum," *IEEE Trans. Ind. Informatics*, vol. 10, no. 4, pp. 2135–2145, 2014, doi: 10.1109/TII.2014.2359416.

[35] W. Dong, S. Liu, Z. Fang, X. Yang, Q. Hu, and L. Tao, "Control and optimization of quality cost based on discrete grey forecasting model," Oct. 2017, pp. 149–149, doi: 10.1109/gsis.2017.8077690.

[36] R. J. Eleftheriadis and O. Myklebust, "A guideline of quality steps towards zero defect manufacturing in industry," *2016 Int. Conf. Ind. Eng. Oper. Manag.*, pp. 332–340, 2016.

[37] I. S. Stephan Thangaiah, V. Sharma, and V. N. Sundharam, "Analysing of customer feedback on critical quality parameters to improve productivity in manufacturing – A case study," *Int. J. Product. Qual. Manag.*, vol. 23, no. 3, pp. 349–368, 2018, doi: 10.1504/IJPQM.2018.089805.

[38] D. Mourtzis, E. Vlachou, and N. Milas, "Industrial Big Data as a Result of IoT Adoption in Manufacturing," in *Procedia CIRP*, Jan. 2016, vol. 55, pp. 290–295, doi: 10.1016/j.procir.2016.07.038.

[39] C. F. Chien, S. C. Hsu, and Y. J. Chen, "A system for online detection and classification of wafer bin map defect patterns for manufacturing intelligence," *Int. J. Prod. Res.*, vol. 51, no. 8, pp. 2324–2338, 2013, doi: 10.1080/00207543.2012.737943.

[40] C. F. Kuo, C. T. M. Hsu, C. H. Fang, S. M. Chao, and Y. De Lin, "Automatic defect inspection system of colour filters using Taguchi-based neural network," *Int. J. Prod. Res.*, vol. 51, no. 5, pp. 1464–1476, 2013, doi: 10.1080/00207543.2012.695877.

[41] G. Choi, S. H. Kim, C. Ha, and S. J. Bae, "Multi-step ART1 algorithm for recognition of defect patterns on semiconductor wafers," *Int. J. Prod. Res.*, vol. 50, no. 12, pp. 3274–3287, 2012, doi: 10.1080/00207543.2011.574502.

[42] H. Ahuett-Garza and T. Kurfess, "A brief discussion on the trends of habilitating technologies for Industry 4.0 and Smart manufacturing," *Manuf. Lett.*, vol. 15, pp. 60–63, Jan. 2018, doi: 10.1016/J.MFGLET.2018.02.011.

[43] F. Psarommatis and D. Kiritsis, "A Scheduling Tool for Achieving Zero Defect Manufacturing (ZDM): A Conceptual Framework," Springer, Cham, 2018, pp. 271–278.

[44] H. Z. Jia, J. Y. H. Fuh, A. Y. C. Nee, and Y. F. Zhang, "Integration of genetic algorithm and Gantt chart for job shop scheduling in distributed manufacturing systems," *Comput. Ind. Eng.*, vol. 53, no. 2, pp. 313–320, Sep. 2007, doi: 10.1016/j.cie.2007.06.024.

[45] J. Behnamian and S. M. T. Fatemi Ghomi, "A survey of multi-factory scheduling," *J. Intell. Manuf.*, vol. 27, no. 1, pp. 231–249, 2016, doi: 10.1007/s10845-014-0890-y.

[46] K. B. Kahn, S. E. Kay, R. Slotegraaf, S. Uban, and Product Development & Management Association, *The PDMA handbook of new product development*. Wiley, 2013.

[47] M. L. Pinedo, *Scheduling Theory, Algorithms, and Systems*, Fifth edit. New York: Springer International Publishing, 2016.

[48] T. Gonzalez and S. Sahni, "Flowshop and Jobshop Schedules: Complexity and Approximation," *Oper. Res.*, vol. 26, no. 1, pp. 36–52, 1978, doi: 10.1287/opre.26.1.36.

[49] A. Giret, D. Trentesaux, and V. Prabhu, "Sustainability in manufacturing operations scheduling: A state of the art review," *J. Manuf. Syst.*, vol. 37, pp. 126–140, 2015, doi: 10.1016/j.jmsy.2015.08.002.

[50] B. Naderi, R. Ruiz, and M. Zandieh, "Algorithms for a realistic variant of flowshop scheduling," *Comput. Oper. Res.*, vol. 37, no. 2, pp. 236–246, Feb. 2010, doi: 10.1016/j.cor.2009.04.017.

[51] M. Nawaz, E. E. Enscore, and I. Ham, "A heuristic algorithm for the m-machine, n-job flow-shop sequencing problem," *Omega*, vol. 11, no. 1, pp. 91–95, 1983, doi:

10.1016/0305-0483(83)90088-9.

[52] J. Gao and R. Chen, “A hybrid genetic algorithm for the distributed permutation flowshop scheduling problem,” *Int. J. Comput. Intell. Syst.*, vol. 4, no. 4, pp. 497–508, 2011, doi: 10.1080/18756891.2011.9727808.

[53] E. Rashidi, M. Jahandar, and M. Zandieh, “An improved hybrid multi-objective parallel genetic algorithm for hybrid flow shop scheduling with unrelated parallel machines,” *Int. J. Adv. Manuf. Technol.*, vol. 49, no. 9–12, pp. 1129–1139, Aug. 2010, doi: 10.1007/s00170-009-2475-z.

[54] J. Gao, R. Chen, and W. Deng, “An efficient tabu search algorithm for the distributed permutation flowshop scheduling problem,” *Int. J. Prod. Res.*, vol. 51, no. 3, pp. 641–651, Feb. 2013, doi: 10.1080/00207543.2011.644819.

[55] D. Cao and M. Chen, “Parallel flowshop scheduling using Tabu search,” *Int. J. Prod. Res.*, vol. 41, no. 13, pp. 3059–3073, Sep. 2003, doi: 10.1080/0020754031000106443.

[56] H. Liu and L. Gao, “A discrete electromagnetism-like mechanism algorithm for solving distributed permutation flowshop scheduling problem,” in *Proceedings - 2010 International Conference on Manufacturing Automation, ICMA 2010*, 2010, pp. 156–163, doi: 10.1109/ICMA.2010.17.

[57] Y. Xu, L. Wang, S. Wang, and M. Liu, “An effective hybrid immune algorithm for solving the distributed permutation flow-shop scheduling problem,” *Eng. Optim.*, vol. 46, no. 9, pp. 1269–1283, Sep. 2014, doi: 10.1080/0305215X.2013.827673.

[58] H. Bargaoui, O. Belkahla Driss, and K. Ghédira, “A novel chemical reaction optimization for the distributed permutation flowshop scheduling problem with makespan criterion,” *Comput. Ind. Eng.*, vol. 111, pp. 239–250, Sep. 2017, doi: 10.1016/j.cie.2017.07.020.

[59] R. Ruiz, Q. K. Pan, and B. Naderi, “Iterated Greedy methods for the distributed permutation flowshop scheduling problem,” *Omega (United Kingdom)*, vol. 83, pp. 213–222, Mar. 2019, doi: 10.1016/j.omega.2018.03.004.

[60] C. Low, “Simulated annealing heuristic for flow shop scheduling problems with unrelated parallel machines,” *Comput. Oper. Res.*, vol. 32, no. 8, pp. 2013–2025, Aug. 2005, doi: 10.1016/j.cor.2004.01.003.

[61] M. M. K. and P. T., “A bat algorithm for realistic hybrid flowshop scheduling problems to minimize makespan and mean flow time,” *ICTACT J. Soft Comput.*, vol. 03, no. 01, pp. 428–433, Oct. 2012, doi: 10.21917/ijsc.2012.0066.

[62] M. T. Jensen, “Generating robust and flexible job shop schedules using genetic algorithms,” *IEEE Trans. Evol. Comput.*, vol. 7, no. 3, pp. 275–288, 2003, doi: 10.1109/TEVC.2003.810067.

[63] M. Gholami and M. Zandieh, “Integrating simulation and genetic algorithm to schedule a dynamic flexible job shop,” *J. Intell. Manuf.*, vol. 20, no. 4, pp. 481–498, 2009, doi: 10.1007/s10845-008-0150-0.

[64] M. E. Salveson, “On a Quantitative Method in Production Planning and Scheduling,” *Econometrica*, vol. 20, no. 4, p. 554, Oct. 1952, doi: 10.2307/1907643.

[65] K. Katragjini, E. Vallada, and R. Ruiz, “Flow shop rescheduling under different types of disruption,” *Int. J. Prod. Res.*, vol. 51, no. 3, pp. 780–797, Feb. 2013, doi: 10.1080/00207543.2012.666856.

[66] C. Pascal and D. Panescu, “On Increasing Adaptability of Holonic Systems,” *Stud. Comput. Intell.*, vol. 762, pp. 229–241, 2018, doi: 10.1007/978-3-319-73751-5_17.

[67] C. A. Marín *et al.*, “A conceptual architecture based on intelligent services for manufacturing support systems,” in *Proceedings - 2013 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2013*, 2013, pp. 4749–4754, doi: 10.1109/SMC.2013.808.

[68] J. M. Novas, J. Van Belle, B. Saint Germain, and P. Valckenaers, “A collaborative framework between a scheduling system and a holonic manufacturing execution system,” in *Studies in Computational Intelligence*, 2013, vol. 472, pp. 3–17, doi: 10.1007/978-3-642-35852-4_1.

[69] R. Rangsaritratsamee, W. G. Ferrell, and M. B. Kurz, “Dynamic rescheduling that

simultaneously considers efficiency and stability," *Comput. Ind. Eng.*, vol. 46, no. 1, pp. 1–15, Mar. 2004, doi: 10.1016/j.cie.2003.09.007.

[70] W. P. Wang and Z. Chen, "A neuro-fuzzy based forecasting approach for rush order control applications," *Expert Syst. Appl.*, vol. 35, no. 1–2, pp. 223–234, Jul. 2008, doi: 10.1016/j.eswa.2007.06.038.

[71] H. Li, Z. Li, L. X. Li, and B. Hu, "A production rescheduling expert simulation system," *Eur. J. Oper. Res.*, vol. 124, no. 2, pp. 283–293, Jul. 2000, doi: 10.1016/S0377-2217(99)00381-1.

[72] Y. H. Hu, J. Q. Yan, F. F. Ye, and J. H. Yu, "Flow shop rescheduling problem under rush orders," *J. Zhejiang Univ. Sci.*, vol. 6 A, no. 10, pp. 1040–1046, Oct. 2005, doi: 10.1631/jzus.2005.A1040.

[73] Y. Hu, F. Ye, Z. Guo, and Z. Zheng, "Multi-shop rescheduling problem under rush orders," in *5th International Conference on Natural Computation, ICNC 2009*, 2009, vol. 3, pp. 40–44, doi: 10.1109/ICNC.2009.330.

[74] I. Pereira and A. Madureira, "Self-Optimization module for Scheduling using Case-based Reasoning," *Appl. Soft Comput. J.*, vol. 13, no. 3, pp. 1419–1432, Mar. 2013, doi: 10.1016/j.asoc.2012.02.009.

[75] Y. H. Lee, S. R. T. Kumara, and K. Chatterjee, "Multiagent based dynamic resource scheduling for distributed multiple projects using a market mechanism," *J. Intell. Manuf.*, vol. 14, no. 5, pp. 471–484, Oct. 2003, doi: 10.1023/A:1025753309346.

[76] D. Ouelhadj, P. I. Cowling, and S. Petrovic, "Utility and stability measures for agent-based dynamic scheduling of steel continuous casting," in *Proceedings - IEEE International Conference on Robotics and Automation*, 2003, vol. 1, pp. 175–180, doi: 10.1109/robot.2003.1241592.

[77] A. Madureira, J. Santos, and N. Gomes, "Hybrid Multi-agent System for Cooperative Dynamic Scheduling Through Meta-Heuristics," Apr. 2008, pp. 9–14, doi: 10.1109/isda.2007.50.

[78] A. Madureira and I. Pereira, "Intelligent Bio-Inspired system for manufacturing scheduling under uncertainties," in *2010 10th International Conference on Hybrid Intelligent Systems, HIS 2010*, 2010, pp. 109–112, doi: 10.1109/HIS.2010.5600084.

[79] A. Madureira, J. M. Santos, S. Gomes, B. Cunha, J. P. Pereira, and I. Pereira, "Manufacturing rush orders rescheduling: A supervised learning approach," in *2014 6th World Congress on Nature and Biologically Inspired Computing, NaBIC 2014*, Oct. 2014, pp. 299–304, doi: 10.1109/NaBIC.2014.6921895.

[80] X. He, S. Dong, and N. Zhao, "Research on rush order insertion rescheduling problem under hybrid flow shop based on NSGA-III," *Int. J. Prod. Res.*, 2019, doi: 10.1080/00207543.2019.1613581.

[81] D. J. Wang, F. Liu, J. J. Wang, and Y. Z. Wang, "Integrated rescheduling and preventive maintenance for arrival of new jobs through evolutionary multi-objective optimization," *Soft Comput.*, vol. 20, no. 4, pp. 1635–1652, Apr. 2016, doi: 10.1007/s00500-015-1615-7.

[82] O. Myklebust, "Zero defect manufacturing: A product and plant oriented lifecycle approach," *Procedia CIRP*, vol. 12, pp. 246–251, 2013, doi: 10.1016/j.procir.2013.09.043.

[83] H. Zörrer, R. Steringer, S. Zambal, and C. Eitzinger, "Using business analytics for decision support in zero defect manufacturing of composite parts in the aerospace industry," *IFAC-PapersOnLine*, 2019, doi: 10.1016/j.ifacol.2019.11.405.

[84] E. Kucharska, K. Grobler-Dębska, and K. Rączka, "Algebraic-logical meta-model based approach for scheduling manufacturing problem with defects removal," *Adv. Mech. Eng.*, vol. 9, no. 4, pp. 1–18, 2017, doi: 10.1177/1687814017692291.

[85] B. J. Joo, T. J. Chua, and T. X. Cai, "Dispatching rule-based scheduling algorithm for a dynamic flexible flow shop with time-dependent process defect rate," 2016.

[86] G. Levitin, M. Finkelstein, and H. Z. Huang, "Scheduling of imperfect inspections for reliability critical systems with shock-driven defects and delayed failures," *Reliab. Eng. Syst. Saf.*, 2019, doi: 10.1016/j.ress.2019.04.016.

[87] Y. Xu, X. Han, M. Yang, M. Wang, X. Zhu, and Y. Zhang, "Condition-based midterm maintenance scheduling with rescheduling strategy," *Int. J. Electr. Power Energy Syst.*, 2020, doi: 10.1016/j.ijepes.2019.105796.

[88] R. Barták and M. Vlk, "Hierarchical task model for resource failure recovery in production scheduling," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10061 LNAI, pp. 362–378, 2017, doi: 10.1007/978-3-319-62434-1_30.

[89] O. Battaïa, L. Sanmartin, and C. Pralet, "Dealing with disruptions in low-volume manufacturing: A constraint programming approach," *Procedia CIRP*, vol. 81, pp. 1372–1375, 2019, doi: 10.1016/j.procir.2019.04.030.

[90] G. Mejía and D. Lefebvre, "Robust scheduling of flexible manufacturing systems with unreliable operations and resources," *Int. J. Prod. Res.*, 2019, doi: 10.1080/00207543.2019.1682706.

[91] F. Psarommatis, A. Gharaei, and D. Kiritsis, "Identification of the critical reaction times for re-scheduling flexible job shops for different types of unexpected events," *Procedia CIRP*, vol. 93, pp. 903–908, Jan. 2020, doi: 10.1016/j.procir.2020.03.038.

[92] D. Rahmani and M. Heydari, "Robust and stable flow shop scheduling with unexpected arrivals of new jobs and uncertain processing times," *J. Manuf. Syst.*, vol. 33, no. 1, pp. 84–92, Jan. 2014, doi: 10.1016/j.jmsy.2013.03.004.

[93] D. Bauer, T. Bauernhansl, and A. Sauer, "Enhanced classification of events for manufacturing companies in supply networks," in *Procedia CIRP*, Jan. 2019, vol. 81, pp. 87–92, doi: 10.1016/j.procir.2019.03.016.

[94] H. Stadtler, "Supply Chain Management: An Overview," 2015, pp. 3–28.

[95] D. Rahmani and R. Ramezanian, "A stable reactive approach in dynamic flexible flow shop scheduling with unexpected disruptions: A case study," *Comput. Ind. Eng.*, vol. 98, pp. 360–372, 2016, doi: 10.1016/j.cie.2016.06.018.

[96] K. Z. Gao, P. N. Suganthan, Q. K. Pan, M. F. Tasgetiren, and A. Sadollah, "Artificial bee colony algorithm for scheduling and rescheduling fuzzy flexible job shop problem with new job insertion," *Knowledge-Based Syst.*, vol. 109, pp. 1–16, 2016, doi: 10.1016/j.knosys.2016.06.014.

[97] L. Liu, "Outsourcing and rescheduling for a two-machine flow shop with the disruption of new arriving jobs: A hybrid variable neighborhood search algorithm," *Comput. Ind. Eng.*, 2019, doi: 10.1016/j.cie.2019.02.015.

[98] N. C. O. Silva, C. T. Scarpin, A. Ruiz, and J. E. Pécora, "Rescheduling production on identical parallel machines upon new jobs arrivals," *IFAC-PapersOnLine*, 2019, doi: 10.1016/j.ifacol.2019.11.586.

[99] L. Liu and H. Zhou, "On the identical parallel-machine rescheduling with job rework disruption," *Comput. Ind. Eng.*, 2013, doi: 10.1016/j.cie.2013.02.018.

[100] S. K. Moghaddam and K. Saitou, "On optimal dynamic pegging in rescheduling for new order arrival," *Comput. Ind. Eng.*, 2019, doi: 10.1016/j.cie.2019.07.012.

[101] R. Buddala and S. S. Mahapatra, "Two-stage teaching-learning-based optimization method for flexible job-shop scheduling under machine breakdown," *Int. J. Adv. Manuf. Technol.*, 2019, doi: 10.1007/s00170-018-2805-0.

[102] F. Qiao, Y. M. Ma, M. C. Zhou, and Q. Di Wu, "A Novel Rescheduling Method for Dynamic Semiconductor Manufacturing Systems," *IEEE Trans. Syst. Man, Cybern. Syst.*, pp. 1–11, 2018, doi: 10.1109/TSMC.2017.2782009.

[103] N. Al-Hinai and T. Y. Elmekkawy, "Robust and stable flexible job shop scheduling with random machine breakdowns using a hybrid genetic algorithm," *Int. J. Prod. Econ.*, vol. 132, no. 2, pp. 279–291, 2011, doi: 10.1016/j.ijpe.2011.04.020.

[104] Y. Yin, T. C. E. Cheng, and D. J. Wang, "Rescheduling on identical parallel machines with machine disruptions to minimize total completion time," *Eur. J. Oper. Res.*, 2016, doi: 10.1016/j.ejor.2016.01.045.

[105] S. Ferrer, G. Nicolò, M. A. Salido, A. Giret, and F. Barber, "Dynamic rescheduling in energy-aware unrelated parallel machine problems," 2018, doi: 10.1007/978-3-319-99707-0_29.

[106] M. Nouiri, A. Bekrar, and D. Trentesaux, “Towards Energy Efficient Scheduling and Rescheduling for Dynamic Flexible Job Shop Problem,” *IFAC-PapersOnLine*, 2018, doi: 10.1016/j.ifacol.2018.08.357.

[107] Z. Li, “Multi-task scheduling optimization in shop floor based on uncertainty theory algorithm,” *Acad. J. Manuf. Eng.*, 2019.

[108] G. E. Modoni, A. Trombetta, M. Veniero, M. Sacco, and D. Mourtzis, “An event-driven integrative framework enabling information notification among manufacturing resources,” *Int. J. Comput. Integr. Manuf.*, vol. 32, no. 3, pp. 241–252, Mar. 2019, doi: 10.1080/0951192X.2019.1571232.

[109] T. Vafeiadis, D. Ioannidis, C. Ziazios, I. N. Metaxa, and D. Tzovaras, “Towards Robust Early Stage Data Knowledge-based Inference Engine to Support Zero-defect Strategies in Manufacturing Environment,” *Procedia Manuf.*, vol. 11, pp. 679–685, Jan. 2017, doi: 10.1016/J.PROMFG.2017.07.167.

[110] D. Mourtzis, E. Vlachou, N. Milas, N. Tapoglou, and J. Mehnen, “A cloud-based, knowledge-enriched framework for increasing machining efficiency based on machine tool monitoring,” *Proc. Inst. Mech. Eng. Part B J. Eng. Manuf.*, vol. 233, no. 1, pp. 278–292, Jan. 2019, doi: 10.1177/0954405417716727.

[111] A. Felsberger, B. Oberegger, and G. Reiner, “A Review of Decisions Support Systems for Manufacturing Systems,” Feb. 2017, doi: 10.5281/ZENODO.495120.

[112] K. Gandhi, B. Schmidt, and A. H. C. Ng, “Towards data mining based decision support in manufacturing maintenance,” *Procedia CIRP*, vol. 72, pp. 261–265, Jan. 2018, doi: 10.1016/J.PROCIR.2018.03.076.

[113] B. M. Al-Alawi and A. D. Coker, “Multi-criteria decision support system with negotiation process for vehicle technology selection,” *Energy*, vol. 157, pp. 278–296, Aug. 2018, doi: 10.1016/J.ENERGY.2018.05.142.

[114] M. Yazdani, P. Zarate, A. Coulibaly, and E. K. Zavadskas, “A group decision making support system in logistics and supply chain management,” *Expert Syst. Appl.*, vol. 88, pp. 376–392, Dec. 2017, doi: 10.1016/J.ESWA.2017.07.014.

[115] C. Schuh, J. S. de Bruin, and W. Seeling, “Clinical decision support systems at the Vienna General Hospital using Arden Syntax: Design, implementation, and integration,” *Artif. Intell. Med.*, vol. 92, pp. 24–33, Nov. 2018, doi: 10.1016/J.ARTMED.2015.11.002.

[116] D. Prasad and S. Ratna, “Decision support systems in the metal casting industry: An academic review of research articles,” *Mater. Today Proc.*, vol. 5, no. 1, pp. 1298–1312, Jan. 2018, doi: 10.1016/J.MATPR.2017.11.215.

[117] B. Mrzygłód *et al.*, “Durability analysis of forging tools after different variants of surface treatment using a decision-support system based on artificial neural networks,” *Arch. Civ. Mech. Eng.*, vol. 18, no. 4, pp. 1079–1091, Sep. 2018, doi: 10.1016/J.ACME.2018.02.010.

[118] M. B. Orm and J. Jeunet, “Time Cost Quality Trade-off Problems: A survey exploring the assessment of quality,” *Comput. Ind. Eng.*, vol. 118, pp. 319–328, 2018, doi: https://doi.org/10.1016/j.cie.2018.01.012.

[119] M. Jafarian and M. Bashiri, “Supply chain dynamic configuration as a result of new product development,” *Appl. Math. Model.*, vol. 38, no. 3, pp. 1133–1146, Feb. 2014, doi: 10.1016/J.APM.2013.08.025.

[120] Y. Tanimizu *et al.*, “Computational Evaluation of Order Selection Methods in Dynamic Supply Chains,” *Procedia CIRP*, vol. 3, pp. 281–286, Jan. 2012, doi: 10.1016/J.PROCIR.2012.07.049.

[121] J. Quigley, L. Walls, G. Demirel, B. L. MacCarthy, and M. Parsa, “Supplier quality improvement: The value of information under uncertainty,” *Eur. J. Oper. Res.*, vol. 264, no. 3, pp. 932–947, Feb. 2018, doi: 10.1016/J.EJOR.2017.05.044.

[122] M. Schröder, B. Falk, and R. Schmitt, “Failure Classification and Analysis for Technical Products,” *Procedia CIRP*, vol. 51, no. chapter 4, pp. 116–121, 2016, doi: 10.1016/j.procir.2016.06.103.

[123] L. Fumagalli, M. Macchi, and A. Giacomin, “Orchestration of preventive maintenance

interventions," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 13976–13981, 2017, doi: 10.1016/j.ifacol.2017.08.2417.

[124] I. Petukhov and L. Steshina, "Training Personalization for Operators of Complex Equipment," *Procedia - Soc. Behav. Sci.*, vol. 186, pp. 1240–1247, 2015, doi: 10.1016/j.sbspro.2015.04.067.

[125] S. Ferretti, D. Caputo, M. Penza, and D. M. D'Addona, "Monitoring systems for zero defect manufacturing," *Procedia CIRP*, vol. 12, pp. 258–263, 2013, doi: 10.1016/j.procir.2013.09.045.

[126] W. Cao, G. Dong, Y. B. Xie, and Z. Peng, "Prediction of wear trend of engines via on-line wear debris monitoring," *Tribol. Int.*, vol. 120, no. October 2017, pp. 510–519, 2018, doi: 10.1016/j.triboint.2018.01.015.

[127] S. Moustakidis, A. Anagnostis, P. Karlsson, and K. Hrissagis, "Non-destructive inspection of aircraft composite materials using triple IR imaging," *IFAC-PapersOnLine*, vol. 49, no. 28, pp. 291–296, 2016, doi: 10.1016/j.ifacol.2016.11.050.

[128] M. Vandebroek, L. Lan, and K. Knapen, "An Experimental Diagnostic Procedure to Identify the Source of Defects in Multi-Stage and Multi-Component Production Processes," *J. Qual. Technol.*, vol. 48, no. 3, pp. 213–226, Jul. 2016, doi: 10.1080/00224065.2016.11918162.

[129] M. A. Farooq, R. Kirchain, H. Novoa, and A. Araujo, "Cost of quality: Evaluating cost-quality trade-offs for inspection strategies of manufacturing processes," *Int. J. Prod. Econ.*, vol. 188, no. November 2016, pp. 156–166, 2017, doi: 10.1016/j.ijpe.2017.03.019.

[130] B. Sarkar and S. Saren, "Product inspection policy for an imperfect production system with inspection errors and warranty cost," *Eur. J. Oper. Res.*, vol. 248, no. 1, pp. 263–271, Jan. 2016, doi: 10.1016/J.EJOR.2015.06.021.

[131] G. Levitin, L. Xing, and H.-Z. Huang, "Cost effective scheduling of imperfect inspections in systems with hidden failures and rescue possibility," *Appl. Math. Model.*, vol. 68, pp. 662–674, 2019, doi: https://doi.org/10.1016/j.apm.2018.12.001.

[132] V. Hirsch, P. Reimann, O. Kirn, and B. Mitschang, "Analytical Approach to Support Fault Diagnosis and Quality Control in End-Of-Line Testing," *Procedia CIRP*, vol. 72, pp. 1333–1338, 2018, doi: 10.1016/j.procir.2018.03.024.

[133] D. Soban, D. Thornhill, S. Salunkhe, and A. Long, "Visual Analytics as an Enabler for Manufacturing Process Decision-making," *Procedia CIRP*, vol. 56, pp. 209–214, 2016, doi: 10.1016/j.procir.2016.10.056.

[134] M. Gewohn, T. Usländer, and J. Beyerer, "Production quality control through a user-oriented and characteristic-based quality visualization model," *Procedia CIRP*, vol. 79, no. i, pp. 511–516, 2019, doi: 10.1016/j.procir.2019.02.105.

[135] A. U. Haq and D. Djurdjanovic, "Virtual Metrology Concept for Predicting Defect Levels in Semiconductor Manufacturing," *Procedia CIRP*, vol. 57, pp. 580–584, 2016, doi: 10.1016/j.procir.2016.11.100.

[136] J. Lindström, H. Larsson, M. Jonsson, and E. Lejon, "Towards Intelligent and Sustainable Production: Combining and Integrating Online Predictive Maintenance and Continuous Quality Control," *Procedia CIRP*, vol. 63, pp. 443–448, 2017, doi: 10.1016/j.procir.2017.03.099.

[137] R. Teti, "Advanced IT methods of signal processing and decision making for zero defect manufacturing in machining," *Procedia CIRP*, vol. 28, pp. 3–15, 2015, doi: 10.1016/j.procir.2015.04.003.

[138] K. K. Castillo-Villar, N. R. Smith, and J. L. Simonton, "A model for supply chain design considering the cost of quality," *Appl. Math. Model.*, vol. 36, no. 12, pp. 5920–5935, 2012, doi: 10.1016/j.apm.2012.01.046.

[139] R. Lopes, "Integrated model of quality inspection, preventive maintenance and buffer stock in an imperfect production system," *Comput. Ind. Eng.*, vol. 126, pp. 650–656, 2018, doi: https://doi.org/10.1016/j.cie.2018.10.019.

[140] R. Jafari-Marandi, M. Khanzadeh, W. Tian, B. Smith, and L. Bian, "From in-situ monitoring toward high-throughput process control: cost-driven decision-making

framework for laser-based additive manufacturing," *J. Manuf. Syst.*, vol. 51, pp. 29–41, Apr. 2019, doi: 10.1016/j.jmsy.2019.02.005.

[141] B. Schleich, N. Anwer, L. Mathieu, and S. Wartzack, "Shaping the digital twin for design and production engineering," *CIRP Ann. - Manuf. Technol.*, vol. 66, no. 1, pp. 141–144, Jan. 2017, doi: 10.1016/j.cirp.2017.04.040.

[142] E. H. Glaessgen and D. S. Stargel, "The digital twin paradigm for future NASA and U.S. Air force vehicles," 2012, doi: 10.2514/6.2012-1818.

[143] M. Shafto, M. C. Rich, D. E. Glaessgen, C. Kemp, J. Lemoigne, and L. Wang, "Modeling, simulation, information technology & processing roadmap," 2010.

[144] M. Grieves and J. Vickers, "Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems," in *Transdisciplinary Perspectives on Complex Systems: New Findings and Approaches*, Springer International Publishing, 2016, pp. 85–113.

[145] B. R. Barricelli, E. Casiraghi, and D. Fogli, "A survey on digital twin: Definitions, characteristics, applications, and design implications," *IEEE Access*, vol. 7. Institute of Electrical and Electronics Engineers Inc., 2019, doi: 10.1109/ACCESS.2019.2953499.

[146] D. Jones, C. Snider, A. Nassehi, J. Yon, and B. Hicks, "Characterising the Digital Twin: A systematic literature review," *CIRP J. Manuf. Sci. Technol.*, vol. 29, pp. 36–52, May 2020, doi: 10.1016/j.cirpj.2020.02.002.

[147] E. Negri, L. Fumagalli, and M. Macchi, "A Review of the Roles of Digital Twin in CPS-based Production Systems," *Procedia Manuf.*, vol. 11, pp. 939–948, Jan. 2017, doi: 10.1016/j.promfg.2017.07.198.

[148] W. Kritzinger, M. Karner, G. Traar, J. Henjes, and W. Sihn, "Digital Twin in manufacturing: A categorical literature review and classification," *IFAC-PapersOnLine*, vol. 51, no. 11, pp. 1016–1022, Jan. 2018, doi: 10.1016/j.ifacol.2018.08.474.

[149] F. Tao, J. Cheng, Q. Qi, M. Zhang, H. Zhang, and F. Sui, "Digital twin-driven product design, manufacturing and service with big data," *Int. J. Adv. Manuf. Technol.*, vol. 94, no. 9–12, pp. 3563–3576, Feb. 2018, doi: 10.1007/s00170-017-0233-1.

