FEDERAL UNIVERSITY OF MINAS GERAIS
DEPARTMENT OF PRODUCTION ENGINEERING
SCHOOL OF ENGINEERING

Fernanda de Freitas Alves

PRODUCTION PLANNING PROBLEMS SUBJECT TO MACHINE FAILURES CONSIDERING CHARACTERISTICS OF AN INDUSTRY 4.0 ENVIRONMENT

Belo Horizonte
2022

Fernanda de Freitas Alves

PRODUCTION PLANNING PROBLEMS SUBJECT TO MACHINE FAILURES CONSIDERING CHARACTERISTICS OF AN INDUSTRY 4.0 ENVIRONMENT

Thesis presented to the Graduate Program in Production Engineering of the Federal University of Minas Gerais as a requirement to obtain a Doctoral Degree in Production Engineering.

Supervisor: Prof. Martín Gómez Ravetti

Belo Horizonte
2022

A474p

Alves, Fernanda de Freitas.
Production planning problems subject to machine failures considering characteristics of an Industry 4.0 environment [recurso eletrônico] / Fernanda de Freitas Alves. - 2022.
1 recurso online (81 f. : il., color.) : pdf.

Orientador: Martín Gómez Ravetti.

Tese (doutorado) - Universidade Federal de Minas Gerais, Escola de Engenharia.

Apêndices: f. 73-81.

Bibliografia: f. 68-72.
Exigências do sistema: Adobe Acrobat Reader.

1. Engenharia de produção - Teses. 2. Planejamento da produção - Teses. 3. Processo decisório - Teses. I. Gómez Ravetti, Martín. II. Universidade Federal de Minas Gerais. Escola de Engenharia. III. Título.

CDU: 658.5(043)

Ficha catalográfica elaborada pelo bibliotecário Reginaldo Cesar Vital dos Santos - CRB-6/2165
Biblioteca Prof. Mário Werneck - Escola de Engenharia da UFMG

UNIVERSIDADE FEDERAL DE MINAS GERAIS
Escola de Engenharia
Programa de Pós-Graduação em Engenharia de Produção

FOLHA DE APROVAÇÃO

Problemas de planejamento da produção sujeitos à falhas de máquinas considerando características de um ambiente de Indústria 4.0

FERNANDA DE FREITAS ALVES

Tese submetida à Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em ENGENHARIA DE PRODUÇÃO, como requisito para obtenção do grau de Doutor em ENGENHARIA DE PRODUÇÃO, área de concentração PESQUISA OPERACIONAL E INTERVENÇÃO EM SISTEMAS SOCIOTÉCNICOS, linha de pesquisa Mod. e Algorit. de Otimiz. para Sistemas em Redes e de Prod..

Aprovada em 07 de julho de 2022, pela banca constituída pelos membros:

Prof(a). Martin Gomez Ravetti - Orientador

UFMG

Prof(a). Mauricio Cardoso de Souza

UFMG

Prof(a). Thiago Henrique Nogueira

UFV

Prof(a). Frederico Gadelha Guimaraes

UFMG

Prof(a). Cleber Damião Rocco

Unicamp

Belo Horizonte, 07 de julho de 2022.

Documento assinado eletronicamente por **Thiago Henrique Nogueira**, **Usuário Externo**, em 07/07/2022, às 17:08, conforme horário oficial de Brasília, com fundamento no art. 5º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Martin Gomez Ravetti**, **Professor do Magistério Superior**, em 07/07/2022, às 17:10, conforme horário oficial de Brasília, com fundamento no art. 5º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Mauricio Cardoso de Souza**, **Professor do Magistério Superior**, em 13/07/2022, às 17:53, conforme horário oficial de Brasília, com fundamento no art. 5º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Frederico Gadelha Guimaraes**, **Professor do Magistério Superior**, em 14/07/2022, às 15:40, conforme horário oficial de Brasília, com fundamento no art. 5º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Cleber Damião Rocco**, **Usuário Externo**, em 15/07/2022, às 09:08, conforme horário oficial de Brasília, com fundamento no art. 5º do Decreto nº 10.543, de 13 de novembro de 2020.

A autenticidade deste documento pode ser conferida no site https://sei.ufmg.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **1573931** e o código CRC **2160CFC0**.

Referência: Processo nº 23072.238774/2022-87 SEI nº 1573931

Acknowledgments

First, I would like to thank my supervisor, Prof. Martín Gómez Ravetti, for his guidance, encouragement, and trust during my studies. Our discussions on production planning problems and other topics made me grow personally and professionally.

I would also like to especially thank Prof. Thiago Henrique Nogueira for all the support and availability since graduation.

I thank professors Maurício Cardoso de Souza, Frederico Gadelha Guimarães, Thiago Henrique Nogueira, and Cleber Damião Rocco for accepting to participate in my doctoral defense committee, especially professors Maurício, Frederico, and Thiago, who contributed with their suggestions in the qualification committee. In addition, I would like to thank the remaining professors of PPGEP/UFMG for their knowledge passed during my master's and doctoral studies.

I thank all my family for their affection and understanding during these years. Further, I also thank the friends who accompanied me during this period, especially the friends of Lamos and PPGEP.

Especially, I thank Armando Honório for being an example of a researcher for me and thank for all the support throughout my doctorate.

Finally, I would like to thank CAPES for its financial support during my research.

Resumo

Novas tecnologias e o avanço da Indústria 4.0 tem gerado mudanças significativas nos processos de produção, como maior uso de dados para tomada de decisão, maior velocidade de produção devido à automação e horizontes de planejamento mais curtos. Em breve, os métodos de resolução e cenários consolidados na literatura podem não se adequar a esse novo panorama. Portanto, os tomadores de decisão devem considerar esse novo contexto na fase de planejamento da produção. Neste trabalho, incorporamos características da Indústria 4.0 em nosso estudo, considerando que a entrega dos produtos aos clientes nas datas combinadas será mais relevante do que a minimização de custos. Nós assumimos que, neste contexto, ordens de produção chegarão e serão rapidamente produzidas. Longos horizontes de planejamento e grandes quantidades de estoque não existirão mais. Assim, decisões em tempo real e horizontes curtos devem ser considerados no planejamento da produção. Estudamos as adaptações a serem feitas nos algoritmos e no processo de tomada de decisão do planejamento da produção baseadas nas ferramentas e características de um ambiente de Indústria 4.0. Primeiramente, propomos e comparamos duas abordagens proativas para lidar com falhas de máquinas no chão de fábrica. Os métodos visam reduzir a probabilidade de inviabilidade no nível de sequenciamento após a ocorrência de falhas. Nós verificamos se existe diferença na utilização de ambas as estratégias e, além disso, testamos diversos parâmetros de robustez utilizados para lidar com rupturas. Com base nos experimentos, concluímos que o tomador de decisão decidirá qual parâmetro de robustez utilizar com base nas quantidades de produtos consideradas, visto que a maioria das métricas testadas não apresenta um padrão definido ao modificar este parâmetro. Em relação ao teste de similaridade das abordagens proativas, concluímos que elas exibiram resultados semelhantes para a maioria dos cenários e variações testadas. Este trabalho também propõe abordagens usando algoritmos de aprendizado de máquina para prever rupturas no chão de fábrica, além de apresentar um *framework* que prevê a melhor estratégia para implementação de acordo com a instância específica do problema. Também propomos uma abordagem proativa-online integrando decisões proativas e em tempo real, comparando os resultados com uma estratégia corretiva. Com base em testes computacionais realizados com um *benchmark* proposto, as abordagens proativa e proativa-online resultaram em menor atraso total ponderado em comparação ao método corretivo. Em relação às abordagens proativa e proativa-online, observamos que seus resultados dependem do conjunto de instâncias analisadas, justificando a proposição do *framework*. Por fim, para a maioria dos casos, as estratégias previstas pelo *framework* obtiveram menor atraso total ponderado quando comparadas com os resultados médios obtidos por todas as estratégias estudadas neste trabalho. Nós concluímos que as estratégias e *framework* propostos poderiam melhorar o processo de tomada de decisão, resultando em soluções rápidas e robustas em relação à falhas de máquinas em um ambiente com características da Indústria 4.0.

Palavras-chave: Planejamento da produção. Falhas de máquinas. Indústria 4.0. Abordagens proativas.

Abstract

New technologies and the advancement of Industry 4.0 have led to significant changes in production processes, e.g., increased use of data for decision-making, faster production speed due to automation, and shorter planning horizons. Soon, the resolution methods and scenarios consolidated in the literature may not adapt to this situation. Therefore, decision-makers must consider this new context in the production planning phase. We incorporate characteristics of Industry 4.0 into our study, considering that the delivery of products at their due dates to customers will be more relevant than minimizing costs. We assume that, in this new context, orders arrive and must be quickly produced. Long planning horizons and large inventory quantities will no longer exist. Then, online decisions and short horizons should be considered in production planning. We study the adaptations to be made in the algorithms and decision-making process of production planning based on the tools and characteristics of an Industry 4.0 environment. First, we propose and compare two proactive approaches for dealing with machine failures on the shop floor. The methods aim to reduce the probability of infeasibility at the scheduling level after the occurrence of failures. We verify if exist differences when using both strategies and test several robustness parameters used to deal with the uncertainties. Based on the experiments, we conclude that the decision-maker should decide which robustness parameter to use based on the product quantities, given that most metrics tested do not present clear patterns when modifying this parameter. Regarding the test of similarity of the proactive approaches, we conclude that they showed similar results for most scenarios and variations tested. This work also proposes approaches using machine learning algorithms to predict disruptions on the shop floor, besides introducing a framework that predicts the best strategy for implementation according to the specific problem instance. Further, we propose a proactive-online approach integrating proactive and real-time decisions, comparing the results with a corrective strategy. Based on computational tests performed with a proposed benchmark, the proactive and proactive-online approaches resulted in lower total weighted tardiness in comparison to the corrective method. Regarding the proactive and proactive-online approaches, we observe that their results depend on the set of analyzed instances, justifying the proposition of the framework. Lastly, for most cases, the strategies predicted by the framework achieved lower total weighted tardiness when compared with the average results obtained by all the strategies studied in this work. We conclude that the proposed strategies and framework would improve the decision-making process, resulting in fast and robust solutions regarding machine failures in an environment with Industry 4.0 characteristics.

Keywords: Production planning. Machine failures. Industry 4.0. Proactive approaches.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

In manufacturing, production planning problems are periodically solved to define which products to produce, their quantities, and the production sequences (Pochet, 2001). However, several uncertainties may occur on the shop floor, affecting the initial plan, such as stochastic processing times and machine breakdowns (Aytug et al., 2005; Azizoglu and Alagöz, 2005; Goren and Sabuncuoglu, 2009). One alternative to reduce the impact of unexpected events when performing the plan is to use proactive approaches. In proactive approaches, uncertainties in the production process are considered *a priori* (Goren and Sabuncuoglu, 2009; Wang et al., 2015). Thus, the decision-maker anticipates possible disruptions, planning the use of resources based on the stochastic characteristics of the shop floor. In this thesis, we study more realistic scenarios when compared to works dealing with deterministic planning problems, proposing proactive and proactive-online approaches to handle machine failures in a parallel machine environment. Furthermore, we introduce a framework that predicts the best strategy for implementation based on the instance set.

Our studies and experiments aim to reduce the deviations from the initial production plan after the occurrence of failures. The procedures consider disruptions when solving the short-term production planning problem, allowing the scheduling, solved after the definition of the production quantities, to define sequences with high probabilities of being feasible regarding the capacity even if failures occur. We indicate the work of Aytug et al. (2005) for more information on approaches considering uncertainties. In our study, we assume an environment that may have several failures in each period. A real example of this scenario occurs in manufacturing involving heavy metals. In this setting, the products are subject to specific stresses, increasing the deterioration rate and possibly causing production failures. Our goal is to develop methods that can be adapted to other types of uncertainties and real-world instances. We consider an environment of parallel machines. Nonetheless, the decision-maker can use the algorithms for other types of production, as long as changes are made in the short-term production planning and scheduling implementations. Failure prediction models, the rationale of proactive and proactive-online approaches, and the proposed framework are general and can be used in any production type.

Nowadays, most planning decisions usually consider financial criteria (see, for instance, the models presented in the work of Pochet and Wolsey (2006)). With the advancement of Industry 4.0, which increases efficiency, value creation, and real-time optimization through digital transformations, companies should also take into account other decision-making measures focusing on customer experience. For more information on Industry 4.0, see the works of Hofmann and Rüsch (2017) and Rossit et al. (2019). Therefore, in this thesis, our aim is to improve customer service levels instead of only minimizing costs. When we anticipate possible failures, we provide the company the ability to estimate accurate due dates to customers considering that unforeseen events may occur. The methodology we present try to meet the due dates with minimum delay, besides indirectly reducing costs.

We hypothesize that, in an Industry 4.0 environment, customized product orders arrive and must be quickly produced to enable fast customer service. Long planning horizons will no longer exist since ordering information will be available in online systems that sequence incoming demands. Consequently, the incorporation of short horizons and online decisions to traditional planning should be considered. In this work, we address one week and two weeks of planning in Chapters 4 and 5, respectively. Therefore, we study an intermediate situation between traditional production planning and the Industry 4.0 environment regarding the planning horizon.

1.1 Contributions

This work presents approaches to deal with uncertainties in production planning problems considering the principles of Industry 4.0. In Chapter 4, we propose two proactive methods aiming to minimize the probability of infeasibility after machine failures. We consider that a sequence is infeasible if its respective makespan is greater than the production capacity. One approach performs simulations before the resolution of the planning problems to estimate information on the disruptions. This information is used to find robust solutions when solving the short-term production planning problem. The second method tests if using only the probability distributions of the failures to modify the planning problem is sufficient to guarantee solutions with low probabilities of infeasibility. The procedures iteratively solve the short-term production problem and scheduling until finding solutions less impacted by disruptions. We compare both methods, verifying if they generate similar results.

In Chapter 5, we also propose two proactive approaches. In this case, we integrate failures prediction using machine learning algorithms with planning problems. The motivation for using machine learning algorithms to predict failures is to find tight upper bounds for the production capacity at the short-term production planning problem. Our goal is to dynamically solve production planning problems integrated with maintenance, considering historical data of failures to predict future disruptions. In addition, we aim to consider the dynamics of the instances and the need to adapt forecasts throughout the planning process. The proactive approaches analyze feedback information about the failures to decide the planning of the next period. We also solve the studied problem considering a proactive-online approach and implement a corrective method for comparison purposes. Another contribution of this work consists of a framework based on machine learning algorithms to predict which proposed strategy will result in the lowest total weighted tardiness for a specific instance set. Since industrial data considering machine failures are usually confidential (Kang et al., 2021), we introduce a benchmark that can be used in future works by other researchers, as it will be available to readers.

To the best of our knowledge, there is no work in the literature comparing proactive and proactive-online methods with the same objective and analyzed scenarios as presented in this work. Also, we did not find studies that dealt with all the problems considered in this thesis simultaneously, predicting the failures and using the resulting information to modify the planning problems, as presented in Chapter 5. Usually, machine failures are treated reactively in the literature. In our work, we proactively deal with them at the short-term production planning level. Besides, the resolution methods for each specific problem have particularities that are contributions of this study, as well as the consideration of how characteristics of Industry 4.0 are changing the essence of the planning problems. Listing the main contributions of the work, we have:

- Comparison of different types of information added to production planning problems to find robust solutions in uncertain environments;

- Integration of machine learning with production planning algorithms;
- Proactive and proactive-online approaches to deal with failures;
- A framework which predicts the best approach for implementation;
- Consideration of Industry 4.0 characteristics in production planning.

1.2 Text organization

The remaining of this work is organized as follows: In Chapter 2, we review production planning problems, focusing on approaches and resolution methods that consider uncertainties. Chapter 3 shows the preliminary definitions, describing the studied problem and the methods used in this work for its resolution. In Chapter 4, we propose and compare two iterative proactive approaches for dealing with uncertainties, while in Chapter 5, we integrate failure prediction methods with production planning problems, proposing proactive and proactive-online procedures. Still, in Chapter 5, we also present a proposed framework that predicts the best approach and strategy to be used based on the instance of the problem. Finally, Chapter 6 concludes the thesis, also presenting proposals for future works. Appendix A shows the results obtained with the scheduling heuristic presented in Chapter 3, whereas Appendix B presents the results of the hierarchical approach used for comparison purposes in Chapter 4.

Chapter 2

Literature review

The lot-sizing problem defines which products will be produced in each period of the planning horizon and their respective quantities. It is solved for the medium-term planning, usually minimizing costs, e.g., production, inventory, and backorder costs. Jans and Degraeve (2008) review deterministic single-level dynamic lot-sizing problems. The authors emphasize that one of the drawbacks of the papers discussed in the review is the consideration of deterministic parameters. Unlike the present work, their study presents models minimizing costs. Gruson et al. (2018) investigate the impact of service level constraints in deterministic lot-sizing with backlogging. They report that it is usually hard to determine backorder and backlogging costs. Therefore, in these cases, they conclude that it might make more sense to consider service levels. We refer the interested reader to the works of Aloulou et al. (2014) and Brahimi et al. (2017) for discussions on the problem. In our work, we define production quantities in the short-term, solving what we name short-term production planning problem. In this case, we consider a short-term horizon since we assume that production orders will arrive and will be quickly produced in the Industry 4.0 scenario. To the best of our knowledge, there are no references in the literature considering the problem with the same characteristics in the short-term as we do in this thesis.

Scheduling problems can be defined as the allocation of resources over time to perform a set of tasks (Blazewicz et al., 2019). Numerous works consider parallel machine scheduling; for instance, see the papers of Mokotoff (2004), Vallada and Ruiz (2011), and Kim et al. (2020). In the work of Mokotoff (2004), the authors propose an exact cutting plane algorithm to solve an identical parallel machine scheduling problem. The method resulted in optimal solutions for almost all instances tested when minimizing the makespan. Vallada and Ruiz (2011) introduce a genetic algorithm for solving the unrelated parallel machine scheduling problem. The authors minimize the makespan in a scenario with sequence dependent setup times. The proposed method outperforms the best algorithms known from the literature for the studied problem. Lastly, Kim et al. (2020) present a novel mathematical model and metaheuristics for solving an identical parallel machine scheduling problem aiming to minimize the total tardiness. Unlike this work, the studies of Mokotoff (2004), Vallada and Ruiz (2011), and Kim et al. (2020) deal with deterministic environments, which may lead to infeasible plans due to several types of disruptions occurring on the shop floor. Regarding the objectives and resolution method, this study differs from the works cited since it minimizes the total weighted tardiness as the primary objective, solving the identical parallel machine scheduling problem with a proposed heuristic based on the paper of Park et al. (2012). For recent studies dealing with parallel machine scheduling problems with uncertainties, we refer the reader to the papers of Kim and Kim (2020) and Zhang et al. (2020). For literature reviews on scheduling, see the works of Allahverdi (2015) and Allahverdi (2016).