[150] R. Rosen, G. Von Wichert, G. Lo, and K. D. Bettenhausen, "About the importance of autonomy and digital twins for the future of manufacturing," in *IFAC-PapersOnLine*, May 2015, vol. 28, no. 3, pp. 567–572, doi: 10.1016/j.ifacol.2015.06.141.

[151] J. Lee, E. Lapira, B. Bagheri, and H. an Kao, "Recent advances and trends in predictive manufacturing systems in big data environment," *Manuf. Lett.*, vol. 1, no. 1, pp. 38–41, Oct. 2013, doi: 10.1016/j.mfglet.2013.09.005.

[152] S. Weyer, T. Meyer, M. Ohmer, D. Gorecky, and D. Zühlke, "Future Modeling and Simulation of CPS-based Factories: an Example from the Automotive Industry," *IFAC-PapersOnLine*, vol. 49, no. 31, pp. 97–102, Jan. 2016, doi: 10.1016/j.ifacol.2016.12.168.

[153] T. Gabor, L. Belzner, M. Kiermeier, M. T. Beck, and A. Neitz, "A simulation-based architecture for smart cyber-physical systems," in *Proceedings - 2016 IEEE International Conference on Autonomic Computing, ICAC 2016*, Sep. 2016, pp. 374–379, doi: 10.1109/ICAC.2016.29.

[154] Q. Qi and F. Tao, "Digital Twin and Big Data Towards Smart Manufacturing and Industry 4.0: 360 Degree Comparison," *IEEE Access*, vol. 6, pp. 3585–3593, Jan. 2018, doi: 10.1109/ACCESS.2018.2793265.

[155] F. Tao *et al.*, "Digital twin and its potential application exploration," *Jisuanji Jicheng Zhizao Xitong/Computer Integr. Manuf. Syst. CIMS*, vol. 24, no. 1, pp. 1–18, Jan. 2018, doi: 10.13196/j.cims.2018.01.001.

[156] F. Tao, M. Zhang, Y. Liu, and A. Y. C. Nee, "Digital twin driven prognostics and health management for complex equipment," *CIRP Ann.*, vol. 67, no. 1, pp. 169–172, Jan. 2018, doi: 10.1016/j.cirp.2018.04.055.

[157] F. Tao, H. Zhang, A. Liu, and A. Y. C. Nee, "Digital Twin in Industry: State-of-the-Art," *IEEE Trans. Ind. Informatics*, vol. 15, no. 4, pp. 2405–2415, Apr. 2019, doi: 10.1109/TII.2018.2873186.

[158] F. Tao *et al.*, "Digital twin-driven product design framework," *Int. J. Prod. Res.*, vol. 57, no. 12, pp. 3935–3953, Jun. 2019, doi: 10.1080/00207543.2018.1443229.

[159] Y. Fang, C. Peng, P. Lou, Z. Zhou, J. Hu, and J. Yan, "Digital-Twin-Based Job Shop Scheduling Toward Smart Manufacturing," *IEEE Trans. Ind. Informatics*, vol. 15, no. 12, pp. 6425–6435, Dec. 2019, doi: 10.1109/TII.2019.2938572.

[160] W. Shen, T. Hu, Y. Yin, J. He, F. Tao, and A. Y. C. Nee, "Digital twin based virtual commissioning for computerized numerical control machine tools," in *Digital Twin Driven Smart Design*, Elsevier, 2020, pp. 289–307.

[161] Q. Qi *et al.*, "Enabling technologies and tools for digital twin," *J. Manuf. Syst.*, Oct. 2019, doi: 10.1016/j.jmsy.2019.10.001.

[162] A. Gharaei, J. Lu, O. Stoll, X. Zheng, S. West, and D. Kiritsis, "Systems Engineering Approach to Identify Requirements for Digital Twins Development," in *IFIP Advances in Information and Communication Technology*, Aug. 2020, vol. 591 IFIP, pp. 82–90, doi: 10.1007/978-3-030-57993-7_10.

[163] J. Lu, X. Zheng, A. Gharaei, K. Kalaboukas, and D. Kiritsis, "Cognitive Twins for Supporting Decision-Makings of Internet of Things Systems," *Lect. Notes Mech. Eng.*, pp. 105–115, Dec. 2019, Accessed: Sep. 10, 2020. [Online]. Available: http://arxiv.org/abs/1912.08547.

[164] M. Colledani and T. Tolio, "Integrated analysis of quality and production logistics performance in manufacturing lines," *Int. J. Prod. Res.*, vol. 49, no. 2, pp. 485–518, Jan. 2011, doi: 10.1080/00207540903443246.

[165] M. Dios and J. M. Framinan, "A review and classification of computer-based manufacturing scheduling tools," *Comput. Ind. Eng.*, vol. 99, pp. 229–249, 2016, doi: 10.1016/j.cie.2016.07.020.

[166] F. Psarommatis and D. Kiritsis, "A scheduling tool for achieving zero defect manufacturing (ZDM): A conceptual framework," in *IFIP Advances in Information and Communication Technology*, Aug. 2018, vol. 536, pp. 271–278, doi: 10.1007/978-3-319-99707-0_34.

[167] M. C. Ferris, C. T. Maravelias, and A. Sundaramoorthy, "Simultaneous batching and scheduling using dynamic decomposition on a grid," *INFORMS J. Comput.*, vol. 21, no. 3, pp. 398–410, 2009, doi: 10.1287/ijoc.1090.0339.

[168] F. Tao, Q. Qi, A. Liu, and A. Kusiak, "Data-driven smart manufacturing," *J. Manuf. Syst.*, vol. 48, pp. 157–169, Jul. 2018, doi: 10.1016/j.jmsy.2018.01.006.

[169] C. Zhang, D. Chen, F. Tao, and A. Liu, "Data driven smart customization," in *Procedia CIRP*, Jan. 2019, vol. 81, pp. 564–569, doi: 10.1016/j.procir.2019.03.156.

[170] R. Sahal, J. G. Breslin, and M. I. Ali, "Big data and stream processing platforms for Industry 4.0 requirements mapping for a predictive maintenance use case," *J. Manuf. Syst.*, vol. 54, pp. 138–151, Jan. 2020, doi: 10.1016/j.jmsy.2019.11.004.

[171] G. Baranauskas, "Application of Big Data Analytics in Customization of E-mass Service: Main Possibilities and Obstacles," *Manag. Organ. Syst. Res.*, vol. 82, no. 1, pp. 1–11, Apr. 2020, doi: 10.1515/mosr-2019-0009.

[172] R. C. . Smith, *Uncertainty quantification : theory, implementation, and applications* . Philadelphia : SIAM, 2014.

[173] Y. Liu, "Uncertain random variables: A mixture of uncertainty and randomness," *Soft Comput.*, vol. 17, no. 4, pp. 625–634, Apr. 2013, doi: 10.1007/s00500-012-0935-0.

[174] Y. Zhang, Y. Cheng, X. V. Wang, R. Y. Zhong, Y. Zhang, and F. Tao, "Data-driven smart production line and its common factors," *Int. J. Adv. Manuf. Technol.*, vol. 103, no. 1–4, pp. 1211–1223, Jul. 2019, doi: 10.1007/s00170-019-03469-9.

[175] T. Y. ElMekkawy, P. A. Hachkowski, D. Strong, and D. D. Mann, "Defect analysis for quality and productivity improvements in a manufacturing system," *Can. Biosyst. Eng. / Le Genie des Biosyst. au Canada*, vol. 48, pp. 9–12, 2006.

[176] J. Freiesleben, "A proposal for an economic quality loss function," *Int. J. Prod. Econ.*, vol. 113, no. 2, pp. 1012–1024, Jun. 2008, doi: 10.1016/J.IJPE.2007.12.005.

[177] D. Talia, R. Yahyapour, and W. Ziegler, "Grid Middleware and Services," no. January, 2008, doi: 10.1007/978-0-387-78446-5.

[178] A. Gharaei, M. Karimi, and S. A. H. Shekarabi, “An integrated multi-product multi-buyer supply chain under the penalty, green, and VMI-CS policies: OA/ER/AP algorithm,” *Appl. Math. Model.*, vol. 69, pp. 223–254, 2018, doi: 10.1016/j.apm.2018.11.035.

[179] S. Shahbazi, M. Salloum, M. Kurdve, and M. Wiktorsson, “Material Efficiency Measurement: Empirical Investigation of Manufacturing Industry,” *Procedia Manuf.*, vol. 8, no. October 2016, pp. 112–120, 2017, doi: 10.1016/j.promfg.2017.02.014.

[180] T. L. Johnson, S. R. Fletcher, W. Baker, and R. L. Charles, “How and why we need to capture tacit knowledge in manufacturing: Case studies of visual inspection,” *Appl. Ergon.*, vol. 74, no. December 2016, pp. 1–9, 2019, doi: 10.1016/j.apergo.2018.07.016.

[181] “Repair or Replace Equipment: Cost Analysis | Fiix.” .

[182] R. Dylewski, A. Jardzioch, and I. Krebs, “The Optimal Sequence of Production Orders, Taking into Account the Cost of Delays,” *Manag. Prod. Eng. Rev.*, vol. 7, no. 2, pp. 21–28, 2016, doi: 10.1515/mper-2016-0013.

[183] J. Korponai, Á. B. Tóth, and B. Illés, “Effect of the Safety Stock on the Probability of Occurrence of the Stock Shortage,” *Procedia Eng.*, vol. 182, pp. 335–341, 2017, doi: 10.1016/j.proeng.2017.03.106.

[184] Y. Li, Q. Lin, and F. Ye, “Pricing and promised delivery lead time decisions with a risk-averse agent,” *Int. J. Prod. Res.*, vol. 52, no. 12, pp. 3518–3537, 2014, doi: 10.1080/00207543.2013.871592.

[185] B. Liu, X. Guan, H. Wang, and S. Ma, “Channel configuration and pay-on-delivery service with the endogenous delivery lead time,” *Omega (United Kingdom)*, vol. 84, pp. 175–188, 2018, doi: 10.1016/j.omega.2018.04.011.

[186] C. F. Chien, J. Z. Wu, and C. C. Wu, “A two-stage stochastic programming approach for new tape-out allocation decisions for demand fulfillment planning in semiconductor manufacturing,” *Flex. Serv. Manuf. J.*, vol. 25, no. 3, pp. 286–309, Sep. 2013, doi: 10.1007/s10696-011-9109-0.

[187] C. Morawetz and W. Sihn, “Model of a decision support system for a least-cost and harmonized capacity adjustment in the short- and medium-term planning horizon,” in *Procedia CIRP*, 2012, vol. 3, no. 1, pp. 20–25, doi: 10.1016/j.procir.2012.07.005.

[188] F. Psarommatis, G. May, P.-A. Dreyfus, and D. Kiritsis, “Zero defect manufacturing: state-of-the-art review, shortcomings and future directions in research,” *Int. J. Prod. Res.*, vol. 7543, pp. 1–17, 2019, doi: 10.1080/00207543.2019.1605228.

[189] G. Zhang and K. Xing, “Differential evolution metaheuristics for distributed limited-buffer flowshop scheduling with makespan criterion,” *Comput. Oper. Res.*, vol. 108, pp. 33–43, Aug. 2019, doi: 10.1016/j.cor.2019.04.002.

[190] X. Zhang, X. Liu, S. Tang, G. Królczyk, and Z. Li, “Solving scheduling problem in a distributed manufacturing system using a discrete fruit fly optimization algorithm,” *Energies*, vol. 12, no. 17, Aug. 2019, doi: 10.3390/en12173260.

[191] Q. K. Pan, L. Gao, L. Wang, J. Liang, and X. Y. Li, “Effective heuristics and metaheuristics to minimize total flowtime for the distributed permutation flowshop problem,” *Expert Syst. Appl.*, vol. 124, pp. 309–324, Jun. 2019, doi: 10.1016/j.eswa.2019.01.062.

[192] B. Naderi and R. Ruiz, “The distributed permutation flowshop scheduling problem,” *Comput. Oper. Res.*, vol. 37, no. 4, pp. 754–768, Apr. 2010, doi: 10.1016/j.cor.2009.06.019.

[193] F. Psarommatis and D. Kiritsis, “Identification of the Inspection Specifications for Achieving Zero Defect Manufacturing,” in *IFIP Advances in Information and Communication Technology*, 2019, vol. 566, pp. 267–273, doi: 10.1007/978-3-030-30000-5_34.

[194] H. Wang, Z. Jiang, Y. Wang, H. Zhang, and Y. Wang, “A two-stage optimization method for energy-saving flexible job-shop scheduling based on energy dynamic characterization,” *J. Clean. Prod.*, vol. 188, pp. 575–588, 2018, doi: 10.1016/j.jclepro.2018.03.254.

[195] D. Mourtzis, M. Doukas, and F. Psarommatis, “A toolbox for the design, planning and

operation of manufacturing networks in a mass customisation environment," *J. Manuf. Syst.*, vol. 36, pp. 274–286, 2015, doi: 10.1016/j.jmsy.2014.06.004.

[196] M. s. Phadke, *Quality engineering using robust design*. Prentice Hall PTR, 1995.

[197] K. Hwang, Ching-Lai; Yoon, *Multiple Attribute Decision Making: Methods and Applications A State-of-the -Art Survey*, vol. 186. Springer Berlin Heidelberg, 1981.

[198] G. Michalos, S. Makris, and D. Mourtzis, "An intelligent search algorithm-based method to derive assembly line design alternatives," *Int. J. Comput. Integr. Manuf.*, vol. 25, no. 3, pp. 211–229, Mar. 2012, doi: 10.1080/0951192X.2011.627949.

[199] A. S. Milani, A. Shanian, R. Madoliat, and J. A. Nemes, "The effect of normalization norms in multiple attribute decision making models: A case study in gear material selection," *Struct. Multidiscip. Optim.*, vol. 29, no. 4, pp. 312–318, Apr. 2005, doi: 10.1007/s00158-004-0473-1.

[200] G. Chryssolouris, *Manufacturing Systems: Theory and Practice*. Springer-Verlag, 2006.

[201] D. C. Montgomery, *Design and Analysis of Experiments*, 7th ed. New York: John Wiley & Sons, 2008.

Annex 1 (simulation results, Utility Values)

- ZDM1: Detect – Repair
- ZDM2: Detect – Prevent
- ZDM3: Predict – Prevent

MFG 1			MFG 2			MFG 3		
ZDM1	**ZDM2**	**ZDM3**	**ZDM1**	**ZDM2**	**ZDM3**	**ZDM1**	**ZDM2**	**ZDM3**
0.6698	0.6665	0.6428	0.6601	0.6572	0.6160	0.6489	0.6766	0.6016
0.6706	0.6665	0.6528	0.6657	0.6642	0.6076	0.6681	0.6698	0.5947
0.6623	0.6517	0.6441	0.6524	0.6493	0.6183	0.6585	0.6539	0.5916
0.6688	0.6766	0.6514	0.6627	0.6638	0.5936	0.6664	0.6695	0.6034
0.6978	0.6974	0.6586	0.6691	0.6644	0.6032	0.6772	0.6723	0.5737
0.6683	0.6500	0.6436	0.6533	0.6161	0.6014	0.6643	0.6288	0.5671
0.6646	0.6456	0.6502	0.6557	0.6398	0.6627	0.6577	0.6234	0.6764
0.6503	0.6388	0.6471	0.6436	0.6529	0.6302	0.6525	0.6662	0.6082
0.6636	0.6600	0.6455	0.6582	0.6561	0.6121	0.6656	0.6621	0.6066
0.6698	0.6701	0.6493	0.6223	0.6257	0.5947	0.6254	0.6362	0.6073
0.6483	0.6429	0.6535	0.6449	0.6524	0.6061	0.6508	0.6641	0.5740
0.6483	0.6394	0.6540	0.6390	0.6199	0.5978	0.6487	0.6202	0.5978
0.6350	0.6273	0.6506	0.6322	0.6281	0.6609	0.6412	0.6355	0.5996
0.6295	0.6335	0.6640	0.6120	0.6107	0.6379	0.6052	0.6099	0.6701
0.6613	0.6615	0.6451	0.6230	0.6198	0.6125	0.6323	0.6273	0.6268
0.6426	0.6309	0.6662	0.6335	0.6079	0.6456	0.6432	0.6081	0.5948
0.6378	0.6319	0.6474	0.6349	0.6360	0.5966	0.6450	0.6460	0.5753
0.6100	0.6040	0.6475	0.5907	0.5928	0.6153	0.5989	0.6181	0.6501
0.6233	0.6245	0.6610	0.6129	0.6102	0.5882	0.6136	0.6220	0.6227
0.6546	0.6511	0.6603	0.6124	0.6140	0.6252	0.6233	0.6213	0.6065
0.6273	0.6055	0.6541	0.6235	0.5755	0.5975	0.6332	0.6242	0.5886
0.6067	0.5901	0.6658	0.5888	0.5780	0.6377	0.5946	0.6011	0.6806
0.5997	0.6012	0.6879	0.5880	0.5973	0.6332	0.5945	0.5889	0.6304
0.6086	0.6082	0.6423	0.6012	0.6020	0.5981	0.6055	0.6024	0.6278
0.6370	0.6315	0.6637	0.5983	0.6016	0.6261	0.6137	0.6117	0.5747

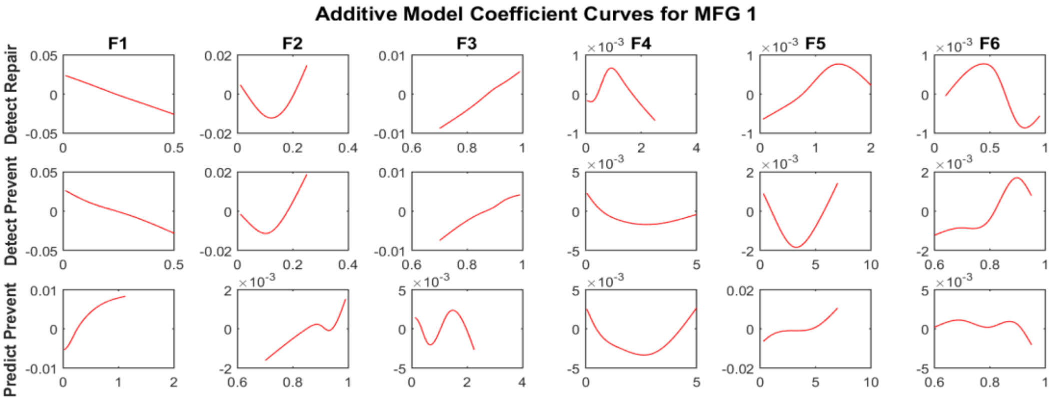
MFG 4			MFG 5			MFG 6		
ZDM1	**ZDM2**	**ZDM3**	**ZDM1**	**ZDM2**	**ZDM3**	**ZDM1**	**ZDM2**	**ZDM3**
0.5972	0.5441	0.6251	0.6853	0.6679	0.6656	0.6845	0.6710	0.6521
0.6238	0.5533	0.6304	0.6878	0.6714	0.6471	0.6930	0.6665	0.6586
0.6147	0.5465	0.6539	0.6736	0.6583	0.6389	0.6761	0.6621	0.6475
0.5936	0.5540	0.6312	0.6737	0.6671	0.6257	0.6751	0.6706	0.6582
0.4876	0.5348	0.6349	0.6565	0.6777	0.6586	0.6565	0.6690	0.6555
0.6283	0.5329	0.6376	0.6761	0.6540	0.6212	0.6793	0.6558	0.6491
0.5826	0.5559	0.6831	0.6810	0.6612	0.6225	0.6842	0.6648	0.6810
0.5753	0.5608	0.6686	0.6664	0.6577	0.6408	0.6702	0.6451	0.6660
0.6135	0.5475	0.6379	0.6773	0.6602	0.6415	0.6763	0.6624	0.6469
0.5555	0.5049	0.5792	0.6400	0.6418	0.6212	0.6509	0.6418	0.6482

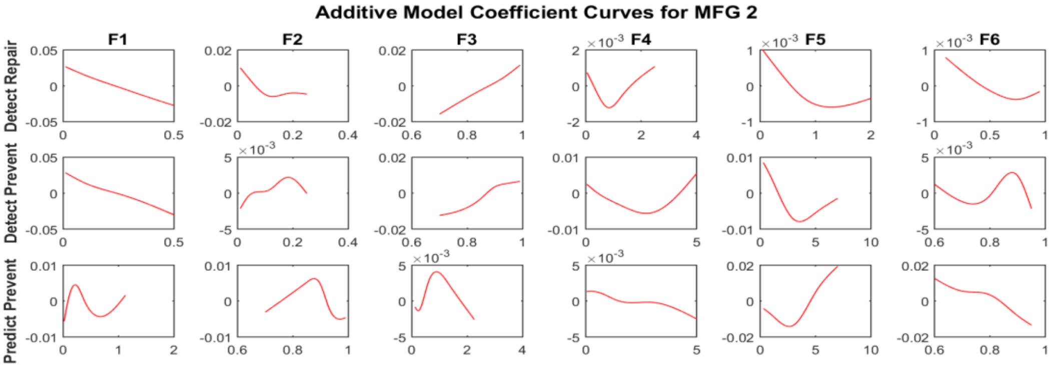
0.5802	0.5525	0.6178	0.6636	0.6536	0.6315	0.6617	0.6457	0.6390
0.6277	0.5036	0.6330	0.6635	0.6427	0.6297	0.6667	0.6431	0.6433
0.5649	0.5018	0.6334	0.6587	0.6365	0.6716	0.6581	0.6352	0.6846
0.5274	0.4983	0.6895	0.6292	0.6243	0.6686	0.6362	0.6226	0.6459
0.5384	0.4961	0.6222	0.6339	0.6304	0.6139	0.6463	0.6345	0.6429
0.5896	0.4726	0.6805	0.6551	0.6382	0.6287	0.6570	0.6376	0.6521
0.6011	0.5260	0.6203	0.6615	0.6414	0.6258	0.6612	0.6368	0.6545
0.5612	0.5027	0.6367	0.6139	0.6012	0.6324	0.6242	0.6033	0.6450
0.5250	0.4984	0.6598	0.6293	0.6123	0.6348	0.6365	0.6198	0.6842
0.5218	0.4921	0.6675	0.6285	0.6236	0.6356	0.6354	0.6242	0.6832
0.5541	0.5075	0.6237	0.6441	0.6058	0.6463	0.6449	0.6169	0.6475
0.5154	0.5068	0.6911	0.6048	0.5924	0.6727	0.6143	0.5960	0.6668
0.5313	0.4669	0.6820	0.6144	0.6100	0.6593	0.6134	0.5995	0.6801
0.5771	0.4801	0.6623	0.6198	0.6032	0.6357	0.6263	0.6062	0.6495
0.4957	0.4666	0.6119	0.6236	0.6110	0.6302	0.6186	0.6045	0.6569
MFG 7			**MFG 8**			**MFG 9**		
ZDM1	**ZDM2**	**ZDM3**	**ZDM1**	**ZDM2**	**ZDM3**	**ZDM1**	**ZDM2**	**ZDM3**
0.6929	0.6937	0.6862	0.6959	0.6915	0.6832	0.6956	0.6956	0.6851
0.6919	0.6906	0.6871	0.6937	0.6910	0.6948	0.6919	0.6914	0.6853
0.6761	0.6751	0.6875	0.6796	0.6790	0.6860	0.6786	0.6782	0.6851
0.6746	0.6747	0.6859	0.6744	0.6732	0.6866	0.6750	0.6739	0.6877
0.6760	0.6741	0.6871	0.6758	0.6735	0.6860	0.6790	0.6741	0.6872
0.6892	0.6851	0.6814	0.6870	0.6836	0.6882	0.6874	0.6849	0.6834
0.6819	0.6738	0.6834	0.6832	0.6798	0.6903	0.6831	0.6771	0.6844
0.6692	0.6637	0.6882	0.6692	0.6702	0.6862	0.6695	0.6673	0.6881
0.6644	0.6659	0.6863	0.6630	0.6633	0.6901	0.6667	0.6655	0.6870
0.6535	0.6573	0.6843	0.6570	0.6564	0.6887	0.6593	0.6589	0.6842
0.6708	0.6658	0.6844	0.6697	0.6652	0.6832	0.6716	0.6650	0.6831
0.6669	0.6622	0.6866	0.6673	0.6633	0.6902	0.6678	0.6617	0.6882
0.6534	0.6486	0.6878	0.6538	0.6535	0.7016	0.6533	0.6510	0.6880
0.6383	0.6364	0.6878	0.6408	0.6430	0.6909	0.6438	0.6404	0.6880
0.6401	0.6445	0.6825	0.6427	0.6382	0.6838	0.6421	0.6490	0.6813
0.6609	0.6527	0.6866	0.6593	0.6551	0.6852	0.6609	0.6543	0.6850
0.6553	0.6534	0.6865	0.6549	0.6558	0.6898	0.6567	0.6544	0.6896
0.6329	0.6297	0.6840	0.6359	0.6333	0.6869	0.6336	0.6358	0.6873
0.6311	0.6328	0.6889	0.6321	0.6311	0.6941	0.6319	0.6326	0.6866
0.6305	0.6284	0.6831	0.6350	0.6323	0.6852	0.6298	0.6329	0.6890
0.6448	0.6378	0.6873	0.6452	0.6387	0.6894	0.6440	0.6392	0.6870
0.6317	0.6228	0.6868	0.6296	0.6284	0.6937	0.6327	0.6240	0.6864
0.6191	0.6171	0.6883	0.6198	0.6220	0.6883	0.6226	0.6209	0.6917
0.6179	0.6154	0.6850	0.6158	0.6164	0.6860	0.6153	0.6144	0.6886
0.6160	0.6167	0.6907	0.6157	0.6154	0.6913	0.6128	0.6167	0.6884
MFG 10			**MFG 11**			**MFG 12**		
ZDM1	**ZDM2**	**ZDM3**	**ZDM1**	**ZDM2**	**ZDM3**	**ZDM1**	**ZDM2**	**ZDM3**
0.6674	0.6588	0.5838	0.6984	0.7011	0.7061	0.6946	0.6971	0.6999
0.6829	0.6446	0.5954	0.6935	0.6956	0.7101	0.6926	0.6926	0.6828

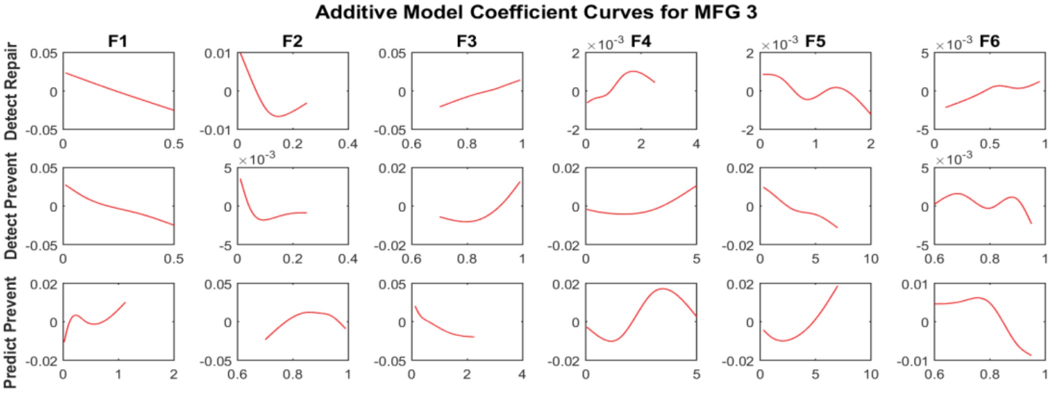
0.6642	0.6300	0.5764	0.6798	0.6814	0.6926	0.6802	0.6794	0.6809
0.6650	0.6556	0.5718	0.6733	0.6737	0.7106	0.6738	0.6728	0.6869
0.6479	0.6496	0.5655	0.6745	0.6738	0.7060	0.6715	0.6771	0.6884
0.6585	0.6111	0.5855	0.6893	0.6905	0.6900	0.6874	0.6802	0.6921
0.6734	0.5785	0.5853	0.6851	0.6846	0.7102	0.6825	0.6791	0.6955
0.6537	0.6381	0.5802	0.6696	0.6709	0.7104	0.6680	0.6696	0.6887
0.6755	0.6562	0.5898	0.6622	0.6633	0.7090	0.6647	0.6634	0.7095
0.6305	0.6150	0.5750	0.6601	0.6646	0.7058	0.6543	0.6509	0.6875
0.6488	0.6305	0.5700	0.6736	0.6735	0.7075	0.6713	0.6710	0.6892
0.6621	0.5893	0.5938	0.6659	0.6677	0.7024	0.6661	0.6652	0.6957
0.6522	0.6093	0.5835	0.6538	0.6530	0.6990	0.6529	0.6535	0.6983
0.6140	0.5969	0.6120	0.6429	0.6466	0.7115	0.6392	0.6402	0.7065
0.6350	0.6197	0.5793	0.6426	0.6445	0.7095	0.6410	0.6399	0.6947
0.6487	0.5637	0.5760	0.6629	0.6623	0.7110	0.6612	0.6561	0.6978
0.6562	0.6312	0.5808	0.6559	0.6563	0.7086	0.6564	0.6545	0.6897
0.5924	0.5936	0.5727	0.6377	0.6400	0.7100	0.6365	0.6353	0.6918
0.6270	0.6001	0.6143	0.6338	0.6331	0.7104	0.6304	0.6285	0.6911
0.6251	0.6114	0.5812	0.6338	0.6326	0.6871	0.6308	0.6280	0.6998
0.6359	0.5438	0.5926	0.6448	0.6457	0.7083	0.6448	0.6413	0.6889
0.5816	0.5431	0.5650	0.6371	0.6386	0.7115	0.6330	0.6313	0.7068
0.5985	0.5697	0.5888	0.6238	0.6253	0.7126	0.6218	0.6191	0.6991
0.6176	0.5960	0.5779	0.6163	0.6165	0.7094	0.6145	0.6155	0.6950
0.6163	0.5989	0.5786	0.6090	0.6084	0.7110	0.6137	0.6142	0.6875
MFG 13			**MFG 14**			**MFG 15**		
ZDM1	**ZDM2**	**ZDM3**	**ZDM1**	**ZDM2**	**ZDM3**	**ZDM1**	**ZDM2**	**ZDM3**
0.6934	0.6949	0.6926	0.6890	0.6877	0.6721	0.6673	0.6711	0.6515
0.6913	0.6904	0.6830	0.6866	0.6805	0.6730	0.6789	0.6771	0.6008
0.6800	0.6753	0.6885	0.6767	0.6706	0.6730	0.6689	0.6589	0.5994
0.6766	0.6733	0.6863	0.6730	0.6754	0.6656	0.6736	0.6736	0.6025
0.6773	0.6775	0.6892	0.6745	0.6784	0.6687	0.6814	0.6808	0.6017
0.6857	0.6812	0.6859	0.6841	0.6716	0.6647	0.6713	0.6529	0.5914
0.6836	0.6782	0.6905	0.6791	0.6696	0.6657	0.6660	0.6495	0.6035
0.6686	0.6677	0.6874	0.6685	0.6617	0.6662	0.6584	0.6542	0.6166
0.6650	0.6671	0.6885	0.6655	0.6666	0.6670	0.6680	0.6683	0.6001
0.6585	0.6607	0.6832	0.6502	0.6510	0.6687	0.6368	0.6349	0.5971
0.6701	0.6667	0.6846	0.6688	0.6561	0.6732	0.6634	0.6473	0.6013
0.6649	0.6636	0.6875	0.6656	0.6525	0.6669	0.6583	0.6331	0.5908
0.6536	0.6511	0.6882	0.6532	0.6481	0.6740	0.6500	0.6381	0.6081
0.6374	0.6440	0.6899	0.6321	0.6375	0.6919	0.6067	0.6220	0.5989
0.6464	0.6455	0.6827	0.6466	0.6451	0.6647	0.6352	0.6366	0.6215
0.6604	0.6557	0.6898	0.6596	0.6448	0.6656	0.6521	0.6349	0.5982
0.6555	0.6539	0.6866	0.6563	0.6479	0.6711	0.6528	0.6452	0.6058
0.6356	0.6351	0.6902	0.6237	0.6223	0.6629	0.6005	0.5974	0.5983
0.6273	0.6312	0.6887	0.6299	0.6272	0.6620	0.6176	0.6143	0.6320
0.6334	0.6373	0.6865	0.6350	0.6341	0.6694	0.6298	0.6297	0.6017
0.6454	0.6399	0.6873	0.6439	0.6319	0.6649	0.6406	0.6150	0.6119