In the remainder of this chapter, we focus on methods to solve planning problems with disruptions. In Section 2.1, we introduce works considering uncertainties, specifically when solving planning problems. We present proactive approaches in Section 2.2, reviewing works dealing with robustness and simheuristics. On the other hand, Section 2.3 exhibits works using reactive approaches to solve planning problems subject to uncertainties. In Section 2.4, we discuss papers analyzing the Industry 4.0 scenario, considering especially machine learning algorithms to solve planning problems. Lastly, Section 2.5 exhibits similar works regarding the studied problem and resolution methods considered in this thesis. We emphasize that we do not focus on the definitions and applications of production planning problems; for this, we guide the interested reader to the paper of Alves et al. (2021).

2.1 Uncertainties

Scheduling problems under uncertainties are considered by Aytug et al. (2005), which review predictive-reactive scheduling, robust approaches, and completely reactive approaches. According to the authors, in predictive-reactive scheduling, an initial sequence is determined considering certain assumptions and is released to the shop floor. However, several uncertainties may occur when executing the plan. Thus, reactive scheduling, also called rescheduling, has to be considered. In this case, the initial schedule is modified to deal with the disruption. This modification may consist of changes in the original sequence or the generation of a new schedule. Aytug et al. (2005) also state that, on the other hand, in robust approaches, the aim is to create a sequence that minimizes the effects of disruptions on the primary performance measure when executed on the shop floor. In completely reactive approaches, the production sequence is defined for the immediate future, usually using dispatching rules. The authors emphasize that disruptions may lead to opportunities to improve the initial solution based on the scenario after the occurrence of the uncertainty. However, a significant amount of reorganization may be required if the original schedule suffers substantial changes.

Vieira et al. (2003) present a framework of strategies, policies, and methods considering rescheduling. According to the authors, rescheduling can be performed periodically to update the original sequence, planning the next periods based on the current situation. It can also be performed when disruptions affect the production state, e.g., when machine failures and new jobs arrive at the system (Vieira et al., 2003). Reactive methods have the disadvantage of having expensive computation since the problem may need to be solved several times to mitigate possible infeasibilities caused by uncertainties. For examples of rescheduling methods, see Section 2.3. On the other hand, preventive approaches like robust optimization consider disruptions *a priori*. In this case, information about the uncertainty is added to the mathematical model of the studied problem, aiming to find a solution possibly close to the optimal solution that remains feasible even with unexpected events (Dias and Ierapetritou, 2016).

In the work of Figueira and Almada-Lobo (2014), the authors present several methods integrating simulation and optimization that could be used to solve uncertainty problems, also providing a taxonomy for the methods proposed in the literature. Their paper aims to guide researchers when applying simulation-optimization approaches and provide ideas on cross-fertilization between them. According to the authors, the integration of simulation and optimization takes advantage of a large amount of detail provided by the simulation methods and their capacity to deal with complex systems, while optimization techniques can find good or optimal solutions. Usually, the difficulty in solving exact optimization methods leads to the use of simulation. The authors aim to create better communication among the academic community by creating a standard classification for such approaches.

Lee and Yu (2008) study a parallel machine scenario subject to disruptions. The authors consider that when suffering rupture, all the machines are unavailable for a certain period. To solve the problem, they propose pseudo-polynomial time algorithms considering resumable and non-resumable cases. Talay and Özdemir-Akyıldırım (2019) investigate yield uncertainties in a multi-product multi-stage make-to-order production system. They present a stochastic optimization model, which is solved with a novel solution algorithm. In the first stage, the model aims to define the optimal amounts of input materials, while in the second stage, the output consists of production quantities for each product. Sensitivity analysis is performed, aiming to evaluate the influence of the parameters on the results.

In the paper of Verderame et al. (2010), the authors review planning and scheduling problems under uncertainties in chemical, petrochemical, and pharmaceutical industries, energy planning, power generation system planning, and other applications. They emphasize that dealing with the integration of planning and scheduling is challenging and even more complicated with uncertainties. According to the authors, the main difficulty is to model it realistically and at the same time solve it in acceptable computational times. They also state that the robustness of a solution can be significantly improved if uncertainties features are considered in problem modeling.

Kim and Kim (2021) consider forecasted machine breakdowns in an unrelated parallel machines environment. The authors propose a rescheduling algorithm in which jobs are re-assigned before the occurrence of machine breakdowns based on the forecasting. They claim that due to sensors and artificial intelligence methods, machine disruptions can be predicted with high accuracy. In this context, preventive maintenance can be performed before the failures, allowing time and cost savings. In the work of Chrétienne (2020), a reactive schedule is also determined to deal with machine failures. In this case, the authors minimize the number of non-rescheduled jobs, i.e., maximize the number of jobs respecting their original starting times. Schmidt (2000) review scheduling problems with limited machine availability, including papers using online algorithms to deal with unexpected machine breakdowns.

In the paper of Peng and van Houtum (2016), the authors integrate condition-based maintenance with production lot-sizing. They emphasize that maintenances should be coordinated with production to reduce the number of interruptions. At the same time, the lot sizes should be optimized considering the maintenances periods. The authors propose a new model aiming to minimize the total cost of lot-sizing and condition-based maintenances. de Jonge and Scarf (2020) present a review on maintenance optimization. The authors state that the organizations realize that planning maintenances effectively can improve efficiency and reliability, allowing more maintenances and alignment with other business functions like scheduling. See the works of Crowder and Lawless (2007) and Leukel et al. (2021) for examples of papers considering maintenances.

Dolgui et al. (2005) propose a decomposition approach to solve a lot-sizing and sequencing problem subject to uncertainties, consisting of rejected items and machine breakdowns. Planned safety time is used when solving the problem to guarantee a period for breakdown repairs. The resolution method consists of an iterative optimization approach, which decomposes the studied problem into three subproblems: an enumeration, a Travelling Salesman Problem (TSP), and a Knapsack problem. Other studies considering uncertainties in lot-sizing and scheduling problems can be seen in the works of Dolgui et al. (2011) and Hu and Hu (2016).

2.2 Proactive Approaches

In a proactive approach, the management acquires capabilities before their needs (Lindberg, 1990). Goren and Sabuncuoglu (2009) propose a proactive method specifically for scheduling. The authors state that such approaches consider future disruptions when generating the sequences, whereas reactive methods revise schedules when uncertainties occur. Wang et al. (2015) also present a proactive scheduling approach considering machine breakdowns in a deteriorating scenario. After the failure, the processing times are compressed, aiming to match the baseline schedule as early as possible. The authors propose a multi-objective evolutionary algorithm, presenting the set of Pareto optimal solutions considering the robustness related to machine breakdowns versus the initial operational costs of scheduling.

A non-identical parallel machine scenario with controllable processing times is studied by Gürel et al. (2010). The authors show that considering failures and repair distributions as well as the flexibility of jobs in anticipated scheduling can reduce manufacturing costs. The proposed anticipative approach uses flexibility measures that estimate the jobs that absorb the effects of disruptions at the lowest costs. In their study, the breakdown occurs in one machine in an uncertain moment, becoming unavailable during repair time. A preempt-repeat case is considered. The proposed method was efficient for the instances tested, given that lower rescheduling costs were obtained.

Fazayeli et al. (2016) propose a proactive strategy to deal with uncertainties consisting of machine breakdowns. The procedure aims to maximize the probability that the makespan will not exceed a specific value. The authors state that a proactive approach is necessary since when a disruption occurs in practice, it is crucial to quickly deal with it. In certain situations, rescheduling methods may require more time to solve complex problems. They propose a hybrid metaheuristic based on a Genetic Algorithm and Simulated Annealing (GA-SA algorithm) to solve a flowshop scheduling problem. Six heuristics and metaheuristics are implemented for comparison. Then, based on the best sequence, a simulation of failures is performed. The schedule with the maximum probability that the makespan will not exceed a specific value is chosen as the robust sequence. Results showed a significant difference between the robustness of the algorithms tested, with the GA-SA algorithm presenting the best performance.

In the paper of Cui et al. (2018), the authors propose a proactive approach to solve an integrated production scheduling and maintenance planning problem. They aim to optimize quality robustness and solution robustness considering failure uncertainty in a flowshop scenario. A model is introduced for solving the integrated problem, which considers preventive and corrective maintenances. The authors use a Monte Carlo sampling method to find the objective function for feasible solutions. An efficient surrogate measure is also devised to approximate the objective function in a lower computational time. Furthermore, an algorithm is proposed to solve the integrated model, finding better results than a traditional baseline algorithm.

Brčić et al. (2019) study a stochastic resource-constrained project scheduling problem focusing on uncertain activity durations. The authors propose a proactive rescheduling approach and introduce a threshold cost-based flexibility measure, which allows proactive-reactive methods to function as hybrids between proactive-reactive and pure reactive approaches according to the planning horizon. Proactive approaches are also studied by King and Teo (2000) and Topan and van der Heijden (2020). King and Teo (2000) compare proactive and reactive methods when integrating business planning with information systems planning, whereas Topan and van der Heijden (2020) consider reactive and proactive interventions in the operational level planning of a multi-item two-echelon spare parts inventory system.

In the next sections, we focus on the methods used to deal with uncertainties in this work. In Section 2.2.1, we review robust approaches, while Section 2.2.2 presents simheuristics methods. We focus on classical papers of each method and applications in production planning problems.

2.2.1 Robustness

Solutions of the original model could no longer be optimal or even feasible if the initial parameters change, which justifies the creation of robust models immune to these modifications (Bertsimas and Sim, 2004). A robust approach is proposed by Bertsimas and Sim (2004) to ensure that the solution remains feasible when uncertainties occur. The proposed method aims to decrease what the authors call the price of robustness, i.e., the trade-off between the probability of violation of the constraints and the effects on the objective function of the original model. This way, we can control the conservatism level of the solution. As opposed to Soyster (1973), which guarantees protection considering that all the uncertain parameters can change (ultraconservative strategy), Bertsimas and Sim (2004) allow that a maximum number of coefficients could be modified. If more than the prespecified number of parameters change, the authors ensure that the solution will still have a high probability of being feasible. Bertsimas et al. (2011) discuss the tractability and conservativeness of robust formulations, also considering the flexibility to apply them in different scenarios, presenting applications in statistics, supply chain management, engineering, and portfolio optimization. An application of a robust approach for dealing with data uncertainties is studied by Rocco and Morabito (2016). The authors consider a problem involving production and logistics decisions in the planning of a tomato processing industry, presenting a deterministic model and three robust formulations to solve the problem.

According to Ghezail et al. (2010), several measures are usually used to consider robustness in quantitative methods: Some approaches aim to minimize the difference between the expected and the real performance with the uncertainty, which is denominated *regret*. Some approaches consider the *worst case measure*. In this case, we want to minimize the worst performance, taking into account the uncertainty under different scenarios. Some approaches consider *slacks* in the deterministic model to guarantee the robustness of the solution when uncertainties occur. For instance, if we add slacks into the initial sequence, we expand the processing times, possibly reducing the effects of disruptions. Lastly, some approaches assume as robustness measure the capacity to determine alternative solutions based on the initial solution, e.g., considering neighbor solutions when uncertainty occurs.

Feng et al. (2012) state that robust models are usually studied based on the statistical point of view rather than analyzing the production system. Seven scheduling policies are examined by the authors, aiming to determine which one is more robust under different scenarios. According to Ghezail et al. (2010), scheduling robustness is usually considered quantitatively, using measures that indicate the capacity of the schedule to remain feasible in uncertain environments. The authors emphasize that quantitative methods may be convenient for optimization objectives. However, when comparing several sequences, using a single measure to evaluate their performance may be too restrictive. In their work, they propose a qualitative approach for solving uncertain problems. In this case, graphical representations of the solutions are used to guide the decision-maker.

Li and Floudas (2014) propose an iterative framework for solving problems with uncertain parameters. The proposed method aims to improve robust solutions, combining the traditional robust optimization with *a posteriori* probability bounds. The authors state that traditional robust optimization uses *a priori* probabilistic bounds, generating a convex problem, which makes the method computationally efficient. However, in most cases, it is too conservative. On the other hand, the *a posteriori* probability bounds approach is less conservative, but a nonconvex problem has to be solved. The proposed iterative framework improved the solutions of traditional robust methods when solving applications in production planning and scheduling.

A resource-constrained project scheduling problem considering breakdowns is studied by Lambrechts et al. (2011) and solved with a proactive robust strategy. According to the authors, robustness can be achieved by adding extra idle time to each activity of the project schedule. In this case, the breakdowns are compensated by not using the total available capacity. However, they emphasize that disruptions predictions will not always be correct. Therefore, it would be necessary to use rescheduling to repair the initial schedule. Nourelfath (2011) studies a multi-period multi-product problem considering machine breakdowns. Constraints are added to the production plan model to guarantee a high probability of obtaining a pre-specified service level, aiming to find robust production plans. The model is solved using a two-step optimization approach.

Aghezzaf et al. (2011) study a two-stage production system proposing a robust hierarchical planning approach for resolution. In the first stage, the authors deal with semi-finished products with stable annual demands. In the second stage, these semi-finished products are disaggregated into finished products subject to unstable weekly demands. The approach aims to find robust plans taking into account demand variability. According to the authors, a robust aggregate plan for their studied problem is the one that leads to at least one feasible disaggregated solution for any demand uncertainty. In the first stage, they solve the problem with a cyclic planning model, while at the second stage, they implement a periodic review policy. A coupling plan is proposed for integrating both stages, using for this the ABC classification. In this case, each semi-finished product produces three types of finished products, A, B, and C. The coupling plan is proposed to guarantee the feasibility of the disaggregation. Another paper considering robustness in production planning problems can be seen in the work of Kleijnen and Gaury (2003).

2.2.2 Simheuristics

The simheuristic method consists of the combination of metaheuristics with simulations designed to solve large-scale stochastic optimization problems. The simulation deals with the model uncertainties, while the metaheuristic searches for a near-optimal solution in the solution space. Feedback information may be transferred between the simulation and the metaheuristic levels (Chica et al., 2017). Juan et al. (2015) state that metaheuristics are usually used to solve combinatorial optimization problems given their large-scale sizes and the need for high-quality solutions in short computational times. However, metaheuristics generally consider deterministic input parameters, objective function, and constraints, which may generate oversimplified models. Simheuristics can be seen as an extension of the metaheuristics methods through simulations, which can then be used to solve more realistic stochastic problems. They can also be applied to complex deterministic problems requiring the use of simulation. The authors present a review of simheuristics, showing some applications of hybrid simulation-optimization (Sim-Opt) methods in manufacturing and production, logistics and supply chain management, and healthcare.

According to Chica et al. (2017), the benefits of using simheuristics include the possibility to study valid models of complex systems and the possibility of using the simheuristic output in risk/reliability and sensitivity analysis. Unlike exact methods, simheuristics can find good results in low computational times since the simulation is used only for promising solutions. Furthermore, any probability distribution can be used to model random variables, which allows the simheuristics to be applied to different types of problems and scenarios (Hatami et al., 2018). Some limitations are that simheuristics do not ensure the optimality of the solution, require additional effort to define the simulation system and analyze the results of the simheuristic, besides requiring more computational effort compared to traditional techniques (Chica et al., 2017). A parallel flowshop scheduling problem with stochastic processing times is solved with a simheuristic algorithm by Hatami et al. (2018). The authors consider three objective functions: the minimization of the makespan, the minimization of the expected makespan, and the minimization of the makespan percentile. The proposed algorithm integrates the Iterated Local Search (ILS) metaheuristic with the Monte Carlo Simulation (MCS).

A simheuristic algorithm is proposed by Juan et al. (2014a) considering a permutation flowshop scenario with stochastic processing times. The proposed methodology assumes that in cases with moderate uncertainty, high-quality solutions for the stochastic problem can be obtained based on high-quality solutions for the deterministic problem. First, the method transforms the stochastic problem into a deterministic one by considering that the processing times are equal to the expected stochastic processing times. The Nawaz-Enscore-Ham (NEH) heuristic is then used to solve the deterministic problem. A local search procedure is applied to the solution, generating the best deterministic solution found. A short simulation estimates the expected makespan. The next step consists of applying the ILS heuristic to the stochastic solution, initialized as the best deterministic solution found. The ILS is iteratively solved, improving the deterministic and stochastic solutions until reaching a stopping condition. Long simulations are then used to estimate more accurately the expected makespan. Different scenarios are tested, considering variations in the stochastic processing times. The proposed algorithm was able to find solutions very close to the optimal solution of the deterministic problem.

Juan et al. (2014b) propose a simheuristic for solving the Single-Period Stochastic Inventory-Routing problem for a scenario with stochastic demands, minimizing the expected routing and inventory costs. The proposed method aims to find refill policies and a routing plan. For this, the authors use MCS to estimate the expected inventory costs and apply a heuristic to determine the routing costs related to each policy. The solution with the lowest cost is used as an initial solution for a multi-start randomized heuristic, which iteratively improves the result. Furthermore, the authors propose a benchmark to enable the reproduction of the experiments. A simheuristic approach is also presented by Gonzalez-Neira et al. (2017), considering stochastic processing and assembly times in a distributed assembly permutation flowshop scenario. The expected makespan is minimized using a simulation procedure integrated with the Greedy Randomized Adaptive Search Procedure (GRASP).

2.3 Reactive scheduling

Reactive scheduling, or rescheduling, consists of a modification in the original sequence that may be used to deal with disruptions. When performing a rescheduling method, we must consider the trade-off between the costs of the new solution and the changes concerning the original scheduling (Yin et al., 2016). Yin et al. (2016) study machine disruption on identical parallel machines, proposing a rescheduling method to solve the problem. Initially, the original schedule is optimally determined using the Shortest Processing Time (SPT) rule when minimizing the total completion time. The authors emphasize the importance of considering the original objective in rescheduling, given that several preparations may have been initiated before scheduling is performed. Then, when rescheduling, they assume the maximum time deviation or the total virtual tardiness for minimizing the disruption concerning the SPT rule. Furthermore, they minimize the completion times, which consists of the original objective. In this case, the due dates of each product are defined as the completion times of the original schedule. Based on these two objectives, they found the set of Pareto optimal solutions of the problem using pseudo-polynomial-time algorithms for resolution.