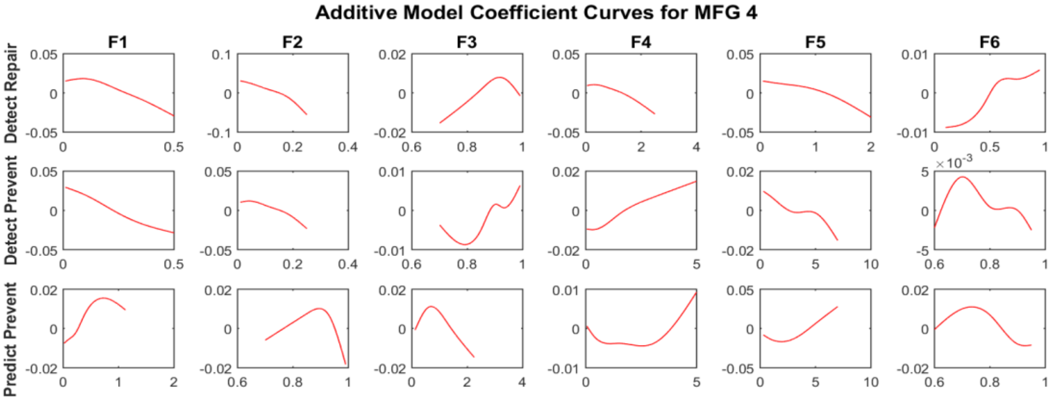
0.6330	0.6284	0.6928	0.6210	0.6117	0.6779	0.6043	0.5943	0.6104
0.6213	0.6181	0.6970	0.6138	0.6109	0.6784	0.6017	0.6026	0.6119
0.6175	0.6196	0.6864	0.6151	0.6138	0.6711	0.6136	0.6080	0.5964
0.6176	0.6197	0.6895	0.6140	0.6182	0.6695	0.6143	0.6149	0.6136

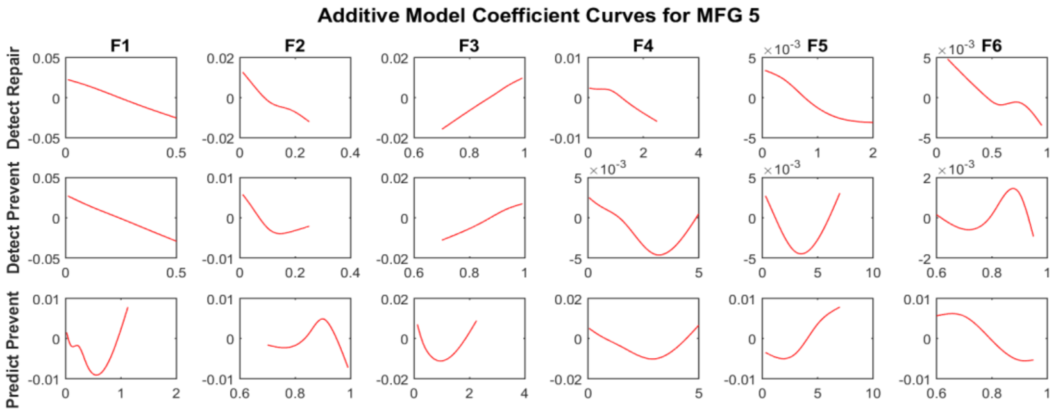
Annex 2 (Utility Value digital twin model coefficients plots)

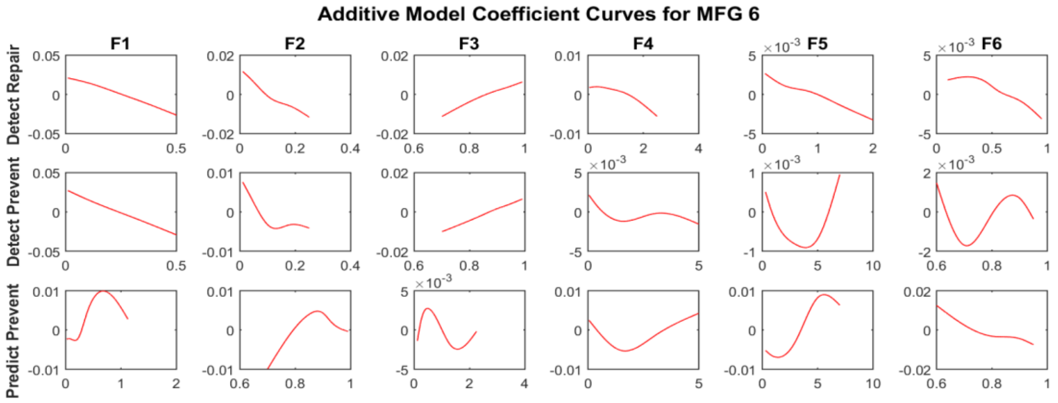


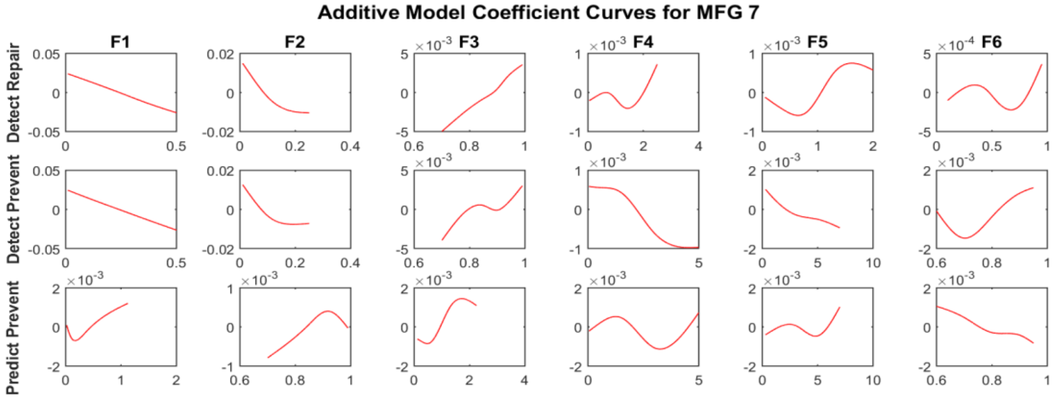


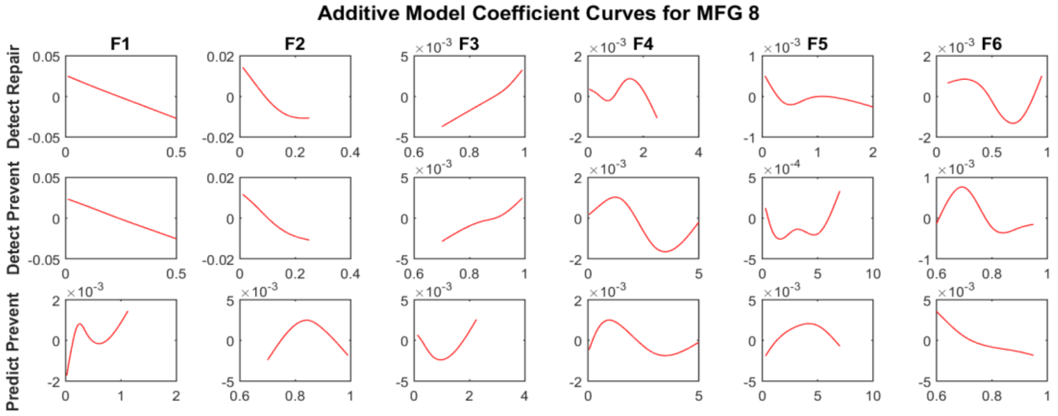




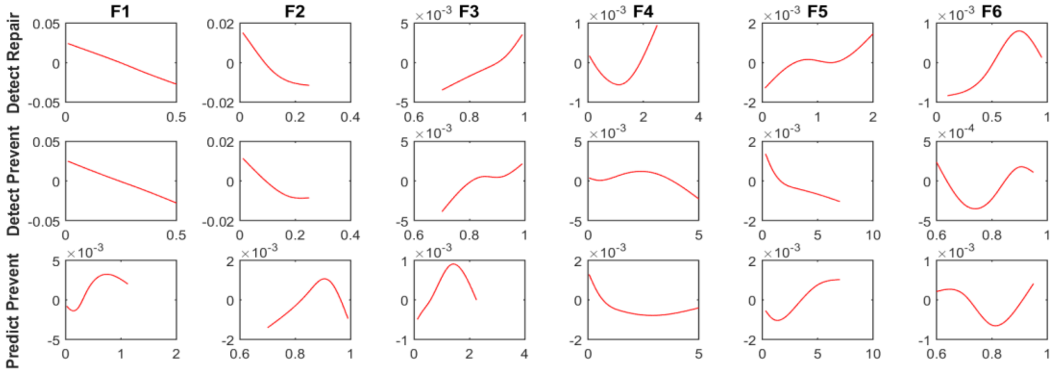

Additive Model Coefficient Curves for MFG 5
F1
F2
F3
F4
F5
F6
Detect Repair
Detect Prevent
Predict Prevent
×10-3
0.05
0
-0.05
0.02
-0.02
0.01
-0.01
5
-5
2
-2
0.5
0.2
0.4
0.6
0.8
1
4
10


Additive Model Coefficient Curves for MFG 6
F1
F2
F3
F4
F5
F6
Detect Repair
Detect Prevent
Predict Prevent
×10-3
0.05
0
-0.05
0.02
-0.02
0.01
-0.01
5
-5
1
-1
2
-2
0.5
0.2
0.4
0.6
0.8
4
10

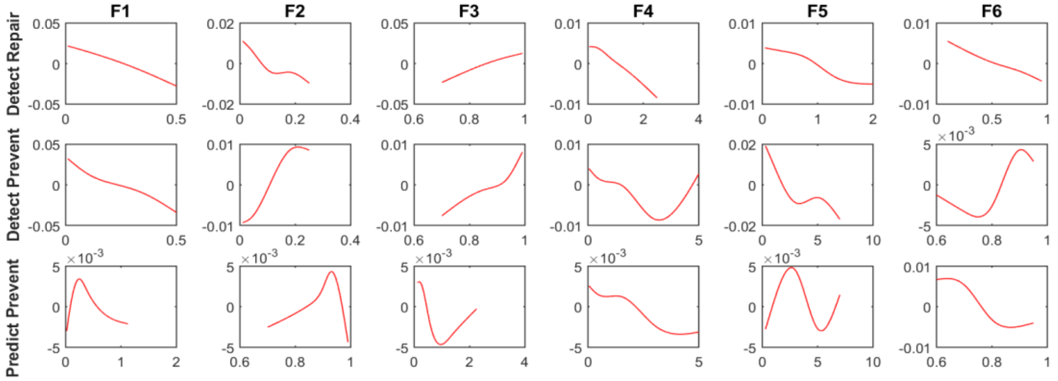

Additive Model Coefficient Curves for MFG 7
F1
F2
F3
F4
F5
F6
Detect Repair
Detect Prevent
Predict Prevent
×10-3
×10-4
0.05
0
-0.05
0.02
-0.02
5
-5
1
-1
2
-2
0.5
0.2
0.4
0.6
0.8
4
10


Additive Model Coefficient Curves for MFG 8
F1
F2
F3
F4
F5
F6
Detect Repair
Detect Prevent
Predict Prevent
×10-3
×10-4
0.05
0
-0.05
0.02
-0.02
5
-5
2
-2
1
-1
0.5
0.2
0.4
0.6
0.8
4
10

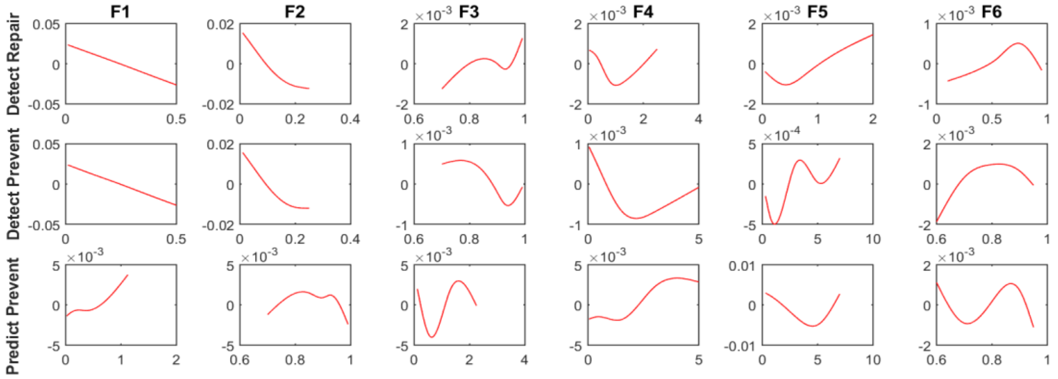
Additive Model Coefficient Curves for MFG 9



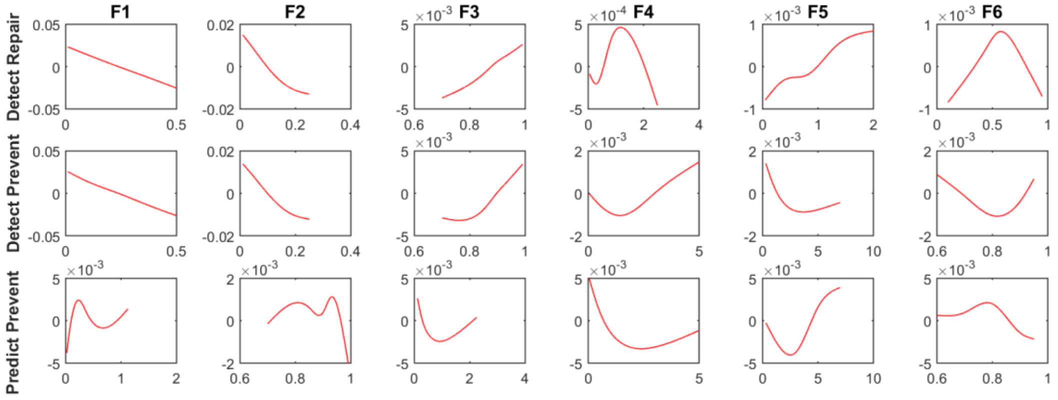
Additive Model Coefficient Curves for MFG 10

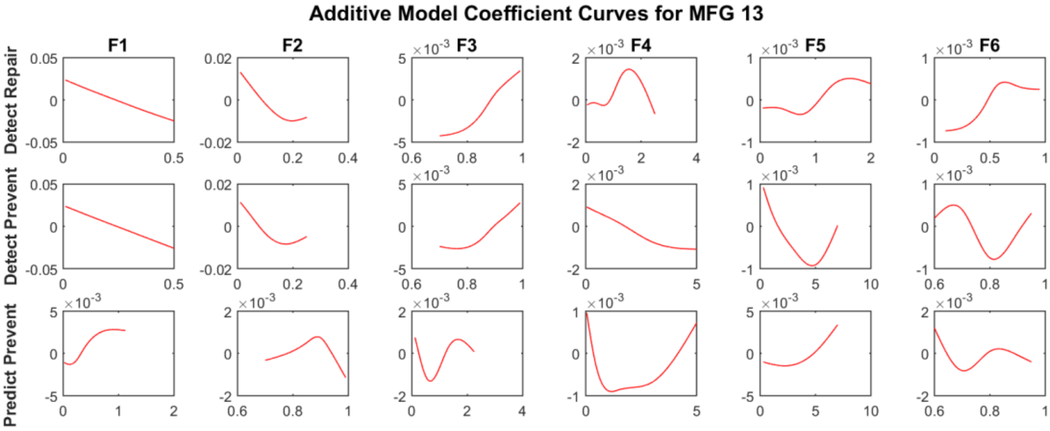


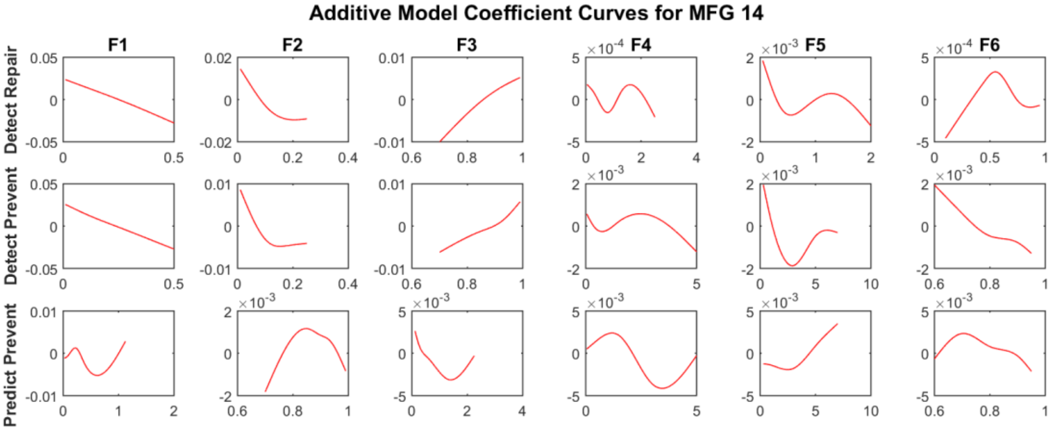
Additive Model Coefficient Curves for MFG 11

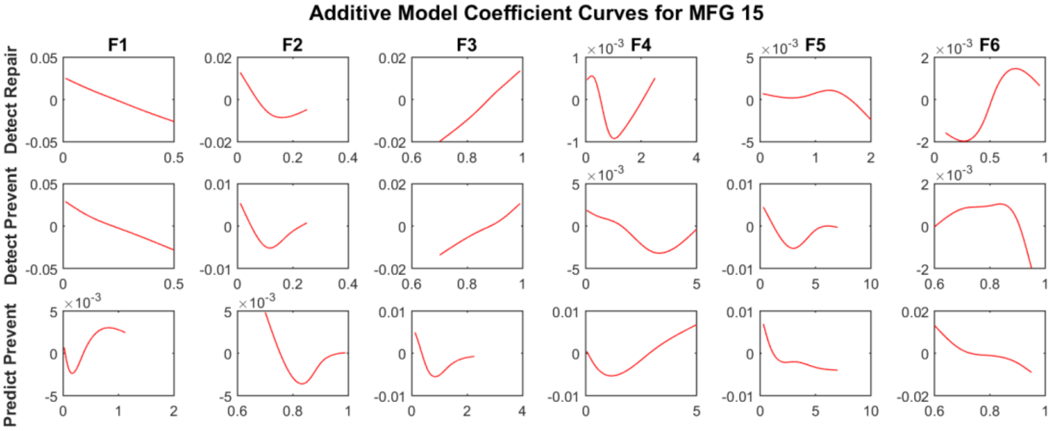


Additive Model Coefficient Curves for MFG 12



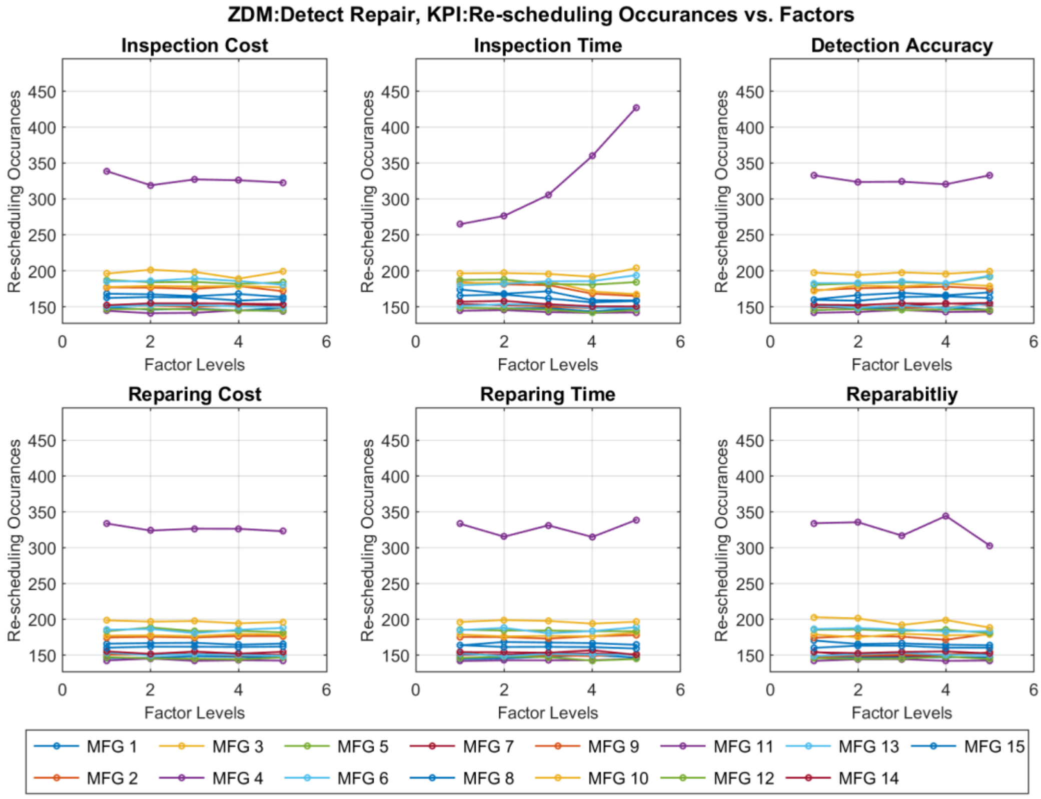

Additive Model Coefficient Curves for MFG 13
F1
F2
F3
F4
F5
F6
Detect Repair
Detect Prevent
Predict Prevent
×10-3
×10-4

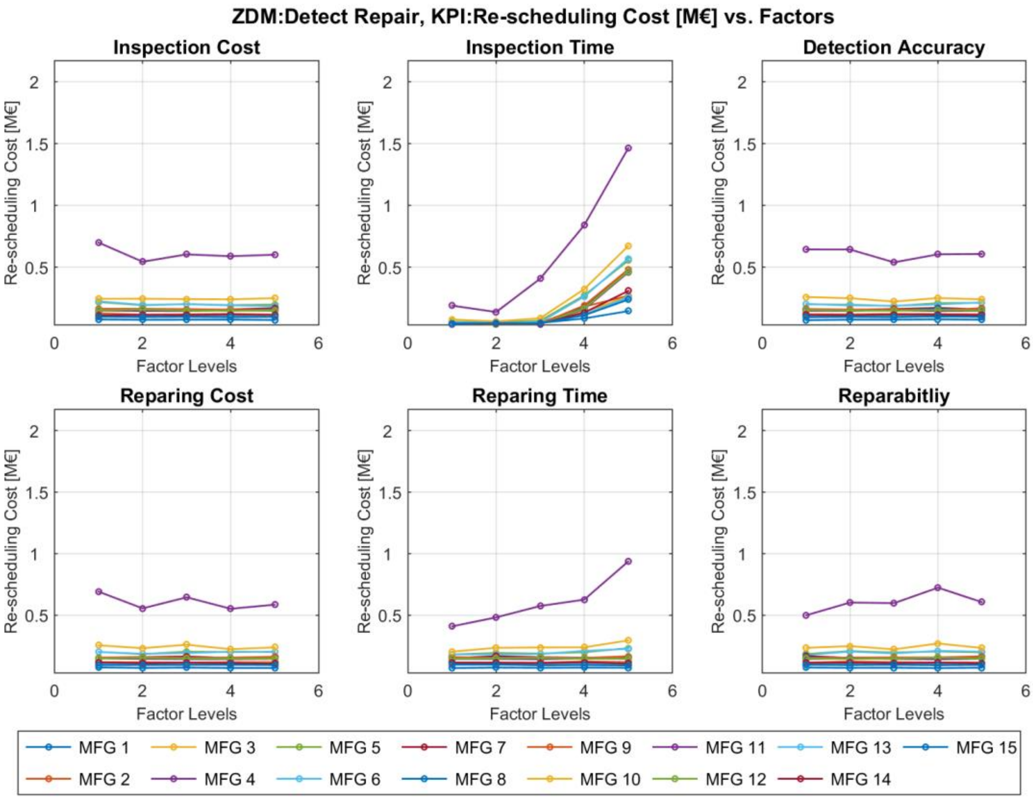

Additive Model Coefficient Curves for MFG 14
F1
F2
F3
F4
F5
F6
Detect Repair
Detect Prevent
Predict Prevent
×10-3
×10-4

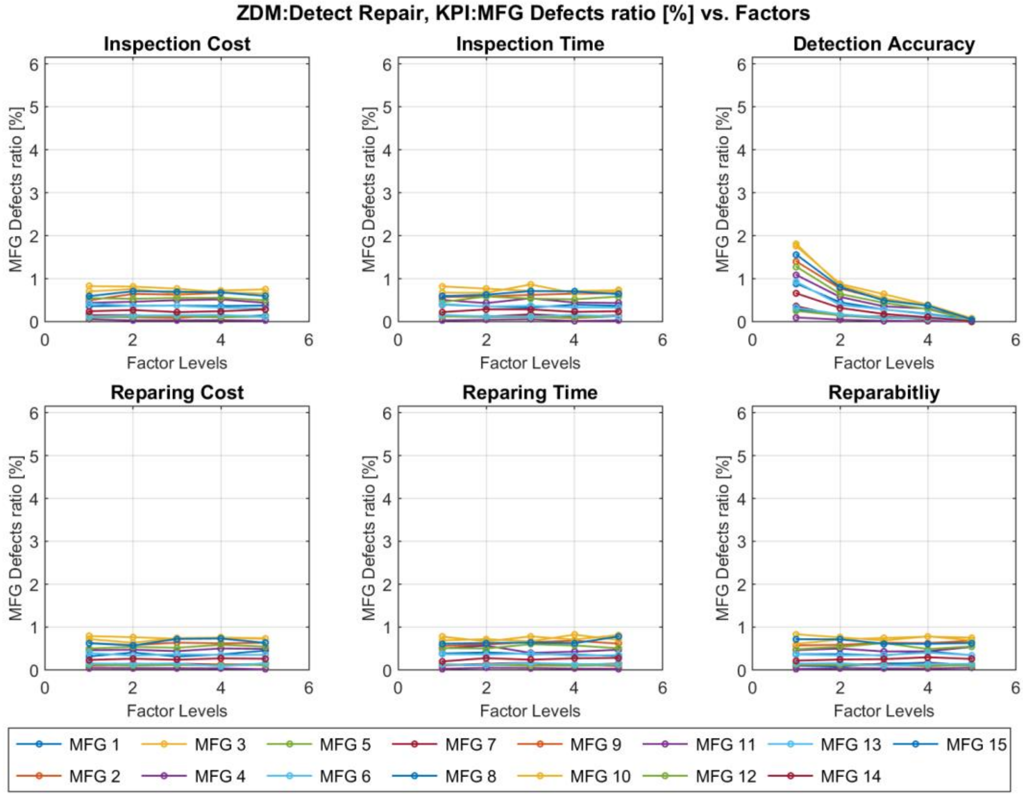

Additive Model Coefficient Curves for MFG 15
F1
F2
F3
F4
F5
F6
Detect Repair
Detect Prevent
Predict Prevent
×10-3

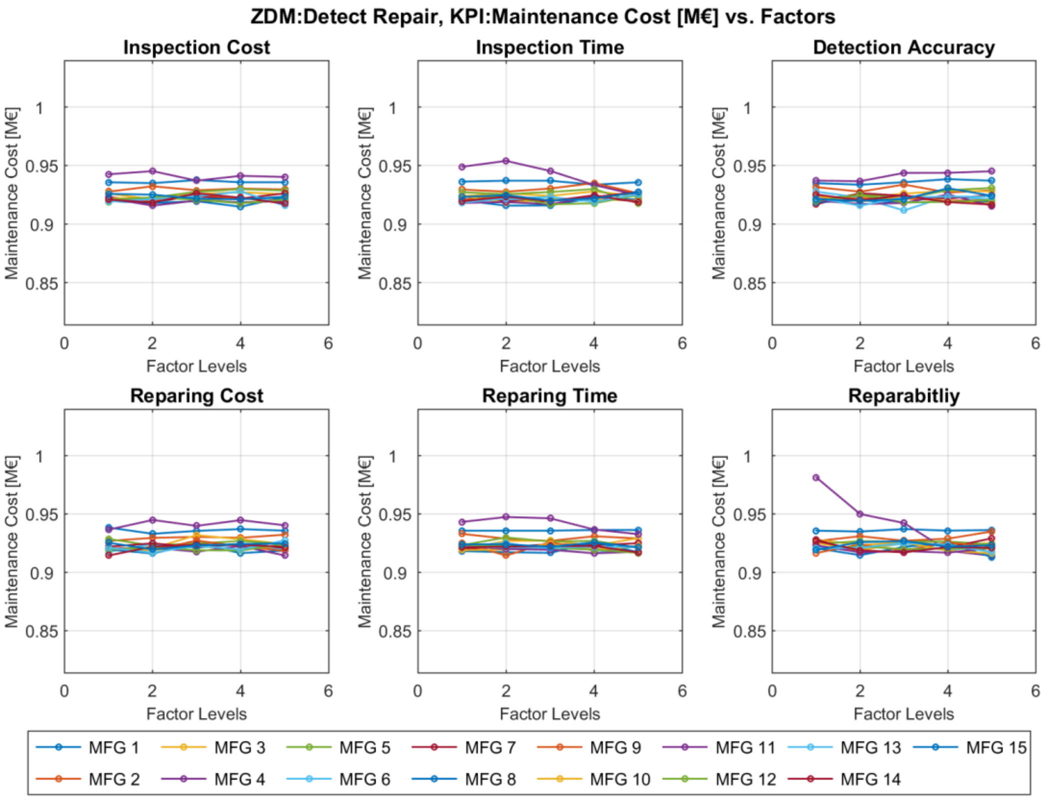
Annex 3 (KPIs ANOM diagrams)