A dynamic rescheduling in a jobshop scenario is considered in the work of Rangsaritratsamee et al. (2004), having as efficiency measures the minimization of makespan and tardiness. As for stability measures, the authors assume the starting times variation between the two schedules and a penalty for the total deviation. A genetic local search algorithm is proposed to solve the problem using simulation to test the two objectives regarding efficiency and stability. Rescheduling is performed periodically as new jobs arrive at the system. For the new scheduling, all the remaining jobs that did not start production until the date of the periodic scheduling are considered. Vieira et al. (2000) propose analytical models to predict the performance of rescheduling methods. The models are used instead of simulations or running experiments, given that these strategies require more computational effort. The authors study a parallel machine scenario, considering three different rescheduling strategies: periodic, hybrid, and event-driven based on the queue size. Results showed that the proposed analytical models accurately estimate the rescheduling performance measures compared to simulation experiments.

Liu and Ro (2014) perform rescheduling on a single machine scenario to deal with machine disruption. At time zero, the start and the end of the unavailable period are known. The authors consider that the scheduling was already planned at this point since several activities like commitments to customers and the allocation of resources must be determined *a priori* based on the original plan. Therefore, in the new schedule, they minimize the disruption from the initial sequence, considering the maximum time deviation in the objective function of the scheduling problem. The authors also emphasize the importance of considering the initial objective function when rescheduling. They present several properties, proposing algorithms to solve the problem. The algorithms also consider the minimization of the makespan and maximum lateness.

Rescheduling on a scenario of unrelated parallel machines is studied by Ozlen and Azizoğlu (2011). The authors minimize the total flow time as the efficiency measure, whereas total disruption cost is taken as the stability measure. They assume that a disrupted job consists of a job allocated in a different machine from its initial schedule. Liu and Zhou (2013) also consider a bi-criteria rescheduling problem. In this case, the authors study an identical parallel machine problem with job rework disruption. For the initial scheduling, they minimize the total completion time, while the stability measure considers the minimization of the number of disrupted jobs. Algorithms are proposed to solve the problem considering both objectives hierarchically and simultaneously.

In the work of Tang and Zhang (2009), the authors study a scheduling problem on identical parallel machines considering machine breakdown. After the disruption, new sequences should be determined minimizing the original objective function of the problem, which consists of the total weighted completion times and the deviation from the initial scheduling, i.e., positive and negative deviations of job completion times. An integer programming model is presented to solve the rescheduling problem taking into account the new number of machines available for processing, using the Lagrangian Relaxation method to solve it. In this case, the capacity constraints are relaxed. Then, to respect the relaxed constraints, a heuristic is proposed to find feasible solutions based on the Lagrangian Relaxation results. The method resulted in low gaps concerning the initial scheduling and low rescheduling computational times.

Another study considering rescheduling on parallel machines subject to disruptions can be seen in the work of Azizoglu and Alagöz (2005). In this case, the objective function of the original problem consists of minimizing the total flow time. In contrast, the stability measure minimizes the number of jobs processed in different machines compared to the initial scheduling. According to the authors, there are two approaches to solve bicriterion problems: the hierarchical approach, in which the most important objective is considered in optimality and restricts the least important objective, and the simultaneous approach, in which both objectives are considered simultaneously. Two hierarchical approaches and a simultaneous approach for solving the problem are proposed.

2.4 Industry 4.0

The term Industry 4.0, also known as the “Fourth Industrial Revolution”, is first introduced in 2011 as a high-tech strategy proposed by Germany to allow growth in the industrial sector (Hofmann and Rüsch, 2017). In Industry 4.0, traditional hierarchical and centralized structures are transformed into autonomous and decentralized ones, triggering changes in production planning (Rossit et al., 2019). According to Mourtzis and Vlachou (2018), Industry 4.0 enables the digitalization of traditional systems, resulting in possible economic opportunities. They emphasize that this transformation requires high-performance processes and flexible production systems. Hofmann and Rüsch (2017) point out that, although the term Industry 4.0 is widely used, there is no consensus on its definition. They define Industry 4.0 as the connectivity of products and services to the internet or other networks, automated and self-optimized production, decentralized control, and autonomous decisions.

Due to the increased availability of data, the ability to store them, the advancement of computational techniques, and the growth of Industry 4.0, machine learning techniques have become an option for dealing with manufacturing problems (Cadavid et al., 2020). Cadavid et al. (2020) state that machine learning techniques can enable Production Planning and Control (PPC) to learn based on historical or real-time data and react to both predicted and unexpected events. The authors perform a systematic literature review, proposing a methodology to implement machine learning techniques for PPC problems. In addition, they map the existing literature papers to identify further research. Trunk et al. (2020) also state that companies deal with a large amount of data that needs to be collected and interpreted to be used in decision-making. The authors affirm that artificial intelligence (AI) can support these activities. However, they emphasize that research is still being developed regarding the use of AI techniques in the company's strategic decisions. Their paper presents a systematic literature review, considering the integration of AI with the strategic decision-making process under uncertainties. The article also highlights the limitations of AI, including the lack of tacit knowledge and the dependence on patterns learned from historical data.

Bueno et al. (2020) present a systematic literature review considering the integration of Industry 4.0 tools and PPC problems. According to their research, the Internet of Things (IoT) and Big Data and Analytics with Artificial Intelligence (BDA/AI) are the Industry 4.0 technologies most used when solving PPC problems, being mainly employed in shop floor control and scheduling, and inventory planning and control. Regarding the smart capabilities provided by Industry 4.0 to PPC, the authors cite the ability to manage resources, logistic flows, products, and support decisions in real-time, the adaptability to the production environment, the ability to synchronize the PPC activities in a physical manufacturing environment, among others. They also state that another smart capability consists of the ability to predict demands and production events, reacting to them. The authors highlight that new capabilities provided by Industry 4.0 are poorly explored when considering Sales & Operations Planning (S&OP)/Aggregate planning, Master Production Scheduling (MPS), and Material Requirements Planning (MRP). Some of the research topics proposed by the authors include studies regarding how Industry 4.0 support PPC problems for medium and long term horizons, the integration of S&OP, MRP, and MPS through frameworks and models based on the concepts of Industry 4.0, and studies considering the effects of distributed manufacturing on traditional hierarchical PPC.

Ruiz-Sarmiento et al. (2020) propose a predictive maintenance model based on a machine learning tool for the stainless steel industry considering an Industry 4.0 scenario. Based on the degradation level measured by monitoring the machine, they predict the maintenances. The proposed approach learns from new data, adapting its operations. The results are compared with a preventive approach and with regression models. According to the authors, in Industry 4.0, scheduled and control-based processes and systems will evolve to smart ones, able to predict the behavior of the system actors and self-adapt their operations. Rossit et al. (2019) also consider the Industry 4.0 scenario. The authors introduce Smart Scheduling, a framework based on dynamic scheduling combining traditional scheduling with Smart Manufacturing. They emphasize that one of the main contributions of this approach is dealing with unforeseen events. For this, they use the Tolerance Scheduling problem to identify the intervals in which a schedule remains optimal or leastwise feasible in practice. Therefore, rescheduling is performed only if the production sequence violates these intervals.

In the work of Hofmann and Rüsch (2017), the authors study Industry 4.0 applied to logistics management, presenting the main changes in this new scenario and real examples. They point out that the integration of Cyber-Physical Systems (CPS) and IoT may enable materials tracking, improve transportation management, among other benefits. According to the authors, Industry 4.0 can only be completely achieved if logistics deliver materials at the right time and place with the desired quality. Experts of logistics and supply chain management were interviewed, analyzing the main findings of the study and their applicability in real future situations.

Mourtzis and Vlachou (2018) propose a cloud-based CPS to solve an adaptive scheduling and a condition-based maintenance problem, which is validated in a mold-making industry. The proposed system uses real-time data gathered by a monitoring module from the shop floor. The data is analyzed and used as an input in the scheduling algorithm and a condition-based maintenance approach. The adaptive scheduling modifies the production sequence based on real-time information. The monitoring of the schedule allows that, if any uncertainty occurs, the sequence can be revised by performing a rescheduling. The proposed CPS system guarantees integration and interconnection with other systems, facilitating adaptive decision-making and improving the information flow. The CPS system was installed in the company, considering the monitoring module's hardware and the software of the scheduling algorithm and condition-based maintenance. The proposed approach presented a lower resolution time compared to the original method used by the company. Further, the proposed system showed highly satisfying results compared to traditional dispatching rules.

In the paper of Ghaleb et al. (2020), the authors consider real-time scheduling for the flexible jobshop scheduling problem in an Industry 4.0 scenario. The proposed approach deals with unexpected job arrivals and machine random breakdowns. The authors emphasize that in order to take full advantage of information technology from Industry 4.0, we should consider models using real-time information in all manufacturing processes, including production planning and scheduling. They investigate whether real-time information can improve scheduling decisions and when and how this information can be used. Several rescheduling policies are also analyzed to compare the use of continuous scheduling with event-driven rescheduling.

Bueno et al. (2020) emphasize that smart capabilities related to scheduling in the Industry 4.0 context include real-time shop floor monitoring, resource traceability, data collection, real-time information sharing for collaborative scheduling, data-driven simulations to perform predictions, among others. In this work, we follow the definition of Industry 4.0 presented by Hofmann and Rüsch (2017), considering that Industry 4.0 allows connectivity in the factory and autonomous decisions. Our work studies the changes suffered by the PPC problems in this scenario, focusing on using data collected in the industry for failures prediction, besides considering real-time information to alter the scheduling algorithm, as presented in Section 5.2. The proposed framework (Section 5.4) also allows autonomous decisions, dynamically predicting the best approach for implementation based on the instance set.

2.4.1 Predictive models

In this section, we consider predictive models, also presenting papers that solve PPC problems applying machine learning algorithms. We exhibit works dealing with classification problems and using machine learning methods to solve time series forecasting. We focus on classification and time series forecasting since we implement them to solve the studied problem, as will be explained in Chapter 5. A predictive problem consists of finding a model to predict future values (Fernández et al., 2018). Susto et al. (2015) state that machine learning techniques seem to be the most popular approach to solve predictive maintenance problems. They classify machine learning-based predictive maintenance into two classes: supervised and unsupervised. In the supervised class, the modeling dataset contains information about the failures, while in the unsupervised, information about maintenances is not available. Models that consider quantitative outputs are usually referred to as regression models, while models considering qualitative outputs are called classification models (James et al., 2013).

When there are only two classes in a classification problem, we refer to it as binary classification. In some cases, the class of interest has lower observations than the others, generating an imbalanced dataset (Fernández et al., 2018). In an imbalanced binary classification, the class with few observations but high relevance is referred to as the minority or positive class, while the other class is named majority or negative class (Sun et al., 2009). James et al. (2013) emphasize that when selecting a model for training, the more important it is to choose based on the type of output, not on the predictors. Some reasons for predictive models to fail are related to inadequate data preprocessing and model validation, the application of a model in a space that was never seen, and overfitting (Kuhn and Johnson, 2013). James et al. (2013) also state that no method dominates other learning methods for all possible datasets. Then, choosing the best one for a particular dataset can be one of the most challenging parts of statistical learning. We highlight that this task is even more complicated when considering imbalanced data.

According to Carvalho et al. (2019), the data collected in industrial systems can be processed and analyzed to provide valuable information to the decision-maker. The authors present a literature review of machine learning methods applied to predictive maintenance. Given that predictive maintenance actions are performed when needed based on continuous monitoring, its use can maximize equipment utilization, minimize the number of repairs, and reduce costs. The literature review showed that studies considering predictive maintenance are recent. After 2013 the number of publications on this topic grew significantly, which may be due to the increase in data utilization and advances in machine learning techniques. Their paper shows that, considering the machine learning methods, 33.0% of the literature papers used Random Forest (RF), 27.0% used neural network-based methods, 25.0% used Support Vector Machine (SVM), and 13.0% used k-means. The authors present sets of public data for predictive maintenance strategies. They emphasize that predictive maintenance strategies are usually dependent on the studied problem and application, also stating that before applying them, we need to implement preventive and corrective maintenances in the production process to obtain data for modeling.

Ayvaz and Alpay (2021) propose a predictive maintenance system based on data gathered by sensors in an industrial setting. The study aims to predict failures before their occurrence using machine learning methods, which guide preventive actions. They compare several algorithms and hyperparameters to find the most suitable model for the dataset and apply bagging and boosting methods to deal with imbalanced data. They consider a regression problem since they aim to estimate the remaining useful time before failure to perform preventive maintenances. The authors emphasize that a possible limitation of their study is that models are trained based on a specific dataset from only one factory. Therefore, they may not generalize for other industrial scenarios.

2.4.1.1 Classification problems

Sun et al. (2009) state that most classifier learning algorithms usually consider a balanced dataset and equal misclassification costs. However, they emphasize that in some instances, predicting the events of the minority class has greater value than predicting the majority class events, e.g., in disease diagnostic. The authors review classification considering imbalanced datasets. One example of such imbalance occurrence cited by the authors is in manufacturing plants. In this context, the number of defective cases is significantly lower than normal ones. They present methods to deal with class imbalance consisting of data-level and algorithm-level approaches, cost-sensitive learning, and boosting. In addition, they comment on the advantages and disadvantages of using each strategy.

Branco et al. (2016) also review imbalanced problems, discussing the challenges, the main approaches, and the metrics used to solve them, besides proposing a taxonomy for the resolution methods. They present four strategies for dealing with the imbalance: modifications on the dataset, modifications on the learning algorithms, adjustments on the predictions, and hybrid methods. Santos et al. (2018) study a fault diagnosis application of gearboxes in wind turbines aiming to find the classification technique less affected by the class imbalance rate.

According to Branco et al. (2016), predictive models are obtained through an optimization process, aiming to find the "optimal" parameters guided for some criteria. The authors also state that for classification the criteria is usually the error rate, while for regression problems, we generally assume the mean squared error. However, they emphasize that when using these traditional metrics on imbalanced problems we may obtain suboptimal models since these metrics do not consider the user preferences related to the minority class. They comment, for example, that the accuracy metric is not indicated when dealing with imbalanced classification classes, given that if we have a sample with 99% of the data composed by the majority class (negative events), we may obtain an accuracy of 99% only predicting the events of this class, i.e., without predicting any positive event. As in the work of Sun et al. (2009), they present several metrics more adequate for imbalanced problems.

After classification, the samples can be categorized as True Positive (TP), False Positive (FP), False Negative (FN), and True Negative (TN), forming the Confusion Matrix; see the paper of Sun et al. (2009). When we plot the False Positive rate on the x-axis and the True Positive rate on the y-axis, considering each threshold, we obtain the Receiver Operating Characteristic (ROC) curve. However, it is hard to define which model performs better when examining several curves. Then, the area under the ROC curve (AUC) is usually used, representing what model performs better on average (Sun et al., 2009). For more details on the ROC curve and other metrics for imbalanced data, see the work of Sun et al. (2009). The true positive rate is also called sensitivity, while the true negative rate is called specificity. When plotting the trade-off of sensitivity and 1 - specificity on a graph, the model with the larger area under the ROC curve would be the most effective (Kuhn and Johnson, 2013).

Burez and Van den Poel (2009) consider four methods to deal with the class imbalance in customer churn prediction. The first method consists of using evaluation metrics more adequate for imbalances, e.g., AUC and lift metrics. They use the ROC curves since the AUC is not biased when considering the minority class; that is, there is no emphasis on a particular class. The lift metric is considered due to its large use by Customer Relationship Management (CRM) managers. The second method consists of cost-sensitive methods, using the weighted RF for prediction. They emphasize that when considering cost-sensitive learning, bias is desirable. When we assign a greater cost to false negatives than false positives, we may improve the performance when classifying rare events. The third method consists of sampling (random and an advanced under-sampling method), and lastly, they consider boosting (stochastic gradient boosting). Prediction models are created for six different datasets using real data, which makes the classification more difficult. To test the prediction models on unseen data, the authors use the two-fold cross-validation, considering five repetitions. The under-sampling method presented good performance, especially when combined with the AUC metric. The advanced under-sampling did not improve the results, while the cost-sensitive method (weighted RF) performed better when compared to the original RF algorithm. Boosting never presented the best performance compared to the other approaches.

In the work of Priore et al. (2018), ensemble methods are used to solve a scheduling problem in Flexible Manufacturing Systems (FMSs). The ensemble methods are employed aiming to improve the accuracy of the individual classifiers and the performance of dispatching rules. The authors state that dispatching rules may work well for a given state of the system but may perform poorly if the state changes. Thus, they emphasize the importance of developing systems capable of adapting to these modifications. According to the authors, this can be done in two ways, simulating which dispatching rule performs better at each moment or using AI techniques. In their work, they initially perform simulations to create examples that feed the machine learning algorithm. These simulations generate the state of the FMS and define the best dispatching rule for each state. The output of the machine learning algorithm consists of knowledge that may be used to determine dispatching rules in a real-time control system. This process generates feedback between the control system and the FMS. Information on the system state and performance may be returned to create new examples. The proposed ensemble methods consider the output of the machine learning algorithms, evaluating their reliability, and proposing a combination of dispatching rules. A similar problem is treated by Priore et al. (2006), which compares three different types of machine learning algorithms for dynamic scheduling in FMSs. Another application of a machine learning classifier in scheduling can be seen in the work of Shiue et al. (2012).

2.4.1.2 Time Series Forecasting problems

Machine learning algorithms are also heavily used for time series forecasting. According to Sagheer and Kotb (2019), time series forecasting consists of predicting the future values of a sequence using historical data. The authors propose a Deep Long-Short Term Memory (DLSTM) approach and test the method in two case studies considering the petroleum industry. They claim that DLSTM can deal with the nonlinearity and complexity of time series data. A multi-layer Long-Short Term Memory (LSTM) is also proposed by Abbasimehr et al. (2020) to predict demand fluctuations. They compare their proposed forecasting model with traditional statistical algorithms and machine learning methods, including the Autoregressive Integrated Moving Average (ARIMA), Exponential Smoothing (ETS), Artificial Neural Network (ANN), K-Nearest Neighbors (KNN), Recurrent Neural Network (RNN), SVM, and single-layer LSTM. The suggested model performed better than the others when applied to a furniture company. According to the authors, the LSTM hyperparameters to optimize are the lag size (number of past observations), the number of hidden layers, the dropout rate, the number of neurons in each layer, the optimization algorithm, the learning rate, the epoch size, and the batch size. They employ a grid search for the definition of these hyperparameters.

Kim and Cho (2019) propose a combination of the Convolutional Neural Network (CNN) with the LSTM, named CNN-LSTM, to predict housing energy consumption. The problem consists of a multivariate time series forecasting, in which sensors capture data of several features used as input to the prediction model. In the CNN-LSTM, the CNN removes noise and deals with the correlation between multivariate features, whereas the LSTM models temporal information and separates the time series into spaces to generate future values. The proposed model presented higher performance when compared with other machine learning algorithms. A CNN-LSTM approach is also proposed by Vidal and Kristjanpoller (2020) to predict gold volatility.

Other papers considering time series forecasting can be seen in the works of Kim (2003) and Taieb et al. (2010). Kim (2003) predict stock prices using the SVM method, also providing a comparison with the ANN and case-based reasoning (CBR) approaches. Taieb et al. (2010) review multiple-output approaches to time series prediction, comparing with single-output methods.

2.5 Concluding remarks

Regarding the studied problem, the paper of Dolgui et al. (2005) is the most similar to our work. In this case, the authors studied lot-sizing and sequencing problems subject to uncertainties. Considering the robust optimization, the paper of Lambrechts et al. (2011) is the one that most resembles this work, given that idle times are considered in the model to guarantee the robustness of the solution. However, the studied problem is different.

When we consider the learning algorithms for prediction, the article of Burez and Van den Poel (2009) is the most similar to this work when dealing with the classification problem, while the paper of Sagheer and Kotb (2019) resembles this work since they also use DLSTM to perform time series forecasting. Nonetheless, we did not find literature papers considering various production planning levels using machine learning techniques to deal with uncertainties.

Therefore, the gap we fill with the proposition of this work consists of studying how the new possibilities brought by Industry 4.0 and the changes and tendencies of the society are affecting production planning problems. We study the impacts of these changes on decision-making and on the algorithms used.

Chapter 3

Preliminary definitions

In this chapter, we describe the characteristics of the studied problem. We also show the notation, the short-term production planning mathematical model, and the proposed scheduling heuristic used to solve the problems in all the chapters of the thesis.

3.1 Production setting under study

We study short-term production planning and scheduling problems subject to uncertainties. These uncertainties are defined by machine failures that can occur while performing the sequence in a two-parallel machine environment. The products can be manufactured on both machines, which have the same limited capacity. We consider sequence-dependent setup times and resumable jobs, i.e., after the maintenance, we may continue producing the product being processed when the machine failed. In the non-resumable situation, we must restart the processing of the product, losing the quantities produced before the failure (Fan et al., 2011). We assume this scenario for analysis, but the approaches proposed in this work are adaptable to other production features.

Since the studied production process handles continuous quantities, we consider that the machines can produce the same product at the same time (job splitting). Thus, percentages of the total amount of the same product can be simultaneously manufactured on different machines. However, to process the same product on both machines, we highlight that this will result in sequence-dependent setup times. This strategy can help to find feasible solutions in cases where one of the machines has reached its capacity limit while the other is idle, matching the makespan of the two machines.

Failures may occur during processing or setup times. In Chapter 4, if failures happen during setups, the preparation must be restarted from the beginning. In addition, we have different durations for repairing the machines. In Chapter 5, after the repair, the production or setup can resume from the point where they stopped due to the disruption. Furthermore, in Chapter 5, we assume that all repairs are performed within an hour. We consider this simplified environment due to the difficulty of predicting the moment and duration of the failures simultaneously, given that these durations may depend on factors other than the characteristics of the machines. We emphasize that different assumptions are adopted for each chapter since we did not aim to compare the strategies proposed for both chapters, as explained in more detail later.

We hypothesize that, in this new scenario, customized orders will arrive and will be quickly produced. Therefore, large quantities will no longer be stocked to meet the demands of customers. Thus, inventory costs are irrelevant on the short-term horizon unless we have specific constraints to stock products, which is not the case in this work. On the other hand, if the factory does not have available capacity, it might not fulfill certain demands. Since products must be delivered in quantities and dates requested by the customers, we maximize the production quantities at the short-term production planning and minimize the total weighted tardiness at the scheduling level.

In this work, we assume that an initial production sequence is found using the short-term production planning problem and, at the lower level, this sequence can be modified by the scheduling method. In this case, at the higher level, we define the quantities to be produced, trying to maximize them, aiming to make good use of the productive capacity. Therefore, we use a mathematical model that already considers real sequence-dependent setups instead of average values. Finally, at the scheduling level, we sequence the quantities of products defined to be produced taking into account the due dates agreed with customers.

3.2 Notation

We present in Tables 3.1, 3.2, and 3.3, respectively, the sets, parameters, and decision variables used in the approaches proposed in this work. In the tables, each period $t \in T$ consists of a week with 112 production hours.

Table 3.1: Sets used in the approaches.

Data	Description
J	Set of products ($j \in J$)
M	Set of machines ($m \in M$)
T	Set of planning periods ($t \in T$)

Table 3.2: Parameters used in the approaches.

Data	Description
t'	Period being currently scheduled
p_j	Processing time of product j
S_{ij}	Setup time for processing product j after the production of product i
D_{jt}	Demand of product j at time period t
C_{mt}	Production capacity of machine m at time period t

Table 3.3: Decision variables used in the approaches.

Data	Description
z_{jm}	Continuous variable indicating the percentage of product j produced on machine m
u_{jm}	Integer variable indicating the position occupied by product j on machine m, such that $0 \leq u_{jm} \leq \|J\| - 1$
y_{jm}	Binary variable indicating if product j is produced ($y_{jm} = 1$) or not ($y_{jm} = 0$) on machine m
x_{ijm}	Binary variable indicating if product i is immediately produced before product j ($x_{ijm} = 1$) or not ($x_{ijm} = 0$) on machine m

3.3 Short-term production planning formulation

The short-term production planning formulation used in this thesis is an adaptation of the models presented in the works of Miller et al. (1960), Parker et al. (1977), and Pochet and Wolsey (2006). The objective function maximizes the production quantities for the period being scheduled (t'), as presented in (3.1):

$$\sum_{j \in J} \sum_{m \in M} D_{jt'} z_{jm} \tag{3.1}$$

Constraints (3.2) enforce the capacity constraints, guaranteeing that the sum of production and setup times are lower than the capacity of the period t' for each machine m.

$$\sum_{j \in J} p_j D_{jt'} z_{jm} + \sum_{i \in J \cup \{0\}} \sum_{j \in J : i \neq j} S_{ij} x_{ijm} \leq C_{mt'}, \qquad \forall m \in M, \tag{3.2}$$

Constraints (3.3) ensure that percentages of product j will only be produced on machine m if variable y_{jm} is equal to 1.

$$z_{jm} \leq y_{jm}, \qquad \forall j \in J, \forall m \in M, \tag{3.3}$$

Equations (3.4) indicate that, if product i is produced on machine m ($y_{im} = 1$), only one variable x_{ijm} is equal to 1, i.e., after processing i only one product j is produced on machine m. In the case of $y_{im} = 0$, then $x_{ijm} = 0$. Equations (3.5) guarantee that the same is valid for product j. In this case, only one product is produced before product j.

$$\sum_{j \in J \cup \{0\} : i \neq j} x_{ijm} = y_{im}, \qquad \forall i \in J, \forall m \in M, \tag{3.4}$$

$$\sum_{i \in J \cup \{0\} : i \neq j} x_{ijm} = y_{jm}, \qquad \forall j \in J, \forall m \in M, \tag{3.5}$$

Equations (3.6) and (3.7) indicate that the dummy product 0 initiates and ends the production sequence.

$$\sum_{j \in J} x_{0jm} = 1, \qquad \forall m \in M, \tag{3.6}$$

$$\sum_{i \in J} x_{i0m} = 1, \qquad \forall m \in M, \tag{3.7}$$

Constraints (3.8) eliminates cycles, while Constraints (3.9) indicate that variable u_{jm} only assumes a value different from zero if variable $y_{jm} = 1$.

$$u_{im} - u_{jm} + |J| x_{ijm} \leq |J| - 1, \qquad \forall i \in J, \forall j \in J : i \neq j, \forall m \in M, \tag{3.8}$$

$$u_{jm} \leq |J| y_{jm}, \qquad \forall j \in J, \forall m \in M, \tag{3.9}$$

Constraints (3.10) ensure that the sum of the production percentages of j on both machines has to be lower or equal to 1.

$$\sum_{m \in M} z_{jm} \leq 1, \qquad \forall j \in J, \tag{3.10}$$

Lastly, Constraints (3.11)-(3.14) define the domain of the decision variables.

$$x_{ijm} \in \{0, 1\}, \qquad \forall i \in J \cup \{0\}, \forall j \in J \cup \{0\}, \forall m \in M, \tag{3.11}$$

$$y_{jm} \in \{0, 1\}, \qquad \forall j \in J, \forall m \in M, \tag{3.12}$$

$$z_{jm} \in \mathbb{R}_+, \qquad \forall j \in J, \forall m \in M, \tag{3.13}$$

$$u_{jm} \in \mathbb{Z}_+, \qquad 0 \leq u_{jm} \leq |J| - 1, \forall j \in J, \forall m \in M. \tag{3.14}$$

Model (3.1)-(3.14) indirectly minimizes the total setup time given that as we maximize production quantities, the model tries to minimize the setup times at the capacity constraints (3.2), allowing us to increase the production quantities focusing on customer service. In this work, we consider that customers accept to place fractional orders, i.e., the factory may not meet the total demand. In the scheduling problem, we aim to deliver these orders before their due dates. We solve the problems hierarchically; first, the short-term production planning is solved using the model presented in this section, and based on the solution found, we schedule the products. Furthermore, in Chapter 5, we consider a rolling horizon since, in this chapter, we assume two weeks of planning. In this case, we solve both problems for the first period and implement the resulting sequence in the factory. Then, we solve the planning and scheduling problems for the next week based on real information from the previous period.

3.4 Scheduling heuristic

In the new scenario considering Industry 4.0 characteristics, it is essential to deliver products respecting the due dates. Therefore, we consider that the main objective at the scheduling level is to minimize the total weighted tardiness. To make better use of production capacity, we also minimize the makespan as a secondary objective. However, we emphasize that the priority objective minimizes the total weighted tardiness, given that we assume characteristics of an Industry 4.0 environment. Then, in the framework proposed in Section 5.4, we will not consider the minimization of the makespan. The heuristic proposed for solving the scheduling problem, presented below, considers job splitting. Steps 1 to 5 of the heuristic are based on the *Heuristic 1 (Slack-based heuristic)* of Park et al. (2012). In Park et al. (2012), the authors consider the slack consisting of the difference between the completion times and the due dates to allocate the jobs. Here we consider the weighted tardiness. Steps 6 to 8 representing the job splitting integrated with the local searches are contributions of this work. We show in Tables 3.4 and 3.5, respectively, the sets and parameters used specifically in the scheduling heuristic. Appendix A presents an analysis of the performance of the proposed heuristic compared to a mathematical model.

Table 3.4: Sets used in the scheduling heuristic.

Set	Description
A	Set of unscheduled products
Seq_m	Production sequence of machine m

Table 3.5: Parameters used in the scheduling heuristic.

Parameter	Description
L_m	Last product allocated to machine m
CT_j	Completion time of product j
m'	Machine with the current lowest makespan
T_j	Tardiness of product j
W_j	Weight of product j
d_{jt}	Due date of product j at time period t
$Cmax$	Makespan

Step 1. Initialize set A with the unscheduled products, $L_m = 0$ ($\forall m \in M$), $CT_0 = 0$, and $Seq_m = \{\emptyset\}$.

Step 2. For each product $j \in A$:

a. Set m' equal to the first available machine, i.e., the one with the lowest CT_{L_m}, to produce j.

b. Calculate the completion time of product j (CT_j) considering that j will be produced on machine m'.

Step 3. Calculate the weighted tardiness of product j, such that $W_jT_j = W_jmax(CT_j - d_{jt'}, 0)$ for all $j \in A$.

Step 4. Choose the product j^* with the largest weighted tardiness to occupy the first available position of the m' machine. Set $L_{m'} = j^*$, i.e., product j^* occupy the last position of machine m'. If more than one product has the same value for the parameter W_jT_j, j^* is defined as the product with the highest weight (W_j) and the lowest due date ($d_{jt'}$). If more than one product has these characteristics, set j^* as the product with the lowest due date. If there are two products with equal due dates, choose the product with the highest weight.

Step 5. Set $A = A - \{j^*\}$ and $Seq_{m'} = Seq_{m'} \cup \{j^*\}$. If $A \neq \emptyset$, return to Step 2. Calculate the total weighted tardiness, saving the solution.

Step 6. [Job Splitting] If any machine has its capacity violated while the other machine is idle, transfer the production quantities from the last product allocated to the machine with violated capacity to the idle machine. Repeat until the violated machine respects the capacity. Based on the sequence obtained, balance the machines in a way that both have the same completion time. Such balancing is carried out to minimize the weighted tardiness of the product allocated to the last position of both machines and also to minimize their makespan. If the new sequence has a lower total weighted tardiness than the saved sequence, replace it with the new solution.

Step 7. Perform local searches on each machine considering swap neighborhoods based on the sequence found before Step 6. If all the products of each sequence were exchanged, go to Step 8. Otherwise, go to Step 6.

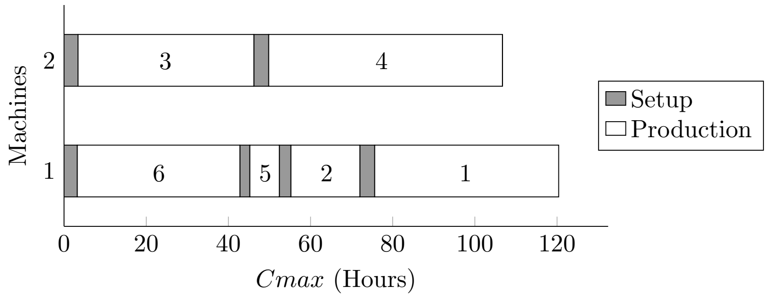
Step 8. Perform local searches procedures in the allocation of products to machines, exchanging products between them. If all the products of each sequence were exchanged, finalize the algorithm. Otherwise, go to Step 6.

The motivation of the scheduling heuristic consists of allocating a product j with the greatest weighted tardiness in the machine with the lowest current makespan. When solving the scheduling problem with a mathematical model at optimality, only small instances could be solved due to its complexity, see Appendix A. However, the instances solved enabled us to observe a specific characteristic of the solutions that were used for the job splitting step of the heuristic. We note that, when solving optimally, the job splitting occurs at the end of the capacity, usually for the last products of the sequences, which is due to the sequence-dependent setup times. We only produce the same product on both machines if there is no available capacity to produce it in only one machine. We implement this strategy along with local searches aiming to improve the results obtained by the scheduling heuristic. The output of the scheduling heuristic is a solution with the lowest total weighted tardiness with makespans for both machines obeying the capacity. However, if the heuristic does not find a solution with makespans respecting the production capacity, the solution with the lowest total weighted tardiness is the scheduling output. We highlight that the approaches proposed in this work are independent of the heuristic procedure. In this case, we choose a simple and efficient heuristic to solve the studied problem.

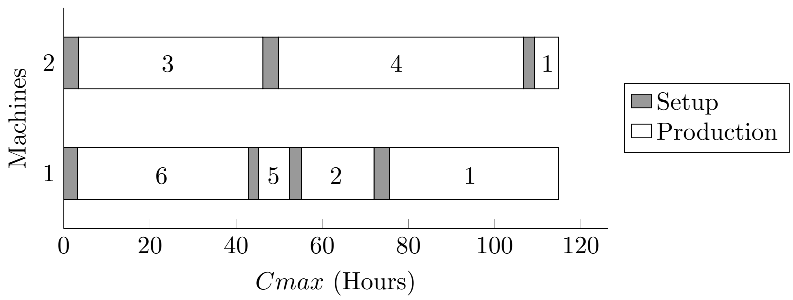
The objective function of the short-term production planning problem correlates with the scheduling objective in that both aim to consider the concepts of Industry 4.0 by prioritizing customer satisfaction. In the short-term production planning problem, we try to deliver as many required products as possible. At the scheduling, we aim to meet the delivery dates of these promised products to the customers.

Figure 3.1 shows an example of an instance with 6 products solved with the proposed scheduling heuristic. Figure 3.1(a) presents the first solution found, reaching a total weighted tardiness of 726.30 units (Steps 1 to 5 of the heuristic). When performing Step 6, we obtain the solution presented in Figure 3.1(b), with total weighted tardiness equal to 709.50 units. After executing

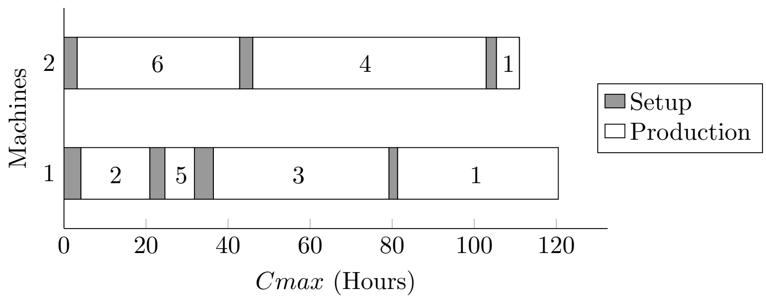
Steps 7 and 8, we obtain the sequences of Figure 3.1(c), with a final total weighted tardiness of 595.60 units. For improving the final solution, units of product 1 produced on machine 1 could be transferred to machine 2 to reduce the completion time of this product. In this example, the solutions presented in Figure 3.1 do not respect the production capacity. Then, quantities of the product causing the infeasibility would not be delivered to the customer, or the factory could use a different strategy to manufacture these quantities.



(a) First solution



(b) Solution after job splitting



(c) Solution after local searches

Figure 3.1: Example of an instance with 6 products solved with the proposed scheduling heuristic.

Chapter 4

Efficiency comparison of proactive approaches

In this chapter, we present two proactive approaches for dealing with machine failures. In these propositions, decision-makers strategically define slacks added to the short-term production planning model to anticipate the capacity lost due to the uncertainties. The idea is to increase the probability of feasibility of the solutions even with failures, reducing the need for rescheduling. A high degree of protection in cuts guaranteed by greater slacks can generate very conservative solutions, which may not be feasible in practice. Furthermore, sometimes the failures will greatly affect the production process, but in other cases, it will have little impact. Therefore, decision-makers should choose a sequence and parameters based on their knowledge of the production system.

The choice of the sequence to be implemented must consider a high probability of feasibility regarding to the production capacity. The trade-off to be analyzed in this case consists of: if failures do not occur, we lose productive capacity. However, if there are any disruptions, we will be able to maintain deliveries to consumers respecting the due dates. This work aims to investigate if there is a difference between the two proposed proactive approaches when considering the studied environment. In addition, different slacks are analyzed. We hypothesize that the main objective when solving planning problems in a future scenario will no longer be cost minimization but the delays in products demanded by customers. Then, we maximize production quantities at the short-term production planning, while at the scheduling problem, we minimize the total weighted tardiness. Tables 4.1 and 4.2 present the new set and parameters used in this chapter. Furthermore, we emphasize that, in this chapter, the parameter indicating the makespan is indexed considering the machine m and solution l ($Cmax_{ml}$).