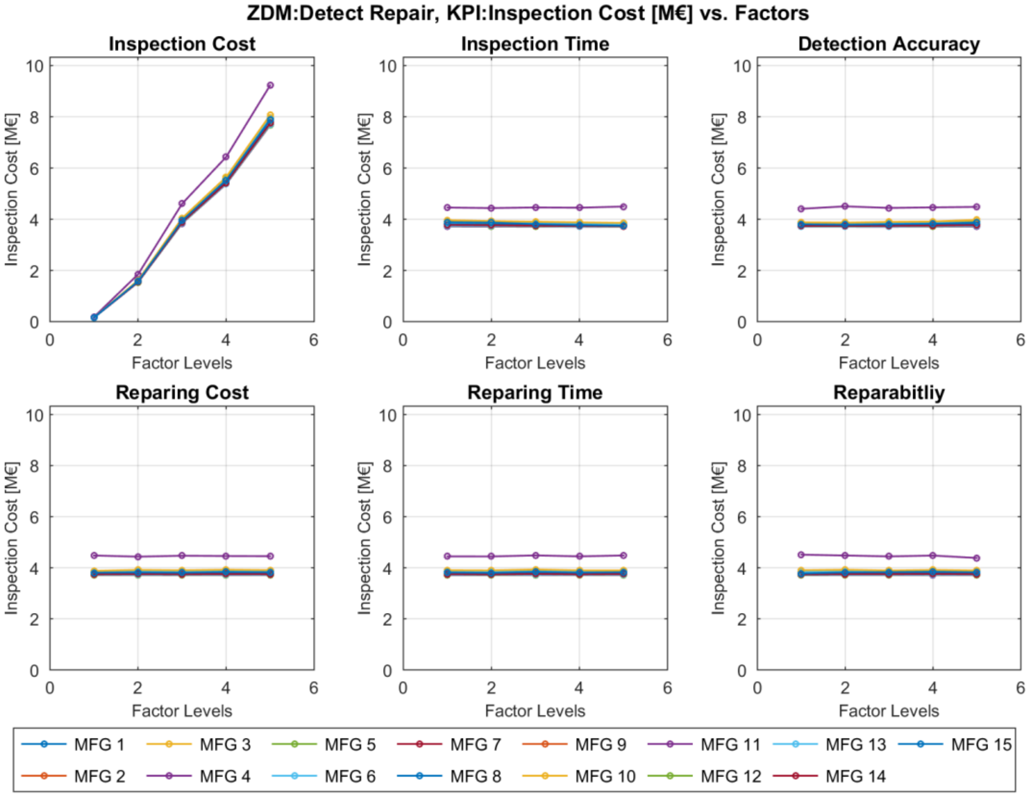
A. Detection Repair ANOM KPIs results

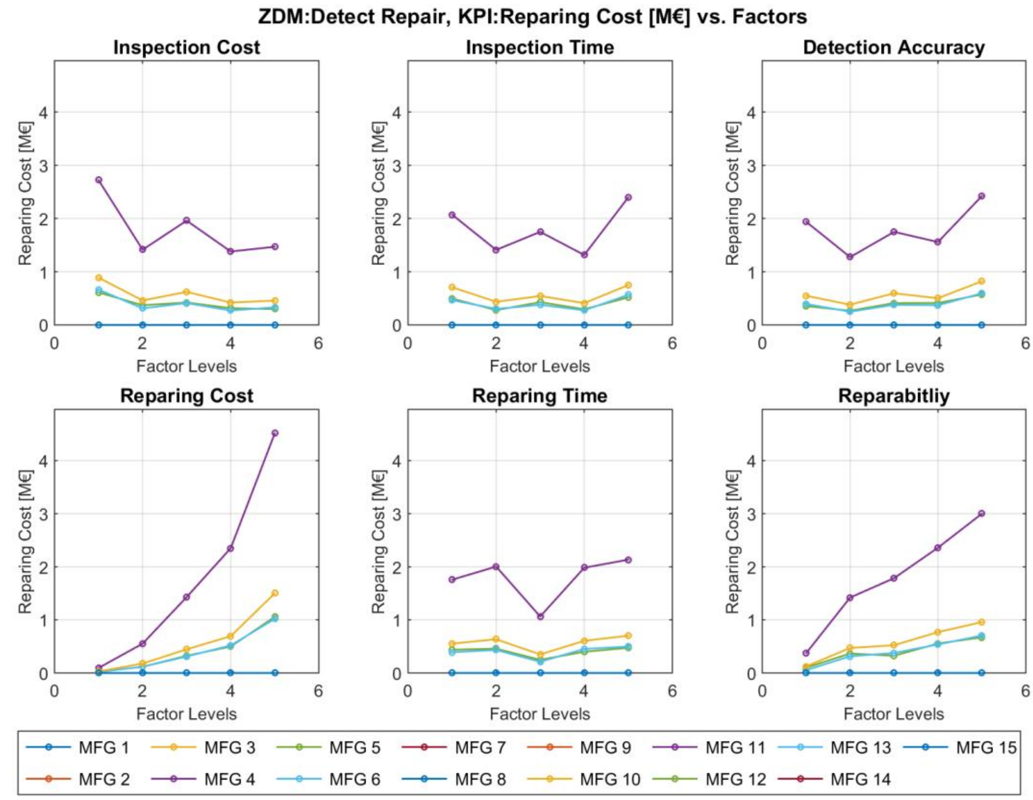


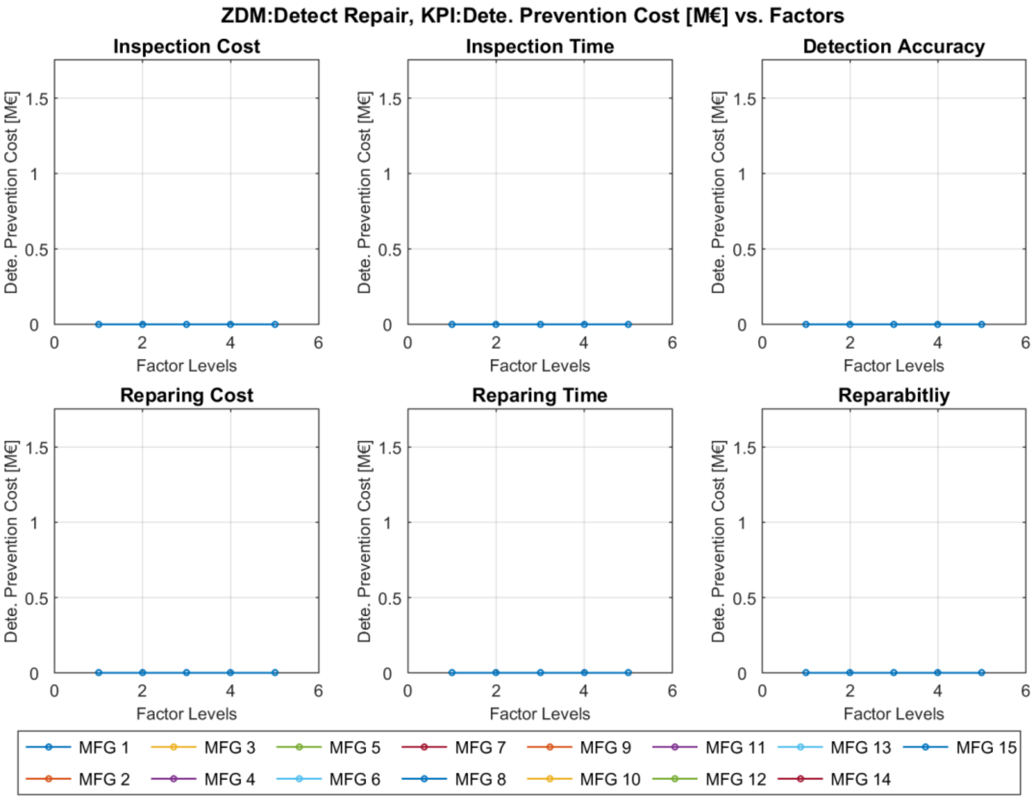

ZDM:Detect Repair, KPI:Re-scheduling Cost [M€] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Reparing Cost
Reparing Time
Reparabitliy
Re-scheduling Cost [M€]
Factor Levels
0
2
4
6
0.5
1
1.5
2
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

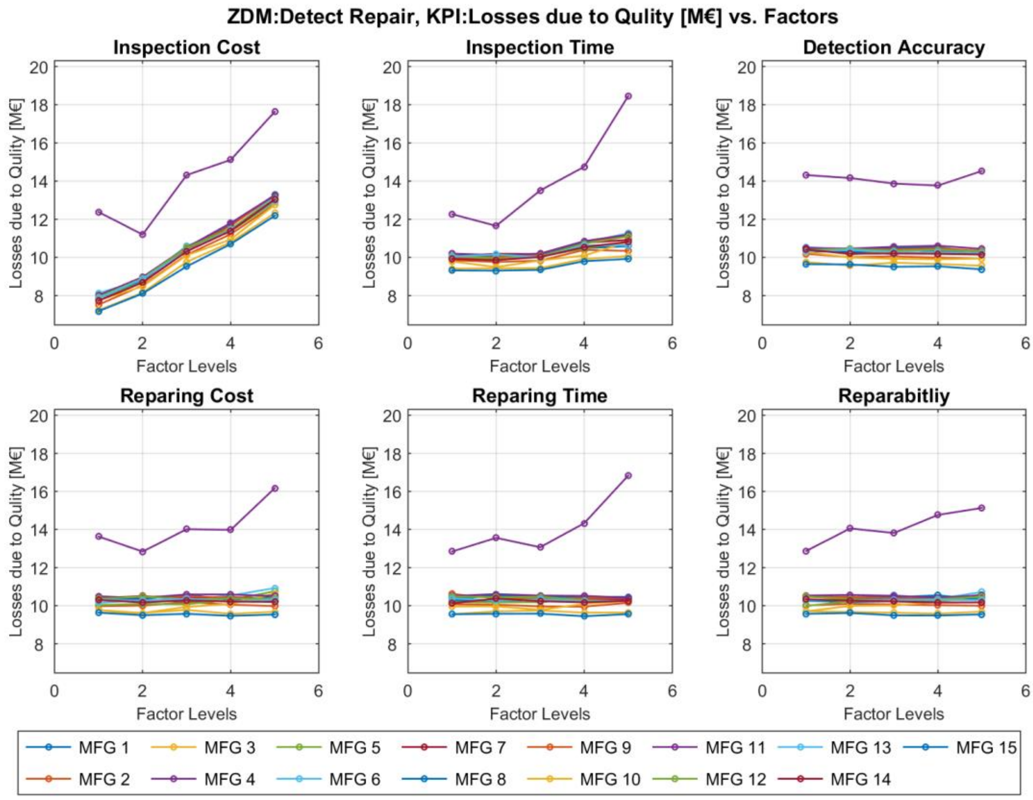

ZDM:Detect Repair, KPI:MFG Defects ratio [%] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Reparing Cost
Reparing Time
Reparabitliy
MFG Defects ratio [%]
Factor Levels
0
1
2
3
4
5
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

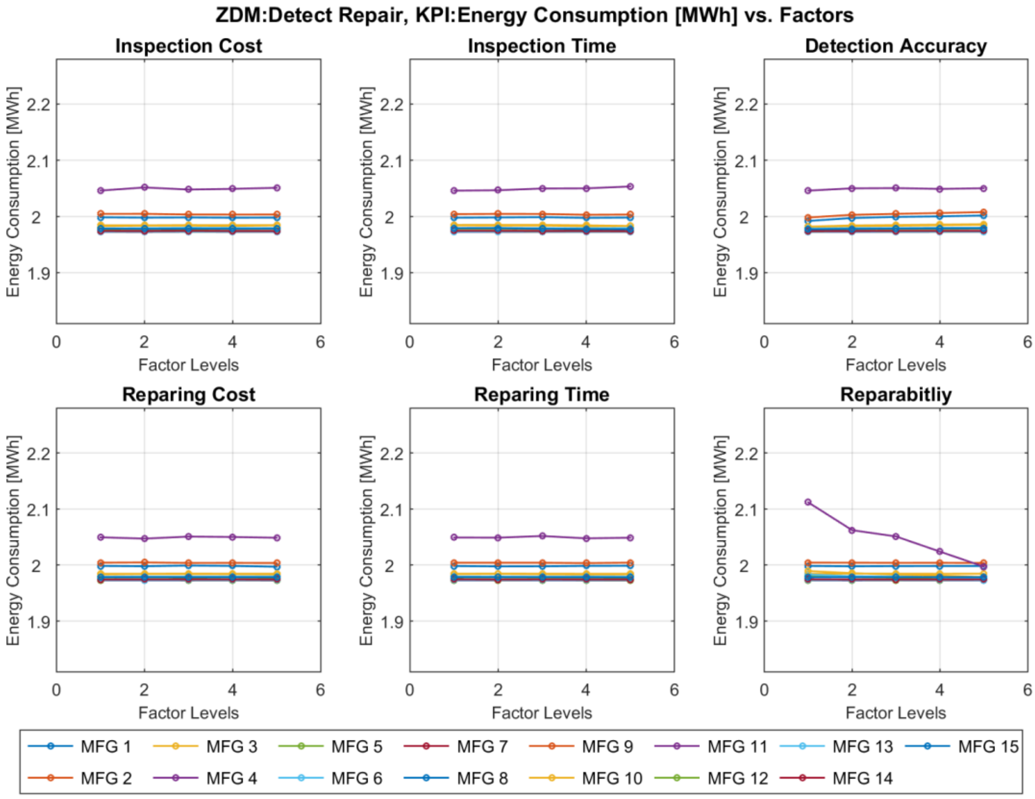

ZDM:Detect Repair, KPI:Maintenance Cost [M€] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Reparing Cost
Reparing Time
Reparabitliy
Maintenance Cost [M€]
0.85
0.9
0.95
1
Factor Levels
0
2
4
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

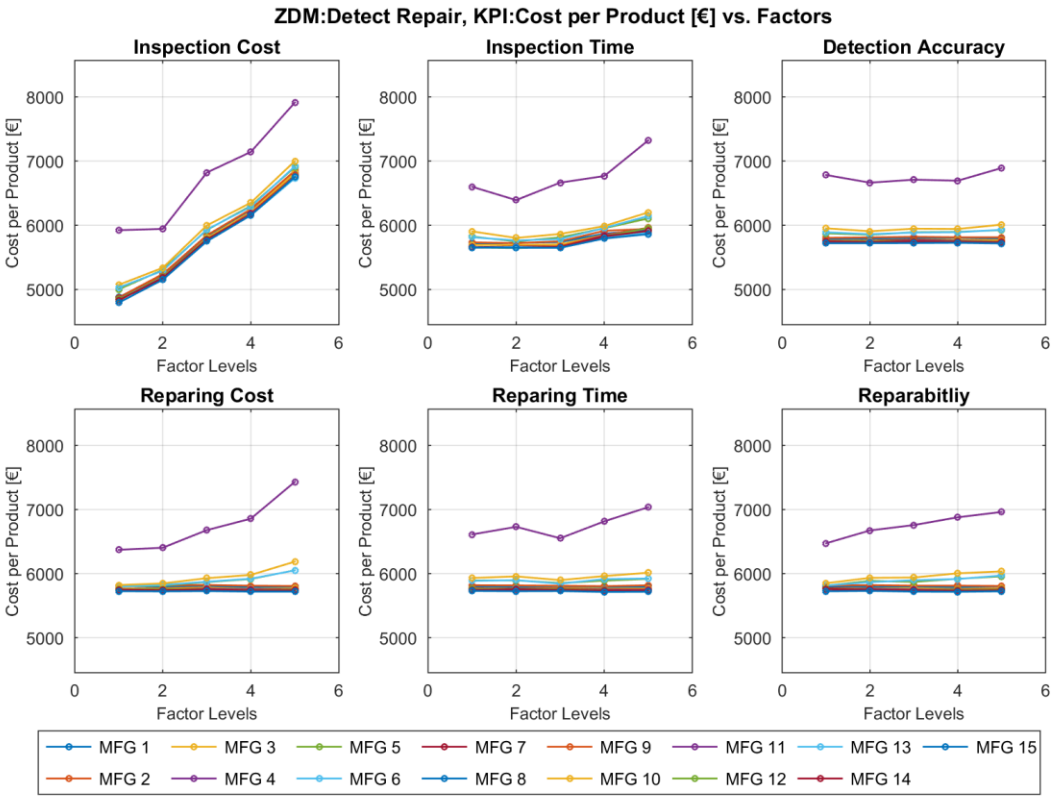

ZDM:Detect Repair, KPI:Inspection Cost [M€] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Reparing Cost
Reparing Time
Reparabitliy
Inspection Cost [M€]
0
2
4
6
8
10
Factor Levels
0
2
4
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

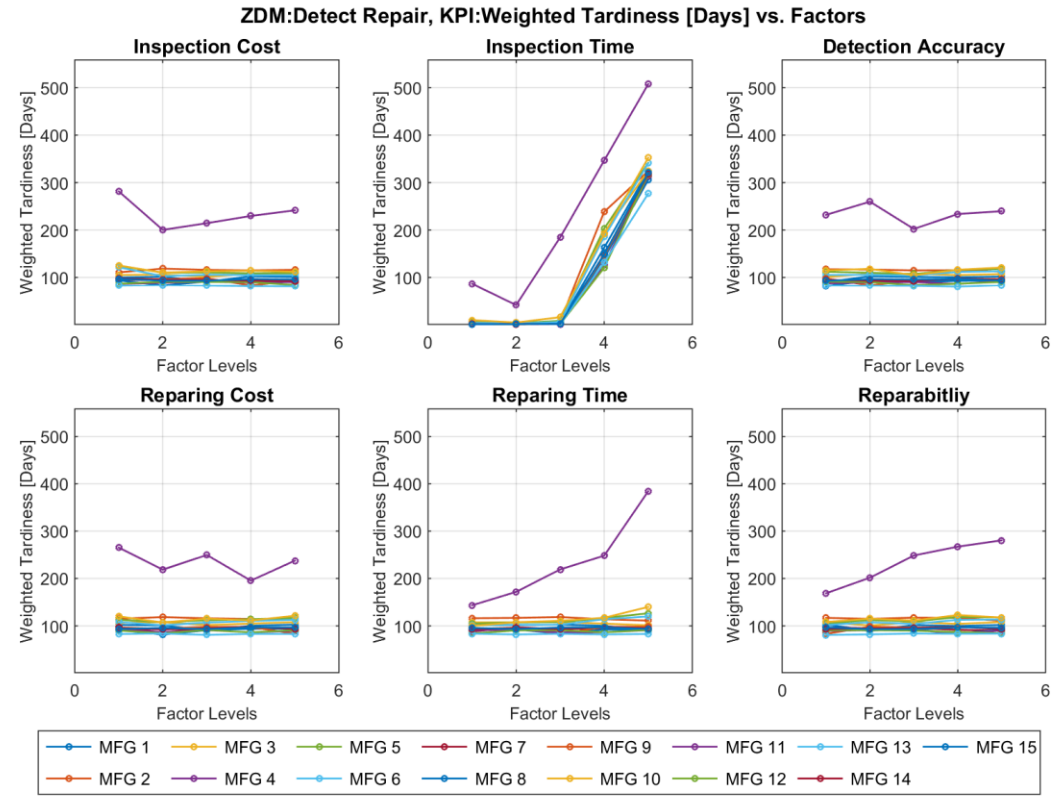

ZDM:Detect Repair, KPI:Reparing Cost [M€] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Reparing Cost
Reparing Time
Reparabitliy
Reparing Cost [M€]
Factor Levels
0
1
2
3
4
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

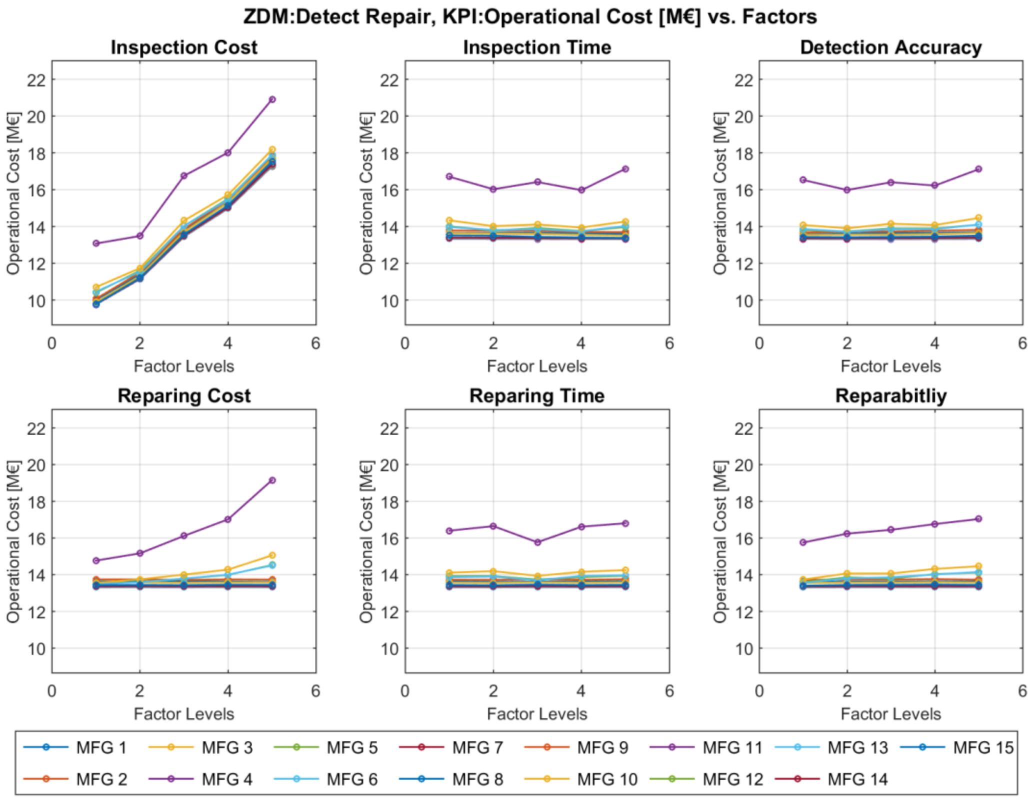

ZDM:Detect Repair, KPI:Dete. Prevention Cost [M€] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Reparing Cost
Reparing Time
Reparabitliy
Dete. Prevention Cost [M€]
Factor Levels
0
0.5
1
1.5
2
4
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

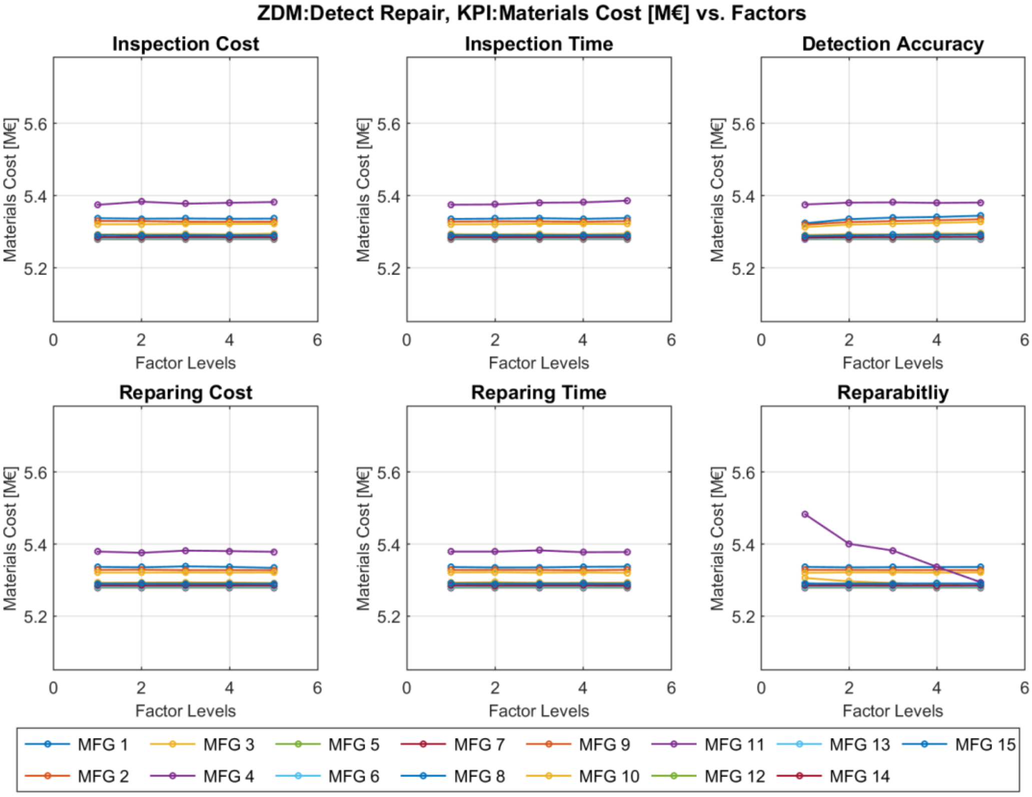

ZDM:Detect Repair, KPI:Losses due to Qulity [M€] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Reparing Cost
Reparing Time
Reparabitliy
Losses due to Qulity [M€]
8
10
12
14
16
18
20
0
2
4
6
Factor Levels
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

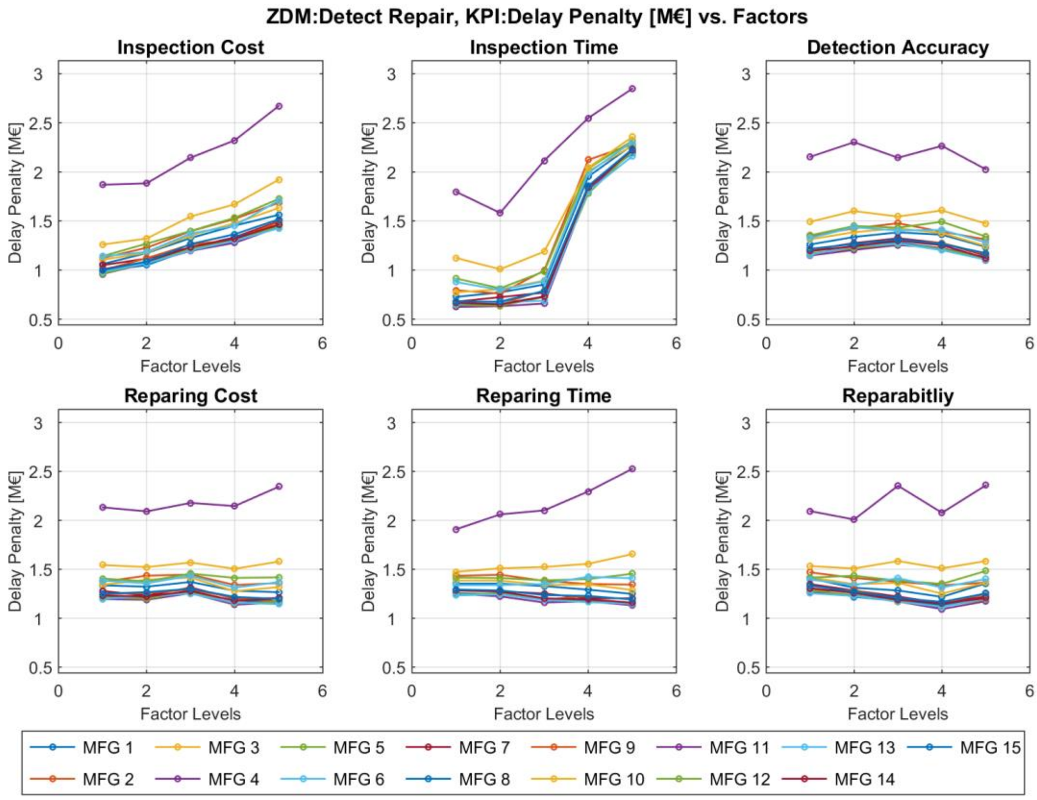

ZDM:Detect Repair, KPI:Energy Consumption [MWh] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Reparing Cost
Reparing Time
Reparabitliy
Energy Consumption [MWh]
1.9
2
2.1
2.2
0
2
4
6
Factor Levels
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

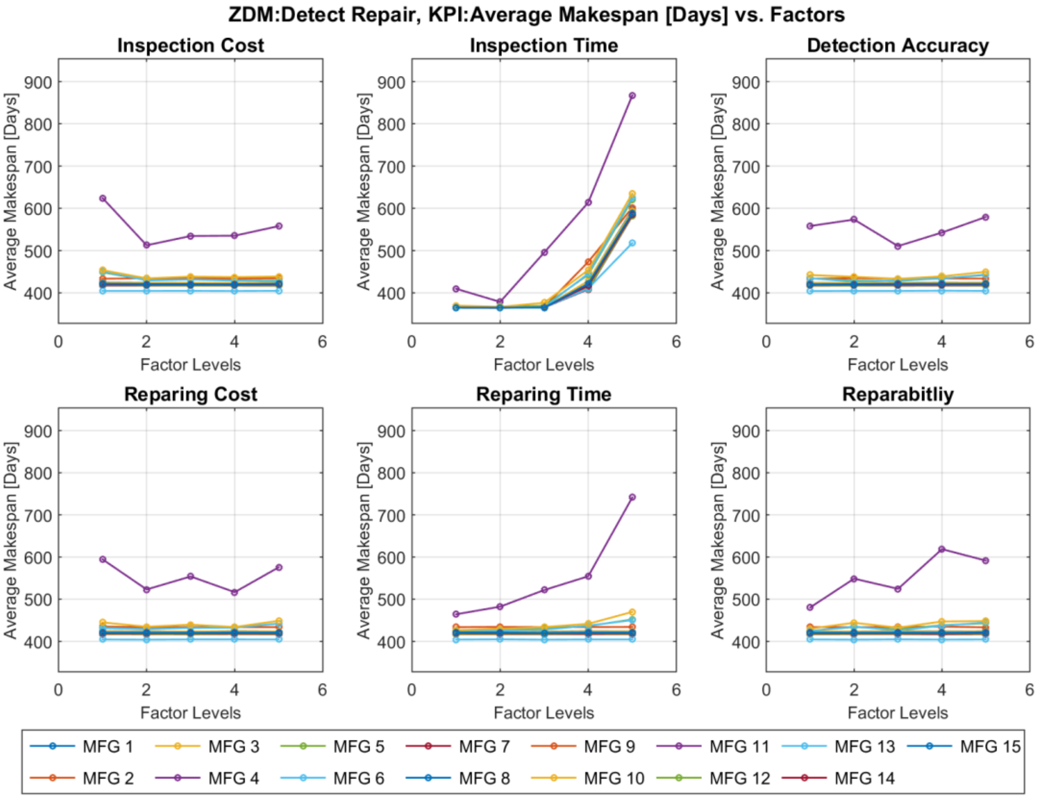

ZDM:Detect Repair, KPI:Cost per Product [€] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Reparing Cost
Reparing Time
Reparabitliy
Cost per Product [€]
5000
6000
7000
8000
Factor Levels
0
2
4
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