Table 4.1: New set used in the proactive approaches.

Set	Description
Φ	Set of solutions with the lowest total weighted tardiness ($l \in \Phi$)

4.1 Proactive Approach I

In this section, we present a proactive approach to solve production planning problems when considering machine failures, named Proactive Approach I (PA-I). In the proposed method, we aim to guarantee a specified probability of feasibility of the scheduling solution when solving the short-term production planning problem. On the shop floor, the production sequence will have a greater chance of resulting in a makespan inferior to the capacity, even with the occurrence of disruptions. The method is based on robust models. For information on robustness, see the

Table 4.2: New parameters used in the proactive approaches.

Data	Description
λ_m	Average duration of failures on machine m determined by simulation
σ_m	Standard deviation of the duration of failures on machine m determined by simulation
ρ_m	Average number of failures on machine m determined by simulation
δ	Average duration of failures based on the probability distribution
ϕ	Average setup time based on the probability distribution
Δ_m	Robustness parameter of machine m
DF	Duration of the failure
TBF	Time Between Failures
η	Degree of conservatism of the solution
$Prob_{ml}$	Probability of infeasibility for machine m considering solution l
$E(\sigma_{ml})$	Expected standard deviation of the makespan for machine m considering solution l
$E(Cmax_{ml})$	Expected makespan of machine m considering solution l
$E(Idle_{ml})$	Expected idle hours of machine m considering solution l
$E(\sum_{j\in J} W_jT_j)$	Expected total weighted tardiness calculated for each solution l

works of Soyster (1973) and Bertsimas and Sim (2004). Soyster (1973) and Bertsimas and Sim (2004) present general models that could be used to solve different types of linear optimization problems with uncertainties. In our work, the robust model is proposed based on the specific characteristics defined for the studied production process. However, we emphasize that proper modifications can be made to address other problems and scenarios.

We use the buffering approach in the proposed proactive methods, which consists of adding idle times in the proactive scheduling to reduce the impacts of the failure when it happens; see the work of Mehta and Uzsoy (1998). However, this strategy may incur in unused periods and waste of capacity if any uncertainty occurs (Wang et al., 2015). In the buffering approach, determining when and how much idle time will be added to the original sequence is crucial, as it will directly affect the scheduling performance (Gürel et al., 2010). In this work, we use the buffering strategy at the short-term production planning level, adding slacks to the capacity constraints to find robust solutions able to absorb the failures at the shop floor.

One advantage of the traditional robust optimization is that the data sets analysis usually provides a good approximation to the uncertainty considered, especially when the uncertain parameters are not stochastic and the probability distribution is unknown (Bertsimas et al., 2011; Li and Floudas, 2014). Then, here we made an *a priori* analysis based on the historical data set of the uncertain parameters to include such information robustly in the short-term production planning problem.

First, we solve the short-term production planning problem optimally with the formulation presented in Section 3.3. To find robust solutions, we add to the formulation the cuts shown in Constraints (4.1). The cuts represent a modification of the capacity constraints (3.2). In this case, we add slacks to the production capacity when solving the short-term production planning, trying to guarantee a high probability of feasibility on the shop floor. Based on a simulation of the failures, we calculate the average duration of the failures (λ_m), the average number of failures (ρ_m), and the standard deviation of the duration of the failures (σ_m) for each machine m. The parameter Δ_m, called robustness parameter, is calculated based on σ_m. We consider three variations to test the robustness of the solutions. In the first case, $\Delta_m = 0$. In the second case, $\Delta_m = 0.5\sigma_m$, while in the third case, $\Delta_m = \sigma_m$. These parameters are added to the Constraints (3.2) to generate robust solutions, as shown in Constraints (4.1). We expect that, with greater values for Δ_m, we find lower probabilities of exceeding the production capacity on the shop floor. However, it is necessary to emphasize that the greater the slack added to Constraints (4.1), the less production capacity will be available, which may affect product deliveries to customers. The objective of the proposed approaches is to provide the decision-makers with several robustness options. Based on this information, they could decide which production plan to implement, considering the risk they are willing to take.

$$\sum_{j \in J} p_j D_{jt'} z_{jm} + \sum_{i \in J \cup \{0\}} \sum_{j \in J: i \neq j} S_{ij} x_{ijm} + \rho_m \lambda_m + \Delta_m \leq C_{mt'}, \forall m \in M \tag{4.1}$$

Based on the results of the short-term production planning problem, we solve the scheduling with the heuristic presented in Section 3.4. In this chapter, we modify the proposed heuristic to have as output the five solutions with the lowest total weighted tardiness ($l \in \Phi$). Each solution l includes the sequences for both machines and the objective function value. Then, a simulation is performed on each of these sequences to find expected results and provide the decision-maker with the best production sequences, i.e., those that guarantee the best customer service even with disruptions. The simulation performs 1000 runs, which differ concerning when the failures occur and their respective durations. The output consists of expected makespans for each machine m and each solution l ($E(Cmax_{ml})$), expected values for the total weighted tardiness ($E(\sum_{j \in J} W_j T_j)$), expected standard deviation of the value found for the makespan ($E(\sigma_{ml})$), and expected idle hours for each machine m and solution l ($E(Idle_{ml})$). Based on these values, the probability of each sequence being infeasible is calculated ($Prob_{ml}$) for each $m \in M$ and $l \in \Phi$.

To choose which of the five solutions will be passed to the decision-maker, we use the following criterion: Considering the two machines, if both sequences of a solution l have the lowest probabilities of infeasibility ($Prob_{ml}$), we choose this solution. Otherwise, we calculate the average probability of each solution l and select the one with the lowest average. The selected solution is represented by l^*. The next step consists of verifying if any of the probabilities of infeasibility of the chosen solution is greater than ϵ. The parameter ϵ indicates the maximum probability of infeasibility accepted for implementing the solution. In this work, we define $\epsilon = 0.50$, denoting that the solution will only be implemented if the sequences of both machines have probabilities of infeasibility lower than 50.00%.

We define the ϵ parameter as 50.00% to compare both proposed approaches. To use the methods in real cases, decision-makers could choose this value based on the information they have about the production process and its failures. If failures do not have much impact on the process, decision-makers could set a high value for this parameter, considering that only adding the slacks in the capacity constraints of the short-term production planning problem would be enough to deal with failures. Otherwise, if the disruptions have a great impact, decision-makers could stipulate a lower value for ϵ, ensuring that solutions with low probabilities of infeasibility are found. Then, the parameter ϵ could be adjusted according to the analyzed scenario.

If the probabilities of infeasibility are lower than $\epsilon = 0.50$ for both machines, these sequences are outputs of the proposed procedure. Otherwise, information is iteratively returned for the short-term production planning problem. The idea is that when performing simulations for the five best solutions found, we acquire information to use in cuts added to the planning problem. In this work, the information returned consists of the difference between the expected makespan ($E(Cmax_{ml})$) and the sum of the production and setup times obtained in the previous iteration ($\sum_{j \in J} p_j D_{jt'} z_{jm} + \sum_{i \in J \cup \{0\}} \sum_{j \in J: i \neq j} S_{ij} x_{ijm}$), i.e., we return the parameter Y_{mi} for each machine m and iteration i, calculated as $Y_{mi} = E(Cmax_{ml}) - (\sum_{j \in J} p_j D_{jt'} z_{jm} + \sum_{i \in J \cup \{0\}} \sum_{j \in J: i \neq j} S_{ij} x_{ijm})$. Then, the cuts showed in Constraints (4.2) are inserted in the short-term production planning model iteractively until reaching a solution l^* with both machines presenting probabilities of infeasibility lower than $\epsilon = 0.50$.

$$\sum_{j \in J} p_j D_{jt'} z_{jm} + \sum_{i \in J \cup \{0\}} \sum_{j \in J: i \neq j} S_{ij} x_{ijm} + Y_{mi} \leq C_{mt'}, \forall m \in M, i \in \Omega \tag{4.2}$$

In Constraints (4.2), Ω indicates the set of total iterations performed until reaching a feasible solution. We emphasize that we are not dealing with real-time feedback. Information will be returned before implementing the sequence on the shop floor, indicating a fully proactive approach.

Figure 4.1 presents a flowchart showing the steps of the proposed approaches. In general, we first solve the short-term production planning model with the cuts (4.1) aiming to find robust solutions. Then, we solve the scheduling with a proposed heuristic, having as output the five best solutions regarding the total weighted tardiness. Simulations of failures are then performed on the best solutions, resulting in expected values for several parameters. Based on these expected values, we choose the best solution (l^*) according to the probabilities of infeasibility. If $Prob_{ml^*} > \epsilon$ for any m, we iteratively return information to the short-term production planning problem, adding the cuts (4.2) to the model until finding a solution respecting $Prob_{ml^*} \leq \epsilon$.

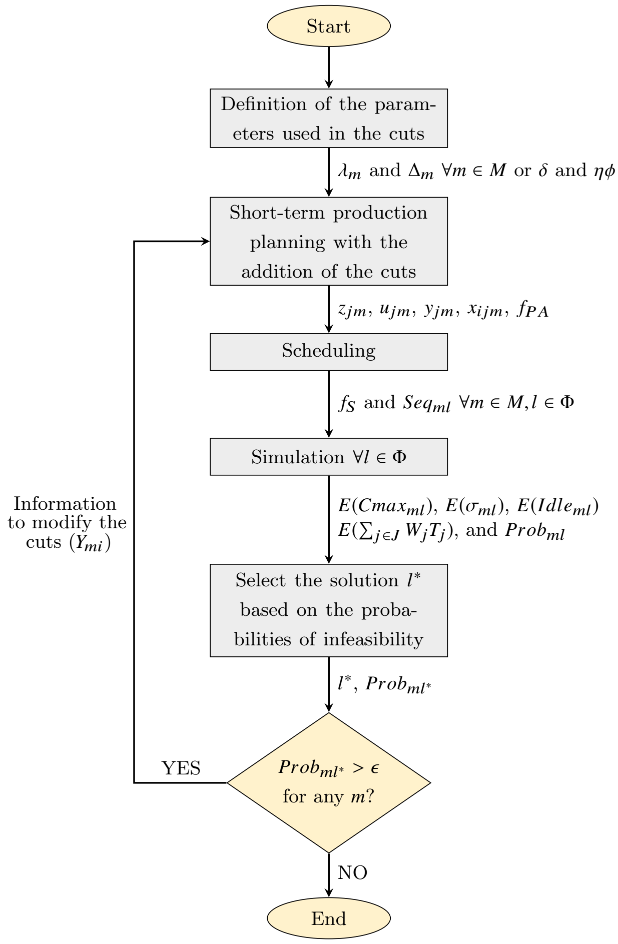
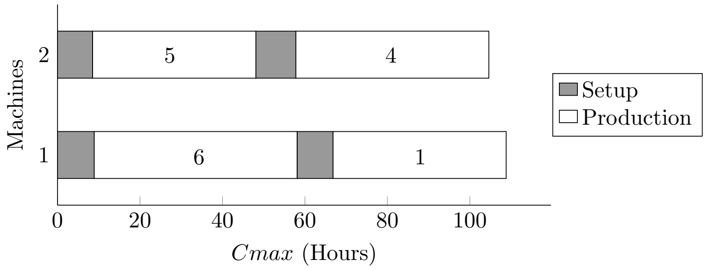


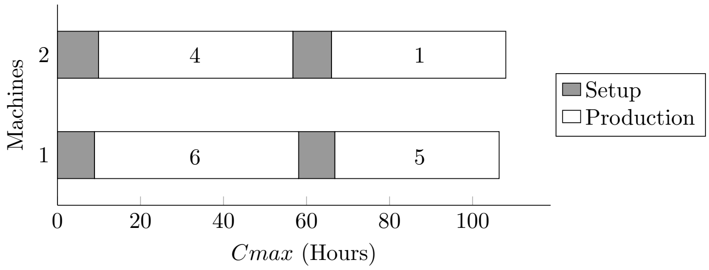
Figure 4.1: Flowchart representing the proposed approaches. f_{PA} indicates the objective function found by the short-term production planning model of the proactive approach. f_S and Seq_{ml} represent, respectively, the objective function of the scheduling problem and the sequence found for each solution $l \in \Phi$ and each machine $m \in M$. The parameters δ and $\eta\phi$ will be used in the cuts presented in Section 4.2.

4.1.1 Example of PA-I

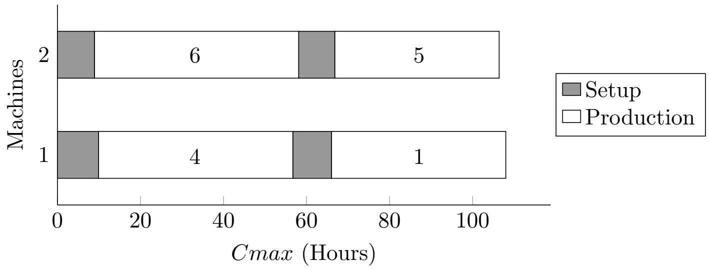
In this section, we present an example of PA-I considering an instance with 6 products and variation 1 ($\Delta_m = 0$ $\forall m \in M$). Initially, we obtain the five solutions shown in Figure 4.2 when solving the short-term production planning and scheduling problems. Figure 4.2 also presents the total weighted tardiness found before the consideration of failures for each solution. When performing the simulation before the resolution of the problems, we obtain an average duration of failures (λ_m) equal to 2.00 and 1.90 hours and an average number of failures (ρ_m) equal to 1.50 and 1.60 for machines 1 and 2, respectively. Therefore, we added the cuts represented by Constraints (4.1) considering the addition of $\rho_m \lambda_m + \Delta_m$ for each machine $m \in M$. The solutions exhibited in Figure 4.2 consider these slacks at the capacity constraint.



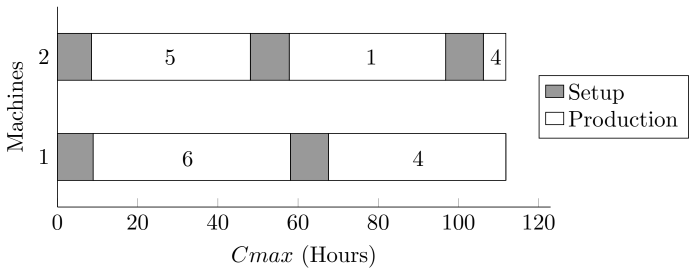
(a) Solution 1 - $\sum_{j \in J} W_j T_j$ = 457.40



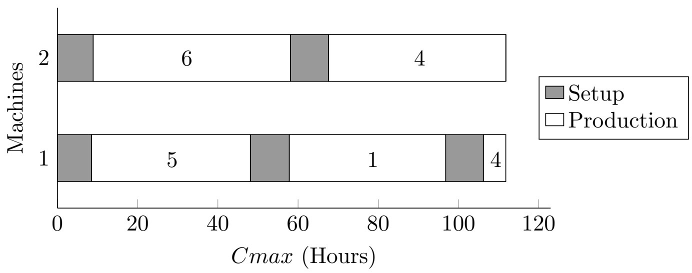
(b) Solution 2 - $\sum_{j \in J} W_j T_j$ = 470.70



(c) Solution 3 - $\sum_{j \in J} W_j T_j$ = 470.70



(d) Solution 4 - $\sum_{j \in J} W_j T_j$ = 450.20



(e) Solution 5 - $\sum_{j \in J} W_j T_j$ = 450.20

Figure 4.2: Example of an instance with 6 products solved with PA-I.

Table 4.3 presents the expected results after the simulations for each solution l. As shown in the table, neither solution has the lowest probabilities of infeasibility for both machines. Then, we calculate the average probabilities for choosing the solution l to be implemented on the shop floor. We obtain average probabilities equal to 0.35, 0.38, 0.37, 0.85, and 0.84 for solutions 1, 2, 3, 4, and 5, respectively. Therefore, solution $l = 1$ is chosen for implementation. However, we first test if any of the sequences of solution $l = 1$ present probabilities of infeasibility greater than $\epsilon = 50.00\%$. As $Prob_{11} = 53.00\%$, we conclude that solution $l = 1$ is not feasible and return information to the short-term production planning model, adding the cuts represented by Constraints (4.2).

Table 4.3: Expected results after the simulations for each solution l and each machine m - Iteration 1.

Data	Solution 1		Solution 2		Solution 3		Solution 4		Solution 5	
	$m = 1$	$m = 2$	$m = 1$	$m = 2$	$m = 1$	$m = 2$	$m = 1$	$m = 2$	$m = 1$	$m = 2$
$E(Cmax)$	112.30	108.30	109.90	111.70	111.70	109.90	115.60	116.20	116.00	115.60
$E(\sigma)$	3.60	3.70	3.60	3.70	3.70	3.50	3.70	4.00	4.00	3.60
$E(Idle)$	1.10	4.10	2.80	1.50	1.50	2.80	0.00	0.00	0.00	0.00
$Prob$	53.00%	16.00%	28.00%	47.00%	47.00%	27.00%	84.00%	85.00%	84.00%	84.00%

In the first iteration, the objective function of the short-term production planning problem was equal to 48.00 units and the expected total weighted tardiness equal to 499.00, 504.80, 504.40, 501.60, and 501.30 units for solutions 1, 2, 3, 4, and 5, respectively. Further, the sum of processing and setup times ($\sum_{j \in J} p_j D_{jt'} z_{jm} + \sum_{i \in J \cup \{0\}} \sum_{j \in J : i \neq j} S_{ij} x_{ijm}$) resulted in 107.50 hours for machine 1 and 106.40 hours for machine 2. Based on the information of expected makespans, $E(Cmax_{11}) = 112.30$ and $E(Cmax_{21}) = 108.30$, we can calculate the parameter Y_{mi} for each machine m and iteration $i = 1$, i.e., $Y_{11} = 4.80$ and $Y_{21} = 1.90$. Then, we use this information in the cuts (4.2). After 8 iterations, we obtain the results presented in Table 4.4.

Table 4.4: Expected results after the simulations for each solution l and each machine m - Final iteration.

Data	Solution 1		Solution 2		Solution 3		Solution 4		Solution 5	
	$m = 1$	$m = 2$	$m = 1$	$m = 2$	$m = 1$	$m = 2$	$m = 1$	$m = 2$	$m = 1$	$m = 2$
$E(Cmax)$	112.30	106.10	109.90	109.60	112.30	107.80	115.60	116.20	116.00	115.60
$E(\sigma)$	3.60	3.70	3.60	3.60	3.60	3.80	3.70	4.00	4.00	3.60
$E(Idle)$	1.10	6.00	2.80	3.10	1.10	4.60	0.00	0.00	0.00	0.00
$Prob$	53.00%	6.00%	28.00%	25.00%	53.00%	14.00%	84.00%	85.00%	84.00%	84.00%

When calculating the averages of the probabilities of infeasibility for each solution, we have 0.30, 0.27, 0.34, 0.85, and 0.84 for solutions 1, 2, 3, 4, and 5, respectively. Then, solution $l = 2$ is selected. As $Prob_{12} \leq \epsilon$ and $Prob_{22} \leq \epsilon$, this solution is the output of the algorithm, with the objective function of the short-term production planning model equal to 47.46 and an expected total weighted tardiness of 496.10 units.