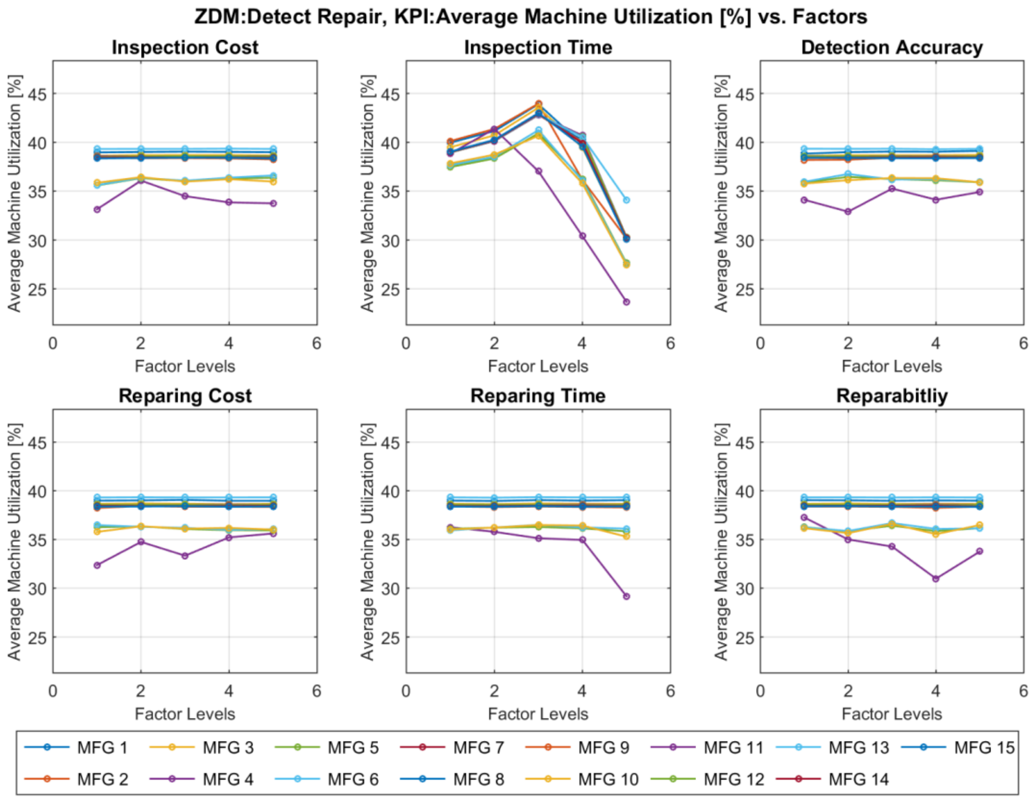

ZDM:Detect Repair, KPI:Weighted Tardiness [Days] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Reparing Cost
Reparing Time
Reparabitliy
Weighted Tardiness [Days]
100
200
300
400
500
Factor Levels
0
2
4
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

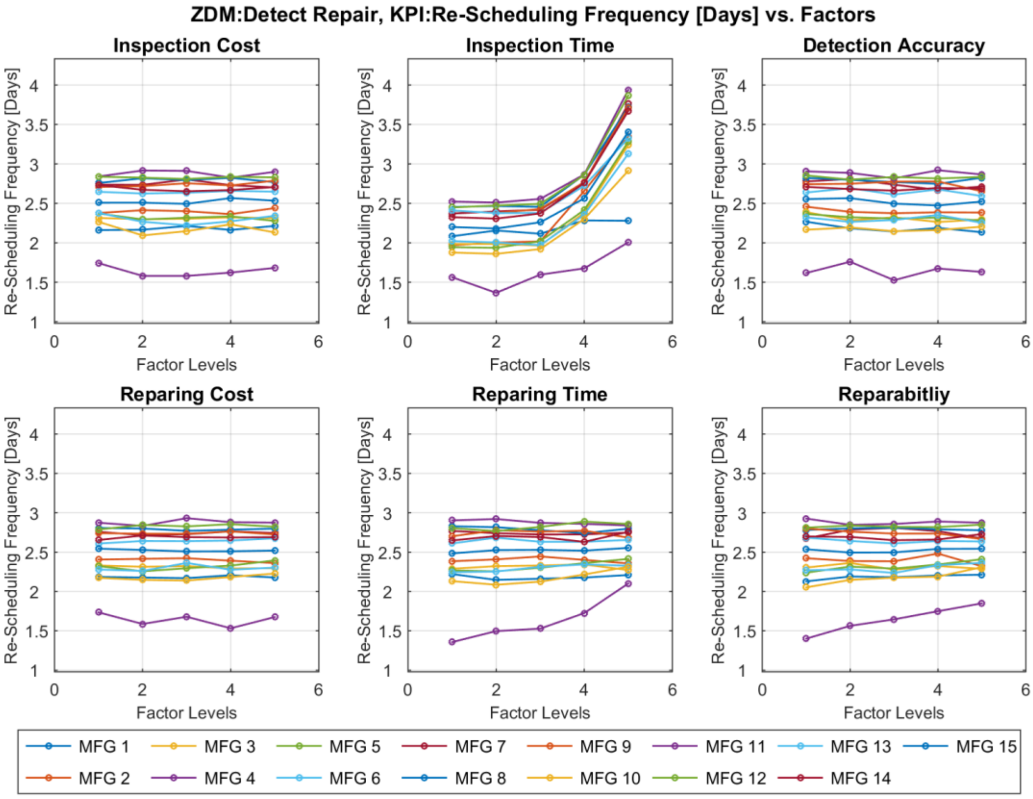

ZDM:Detect Repair, KPI:Operational Cost [M€] vs. Factors


ZDM:Detect Repair, KPI:Materials Cost [M€] vs. Factors

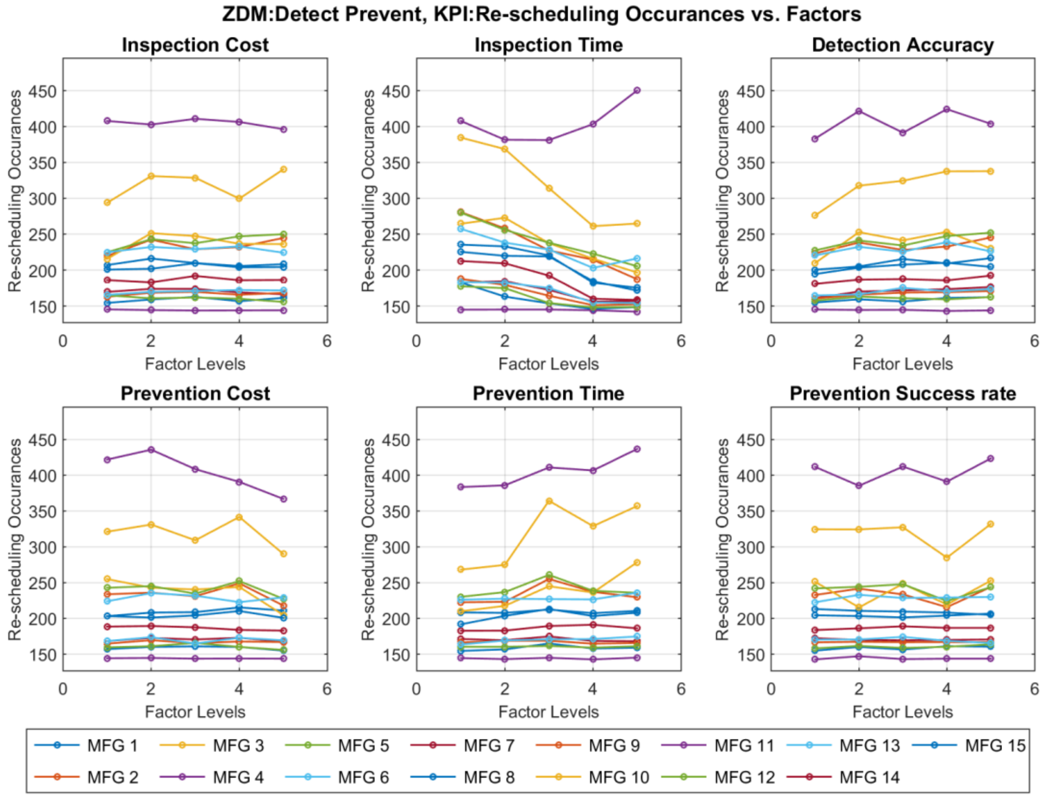

ZDM:Detect Repair, KPI:Delay Penalty [M€] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Reparing Cost
Reparing Time
Reparabitliy
Delay Penalty [M€]
0.5
1
1.5
2
2.5
3
Factor Levels
0
2
4
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

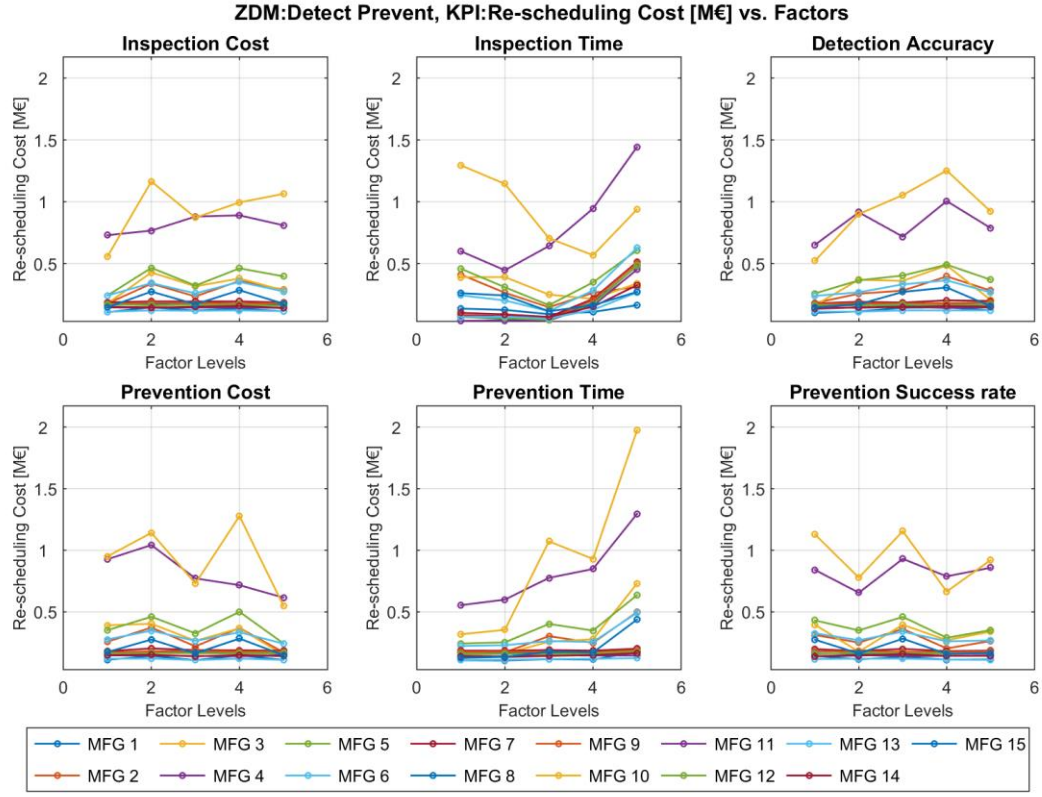

ZDM:Detect Repair, KPI:Average Makespan [Days] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Reparing Cost
Reparing Time
Reparabitliy
Average Makespan [Days]
400
500
600
700
800
900
Factor Levels
0
2
4
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

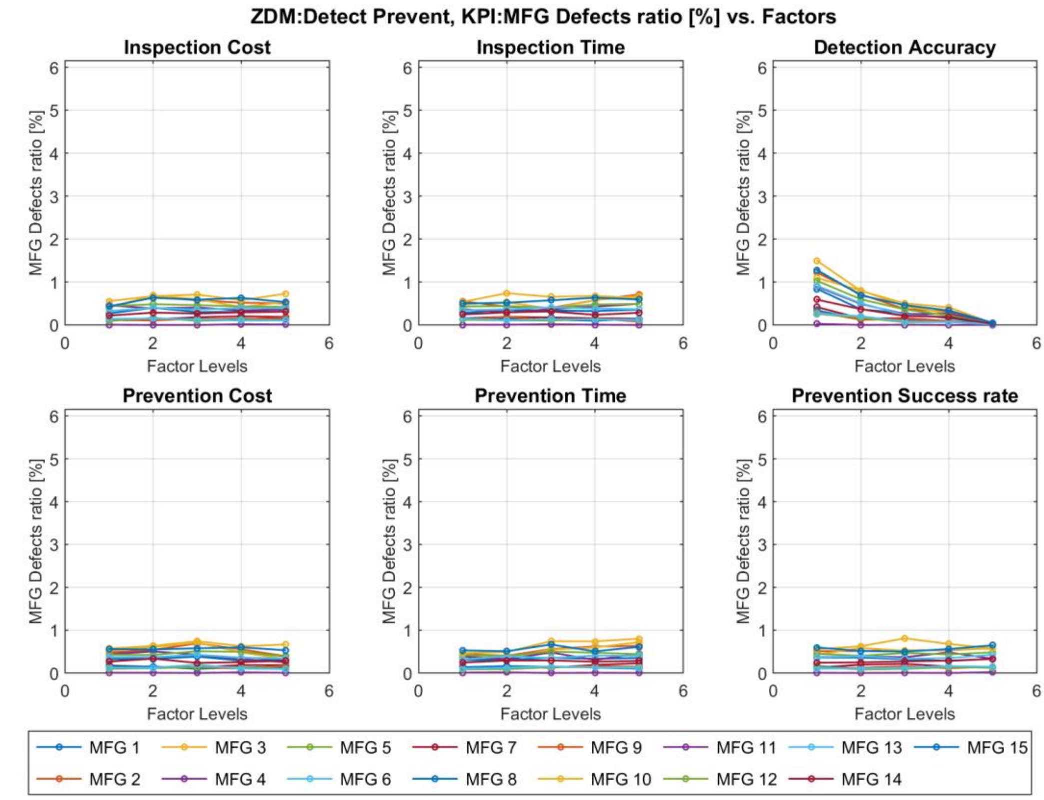

ZDM:Detect Repair, KPI:Average Machine Utilization [%] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Reparing Cost
Reparing Time
Reparabitliy
Average Machine Utilization [%]
25
30
35
40
45
0
2
4
6
Factor Levels
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

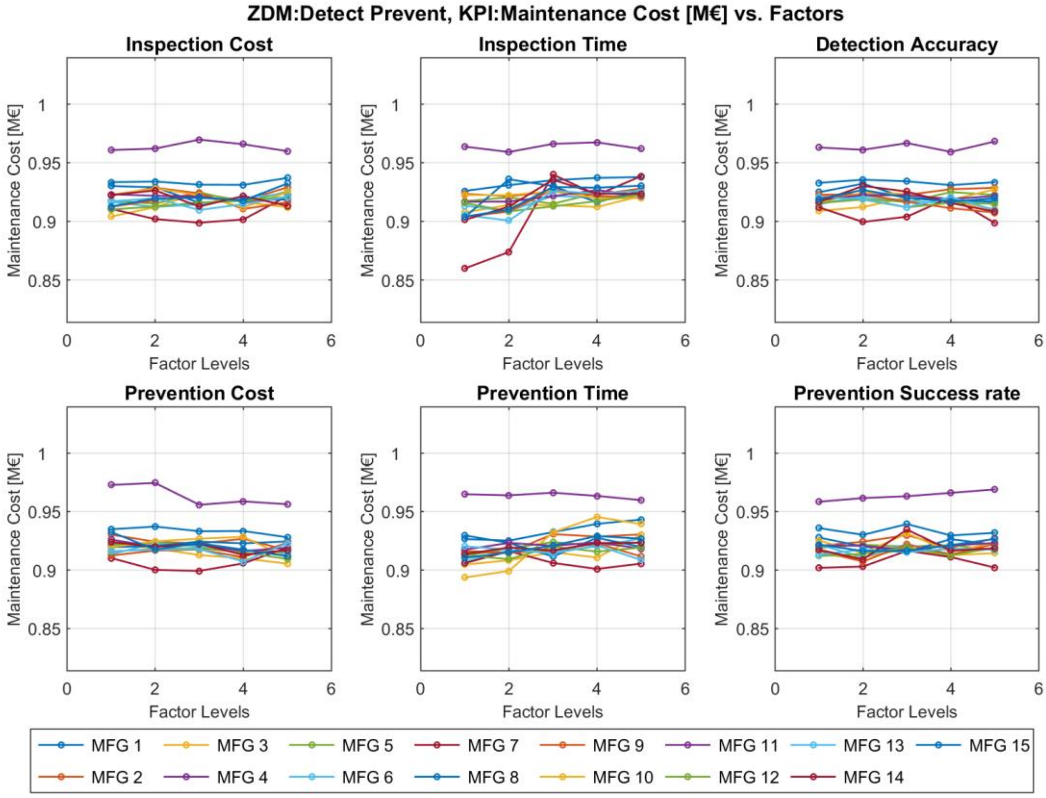

ZDM:Detect Repair, KPI:Re-Scheduling Frequency [Days] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Reparing Cost
Reparing Time
Reparabitliy
Re-Scheduling Frequency [Days]
1
1.5
2
2.5
3
3.5
4
0
2
4
6
Factor Levels
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

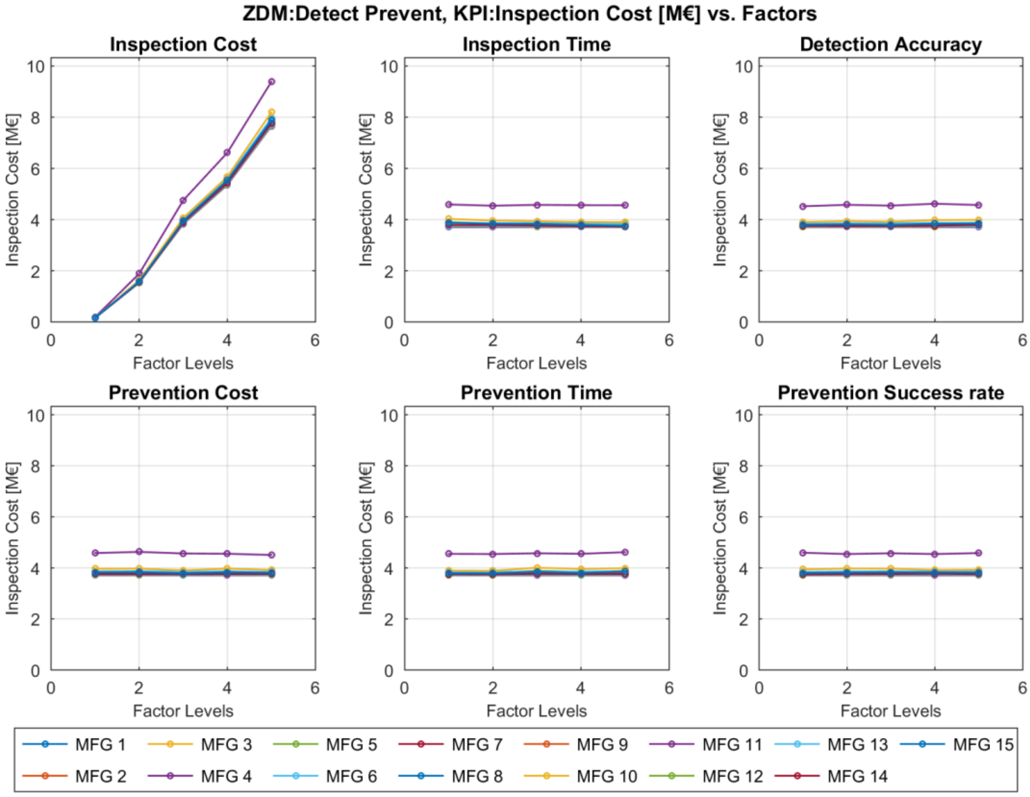
B. Detection Prevention ANOM KPIs results

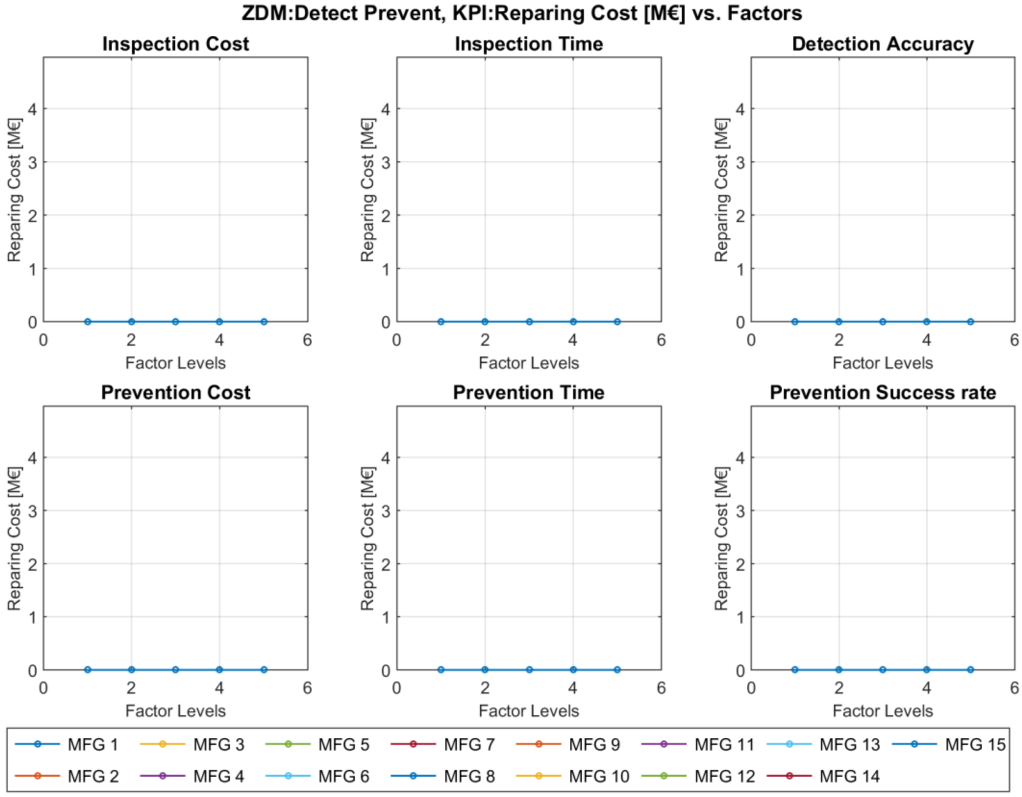


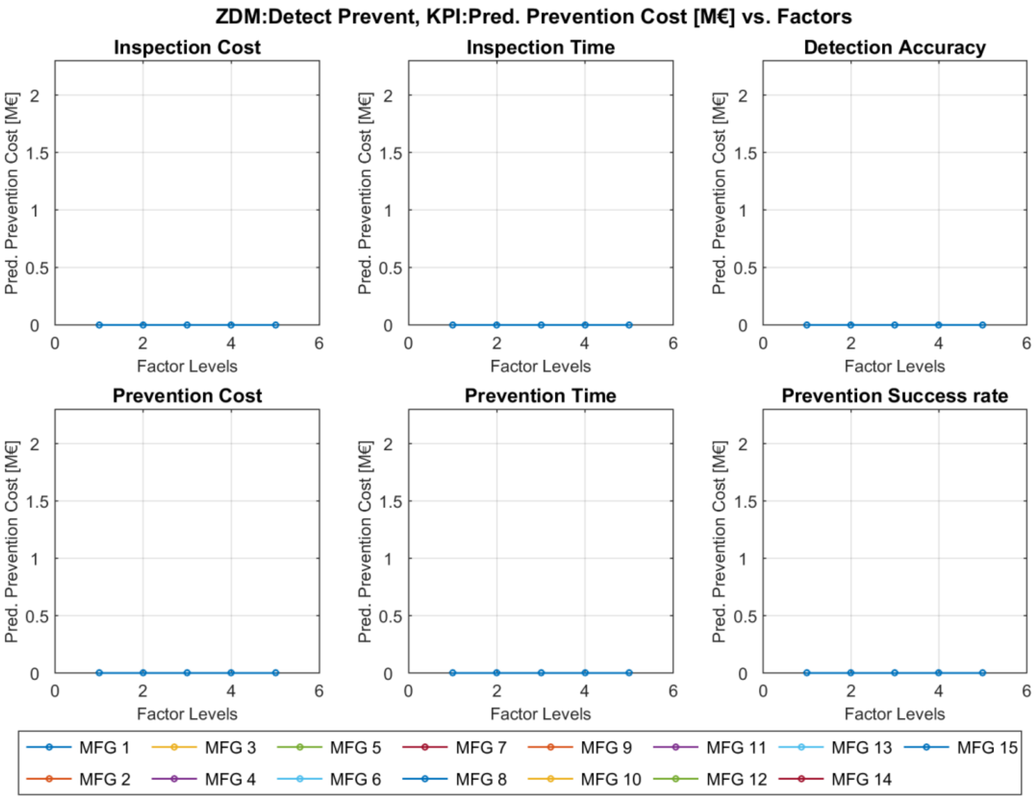


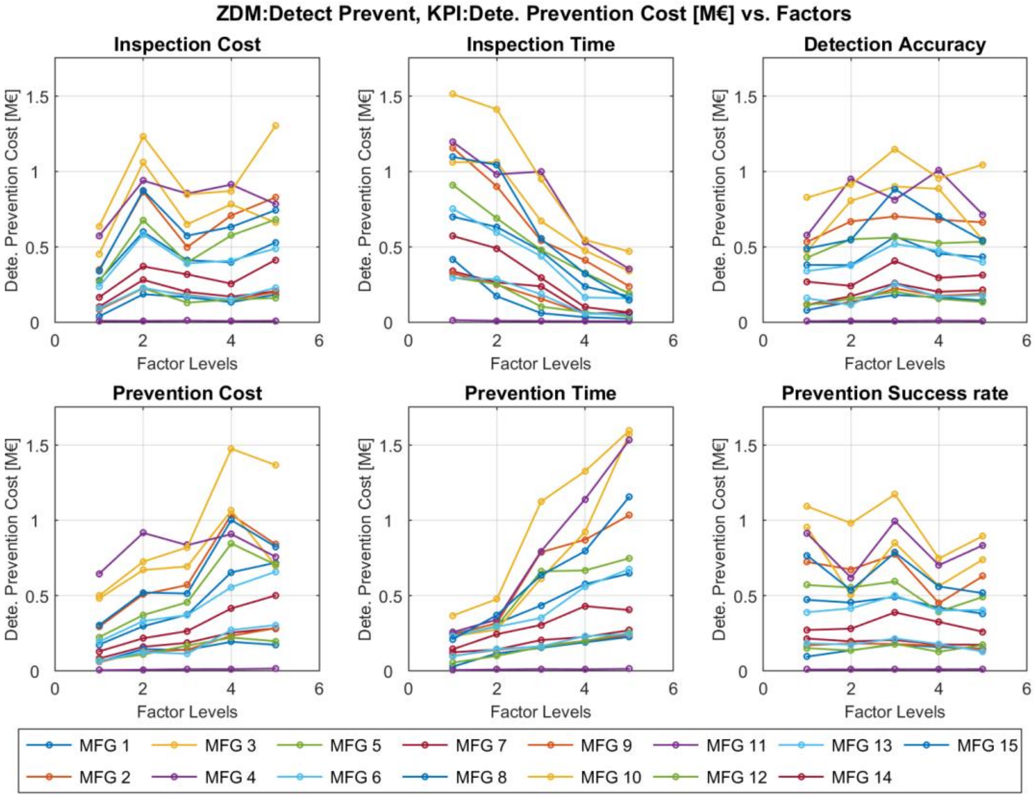

ZDM:Detect Prevent, KPI:MFG Defects ratio [%] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Prevention Cost
Prevention Time
Prevention Success rate
MFG Defects ratio [%]
Factor Levels
0
1
2
3
4
5
6
0
2
4
6
MFG 1
MFG 3
MFG 5
MFG 7
MFG 9
MFG 11
MFG 13
MFG 15
MFG 2
MFG 4
MFG 6
MFG 8
MFG 10
MFG 12
MFG 14

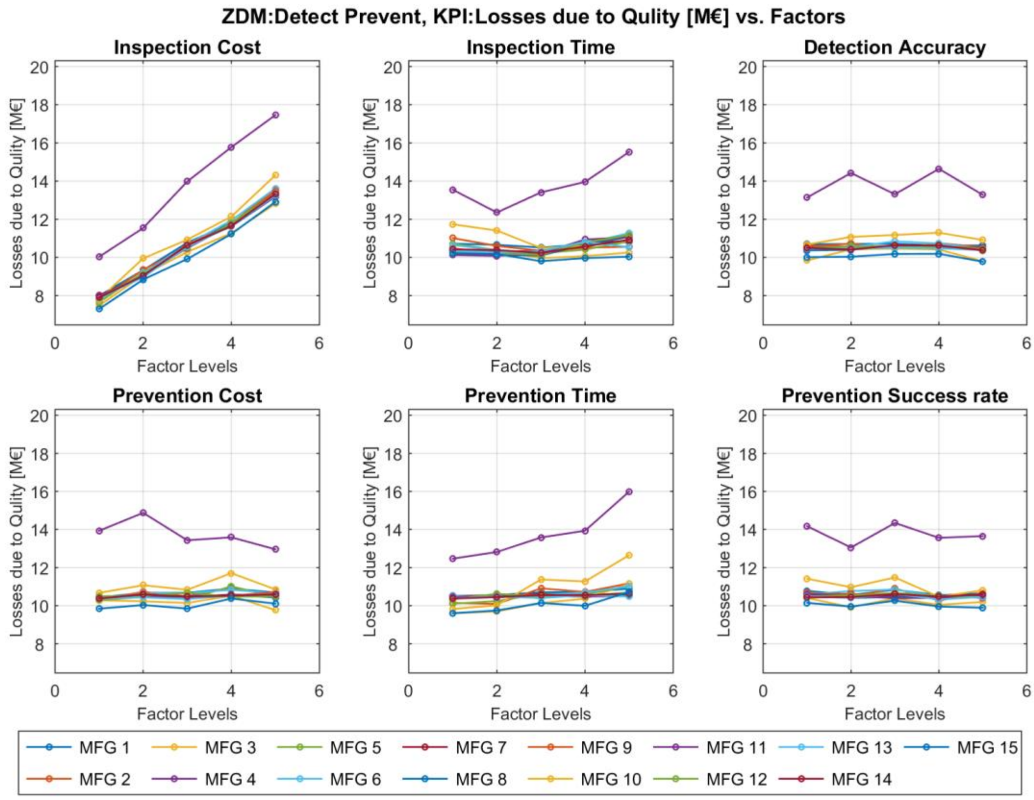

ZDM:Detect Prevent, KPI:Maintenance Cost [M€] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Prevention Cost
Prevention Time
Prevention Success rate
Maintenance Cost [M€]
Factor Levels
0.85
0.9
0.95
1
0
2
4
6
MFG 1
MFG 3
MFG 5
MFG 7
MFG 9
MFG 11
MFG 13
MFG 15
MFG 2
MFG 4
MFG 6
MFG 8
MFG 10
MFG 12
MFG 14