In this example, the probability of infeasibility of machine 1 in the initially chosen solution $l = 1$ (iteration $i = 1$) was greater than ϵ. However, in the final iteration, we see from Table 4.4 that this probability remains the same, as well as the expected values found for the makespan, standard deviation, and idle hours. This behavior is due to the fact that, after solving the short-term production planning problem, the products to produce and their respective quantities are transferred to the scheduling heuristic, which may change the sequences and the products between both machines. When solving the short-term production planning for the tested instance, the quantities to manufacture of product 4 (defined to be produced on machine 1) are reduced over the iterations to respect the cuts (4.2). Nonetheless, when solving the scheduling, the heuristic alters the allocation of product 4 to machine 2, reducing the expected values and the probability of infeasibility for this machine. Therefore, we observe a modification of the values for machine 2, which initially obeyed the criterion of having a probability of infeasibility lower or equal to ϵ.

We emphasize that this behavior was observed in this example; it does not represent all tested instances, as there may be cases in which the values of the machine that was initially infeasible ($Prob_{ml} > \epsilon$) change.

4.2 Proactive Approach II (PA-II)

In the Proactive Approach II (PA-II), the robustness is based on values obtained *a priori* considering the probability distribution instead of simulations. In this case, we modify the capacity constraint considering the average duration of the failures and their impact on the setup times. Assuming that the duration of the failures follow an exponential distribution $DF \sim Exp(\lambda = 1/2)$, while the setup times follow the distribution $S_{ij} \sim U(50/n, 50/n + 10/n)$, with n indicating the number of products. Then, in PA-II, the robust production capacity is calculated by reducing the quantities δ and $\eta\phi$. Considering the example, $\delta = 2$, i.e., it is equal to the average duration of the failure and $\phi = trunc((50/n + (50/n + 10/n))/2, 1)$. Here we truncate the average setup in a decimal place. We use the parameter η to indicate the degree of conservatism of the solution; η weighs the importance of the variation in setup times due to the failures. We intend to define a value for this parameter so that it is not too conservative, possibly leading to a loss of production capacity. Constraints (4.3) show the cuts added in PA-II to the short-term production planning model presented in Section 3.3.

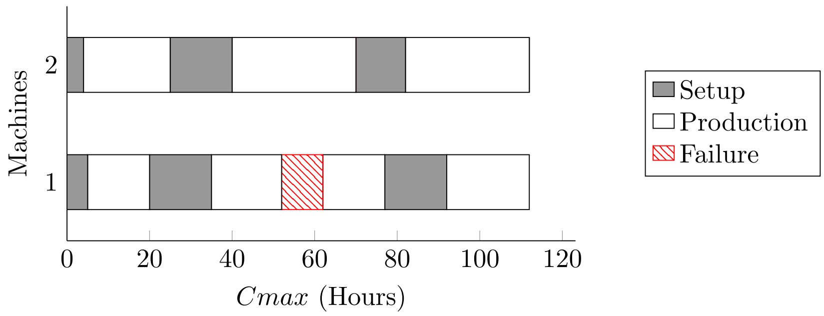
$$\sum_{j \in J} p_j D_{jt'} z_{jm} + \sum_{i \in J \cup \{0\}} \sum_{j \in J : i \neq j} S_{ij} x_{ijm} + \delta + \eta\phi \leq C_{mt'}, \forall m \in M \tag{4.3}$$

We performed several preliminary tests to determine the best way to modify the production capacity to deal with failures. We remove only the average duration of the failure from the production capacity, we remove a certain percentage of the capacity, e.g., $C_{mt'} = 0.8C_{mt'}$, and also remove the average duration of the failure plus a robust parameter (η) multiplied by the average setup, as explained earlier. To choose which alternative to implement, we calculate the probability of the final sequence to be infeasible related to the capacity ($Cmax > C_{mt'}$) for each case. Results showed that removing from the production capacity the average duration of the failure plus η multiplied by the average setup, as shown in Constraints (4.3), results in lower probabilities of infeasibility. In this work, we consider three variations for the η parameter: $\eta = 0$, $\eta = 0.3$, and $\eta = 0.5$ named Variations 1, 2, and 3, respectively.

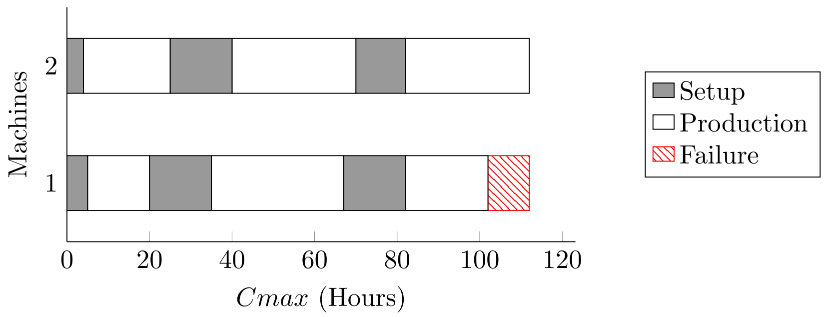
As in PA-I, if the probabilities of infeasibility of the chosen solution l^* are greater than ϵ, we return the information Y_{mi} to the short-term production planning. Then, we solve the problem iteratively until finding a solution such that $Prob_{ml^*} \leq \epsilon$.

4.3 Hypothesis

When uncertainties do not cause changes in the order in which the products were sequenced, we hypothesize that using PA-I or PA-II entails similar results. Considering machine failures, we suppose that we do not need to know the exact production sequence and perform simulations; adding a certain amount of time in the capacity constraints of the short-term production planning problem will guarantee low probabilities of infeasibility even with disruptions. Figure 4.3 presents an example of this hypothesis. Assuming that a failure occurs at time $t = 52.00$ on machine 1. Since the disruption duration is 10.00 hours, the product interrupted by the failure will continue to be produced only in the instant $t = 62.00$, resulting in a makespan of 112.00 hours. Case 1 of Figure 4.3 presents the final planning, considering the machine failure of a generic situation.



(a) Case 1



(b) Case 2

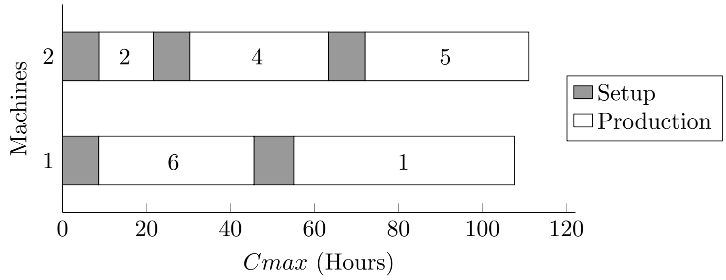
Figure 4.3: Example of the hypothesis that both approaches PA-I and PA-II result in similar solutions.

In Case 1 of Figure 4.3, the initial production sequence does not change after the failure occurs. In this case, we can consider that the final makespan will be the initial deterministic $Cmax_{ml^*}$ $\forall m \in M$, without the disruption, plus the duration of the failure at the end, as shown in Case 2, also resulting in $Cmax_{ml^*} = 112$ hours $\forall m \in M$. Therefore, in this work, we test if solving the problem considering Cases 1 or 2 results in similar objective functions, i.e., we examine if considering the probability distribution *a priori* is sufficient to deal with the disruption or if it is necessary to perform simulations.

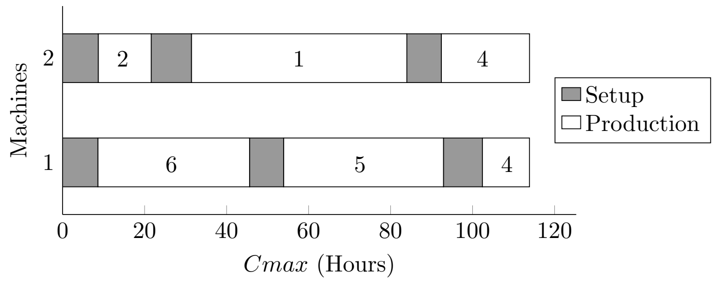
4.4 Analysis based on the five best solutions

In this section, we present an example of the analysis based on the simulation results, justifying the use of the five best deterministic solutions. Suppose that when solving an instance using PA-I, the scheduling problem resulted in the sequences shown in Figure 4.4. As mentioned, the scheduling heuristic has as its output the five best sequences found, i.e., those with the lowest total weighted tardiness. However, for simplicity of the example, the analysis will be performed based on the three sequences presented in Figure 4.4. Initially, without considering the failures, Solution 1 resulted in total weighted tardiness equal to 357.93, while Solution 2 obtained a value equal to 435.41, and Solution 3 resulted in a value of 386.59 units. When running the simulations, we obtain the results summarized in Table 4.5. The table shows the results obtained for the expected makespan ($E(Cmax_{ml})$), expected standard deviation of the makespan ($E(\sigma_{ml})$), and the probability of infeasibility after failures ($Prob_{ml}$). We present the results for each machine m and each solution l. Furthermore, the simulation output values for the expected total weighted tardiness equal to 724.40, 1174.40, and 674.90 units for Solutions 1, 2, and 3, respectively.

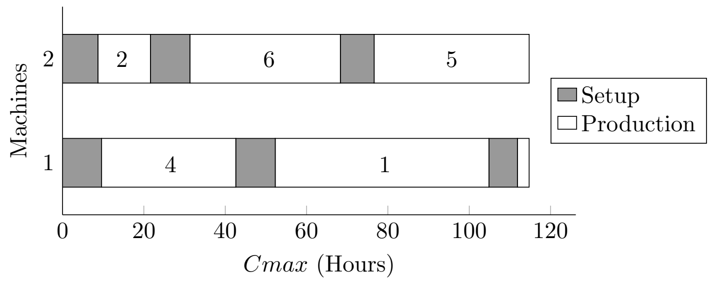
When analyzing Table 4.5, we can observe that despite the values found for the total weighted



(a) Solution 1



(b) Solution 2



(c) Solution 3

Figure 4.4: Example considering an instance with six products.

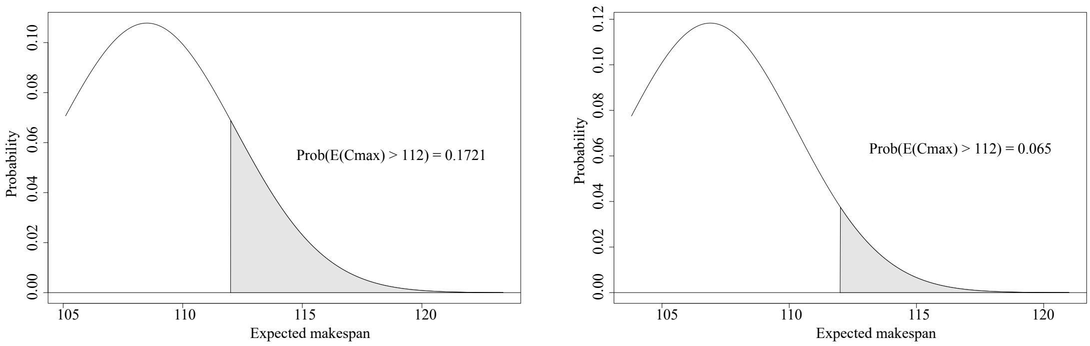
tardiness without considering the failures being very close for Solutions 1 and 3, the expected results after the simulations were quite different for the metrics examined. For this reason, we choose to consider the best five deterministic sequences for the analysis.

For the cases in which we have only two products to be produced in a specific machine, we consider the two possible sequences determined by the scheduling level to calculate the probability of infeasibility after the failures. Thus, the decision-maker could choose which sequence to implement based on these probabilities. We emphasize that the probabilities of infeasibility are calculated based on simulations. At each run, we save the resulting makespan considering the failures simulation. After the runs, the distribution of expected makespans will exhibit a Gaussian behavior, or we expect it to be at least symmetric. Thus, we can calculate the probability of the expected makespan to be greater than the production capacity.

Assuming that two sequences were found when considering the resolution of an instance with four jobs at the scheduling level. Then, when performing the simulation for each of these sequences, we obtain the distributions shown in Figure 4.5. Supposing that the production capacity is equal to 112 hours, as we can see, Sequence 1 has a probability of approximately 17.21% to be infeasible after the failure (makespan greater than the production capacity), while Sequence 2 has a probability of approximately 6.50%. Therefore, Sequence 2 may be a better choice considering the proposed analysis.

Table 4.5: Results after simulations considering the example shown in Figure 4.4.

Data	Solution 1		Solution 2		Solution 3	
	$m = 1$	$m = 2$	$m = 1$	$m = 2$	$m = 1$	$m = 2$
$E(Cmax)$	111.30	115.30	118.10	118.20	118.80	119.00
$E(\sigma)$	3.70	4.00	4.00	4.20	3.70	4.20
$Prob$	42.00%	79.00%	94.00%	93.00%	97.00%	95.00%



(a) Sequence 1

(b) Sequence 2

Figure 4.5: Probability distributions for each sequence. The highlighted area indicates the probability of the sequence being infeasible after the occurrence of the failure.

4.5 Computational Experiments

In this section, we present the instances and results obtained. Implementations were carried out with software AMPL R API and solved with CPLEX 12.10. The tests were executed on an Intel Core i5 processor computer with a Windows operating system with 8 GB of RAM. The resolution time for the short-term production planning model was limited to 120 seconds. We performed preliminary tests to analyze this time limit, which indicated that solutions close to the optimal solutions are found in this time interval for most of the analyzed instances.

4.5.1 Instances

We consider the following product quantities for the experiments: 4, 6, 8, 10, 12, 15, 20, 50, and 100, generating fifteen instances for each product quantity. The planning horizon consists of one period with 112 hours. We assume only one planning period to incorporate into the study the characteristics of Industry 4.0. Further, we consider two parallel machines ($|M| = 2$) presenting the same capacity ($C_{mt'} = 112 \ \forall m \in M$). Sequence-dependent setup times obey the triangle inequality. For data generation, we use the uniform distribution considering two distinct cases presented in Table 4.6. Case I assumes that setup times are greater than production times, whereas Case II considers the opposite. In addition, the demand has a larger variation for Case II, resulting in instances in which all the demand is produced without violating the available capacity. We aim to verify if the approaches behave differently when two different situations are considered. We emphasize that we generate the setup and processing time parameters considering its dependency on the number of products n aiming to guarantee that the instances present objective functions with the same order of magnitude for the short-term production planning problem when considering any product quantity. For more information on this, see the work of Alves et al. (2021). The processing and setup times were truncated to one decimal place, while the demand was rounded up to the next largest integer. The values of the weight and due date were rounded to the nearest integer.

Table 4.6: Data distributions, with n indicating the product quantities.

Parameter	Case I	Case II
Processing time (p_j)	$U(20/n, 20/n + 10/n)$	$U(20/n, 20/n + 20/n)$
Setup time (S_{ij})	$U(50/n, 50/n + 10/n)$	$U(10/n, 10/n + 20/n)$
Demand (D_{jt})	$U(8, 12)$	$U(1, 10)$
Weight (W_j)	$U(2, 5)$	$U(2, 5)$
Due date (d_{jt})	$U(max\{j \in J\}S_{0j} + p_j, C_{mt})$	$U(max\{j \in J\}S_{0j} + p_j, C_{mt})$

Vieira et al. (2000), Dolgui et al. (2005), and Lambrechts et al. (2011) assume that times to failure are exponentially distributed. One advantage of using exponential distributions is that we only need the mean time to failures to know their distribution function (Lambrechts et al., 2011). Therefore, for the simulations, we generate time between failures for each machine following the exponential distribution $TBF \sim Exp(\lambda = 1/112)$, determining the moments in which failures occur. The parameters DF were generated based on three scenarios: considering the distribution $DF \sim Exp(\lambda = 1/2)$ for Scenario 1, $DF \sim Exp(\lambda = 1/5)$ for Scenario 2, and $DF \sim Exp(\lambda = 1/7)$ for Scenario 3.

4.5.2 Computational results

We implement a hierarchical approach without disruptions to compare the results of the proposed proactive approaches. In this case, we first solve the short-term production planning problem at optimality with the model presented in Section 3.3. Based on the results, we solve the scheduling problem with the proposed heuristic. Equation (4.4) calculates the deviation of the results obtained in the short-term production planning with the proposed strategies related to the results achieved by the hierarchical approach without disruptions. The deviation is represented by Dev. In the equation, f_{PA} indicates the objective function of the proactive approaches, while f denotes the objective function of the short-term production planning problem of the hierarchical approach. Appendix B shows the average results obtained by the hierarchical approach.

$$Dev = \frac{f_{PA} - f}{f_{PA}} \tag{4.4}$$

Table 4.7 presents the average results using PA-I to solve the instances of Case I, whereas Table 4.8 exhibits the average results for Case II. The averages consider the fifteen instances generated for each product quantity. The first and second columns of the tables show the scenario and the number of products considered, respectively. Columns 3, 4, 5, 6, and 7 present the results obtained for Variation 1 ($\Delta_m = 0$). In column 3, we have the average deviation, calculated as shown in Equation (4.4), while column 4 presents the average expected total weighted tardiness, represented by $E(\sum_{j \in J} W_j T_j)$. Columns 5 and 6 show the average probabilities of infeasibility for machines 1 and 2, respectively, which are represented by $P_m \; \forall m \in M$. In column 7, we present the average expected total idle time ($E(Idle)$). In this case, for each simulation run, we calculate the idle time of both machines resulting from the addition of cuts in the short-term production planning problem. In the same way, columns 8, 9, 10, 11, and 12 present the results for Variation 2 ($\Delta_m = 0.5\sigma_m$), and columns 13, 14, 15, 16, and 17 show the results for Variation 3 ($\Delta_m = \sigma_m$).

As we observe in Tables 4.7 and 4.8, the deviation (Dev) usually increases as we move from Scenario 1 to Scenario 3 considering all product quantities, which can be due to the greater duration of failures for Scenario 3. As we use the average duration of failures in the cuts added to the production planning model, when we increase these durations, a smaller amount of products is produced because of the capacity constraint. Thus, the short-term production planning model objective function will be smaller, distancing itself from the objective function of the hierarchical approach, which does not consider failures. In our analysis, the closer the deviations are to zero, the better, which indicates proximity to the objective function of a non-disruptive environment.

Table 4.7: Average results obtained for PA-I considering Case I.

Scenario	n	Variation 1					Variation 2					Variation 3				
		Dev	$E(\sum_{j\in J} W_jT_j)$	P_1	P_2	$E(Idle)$	Dev	$E(\sum_{j\in J} W_jT_j)$	P_1	P_2	$E(Idle)$	Dev	$E(\sum_{j\in J} W_jT_j)$	P_1	P_2	$E(Idle)$
	4	**-12.50%**	220.47	17.33%	21.07%	13.61	-12.85%	207.48	15.67%	19.60%	15.28	-9.28%	236.30	20.27%	30.27%	12.36
	6	**-6.22%**	348.99	22.40%	19.13%	9.18	-5.96%	356.11	26.33%	23.20%	8.02	-5.75%	357.21	25.80%	22.33%	7.95
	8	**-7.97%**	283.30	23.20%	21.27%	7.61	-7.63%	288.81	25.87%	22.27%	7.01	-7.75%	292.49	27.40%	23.93%	6.62
	10	**-7.12%**	251.39	30.27%	23.67%	6.53	-6.92%	259.97	34.07%	26.87%	5.65	-6.83%	255.82	33.13%	30.13%	5.56
1	12	**-6.53%**	237.11	29.87%	38.40%	4.93	-6.66%	233.40	28.73%	34.73%	5.21	-6.66%	231.60	29.93%	29.73%	5.49
	15	**-6.07%**	236.36	30.40%	37.47%	5.04	-6.07%	220.95	28.60%	36.53%	4.68	-5.89%	226.84	35.13%	36.07%	4.55
	20	**-5.45%**	288.81	35.47%	38.07%	4.09	-5.66%	310.95	33.47%	32.20%	4.90	-5.41%	317.25	33.73%	38.60%	4.19
	50	**-4.40%**	578.67	34.27%	37.67%	3.91	-4.59%	485.19	30.80%	36.00%	4.25	-4.41%	584.00	32.93%	37.33%	3.87
	100	**-7.81%**	294.34	22.20%	22.27%	13.03	-9.67%	248.13	17.13%	18.53%	14.77	-4.26%	308.87	21.53%	25.20%	7.27
Average		-7.12%	304.38	27.27%	28.78%	7.55	-7.33%	290.11	26.74%	27.77%	7.75	-6.25%	312.26	28.87%	30.40%	6.43
SD		2.31%	110.12	6.19%	8.74%	3.69	2.51%	86.85	6.51%	7.22%	4.29	1.60%	111.27	5.46%	5.94%	2.62
	4	-20.37%	219.95	15.73%	24.40%	26.32	-13.99%	228.79	25.40%	34.20%	16.91	-15.44%	233.55	20.13%	27.73%	17.29
	6	-9.58%	362.98	40.93%	30.20%	10.38	-9.26%	368.03	33.07%	36.07%	10.67	-11.13%	328.16	25.00%	26.00%	14.18
	8	-13.16%	277.73	34.53%	33.27%	10.55	-12.79%	308.90	34.00%	34.73%	10.59	-13.24%	301.91	36.60%	30.40%	10.75
	10	-12.86%	253.35	30.00%	35.07%	11.73	-11.34%	278.92	37.07%	41.67%	9.15	-12.39%	283.29	32.47%	34.00%	11.13
2	12	-10.78%	259.63	36.00%	31.33%	10.78	-11.07%	278.56	36.20%	35.53%	11.06	-11.44%	273.27	31.87%	29.73%	11.65
	15	-11.06%	254.72	37.67%	31.60%	10.68	-10.86%	255.28	35.73%	34.80%	10.49	-11.68%	271.96	33.93%	31.53%	11.15
	20	-10.30%	309.71	31.73%	37.73%	10.24	-10.33%	323.57	38.33%	32.00%	10.19	-11.61%	285.55	30.73%	27.13%	12.27
	50	-8.56%	613.37	40.40%	44.80%	7.52	-9.30%	583.32	42.20%	34.87%	8.60	-11.21%	557.42	29.33%	28.07%	11.87
	100	-7.99%	580.51	39.20%	37.93%	9.45	-15.77%	427.19	26.07%	25.07%	19.61	-15.79%	380.65	21.07%	19.33%	20.25
Average		-11.63%	348.00	34.02%	34.04%	11.96	-11.63%	339.17	34.23%	34.33%	11.92	-12.66%	323.97	29.01%	28.21%	13.39
SD		3.70%	147.04	7.82%	5.78%	5.51	2.17%	109.47	5.48%	4.33%	3.74	1.80%	96.69	5.73%	4.13%	3.29
	4	-21.44%	231.93	26.47%	27.47%	23.23	-17.75%	230.67	22.87%	38.07%	21.14	-20.64%	271.35	16.20%	25.60%	25.24
	6	-14.32%	327.02	32.80%	34.00%	18.29	-12.96%	341.52	29.80%	36.33%	15.46	-15.55%	323.45	20.93%	29.80%	19.09
	8	-17.61%	275.55	37.20%	38.67%	14.41	-17.96%	292.57	37.33%	31.40%	16.29	-20.03%	276.37	25.33%	28.80%	19.05
	10	-15.89%	286.23	33.33%	41.07%	14.15	-14.93%	255.17	34.40%	39.47%	13.66	-17.73%	269.76	26.07%	28.60%	18.16
3	12	-14.23%	265.77	34.13%	35.53%	14.53	-13.46%	254.01	39.93%	33.07%	13.46	-16.03%	262.63	29.27%	26.00%	17.51
	15	-13.82%	259.91	32.27%	39.53%	13.74	-13.83%	261.53	38.13%	35.27%	13.49	-16.57%	221.31	25.20%	29.73%	17.42
	20	-13.06%	320.64	39.53%	39.07%	12.32	-13.80%	305.11	37.40%	35.67%	13.29	-16.25%	280.93	29.87%	26.80%	17.23
	50	-11.49%	716.90	43.87%	43.93%	10.04	-12.73%	691.57	38.80%	39.13%	12.05	-16.35%	486.94	27.73%	24.40%	17.93
	100	-13.62%	699.51	38.93%	37.07%	15.26	-12.53%	673.34	35.33%	33.80%	13.76	-17.22%	524.85	21.53%	20.73%	20.61
Average		-15.05%	375.94	35.39%	37.37%	15.11	-14.44%	367.28	34.89%	35.80%	14.73	-17.38%	324.18	24.68%	26.72%	19.14
SD		2.95%	190.68	5.08%	4.73%	3.76	2.06%	181.73	5.41%	2.76%	2.71	1.80%	106.68	4.41%	2.95%	2.53

Table 4.8: Average results obtained for PA-I considering Case II.

Scenario	n	Variation 1					Variation 2					Variation 3				
		Dev	$E(\sum_{j\in J} W_jT_j)$	P_1	P_2	$E(Idle)$	Dev	$E(\sum_{j\in J} W_jT_j)$	P_1	P_2	$E(Idle)$	Dev	$E(\sum_{j\in J} W_jT_j)$	P_1	P_2	$E(Idle)$
	4	-1.34%	273.99	12.40%	8.47%	31.23	-1.33%	270.07	14.53%	10.07%	31.01	-1.28%	264.33	14.27%	10.20%	30.95
	6	-1.04%	285.74	8.20%	8.20%	38.48	-1.10%	284.82	7.67%	7.73%	38.59	-1.17%	280.89	7.67%	7.73%	38.78
	8	-1.17%	225.85	11.80%	9.07%	27.67	-1.16%	220.81	12.27%	9.53%	27.65	-1.22%	225.50	10.67%	8.80%	27.93
	10	-0.83%	241.96	**8.93%**	**8.60%**	29.85	-0.83%	246.16	**11.33%**	**9.60%**	29.76	-0.67%	248.58	**12.60%**	**11.47%**	29.45
1	12	-0.09%	162.31	6.33%	4.93%	31.35	-0.09%	161.37	6.47%	5.00%	31.30	-0.09%	158.56	7.40%	3.67%	31.33
	15	-0.45%	207.67	**6.20%**	**5.80%**	31.97	-0.54%	209.83	**5.00%**	**4.87%**	32.22	-0.59%	200.64	**4.60%**	**4.13%**	32.27
	20	-0.47%	201.35	4.93%	2.40%	32.17	-0.43%	200.89	5.20%	2.67%	32.03	-0.35%	203.12	3.20%	5.40%	32.03
	50	0.00%	203.63	0.07%	0.00%	35.09	0.00%	204.41	0.00%	0.00%	35.11	0.00%	204.09	0.00%	0.07%	35.06
	100	0.00%	67.98	0.00%	0.00%	55.94	0.00%	67.21	0.00%	0.00%	55.95	0.00%	67.77	0.00%	0.00%	56.05
Average		-0.60%	207.83	6.54%	5.27%	34.86	-0.61%	207.29	6.94%	5.50%	34.85	-0.60%	205.94	6.71%	5.72%	34.87
SD		0.52%	64.83	4.44%	3.68%	8.48	0.52%	64.65	5.11%	4.00%	8.51	0.53%	63.84	5.19%	4.16%	8.55
	4	-3.12%	283.69	22.40%	21.87%	29.98	-2.80%	286.32	25.27%	22.53%	29.44	-3.39%	276.61	21.67%	18.80%	30.49
	6	-2.27%	318.19	12.33%	13.47%	35.31	-2.04%	315.95	14.67%	13.87%	34.83	-2.32%	314.61	12.53%	11.80%	35.35
	8	-2.55%	275.31	20.60%	17.67%	25.17	-2.42%	269.51	21.07%	18.27%	25.10	-2.93%	265.80	17.80%	18.07%	26.03
	10	-2.13%	280.21	19.73%	19.40%	27.19	-2.51%	271.81	15.40%	19.67%	27.93	-2.50%	276.15	17.33%	18.67%	27.73
2	12	-1.19%	204.79	15.87%	12.67%	27.82	-1.17%	203.65	16.53%	13.13%	27.77	-1.14%	202.72	15.40%	14.07%	27.69
	15	-1.40%	251.85	15.47%	11.33%	28.19	-1.11%	250.87	13.13%	13.73%	27.55	-1.30%	247.91	12.20%	11.87%	28.10
	20	-0.97%	293.49	14.73%	13.27%	27.08	-1.11%	289.43	12.33%	14.20%	27.49	-1.18%	289.08	11.73%	14.60%	27.58
	50	0.00%	402.53	6.53%	8.80%	28.06	0.00%	399.60	6.87%	8.40%	28.18	0.00%	400.65	7.47%	7.80%	28.14
	100	0.00%	220.83	0.07%	0.07%	48.91	0.00%	220.85	0.07%	0.07%	48.79	-4.03%	217.11	0.07%	0.07%	52.88
Average		-1.52%	281.21	14.19%	13.17%	30.86	-1.46%	278.67	13.93%	13.76%	30.79	-2.09%	276.74	12.91%	12.86%	31.55
SD		1.10%	57.69	7.12%	6.41%	7.34	1.05%	57.08	7.37%	6.60%	7.24	1.28%	58.02	6.35%	6.04%	8.44
	4	-4.38%	315.86	27.87%	28.40%	29.56	-4.83%	296.87	23.67%	25.20%	30.77	-6.24%	286.83	15.60%	21.33%	33.47
	6	-3.71%	336.11	15.20%	18.80%	34.46	-3.38%	338.13	18.67%	17.00%	33.79	-3.94%	329.53	15.33%	14.60%	35.23
	8	-4.69%	303.43	22.47%	20.67%	26.79	-4.47%	304.07	24.53%	21.53%	25.94	-5.17%	274.21	21.67%	16.67%	27.24
	10	-3.63%	302.27	20.47%	26.20%	26.79	-3.32%	298.65	22.93%	26.73%	26.05	-4.40%	289.06	22.47%	18.00%	28.51
3	12	-1.94%	239.81	23.73%	20.20%	25.81	-2.10%	227.18	21.80%	21.13%	26.13	-2.78%	225.37	16.07%	22.47%	27.55
	15	-1.72%	304.88	19.73%	22.00%	25.51	-2.15%	289.67	17.73%	21.53%	26.15	-2.34%	279.21	18.20%	18.47%	26.61
	20	-1.61%	350.93	17.47%	25.87%	25.09	-1.73%	348.23	18.07%	25.07%	25.43	-1.79%	341.56	16.53%	25.13%	25.44
	50	0.00%	596.59	16.53%	21.47%	24.78	0.00%	593.91	18.67%	19.33%	24.75	-0.31%	583.36	17.33%	18.00%	25.33
	100	0.00%	436.27	2.27%	2.33%	44.52	-2.05%	417.31	2.20%	2.20%	47.17	0.00%	435.41	2.47%	2.13%	44.44
Average		-2.41%	354.02	18.41%	20.66%	29.26	-2.67%	346.00	18.70%	19.97%	29.58	-3.00%	338.28	16.19%	17.42%	30.42
SD		1.77%	104.93	7.21%	7.58%	6.48	1.49%	106.25	6.71%	7.33%	7.23	2.13%	108.97	5.75%	6.56%	6.29

We also note in Tables 4.7 and 4.8 greater deviations for Case I compared to Case II for all scenarios and variations. As in Case II several instances present expected makespans lower than the production capacity resulting in idle hours, failures do not have as much impact on quantities produced as in Case I. Lastly, we observe that the deviations tend to decrease as we increase the product quantities, which is probably related to the lower processing and setup times, as highlighted in Table 4.7 when considering Variation 1 and Scenario 1.

Assuming only one iteration to solve the problem, considering Variation 1, we have a greater available production capacity than variations 2 and 3. Therefore, the objective function would increase, reducing the deviation between this value and the objective function obtained by the hierarchical approach. On the other hand, when considering Variation 3, we have a lower available productive capacity, which would reduce the objective function of the short-term production planning model and increase the deviation Dev. However, as we can see in the Tables 4.7 and 4.8, this behavior does not occur in all scenarios and product quantities, which may be due to the several iterations performed by the algorithms until reaching the final solution. In this case, the objective function would also be affected by the cuts (4.2), which can modify the quantities produced and, consequently, alter the behavior of the objective function in the different variations.

Regarding the expected total weighted tardiness ($E(\sum_{j\in J} W_j T_j)$), we observe an increase in these values when considering Scenario 3 given the high duration of the failures, which affect the completion times, increasing the tardiness of each product. If we have considered only one iteration, we expected that the $E(\sum_{j\in J} W_j T_j)$ values were greater for Variation 1 since it is less protected from failures when compared to variations 2 and 3. However, as discussed when analyzing Dev, this behavior may be changed due to the several iterations performed for each instance until finding a feasible solution ($Prob_{ml^*} \leq \epsilon$). We note in Tables 4.7 and 4.8 that, for several cases, variations 2 and 3 present the highest expected total weighted tardiness. Lastly, we observe high $E(\sum_{j\in J} W_j T_j)$ values considering all the scenarios and variations. In the next chapter, we propose a strategy to predict the machine failures aiming to reduce the expected total weighted tardiness.

When analyzing the probabilities of infeasibility (P_m $\forall m \in M$), we do not observe a pattern when increasing the duration of the failures or modifying the variation; in some cases, the probabilities increase and, in others, they decrease, as highlighted in Table 4.8. In this context, decision-makers could analyze which slack to add to the short-term production planning model based on each specific product quantity. Observing the average probabilities of infeasibility for all the product quantities, we can note an increasing trend when considering Scenario 3 for most variations. These greater probabilities are more evident for Case II, especially compared with the low probabilities of Scenario 1.

We noticed that high average values for the expected number of idle hours ($E(Idle)$) are found, especially when we have long failure durations (see, for instance, Scenarios 2 and 3 of Case I). This increase can be observed especially for Case II, in which the PA-I approach obtains low probabilities of infeasibility and high expected values of idle capacity. As explained, some instances of Case II do not spend all the production capacity for producing the demands in the original scheduling. Therefore, in these cases, the idle hours are not caused by the slacks inserted into the short-term production planning problems. In case they are caused by inserted slacks, it could be necessary to change the parameter ϵ or modify the cuts added to the short-term production planning model to reduce the value of Y_{mi} so that large slacks are not inserted into the cuts at once.

Tables 4.9 and 4.10 show the average results for PA-II considering Cases I and II, respectively. All the columns have the same meaning as Tables 4.7 and 4.8. Nonetheless, in this context, the variations are related to the robustness parameter, named η, as explained in Section 4.2.

Variation 1 refers to $\eta = 0$, whereas Variation 2 assumes $\eta = 0.3$, and Variation 3 considers $\eta = 0.5$. As in PA-I, we also observe an increase in the Dev and $E(\sum_{j \in J} W_j T_j)$ values when the duration of the failures grows (Scenario 1 to Scenario 3) for Cases I and II. In the same way, lower Dev values were found for Case II (compare the highlighted results for Variation 1 and Scenario 1). Lastly, as in PA-I, we observe an increase in the expected idle hours when augmenting the duration of the failures for Case I. When considering Case II, the average considering all product quantities decreases, which also occurred when solving the studied problem with PA-I.

Comparing PA-I and PA-II, we can see that their results were very similar when considering the average values for the different quantities of products and their standard deviations for Cases I and II. The only discrepancy can be seen in Variation 3 for the PA-II for all analyzed scenarios of Case I. In this case, the values presented in Table 4.9 showed a greater difference compared to the results obtained with PA-I. For the other variations and scenarios, the decision-maker could use any proposed approach for implementation on the shop floor.

The approaches presented in this chapter would help the decision-maker to find robust solutions regarding the failures in an environment with characteristics of Industry 4.0. With the proper choice of the robustness parameter, based on experiences with previous disruptions, the decision-maker would have a production sequence that made the best use of production capacity, with few idle hours and low probability of infeasibility in case of failures. Thus, it would be able to promise more accurate delivery dates to the customers and consequently improve the planning of the moments when each production resource would be used.

Table 4.9: Average results obtained for PA-II considering Case I.