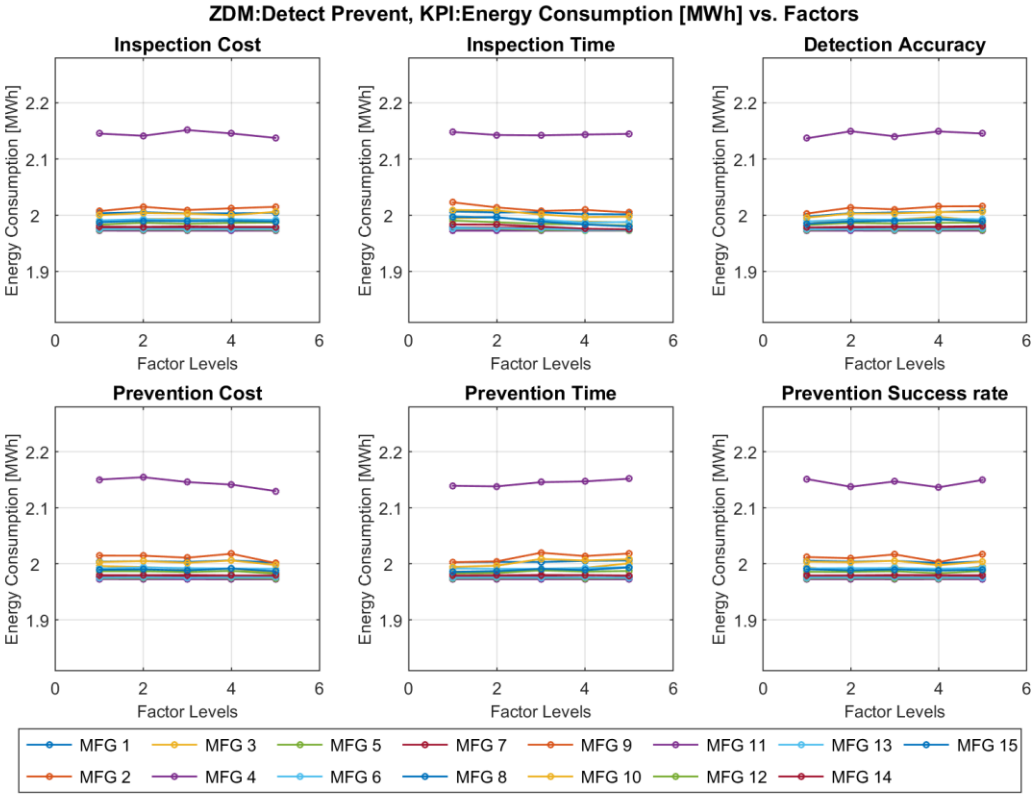

ZDM:Detect Prevent, KPI:Inspection Cost [M€] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Prevention Cost
Prevention Time
Prevention Success rate
Inspection Cost [M€]
Factor Levels
0
2
4
6
8
10
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

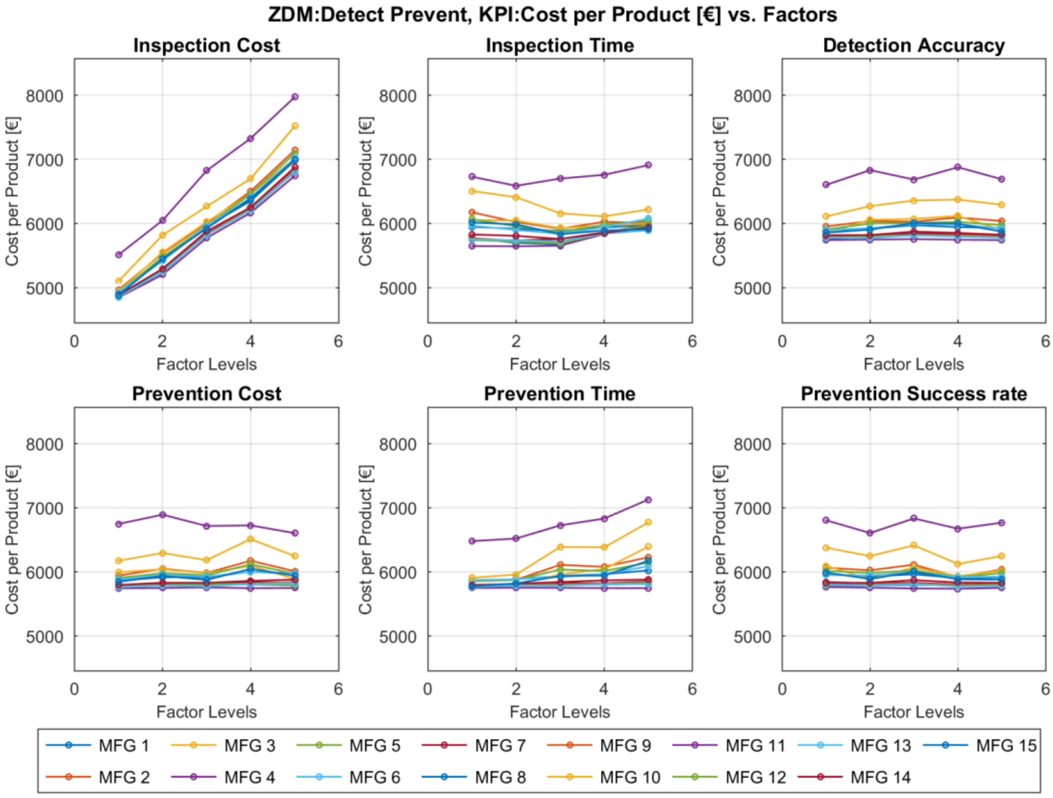

ZDM:Detect Prevent, KPI:Reparing Cost [M€] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Prevention Cost
Prevention Time
Prevention Success rate
Reparing Cost [M€]
Factor Levels
0
1
2
3
4
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

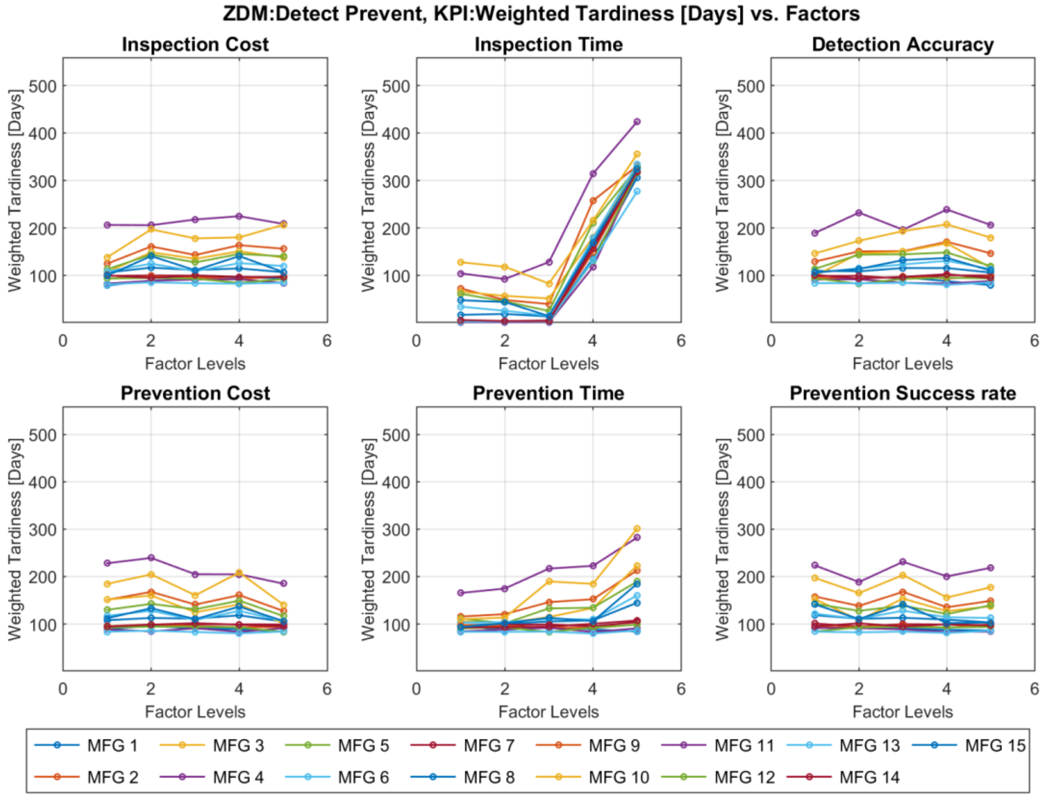

ZDM:Detect Prevent, KPI:Pred. Prevention Cost [M€] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Prevention Cost
Prevention Time
Prevention Success rate
Pred. Prevention Cost [M€]
Factor Levels
0
0.5
1
1.5
2
0
2
4
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

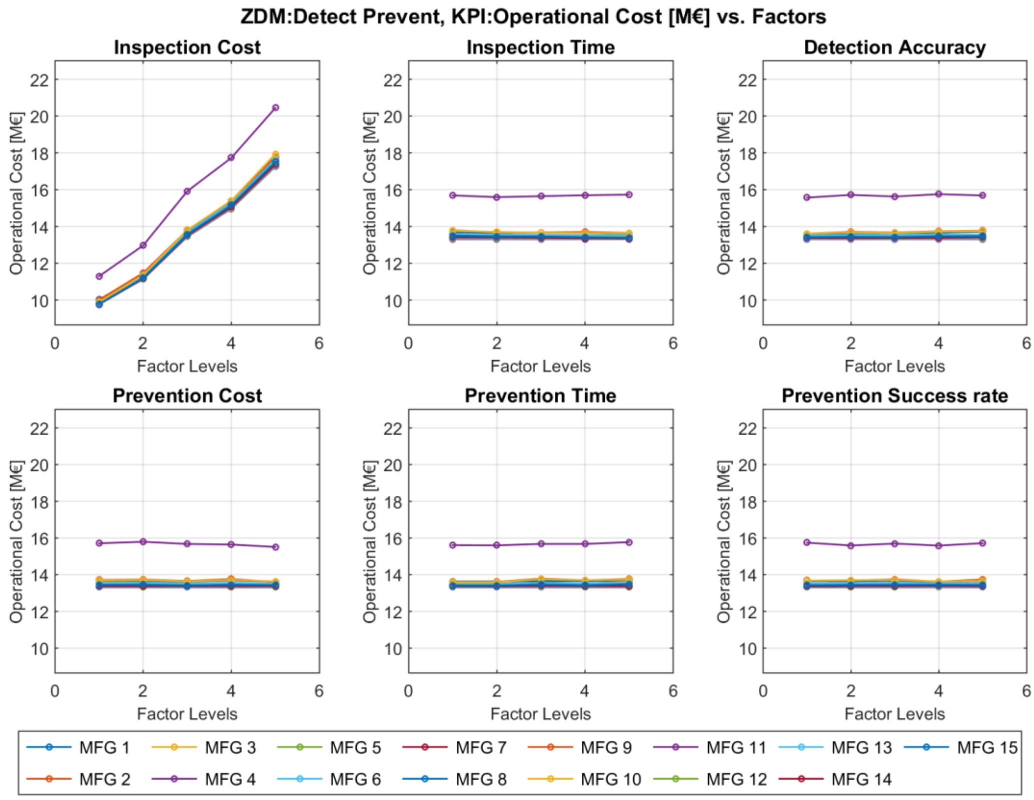

ZDM:Detect Prevent, KPI:Dete. Prevention Cost [M€] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Prevention Cost
Prevention Time
Prevention Success rate
Dete. Prevention Cost [M€]
Factor Levels
0
0.5
1
1.5
0
2
4
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

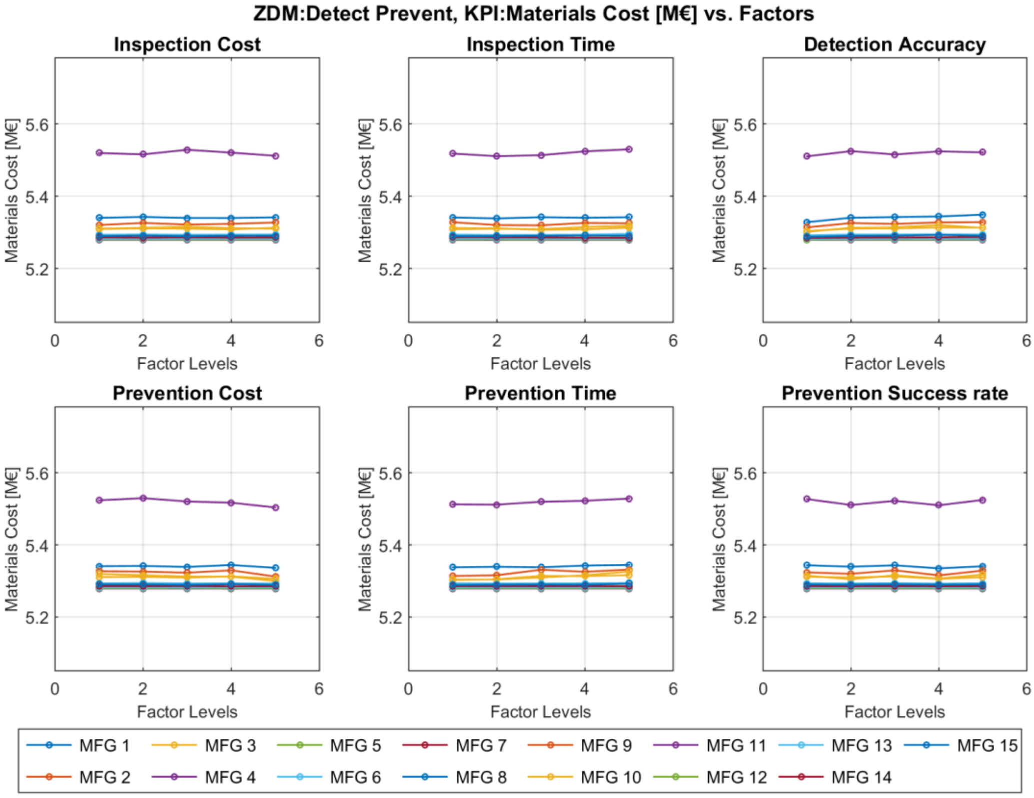

ZDM:Detect Prevent, KPI:Losses due to Qulity [M€] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Prevention Cost
Prevention Time
Prevention Success rate
Losses due to Qulity [M€]
Factor Levels
8
10
12
14
16
18
20
0
2
4
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

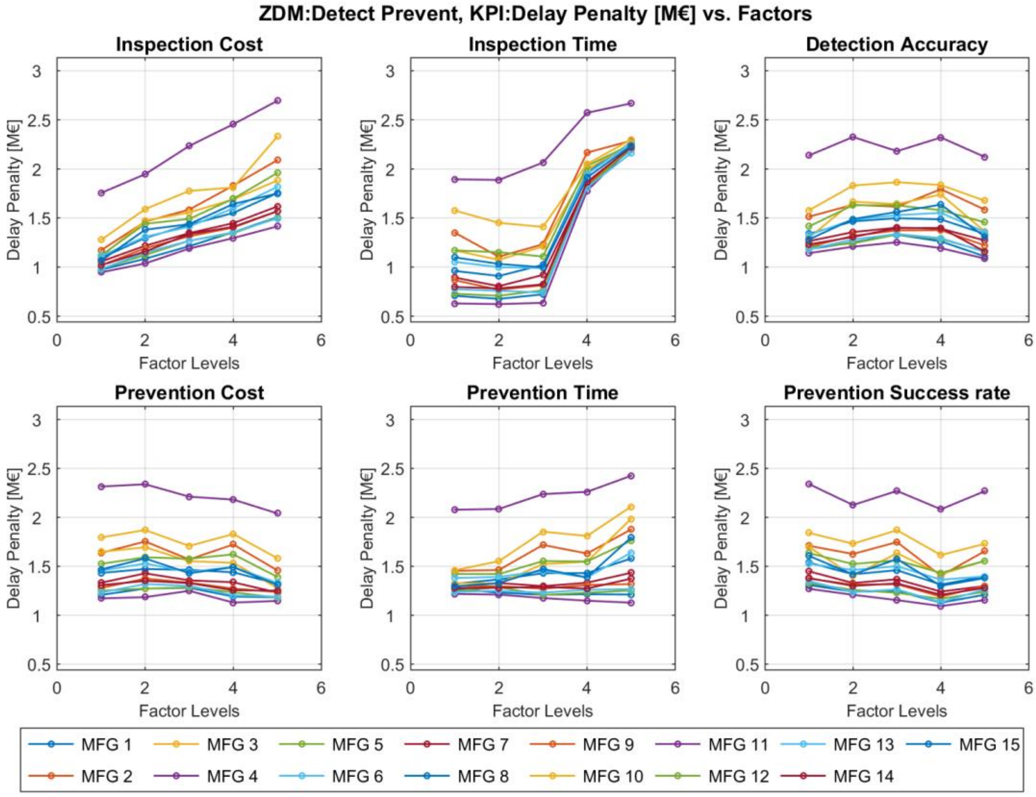

ZDM:Detect Prevent, KPI:Energy Consumption [MWh] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Prevention Cost
Prevention Time
Prevention Success rate
Energy Consumption [MWh]
Factor Levels
1.9
2
2.1
2.2
0
2
4
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

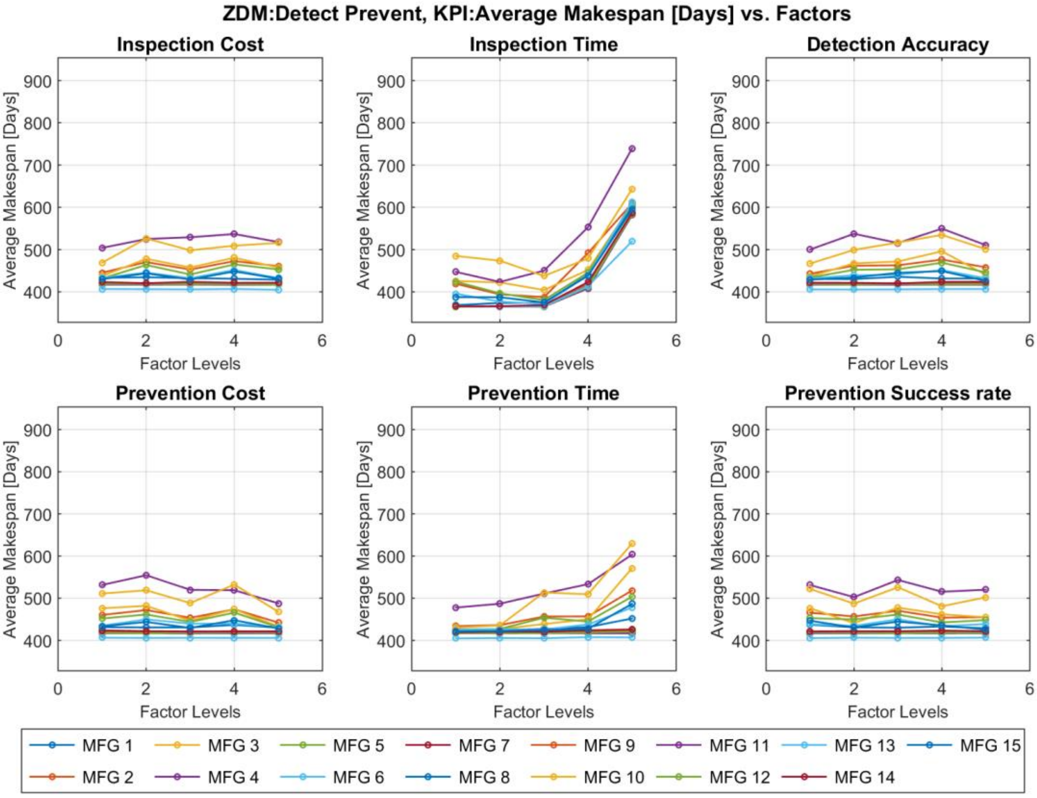

ZDM:Detect Prevent, KPI:Cost per Product [€] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Prevention Cost
Prevention Time
Prevention Success rate
Cost per Product [€]
8000
7000
6000
5000
Factor Levels
0
2
4
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

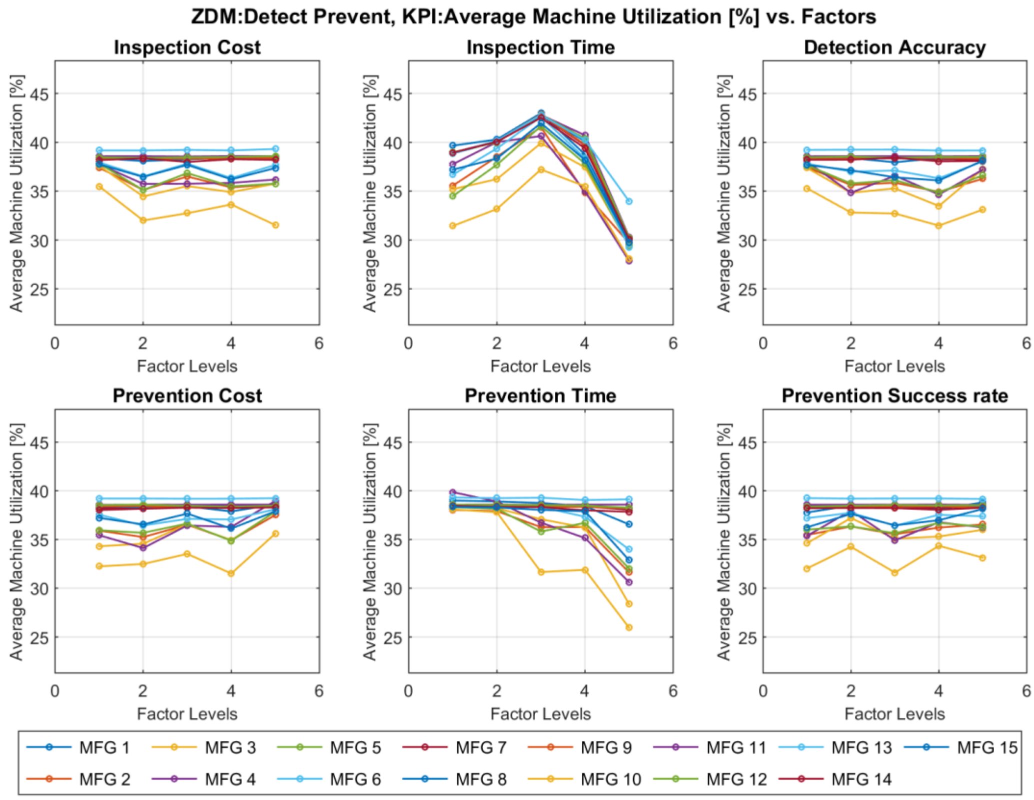

ZDM:Detect Prevent, KPI:Weighted Tardiness [Days] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Prevention Cost
Prevention Time
Prevention Success rate
Weighted Tardiness [Days]
500
400
300
200
100
Factor Levels
0
2
4
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

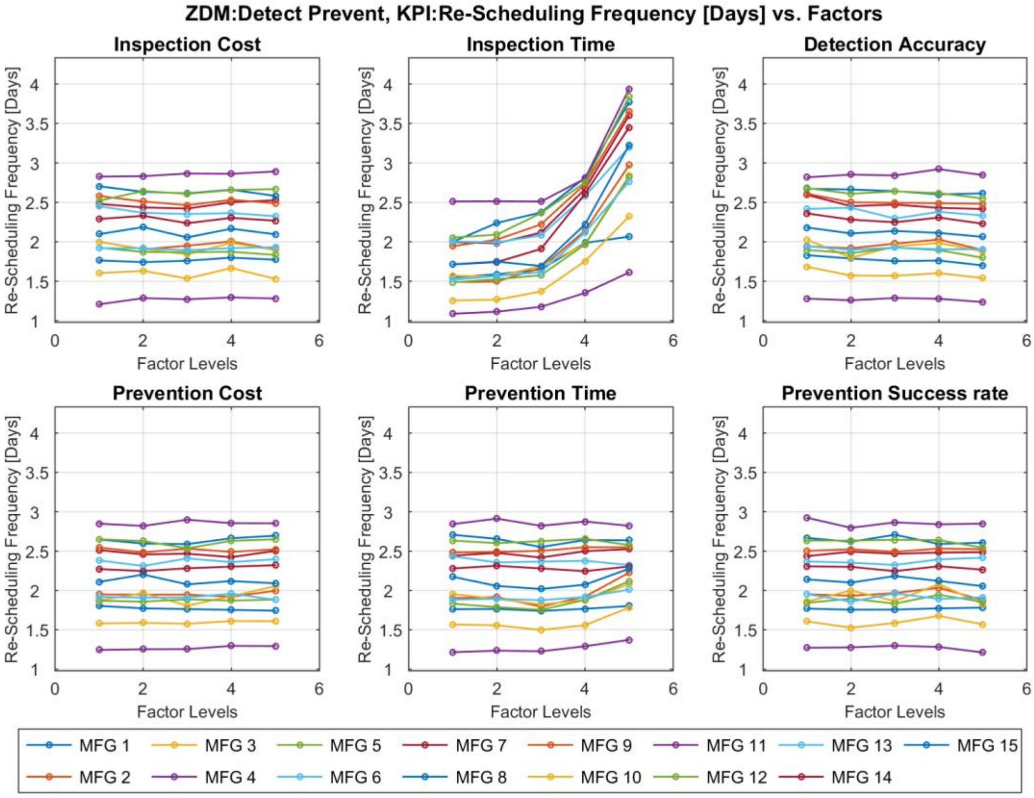

ZDM:Detect Prevent, KPI:Operational Cost [M€] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Prevention Cost
Prevention Time
Prevention Success rate
Operational Cost [M€]
10
12
14
16
18
20
22
0
2
4
6
Factor Levels
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15


ZDM:Detect Prevent, KPI:Materials Cost [M€] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Prevention Cost
Prevention Time
Prevention Success rate
Materials Cost [M€]
5.2
5.4
5.6
0
2
4
6
Factor Levels
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

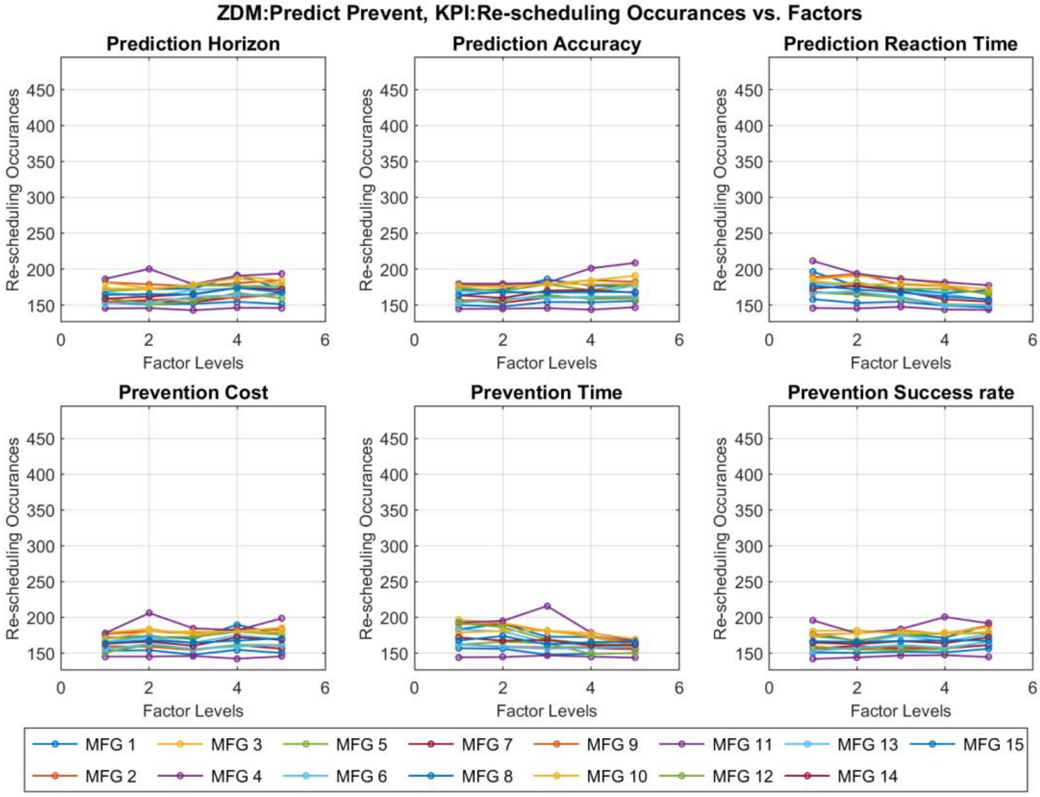

ZDM:Detect Prevent, KPI:Delay Penalty [M€] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Prevention Cost
Prevention Time
Prevention Success rate
Delay Penalty [M€]
0.5
1
1.5
2
2.5
3
0
2
4
6
Factor Levels
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

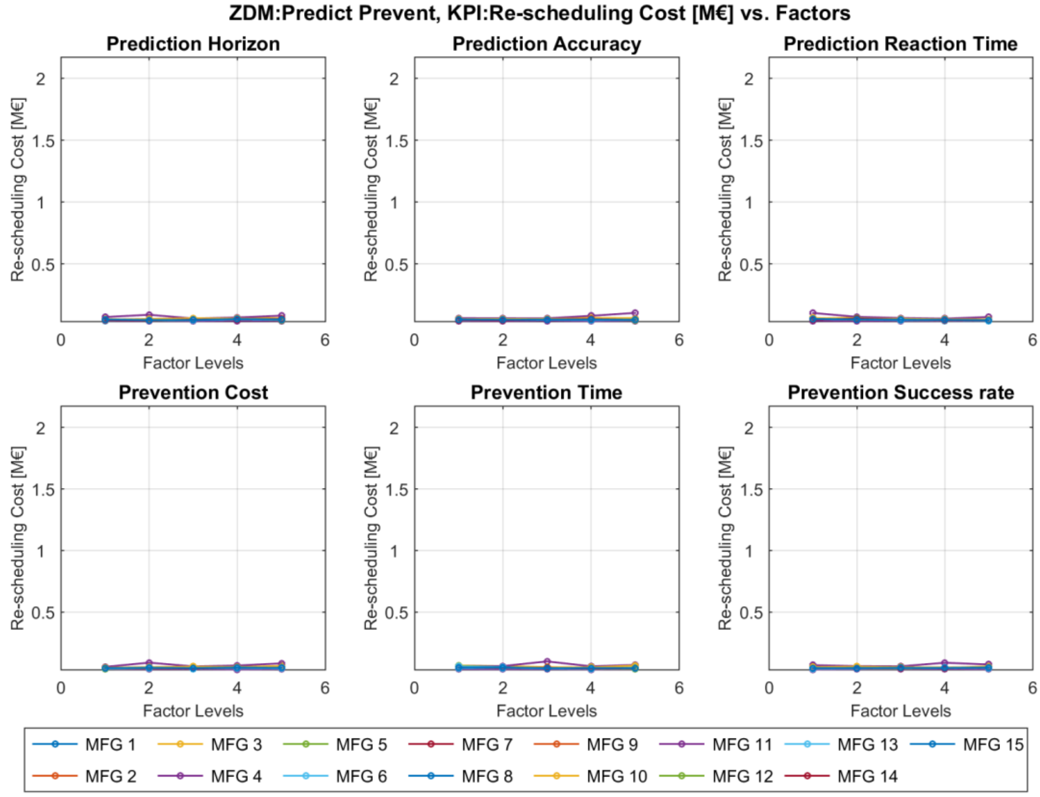

ZDM:Detect Prevent, KPI:Average Makespan [Days] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Prevention Cost
Prevention Time
Prevention Success rate
Average Makespan [Days]
400
500
600
700
800
900
0
2
4
6
Factor Levels
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

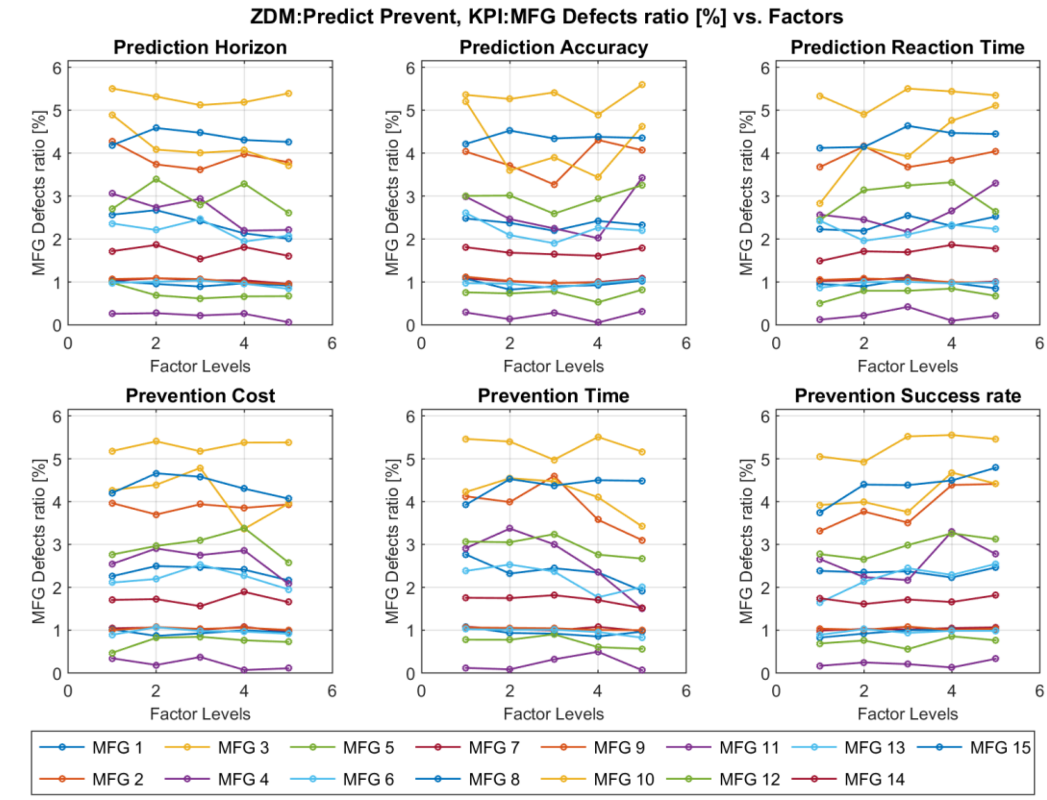

ZDM:Detect Prevent, KPI:Average Machine Utilization [%] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Prevention Cost
Prevention Time
Prevention Success rate
Average Machine Utilization [%]
25
30
35
40
45
Factor Levels
0
2
4
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

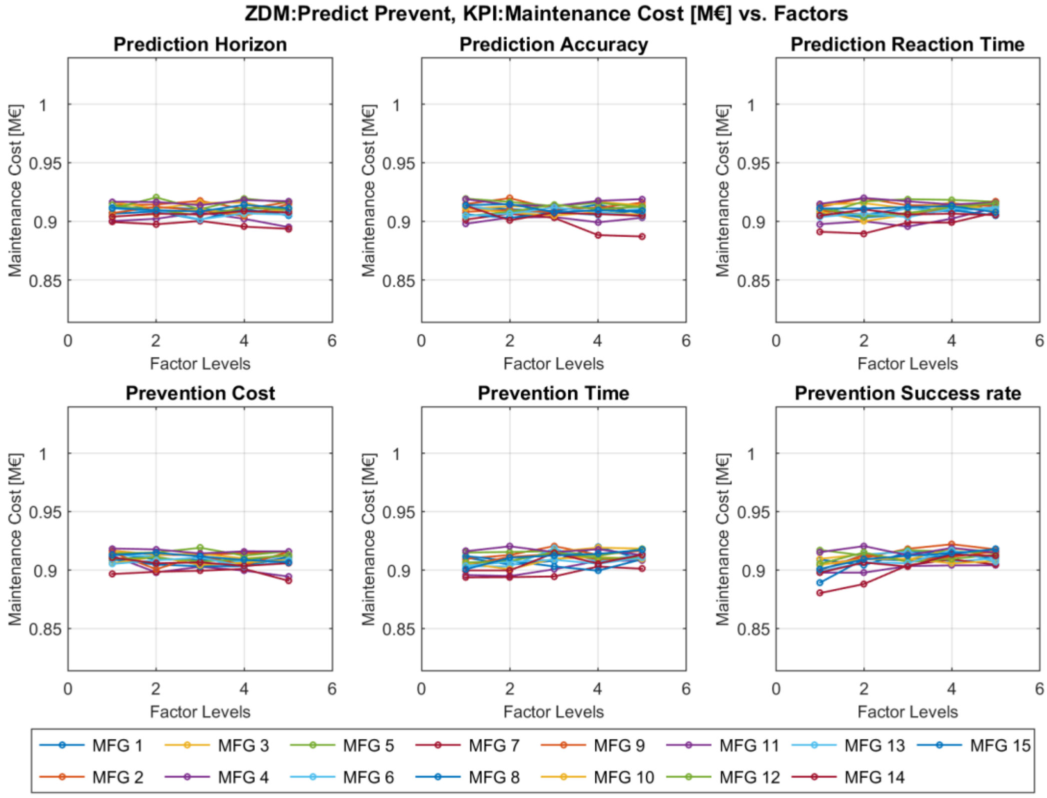

ZDM:Detect Prevent, KPI:Re-Scheduling Frequency [Days] vs. Factors
Inspection Cost
Inspection Time
Detection Accuracy
Prevention Cost
Prevention Time
Prevention Success rate
Re-Scheduling Frequency [Days]
1
1.5
2
2.5
3
3.5
4
Factor Levels
0
2
4
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

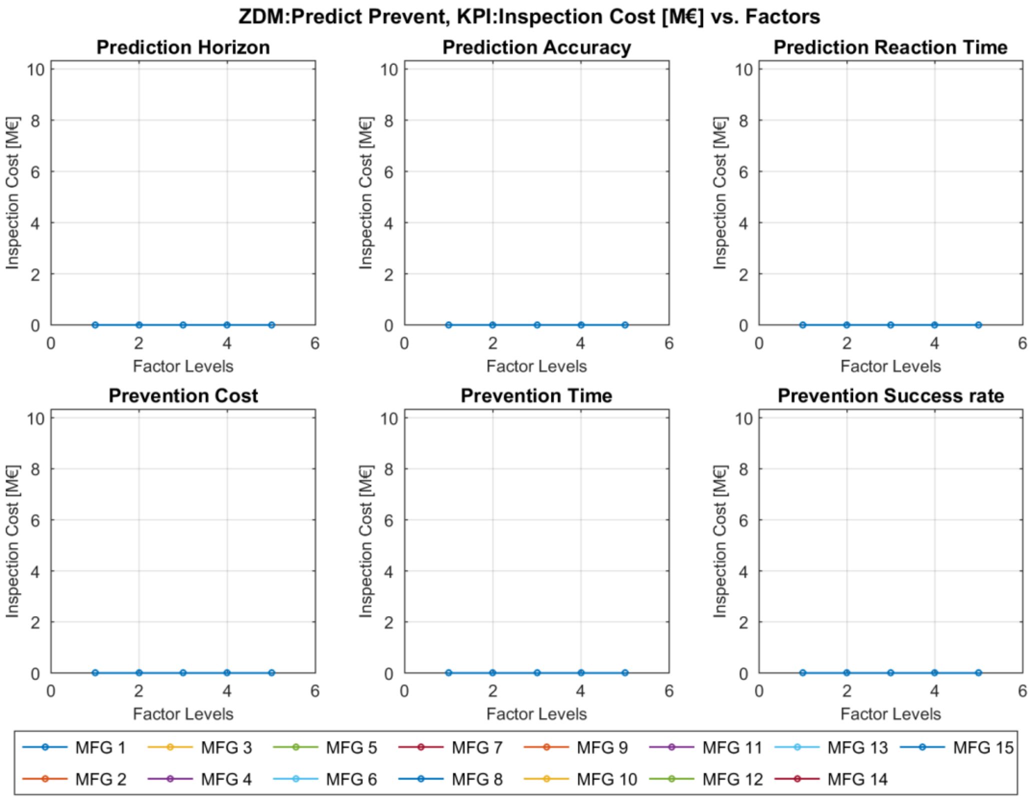
C. Prediction Prevention ANOM KPIs results

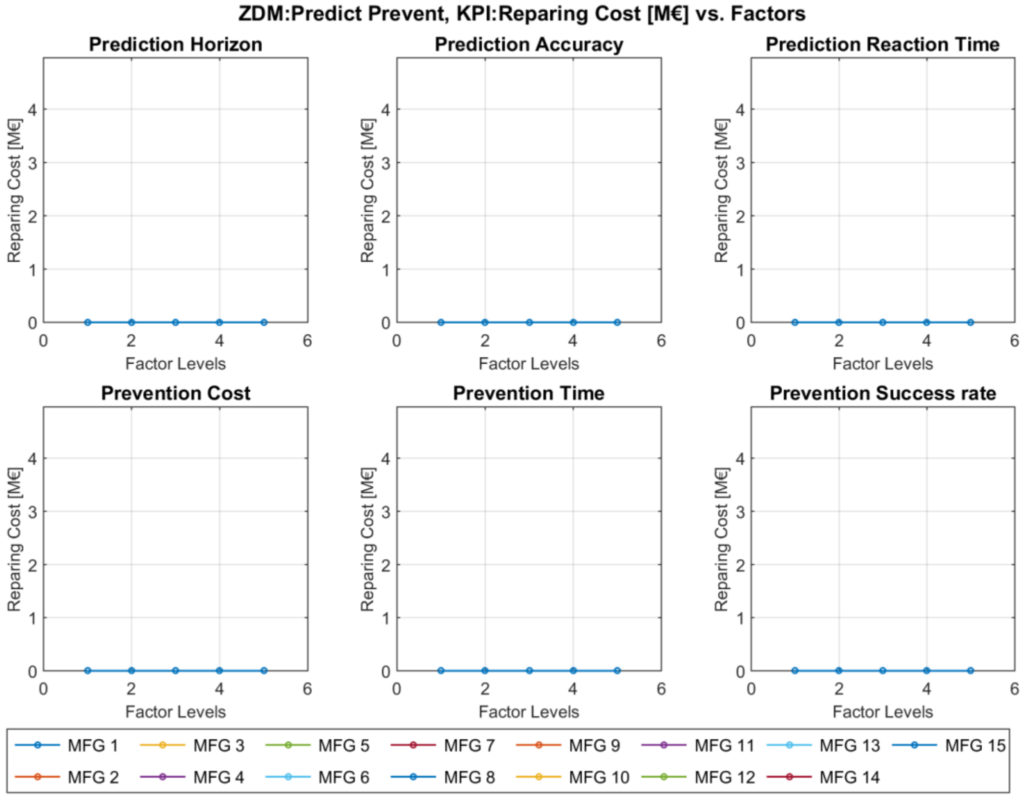


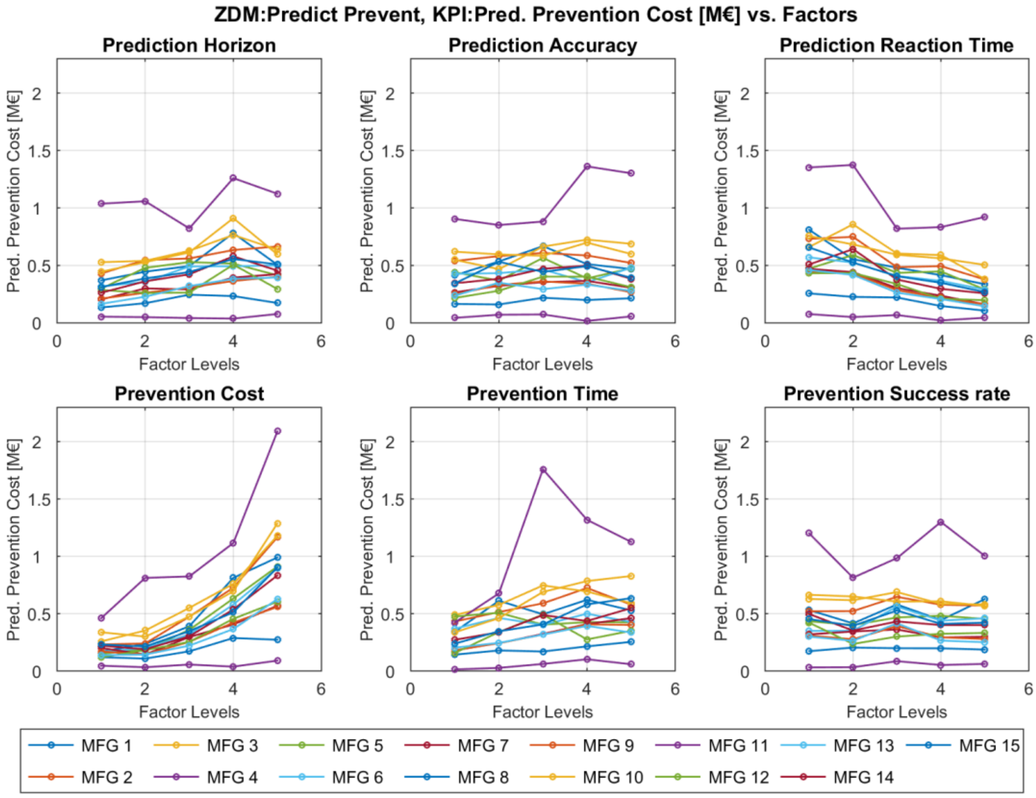


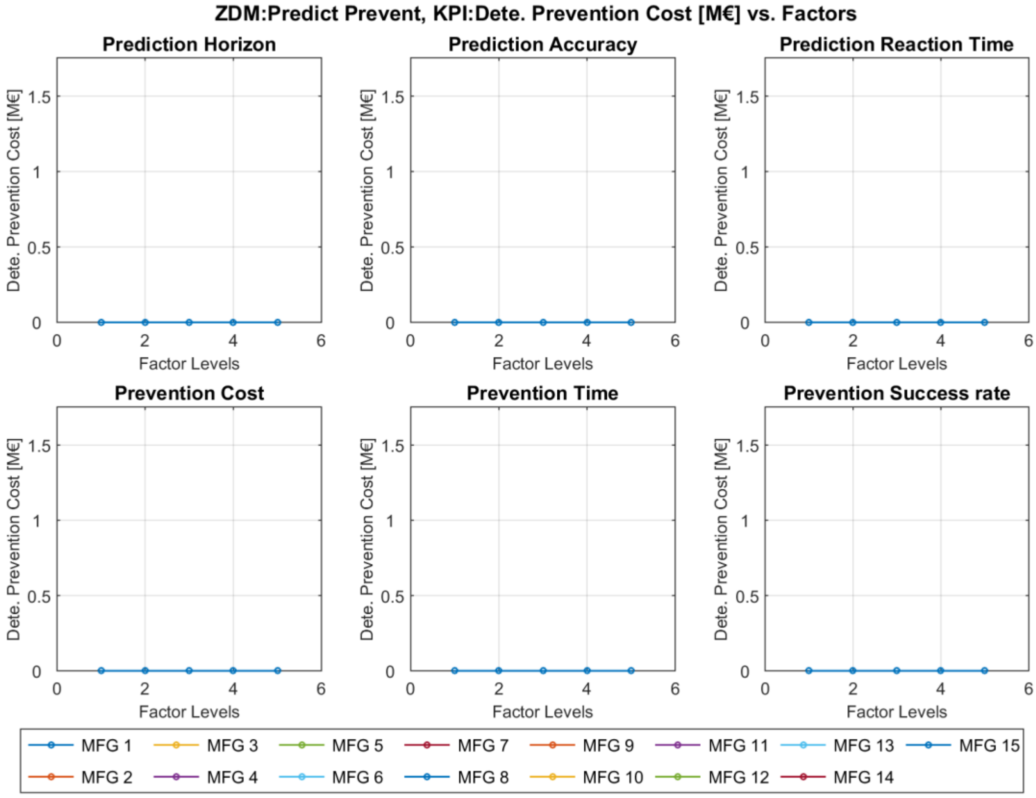

ZDM:Predict Prevent, KPI:MFG Defects ratio [%] vs. Factors
Prediction Horizon
Prediction Accuracy
Prediction Reaction Time
Prevention Cost
Prevention Time
Prevention Success rate
MFG Defects ratio [%]
Factor Levels
0
1
2
3
4
5
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

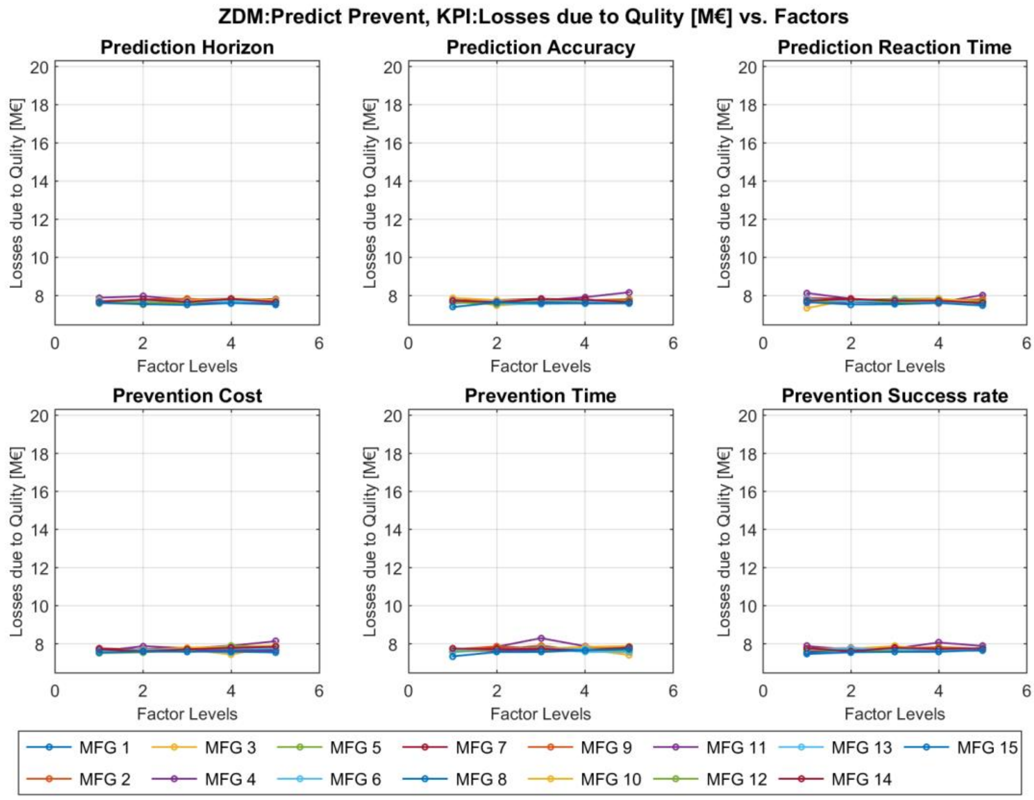

ZDM:Predict Prevent, KPI:Maintenance Cost [M€] vs. Factors
Prediction Horizon
Prediction Accuracy
Prediction Reaction Time
Prevention Cost
Prevention Time
Prevention Success rate
Maintenance Cost [M€]
0.85
0.9
0.95
1
Factor Levels
0
2
4
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

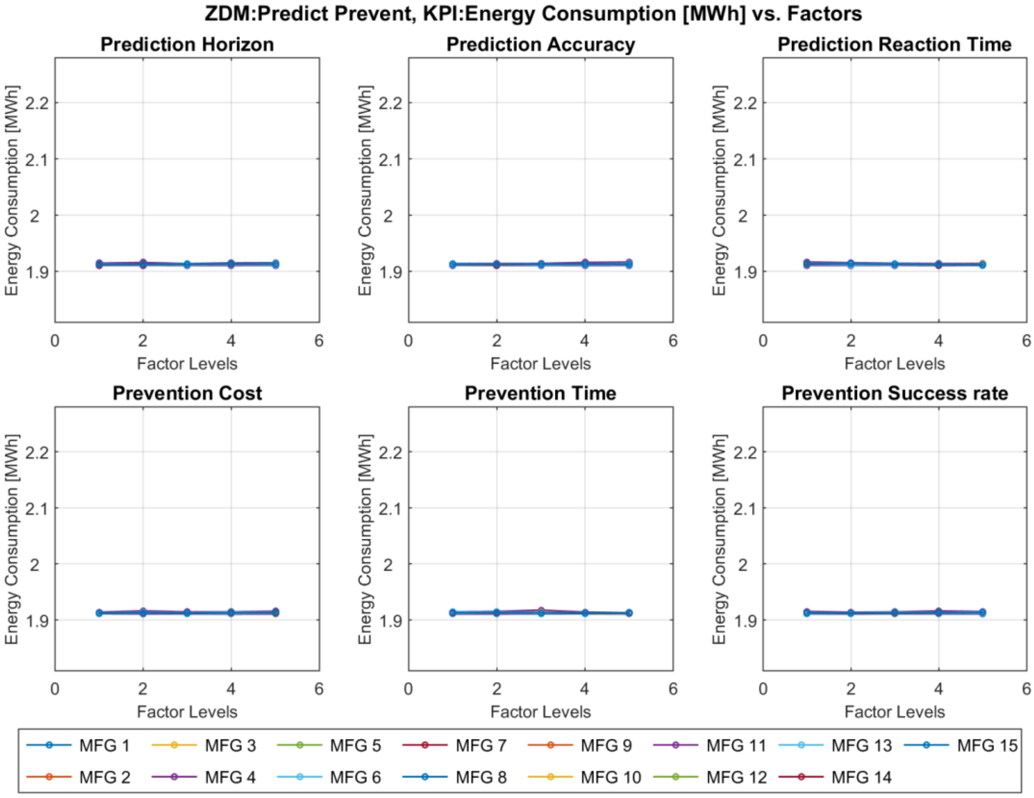

ZDM:Predict Prevent, KPI:Inspection Cost [M€] vs. Factors
Prediction Horizon
Prediction Accuracy
Prediction Reaction Time
Prevention Cost
Prevention Time
Prevention Success rate
Inspection Cost [M€]
Factor Levels
0
2
4
6
8
10
0
2
4
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

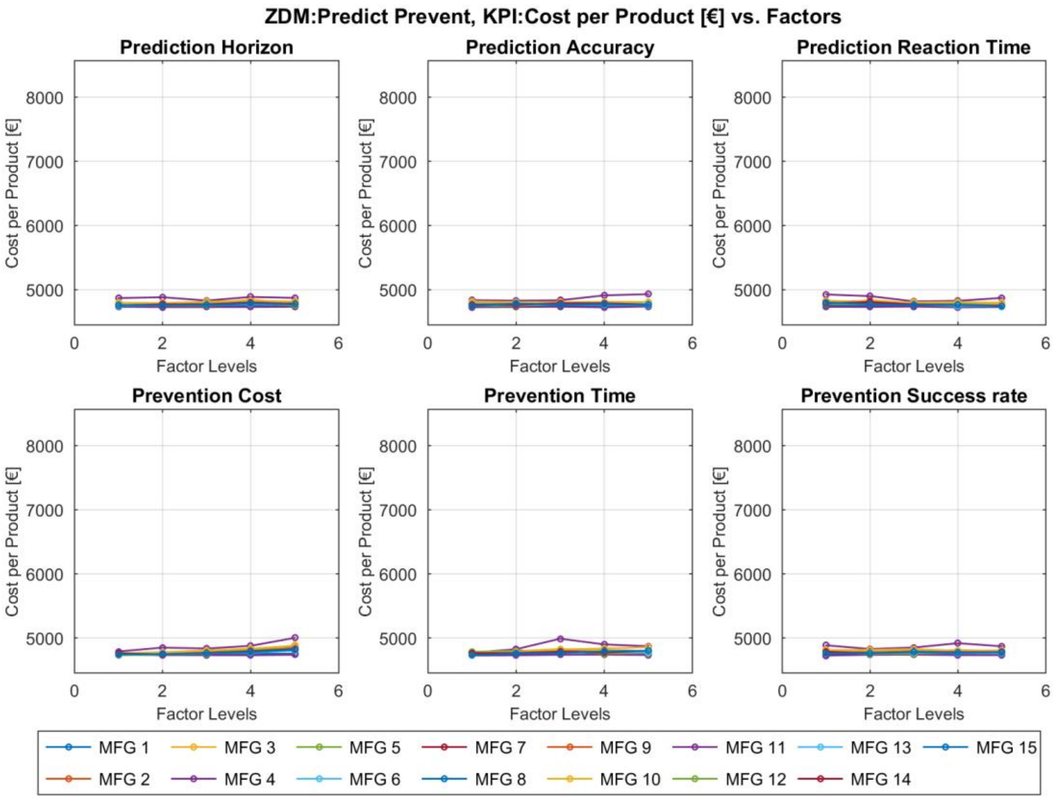

ZDM:Predict Prevent, KPI:Reparing Cost [M€] vs. Factors
Prediction Horizon
Prediction Accuracy
Prediction Reaction Time
Prevention Cost
Prevention Time
Prevention Success rate
Reparing Cost [M€]
Factor Levels
0
1
2
3
4
0
2
4
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

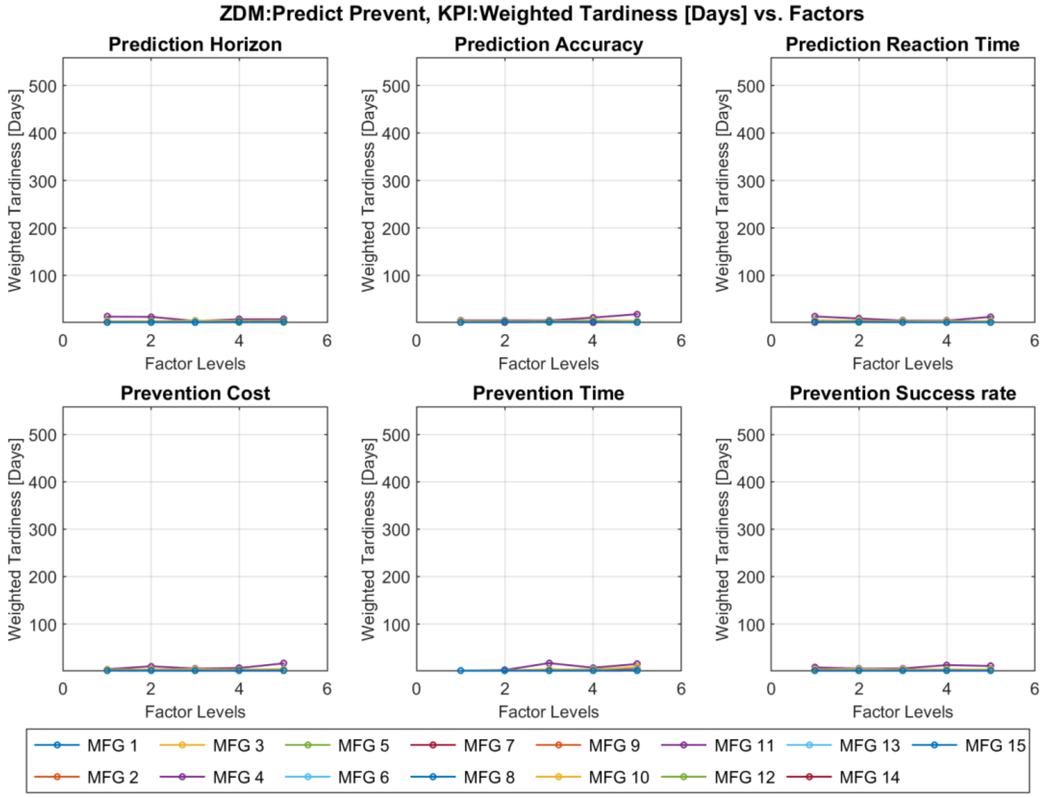

ZDM:Predict Prevent, KPI:Pred. Prevention Cost [M€] vs. Factors
Prediction Horizon
Prediction Accuracy
Prediction Reaction Time
Prevention Cost
Prevention Time
Prevention Success rate
Pred. Prevention Cost [M€]
0
0.5
1
1.5
2
0
2
4
6
Factor Levels
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

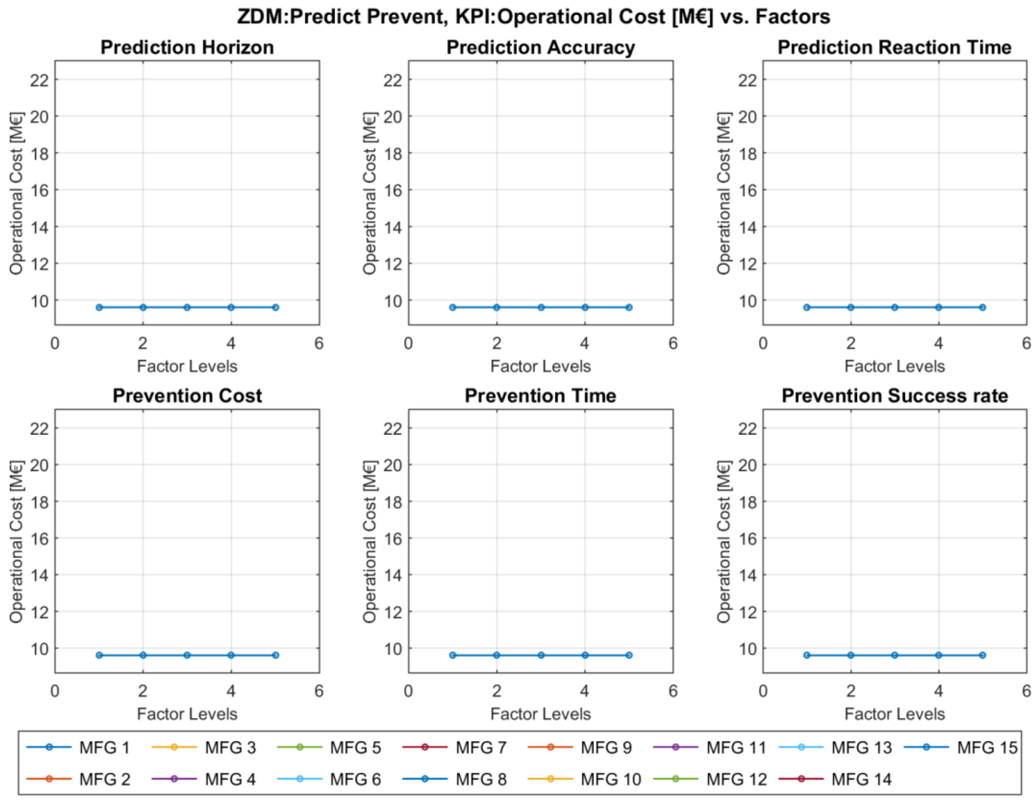

ZDM:Predict Prevent, KPI:Dete. Prevention Cost [M€] vs. Factors
Prediction Horizon
Prediction Accuracy
Prediction Reaction Time
Prevention Cost
Prevention Time
Prevention Success rate
Dete. Prevention Cost [M€]
0
0.5
1
1.5
0
2
4
6
Factor Levels
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

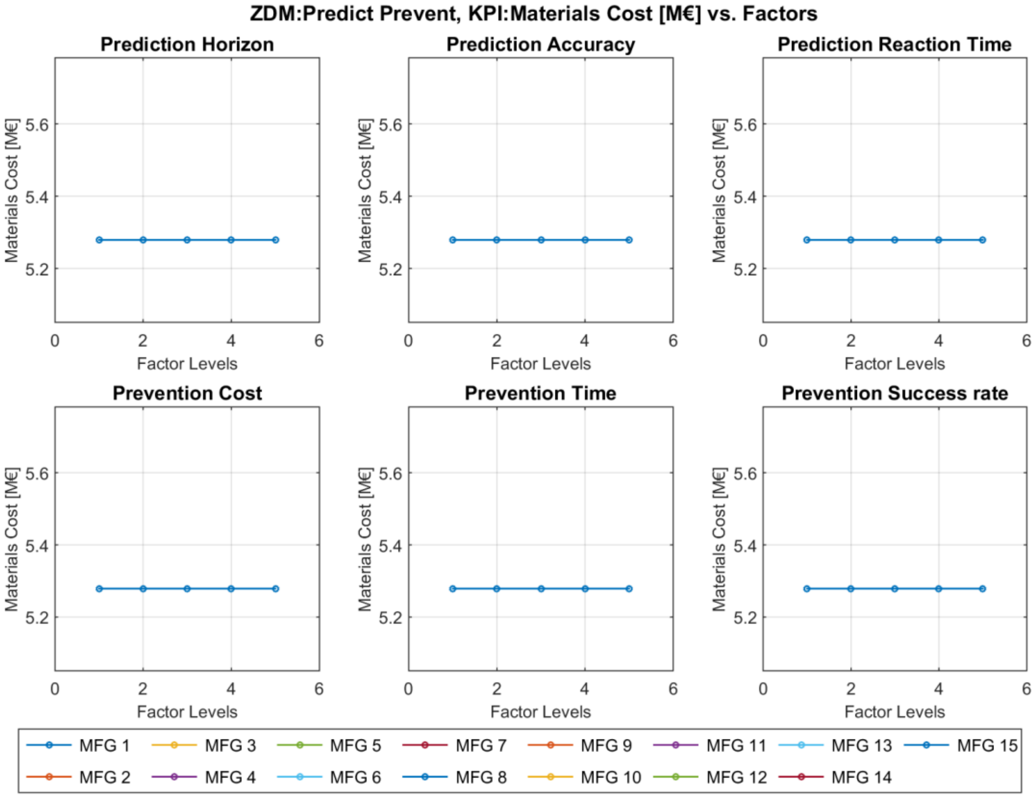

ZDM:Predict Prevent, KPI:Losses due to Qulity [M€] vs. Factors
Prediction Horizon
Prediction Accuracy
Prediction Reaction Time
Prevention Cost
Prevention Time
Prevention Success rate
Losses due to Qulity [M€]
Factor Levels
8
10
12
14
16
18
20
0
2
4
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

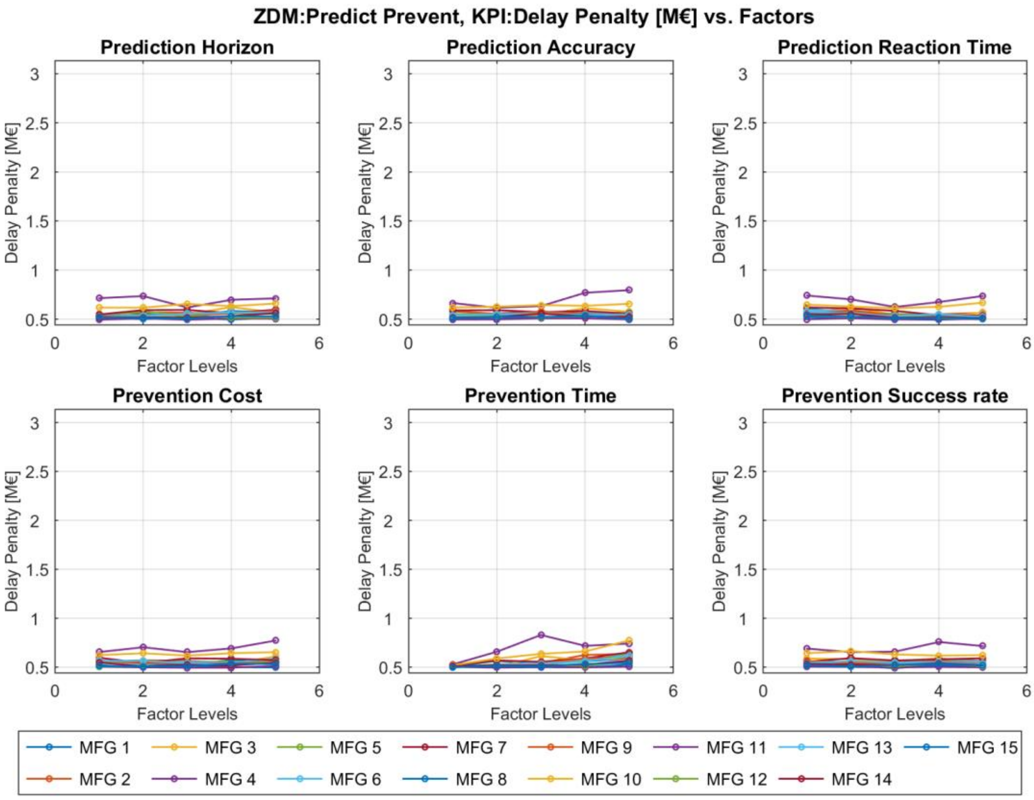

ZDM:Predict Prevent, KPI:Energy Consumption [MWh] vs. Factors
Prediction Horizon
Prediction Accuracy
Prediction Reaction Time
Prevention Cost
Prevention Time
Prevention Success rate
Energy Consumption [MWh]
Factor Levels
1.9
2
2.1
2.2
0
2
4
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

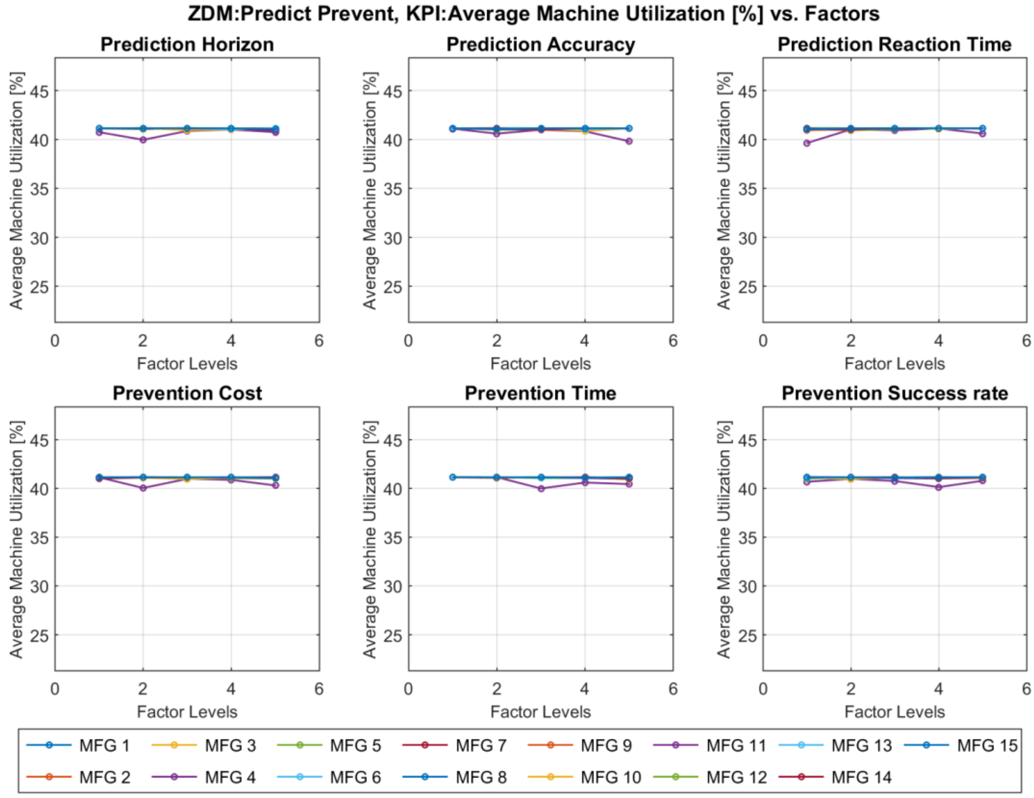

ZDM:Predict Prevent, KPI:Cost per Product [€] vs. Factors
Prediction Horizon
Prediction Accuracy
Prediction Reaction Time
Prevention Cost
Prevention Time
Prevention Success rate
Cost per Product [€]
8000
7000
6000
5000
0
2
4
6
Factor Levels
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

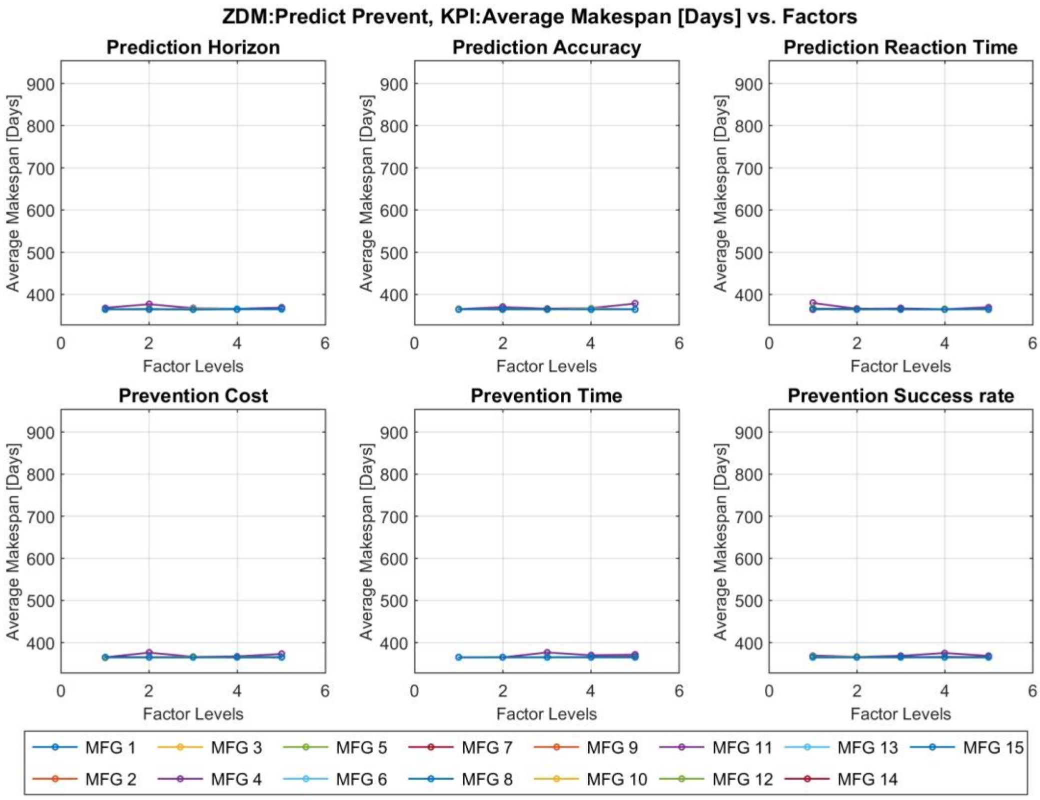

ZDM:Predict Prevent, KPI:Weighted Tardiness [Days] vs. Factors
Prediction Horizon
Prediction Accuracy
Prediction Reaction Time
Prevention Cost
Prevention Time
Prevention Success rate
Weighted Tardiness [Days]
500
400
300
200
100
0
2
4
6
Factor Levels
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15

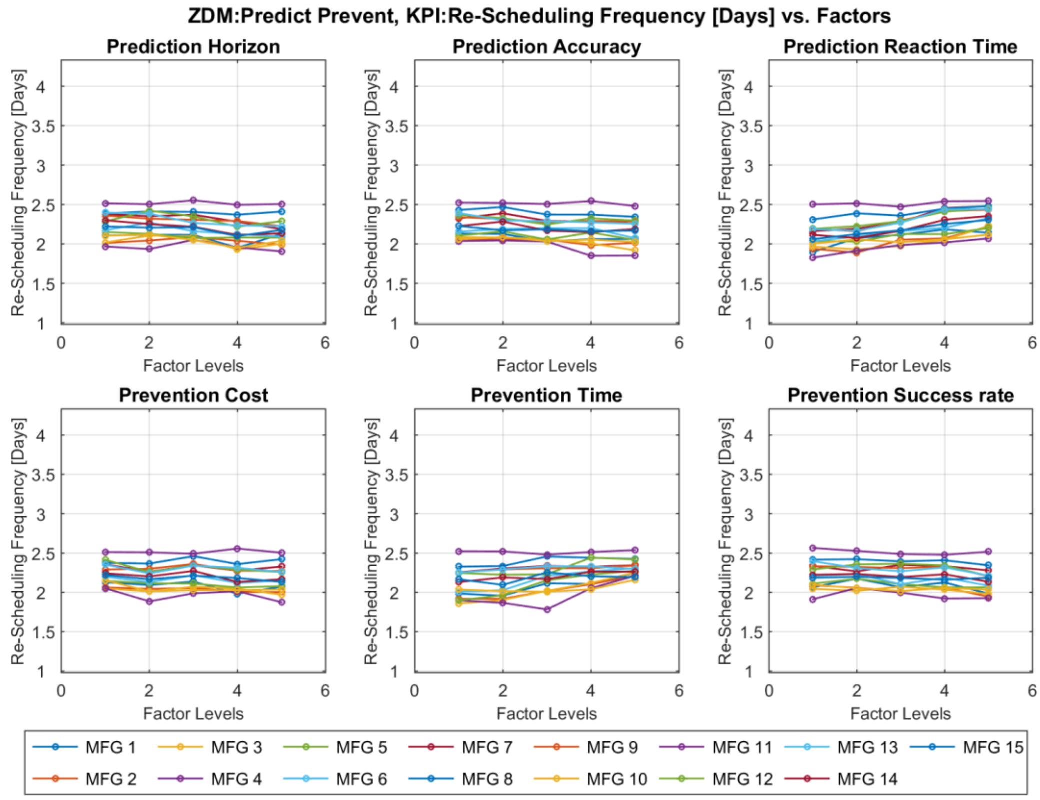

ZDM:Predict Prevent, KPI:Operational Cost [M€] vs. Factors
Prediction Horizon
Prediction Accuracy
Prediction Reaction Time
Prevention Cost
Prevention Time
Prevention Success rate
Operational Cost [M€]
10
12
14
16
18
20
22
Factor Levels
0
2
4
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15


ZDM:Predict Prevent, KPI:Materials Cost [M€] vs. Factors
Prediction Horizon
Prediction Accuracy
Prediction Reaction Time
Prevention Cost
Prevention Time
Prevention Success rate
Materials Cost [M€]
5.2
5.4
5.6
Factor Levels
0
2
4
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15


ZDM:Predict Prevent, KPI:Delay Penalty [M€] vs. Factors
Prediction Horizon
Prediction Accuracy
Prediction Reaction Time
Prevention Cost
Prevention Time
Prevention Success rate
Delay Penalty [M€]
0.5
1
1.5
2
2.5
3
Factor Levels
0
2
4
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15


ZDM:Predict Prevent, KPI:Average Machine Utilization [%] vs. Factors
Prediction Horizon
Prediction Accuracy
Prediction Reaction Time
Prevention Cost
Prevention Time
Prevention Success rate
Average Machine Utilization [%]
25
30
35
40
45
Factor Levels
0
2
4
6
MFG 1
MFG 2
MFG 3
MFG 4
MFG 5
MFG 6
MFG 7
MFG 8
MFG 9
MFG 10
MFG 11
MFG 12
MFG 13
MFG 14
MFG 15


ZDM:Predict Prevent, KPI:Average Makespan [Days] vs. Factors
Prediction Horizon
Prediction Accuracy
Prediction Reaction Time
Prevention Cost
Prevention Time
Prevention Success rate
Average Makespan [Days]
400
500
600
700
800
900
Factor Levels
0
2
4
6
MFG 1
MFG 3
MFG 5
MFG 7
MFG 9
MFG 11
MFG 13
MFG 15
MFG 2
MFG 4
MFG 6
MFG 8
MFG 10
MFG 12
MFG 14


ZDM:Predict Prevent, KPI:Re-Scheduling Frequency [Days] vs. Factors
Prediction Horizon
Prediction Accuracy
Prediction Reaction Time
Prevention Cost
Prevention Time
Prevention Success rate
Re-Scheduling Frequency [Days]
1
1.5
2
2.5
3
3.5
4
Factor Levels
0
2
4
6
MFG 1
MFG 3
MFG 5
MFG 7
MFG 9
MFG 11
MFG 13
MFG 15
MFG 2
MFG 4
MFG 6
MFG 8
MFG 10
MFG 12
MFG 14

Annex 4 (Orthogonal Arrays used in experiments)

A. L_{25} Orthogonal Array

Table 49: L_{25} Orthogonal Array

	Factors					
Exp. No.	**X1**	**X2**	**X3**	**X4**	**X5**	**X6**
1	1	1	1	1	1	1
2	1	2	2	2	2	2
3	1	3	3	3	3	3
4	1	4	4	4	4	4
5	1	5	5	5	5	5
6	2	1	2	3	4	5
7	2	2	3	4	5	1
8	2	3	4	5	1	2
9	2	4	5	1	2	3
10	2	5	1	2	3	4
11	3	1	3	5	2	4
12	3	2	4	1	3	5
13	3	3	5	2	4	1
14	3	4	1	3	5	2
15	3	5	2	4	1	3
16	4	1	4	2	5	3
17	4	2	5	3	1	4
18	4	3	1	4	2	5
19	4	4	2	5	3	1
20	4	5	3	1	4	2
21	5	1	5	4	3	2
22	5	2	1	5	4	3
23	5	3	2	1	5	4
24	5	4	3	2	1	5
25	5	5	4	3	2	1

B. L32 Orthogonal Array and primary results of chapter 4.8

- f1: NORT
- f2: NORDT
- f3: NDRT
- f4: NDDRT
- f5: NPRT
- f6: NPDRT
- f7: Prediction horizon

Table 50: L_{32} orthogonal Array

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	20	22	23	24	31
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2
3	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	1	1	1	1	1	1	2	2
4	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	1	1
5	1	1	1	2	2	2	2	1	1	1	1	2	2	2	2	1	1	1	2	2	2	1	2
6	1	1	1	2	2	2	2	1	1	1	1	2	2	2	2	2	2	2	1	1	1	2	1
7	1	1	1	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	2	2	2	2	1
8	1	1	1	2	2	2	2	2	2	2	2	1	1	1	1	2	2	2	1	1	1	1	2
9	1	2	2	1	1	2	2	1	1	2	2	1	1	2	2	1	1	2	1	2	2	1	2
10	1	2	2	1	1	2	2	1	1	2	2	1	1	2	2	2	2	1	2	1	1	2	1
11	1	2	2	1	1	2	2	2	2	1	1	2	2	1	1	1	1	2	1	2	2	2	1
12	1	2	2	1	1	2	2	2	2	1	1	2	2	1	1	2	2	1	2	1	1	1	2
13	1	2	2	2	2	1	1	1	1	2	2	2	2	1	1	1	1	2	2	1	1	1	1
14	1	2	2	2	2	1	1	1	1	2	2	2	2	1	1	2	2	1	1	2	2	2	2
15	1	2	2	2	2	1	1	2	2	1	1	1	1	2	2	1	1	2	2	1	1	2	2
16	1	2	2	2	2	1	1	2	2	1	1	1	1	2	2	2	2	1	1	2	2	1	1
17	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	1	1	2	1	2
18	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	2	1	2	2	2	1	2	1
19	2	1	2	1	2	1	2	2	1	2	1	2	1	2	1	1	2	1	1	1	2	2	1
20	2	1	2	1	2	1	2	2	1	2	1	2	1	2	1	2	1	2	2	2	1	1	2
21	2	1	2	2	1	2	1	1	2	1	2	2	1	2	1	1	2	1	2	2	1	1	1
22	2	1	2	2	1	2	1	1	2	1	2	2	1	2	1	2	1	2	1	1	2	2	2
23	2	1	2	2	1	2	1	2	1	2	1	1	2	1	2	1	2	1	2	2	1	2	2
24	2	1	2	2	1	2	1	2	1	2	1	1	2	1	2	2	1	2	1	1	2	1	1
25	2	2	1	1	2	2	1	1	2	2	1	1	2	2	1	1	2	2	1	2	1	1	1
26	2	2	1	1	2	2	1	1	2	2	1	1	2	2	1	2	1	1	2	1	2	2	2
27	2	2	1	1	2	2	1	2	1	1	2	2	1	1	2	1	2	2	1	2	1	2	2
28	2	2	1	1	2	2	1	2	1	1	2	2	1	1	2	2	1	1	2	1	2	1	1
29	2	2	1	2	1	1	2	1	2	2	1	2	1	1	2	1	2	2	2	1	2	1	2
30	2	2	1	2	1	1	2	1	2	2	1	2	1	1	2	2	1	1	1	2	1	2	1
31	2	2	1	2	1	1	2	2	1	1	2	1	2	2	1	1	2	2	2	1	2	2	1
32	2	2	1	2	1	1	2	2	1	1	2	1	2	2	1	2	1	1	1	2	1	1	2

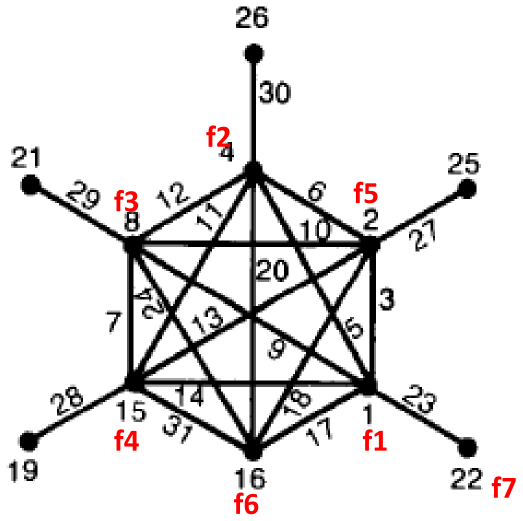


Figure 76: L32 interaction linear graph with experiment factors mapped

Table 51: L_{32} Orthogonal array with factors and utility value

L_{32} Column	**1**	**4**	**8**	**15**	**2**	**16**	**22**	
Exp. No	**F1**	**F2**	**F3**	**F4**	**F5**	**F6**	**F7**	**Utility Value**
1	1	1	1	1	1	1	1	**0.5143**
2	1	1	1	1	1	2	2	**0.2820**
3	1	1	2	2	1	1	1	**0.4661**
4	1	1	2	2	1	2	2	**0.0798**
5	1	2	1	2	1	1	2	**0.2366**
6	1	2	1	2	1	2	1	**0.3729**
7	1	2	2	1	1	1	2	**0.2881**
8	1	2	2	1	1	2	1	**0.3739**
9	1	1	1	2	2	1	2	**0.5315**
10	1	1	1	2	2	2	1	**0.6800**
11	1	1	2	1	2	1	2	**0.5492**
12	1	1	2	1	2	2	1	**0.5280**
13	1	2	1	1	2	1	1	**0.5747**
14	1	2	1	1	2	2	2	**0.4722**
15	1	2	2	2	2	1	1	**0.583**
16	1	2	2	2	2	2	2	**0.4818**
17	2	1	1	2	1	1	1	**0.3509**
18	2	1	1	2	1	2	2	**0.1684**
19	2	1	2	1	1	1	1	**0.3975**
20	2	1	2	1	1	2	2	**0.0710**
21	2	2	1	1	1	1	2	**0.1136**
22	2	2	1	1	1	2	1	**0.3371**
23	2	2	2	2	1	1	2	**0.1930**
24	2	2	2	2	1	2	1	**0.2893**
25	2	1	1	1	2	1	2	**0.3725**
26	2	1	1	1	2	2	1	**0.4418**
27	2	1	2	2	2	1	2	**0.4311**
28	2	1	2	2	2	2	1	**0.3658**
29	2	2	1	2	2	1	1	**0.4778**
30	2	2	1	2	2	2	2	**0.3360**
31	2	2	2	1	2	1	1	**0.3617**
32	2	2	2	1	2	2	2	**0.2645**

Table 52: Interactions mapped to the L_{32} columns

Factor Interactions	**L_{32} columns**
f1xf2	5
f1xf3	9
f1xf4	14
f1xf5	3
f1xf6	17
f1xf7	23
f2xf3	12
f2xf4	11
f2xf5	6
f2xf6	20
f3xf4	7
f3xf5	10
f3xf6	24
f4xf5	13
f4xf6	31
f5xf6	18

Curriculum Vitae

Foivos PSAROMMATIS – GIANNAKOPOULOS

Manousou Koundourou 89, 18533, Athens, Greece

☎ (+41) 78-666-6309 | ✉ foivospsar@gmail.com | in Foivos Psarommatis

Languages: Greek (Native) - English (full professional) - French (elementary) - German (elementary)

Responsible, ethical and enthusiastic individual, able to work under strict deadlines and cope with the daily challenges. Multitasking, with excellent organisational, interpretational and communication skills; easily embodied in any environment. Strong team player, ready to absorb and share knowledge. Seeking to incorporate my experience to a profession, in an Industrial or Academic environment. Main research interests on Zero Defect Manufacturing, design, planning, scheduling, decision-making, control of manufacturing systems and Industrial management as well as on product design and prototyping.

Professional Experience

Research Assistant - *École Polytechnique Fédérale de Lausanne (EPFL)* **Jun 2017-present**

Working on decision support and scheduling systems for manufacturing systems for achieving Zero Defect Manufacturing for the EU funded projects Z-Fact0r, Qua4lity, ZDMterm

Doctoral Assistant - *École Polytechnique Fédérale de Lausanne (EPFL)* **Jun 2017-present**

Working on Zero Defect manufacturing (ZDM) technics and strategies. Design and development of a dynamic process Scheduling tool taking into consideration Zero Defect manufacturing objectives. Methodologies for designing for ZDM

Other responsibilities: teaching CAD-CAM (CATIA), Design for Assembly and supervising students thesis (30 students up to present)

Technology Assistant - *Institut Le Rosey* **Jan 2018 – Jun 2018**

Assisting students with their projects in the fields of Robotics, 3d design and 3d printing, Programming and Mechatronics and teaching all necessary skills and knowledge in these areas, aiming to help them develop an engineering mindset. (https://www.rosey.ch/)

Teaching (voluntarily) *National Technical University of Athens (NTUA)* **Jan 2017 – Mar 2017**

Demonstration, theory and practice of an OKUMA CNC Vertical Milling Centre to undergraduate students. *Laboratory of Manufacturing Technology (supervisor Prof George Vosniakos)*

Machinist *National Technical University of Athens (NTUA)* **Nov 2016 – Dec 2016**

Manufactured a batch of 200 parts for a commercial client, copying a prototype diving equipment fitting, using CNC machinery. *Laboratory of Manufacturing Technology (supervisor Prof George Vosniakos) (http://www.mech.ntua.gr/en/sections/ttk)*

Military service (compulsory 12 months) *Hellenic Navy* **Jan 2014 – Jan 2015**

Designer *DL Automation & Control* **Dec 2013 – Mar 2017**

Electrical and Mechanical drawings for marine automation systems (*http://dlautomation.gr/*)

Research Assistant, *University of Patras* **Mar 2009 – Dec 2013**

Working on designing and testing ICT tools for Decision-making, Design and Planning manufacturing systems and product customization. EU funded projects e-CUSTOM and CAP4SMEs. *Laboratory for Manufacturing Systems and Automation (supervisor Prof. Dimitris Mourtzis)* (http://lms.mech.upatras.gr/)

Academic Education

June 2017 – Present **PhD student Advanced Manufacturing** *École polytechnique fédérale de Lausanne (EPFL)*

Thesis: A dynamic scheduling tool and a methodology for creating digital twin of manufacturing systems for achieving Zero Defect Manufacturing, Thesis Advisor: Prof. Dimitris Kiritsis

Jun 2017 – Jul 2017 **Summer school** *University of Vienna*

"Next-Generation Enterprise Modelling in the Age of Internet of Things" (http://nemo.omilab.org/nemo/)

Oct 2015 – Nov 2016 **MSc Automation Systems (***8.44/10***)** *National Technical University of Athens (NTUA)*

Inter-Departmental course (direction "Manufacturing and Production Systems")

Master Thesis: "*Design and build a powder management system for a prototype SLS machine and use robust techniques for the calibration of the machine*" (http://www.mech.ntua.gr/en/sections/ttk)

Sep 2008 – Jun 2014 **BSc and MSc Mechanical Engineering (***7.71/10***)** *University of Patras*

Department of Mechanical Engineering and Aeronautics

Master Thesis: "*A platform for supporting decentralized manufacturing networks in the era of personalization"*

Awards & Accomplishments

- **Invited** to participate to a project (ZDMterm) for the standardization of Zero Defect Manufacturing terminology lead by CEN-CENELEC and DIN standardization organizations (Kick off, of the ZDMP project October 2020).
- Clinton Global Initiative University **Fellowship 2016** (the concept of the proposed project was to improve health care services and quality of treatment (dental sector as use case), by developing a device which will allow doctors to manufacture personalized surgical implants on site in order to provide highly personalised treatment). The project funded from Angelopoulos foundation fellowship.
- **Patent** 2016, "Methodology and structure for the manufacturing of customized dental implants at the operating room", patent No. 20160100426
- Best Paper Award (***Burbidge Award***) (http://www.mckn.eu/2012/10/s-mc-s-at-the-apms-2012/) *"Simulation-based design of production networks for manufacturing of personalised products" (http://www.springer.com/computer/information+systems+and+applications/book/978-3-642-40351-4)*
- The research project e-CUSTOM (A web based collaboration system for mass customisation), in which was one of the researchers, was evaluated together with 98 other FP7 FoF research projects and was selected as a ***Success Story.*** It was presented at the FoF Impact Workshop (11 and 12-03-2013, Brussels). (http://ec.europa.eu/research/industrial_technologies/events-fp7-draft-programme-2010_en.html & http://ec.europa.eu/research/industrial_technologies/pdf/fof-impact-workshop-11-12032013- agenda_en.pdf)

Conference Attendance (With Paper Presentation)

- 53rd CIRP Conference on Manufacturing Systems (CIRP CMS, 2020).
- 8th CIRP Conference on Assembly Technologies and Systems (CIRP CATS, 2020)
- AMPS 2020 Conference in Advances in Production and Management systems
- AMPS 2019 Conference in Advances in Production and Management systems
- 52nd CIRP Conference on Manufacturing Systems (CIRP CMS, 2019).
- APMS 2018 Conference in Advances in Production and Management systems.
- PALM 2018 Conference on Product and Asset Lifecycle Management
- 9th annual Clinton Global Initiative University meeting from April 1-3, 2016 at the University of California, Berkeley, US
- 45th CIRP Conference on Manufacturing Systems (CIRP CMS, 2012).

Computer Knowledge

Microsoft Office	● ● ● ● ● ●	Lanner *WITNESS*	● ● ● ● ○ ○
INTERNET	● ● ● ● ● ●	KiCAD	● ● ● ● ○ ○
MATLAB	● ● ● ● ● ●	Programming (VB, C, C++, Arduino, Python)	● ● ● ● ○ ○
Dassault Systems *CATIA V5*	● ● ● ● ● ○	Microsoft *VISUAL STUDIO*	● ● ○ ○ ○ ○
Autodesk *AUTOCAD & 360*	● ● ● ● ● ○	Atmel *STUDIO*	● ● ○ ○ ○ ○

Other Interests & Skills

- Technology, Computers, Engineering projects (designed and fabrication of engineering devices).Design and development of Smart home automations. Repair and maintain my family cars.
- Sports (Mountain bike, Sailing, Ski, Table tennis, hiking, Scuba Diving, Model Race cars)
- Traveling, music, table games, cinema, theatre and cultural events
- First aid certified