Scenario	n	Variation 1					Variation 2					Variation 3				
		Dev	$E(\sum_{j\in J} W_jT_j)$	P_1	P_2	$E(Idle)$	Dev	$E(\sum_{j\in J} W_jT_j)$	P_1	P_2	$E(Idle)$	Dev	$E(\sum_{j\in J} W_jT_j)$	P_1	P_2	$E(Idle)$
	4	**-12.11%**	215.40	20.20%	20.67%	12.43	-7.84%	235.13	29.07%	34.73%	7.59	-10.25%	212.21	15.20%	17.73%	12.24
	6	**-6.57%**	345.43	19.93%	20.80%	9.44	-6.11%	350.82	21.33%	20.93%	8.65	-5.96%	360.31	23.07%	18.27%	8.58
	8	**-7.87%**	304.26	27.40%	21.13%	7.37	-7.61%	287.92	20.80%	23.60%	7.55	-7.82%	292.03	28.00%	21.40%	7.32
	10	**-6.62%**	256.49	39.13%	30.20%	4.83	-6.58%	262.82	36.80%	30.00%	5.40	-6.63%	264.86	33.73%	31.00%	5.25
1	12	**-6.30%**	238.33	31.87%	37.40%	4.50	-6.81%	239.45	31.27%	31.40%	5.17	-6.63%	237.97	29.53%	33.93%	5.56
	15	**-6.64%**	211.11	35.47%	27.80%	5.95	-6.30%	221.24	35.20%	29.67%	4.87	-5.84%	236.91	30.80%	38.13%	4.38
	20	**-5.48%**	268.66	38.27%	33.07%	4.35	-5.59%	296.65	33.60%	30.33%	4.99	-5.55%	297.49	34.73%	30.40%	4.95
	50	**-4.73%**	536.53	33.53%	28.47%	4.58	-4.59%	506.28	35.93%	30.00%	4.27	-4.71%	592.46	28.47%	34.67%	4.59
	100	**-7.54%**	209.17	15.13%	19.60%	12.52	-7.54%	258.62	23.13%	20.80%	13.01	-14.80%	204.62	16.67%	16.60%	22.51
Average		**-7.09%**	287.27	28.99%	26.57%	7.33	-6.55%	295.44	29.68%	27.94%	6.83	-7.58%	299.87	26.69%	26.90%	8.38
SD		2.11%	104.02	8.76%	6.36%	3.35	1.05%	88.23	6.42%	4.93%	2.76	3.15%	119.95	6.98%	8.37%	5.87
	4	-18.84%	205.87	20.27%	20.87%	22.85	-17.44%	222.83	17.33%	31.00%	20.79	-14.81%	230.89	23.73%	30.27%	16.19
	6	-9.77%	354.64	33.47%	34.00%	11.13	-9.73%	369.91	37.73%	32.07%	10.63	-8.99%	370.00	35.73%	39.73%	9.47
	8	-13.53%	287.26	33.27%	34.47%	11.01	-13.11%	276.33	34.67%	34.20%	10.39	-12.47%	303.55	36.40%	30.93%	10.70
	10	-11.92%	267.63	36.73%	33.93%	11.12	-12.19%	273.30	36.20%	35.07%	10.52	-11.86%	276.16	36.33%	40.07%	10.14
2	12	-11.51%	268.31	34.73%	30.87%	11.49	-11.05%	272.43	33.93%	36.00%	11.17	-10.62%	254.32	36.87%	32.73%	10.40
	15	-10.34%	271.06	36.07%	36.93%	9.91	-11.05%	242.68	34.67%	34.40%	10.77	-10.93%	243.84	34.93%	37.33%	9.93
	20	-10.33%	307.62	37.53%	33.87%	10.04	-10.02%	338.05	36.13%	36.73%	9.69	-10.49%	319.84	37.93%	31.87%	10.65
	50	-8.71%	611.50	39.67%	43.27%	7.80	-8.71%	593.61	41.47%	41.80%	7.75	-8.48%	711.56	44.67%	40.67%	7.47
	100	-9.34%	531.89	32.27%	34.00%	11.51	-11.86%	499.94	33.40%	35.33%	14.41	-8.49%	519.38	34.93%	37.40%	10.12
Average		-11.59%	345.09	33.78%	33.58%	11.87	-11.69%	343.23	33.95%	35.18%	11.79	-10.79%	358.84	35.73%	35.67%	10.56
SD		3.09%	135.77	5.58%	5.87%	4.28	2.54%	125.98	6.69%	3.08%	3.79	2.07%	159.00	5.39%	4.20%	2.32
	4	-22.77%	228.64	22.47%	27.80%	24.13	-21.69%	217.21	20.33%	27.53%	27.11	-17.96%	232.63	22.73%	37.33%	21.41
	6	-13.31%	336.31	34.47%	34.20%	15.44	-12.49%	339.91	35.13%	36.07%	14.61	-12.20%	351.47	31.67%	41.00%	14.69
	8	-16.83%	299.09	35.07%	38.87%	14.97	-16.91%	300.02	32.53%	40.80%	15.11	-17.16%	299.81	34.87%	42.27%	15.21
	10	-15.63%	268.61	33.33%	39.20%	13.91	-15.18%	276.83	36.00%	39.67%	13.03	-15.38%	281.80	37.13%	38.13%	13.45
3	12	-14.07%	245.11	39.67%	29.60%	14.31	-13.61%	257.27	38.87%	32.87%	13.64	-14.73%	249.83	35.53%	32.27%	15.19
	15	-13.53%	282.05	36.80%	39.67%	12.65	-13.51%	268.92	40.20%	37.60%	12.40	-13.81%	248.11	35.07%	38.73%	13.41
	20	-12.88%	343.16	37.87%	42.07%	12.11	-12.74%	336.69	41.93%	37.47%	11.99	-12.61%	356.04	38.80%	42.53%	11.57
	50	-11.40%	629.51	44.07%	44.53%	9.95	-11.62%	701.89	41.67%	44.73%	10.35	-11.08%	694.06	45.67%	45.47%	9.49
	100	-10.80%	768.87	41.67%	43.20%	11.34	-17.78%	607.14	29.07%	28.40%	21.06	-13.10%	740.84	38.80%	37.93%	14.43
Average		-14.58%	377.93	36.16%	37.68%	14.31	-15.06%	367.32	35.08%	36.13%	15.48	-14.23%	383.84	35.59%	39.52%	14.32
SD		3.60%	189.20	6.21%	5.91%	4.09	3.22%	168.89	7.00%	5.66%	5.30	2.30%	194.37	6.18%	3.82%	3.25

Table 4.10: Average results obtained for PA-II considering Case II.

Scenario	n	Variation 1					Variation 2					Variation 3				
		Dev	$E(\sum_{j\in J} W_jT_j)$	P_1	P_2	$E(Idle)$	Dev	$E(\sum_{j\in J} W_jT_j)$	P_1	P_2	$E(Idle)$	Dev	$E(\sum_{j\in J} W_jT_j)$	P_1	P_2	$E(Idle)$
1	4	**-1.28%**	282.11	13.20%	8.93%	31.10	-1.34%	262.74	12.40%	12.33%	31.26	-1.24%	264.63	15.20%	10.87%	30.88
	6	**-1.43%**	283.23	7.47%	7.60%	39.16	-0.99%	285.71	8.60%	8.73%	38.39	-1.10%	284.79	7.80%	7.80%	38.57
	8	**-1.14%**	225.81	11.80%	9.07%	27.64	-1.13%	221.48	12.47%	9.80%	27.59	-1.14%	221.03	12.40%	9.67%	27.62
	10	**-0.75%**	249.47	11.47%	10.27%	29.56	-0.81%	241.87	9.20%	8.87%	29.82	-0.83%	241.95	8.87%	8.60%	29.85
	12	**-0.06%**	161.93	6.67%	5.27%	31.27	-0.09%	158.53	7.40%	3.67%	31.33	-0.09%	161.21	7.40%	3.80%	31.35
	15	**-0.53%**	200.35	4.73%	4.00%	31.92	-0.62%	202.23	4.13%	5.67%	32.65	-0.55%	208.08	6.13%	7.60%	32.35
	20	**-0.45%**	200.79	1.13%	5.73%	32.30	-0.36%	203.18	1.27%	7.40%	32.16	-0.41%	202.60	5.47%	3.07%	31.97
	50	**0.00%**	203.83	0.00%	0.00%	35.05	0.00%	208.82	0.07%	0.00%	35.01	0.00%	205.93	0.00%	0.00%	35.09
	100	**0.00%**	67.66	0.00%	0.00%	56.10	0.00%	67.77	0.00%	0.00%	55.95	0.00%	68.07	0.00%	0.00%	55.99
Average		**-0.63%**	208.35	6.27%	5.65%	34.90	-0.59%	205.81	6.17%	6.27%	34.91	-0.60%	206.48	7.03%	5.71%	34.85
SD		0.56%	66.27	5.18%	3.78%	8.60	0.51%	63.73	4.98%	4.32%	8.46	0.50%	63.45	5.02%	4.10%	8.52
2	4	-3.27%	274.73	23.40%	21.33%	29.90	-3.11%	284.17	22.53%	21.73%	29.96	-3.37%	278.93	23.00%	21.27%	29.97
	6	-2.37%	315.02	12.47%	11.73%	35.46	-2.23%	314.11	13.93%	11.27%	35.29	-2.39%	314.69	12.53%	11.67%	35.49
	8	-2.38%	271.14	20.33%	17.80%	25.13	-2.47%	270.63	19.80%	18.53%	25.13	-2.42%	268.72	20.47%	18.93%	25.00
	10	-2.07%	283.47	18.53%	20.33%	27.07	-1.99%	278.76	18.87%	22.00%	26.88	-1.96%	280.74	20.00%	21.07%	26.75
	12	-1.21%	206.67	20.40%	9.20%	27.91	-1.17%	207.00	18.93%	10.53%	27.87	-1.12%	205.05	17.93%	12.87%	27.68
	15	-1.14%	256.21	12.40%	15.33%	27.27	-1.35%	244.35	11.60%	12.13%	28.12	-1.31%	254.13	12.00%	12.67%	27.99
	20	-1.01%	292.44	13.67%	13.93%	27.04	-1.07%	291.07	12.93%	14.07%	27.23	-1.06%	291.93	14.47%	13.13%	27.16
	50	0.00%	401.71	5.73%	9.33%	28.16	0.00%	402.07	7.73%	7.87%	28.00	0.00%	400.20	6.40%	8.93%	28.09
	100	0.00%	219.53	0.07%	0.07%	48.86	0.00%	219.97	0.07%	0.07%	48.84	0.00%	219.40	0.07%	0.07%	48.87
Average		-1.49%	280.10	14.11%	13.23%	30.76	-1.49%	279.13	14.04%	13.13%	30.81	-1.51%	279.31	14.10%	13.40%	30.78
SD		1.11%	56.91	7.56%	6.60%	7.39	1.07%	57.54	7.02%	7.00%	7.33	1.13%	56.87	7.35%	6.64%	7.39
3	4	-4.78%	305.55	28.80%	23.87%	30.41	-4.71%	308.01	28.40%	23.93%	30.28	-4.66%	308.28	25.40%	27.73%	30.26
	6	-4.01%	337.27	16.40%	15.67%	35.21	-3.99%	334.96	13.80%	17.60%	35.29	-3.09%	343.45	19.47%	17.93%	33.43
	8	-4.86%	294.98	24.13%	18.07%	26.77	-5.24%	290.55	23.20%	16.60%	27.52	-4.86%	297.49	24.80%	16.80%	27.05
	10	-3.85%	289.93	22.73%	22.20%	27.27	-3.72%	300.73	23.40%	24.07%	26.79	-4.08%	286.37	17.67%	24.93%	27.93
	12	-2.17%	240.59	23.13%	19.20%	26.23	-2.30%	233.59	21.93%	19.40%	26.42	-2.23%	240.57	21.27%	20.60%	26.41
	15	-2.16%	282.33	17.40%	21.73%	26.31	-2.31%	277.54	17.27%	20.60%	26.59	-2.15%	284.24	17.40%	21.53%	26.27
	20	-1.40%	357.13	20.07%	24.87%	24.70	-1.69%	354.68	18.53%	25.07%	24.95	-1.54%	355.59	19.47%	25.40%	24.73
	50	0.00%	597.92	19.53%	19.00%	24.58	0.00%	598.85	17.73%	20.53%	24.70	0.00%	596.71	20.80%	17.60%	24.64
	100	0.00%	436.18	2.07%	2.40%	44.56	0.00%	433.58	2.13%	2.27%	44.63	0.00%	436.27	2.40%	2.20%	44.56
Average		-2.58%	349.10	19.36%	18.56%	29.56	-2.66%	348.05	18.49%	18.90%	29.69	-2.51%	349.89	18.74%	19.41%	29.48
SD		1.90%	108.55	7.50%	6.72%	6.53	1.91%	109.35	7.48%	6.89%	6.48	1.83%	107.95	6.73%	7.50%	6.30

Chapter 5

Machine learning integrated with planning problems

In this chapter, we present two proactive approaches to deal with machine failures. In this case, in both strategies, we aim to anticipate the impacts of disruptions using machine learning algorithms for failures prediction. Based on the results, we modify the production planning problems. Here, our goal is to improve the methods presented in Chapter 4, which resulted in high expected total weighted tardiness. To increase clarity, Figure 5.1 shows the structure of this chapter.

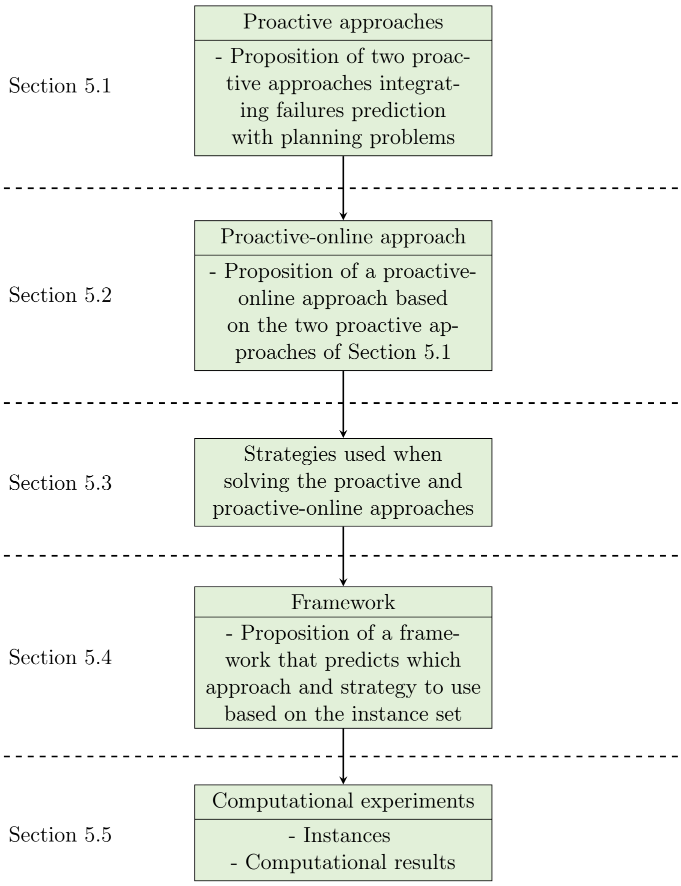


Figure 5.1: Flowchart representing the structure of this chapter.

5.1 Proactive Approaches

In this section, we propose two proactive approaches for dealing with machine failures. The approaches integrate maintenance with planning problems, aiming to anticipate disruptions using machine learning algorithms for failures prediction. Based on the predicted values, we modify the production planning problems. We emphasize that due to random events, not all uncertainties can be predicted in advance. We assume that the forecasting models predict failures based on the deterioration of the machines, which increases the probabilities of failure. After the repair, they return to their initial state.

Figure 5.2 shows a general flowchart representing the proposed approaches. First, based on historical data, we perform a time series forecasting for the features of the next period. The predicted information is input to a classification algorithm, which classifies whether each hour of the next period has a high probability of failure. Based on the number of failures defined by the classification algorithm, we modify the short-term production planning problem, which outputs information about the production plan, consisting of entries for the scheduling problem. The scheduling is then solved considering the predicted failures; its result is compared to a simulated failure environment. In the simulated environment, we generate the features for the following periods based on the distributions used to generate the historical data. This simulation is used for comparison between the predictions and the simulated scenario. Information can be returned to the time series forecasting stage, as will be explained in Section 5.3.

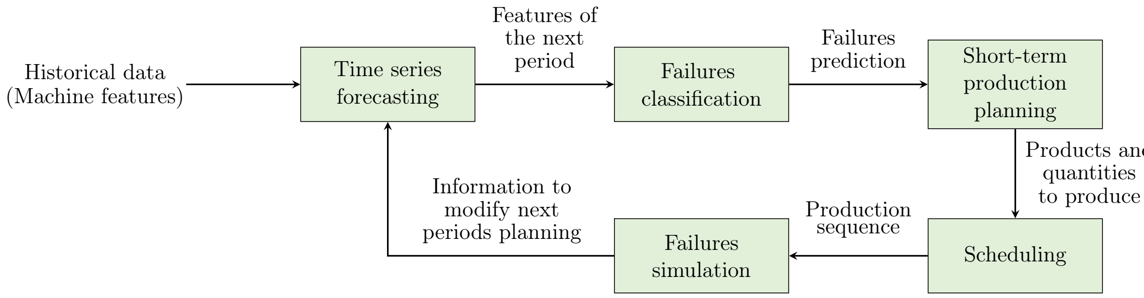


Figure 5.2: Flowchart representing the proactive approaches integrating failures prediction with planning problems.

5.1.1 Failures prediction

This section presents the machine learning algorithms used for failures prediction. Section 5.1.1.1 refers to the time series forecasting, whereas Section 5.1.1.2 explains the classification method used in this work.

5.1.1.1 Time Series Forecasting

We predict the failures for the next planning period, consisting of a week with 112 production hours. For failures prediction, we initially have to predict the behavior of the five features that influence machine operation, which are pressure, speed, temperature, sound, and vibration. In real cases, several other variables may affect the machine; considering the complexity of a real problem, we simulated a simplified version. We used a database with 10752 previous hours (96 periods) to forecast the time series for the next period. To perform the forecast, we use deep learning, specifically the LSTM. For information on the LSTM, see the works of Hochreiter and Schmidhuber (1997) and Yu et al. (2019). We test other traditional forecasting methods, but they were unable to deal with a multivariate and multistep scenario such that the behavior of the variables depends on the values predicted for the remaining variables. Other deep learning algorithms were also tested for training like Multilayer Perceptrons (MLPs), Convolutional Neural Networks (CNNs), and CNN-LSTM. However, we obtained the best results considering the LSTM.

Table 5.1 presents an example of the historical data of machine features. As we can observe, at certain intervals, the values of the features return to the initial values they had at the beginning of the first planning period, indicating that a failure occurred in the previous hour and the machine had to be restarted after maintenance. In this context, since we have to predict the behavior of each feature, which varies according to different distributions, time series forecasting is hard to be performed. Furthermore, the predicted values for certain features influence the value of the other features, as it might indicate a failure. Another difficulty in forecasting is to predict several hours ahead. Figure 5.3 shows the behavior of the historical data of the five features, while Figure 5.4 focuses on a fraction of the data to make it possible to visualize the behavior of the data over time. We emphasize that we consider the five features for forecasting since they did not present high correlations between them.

Table 5.1: Example of historical data - Machine 1.

Hour	Pressure	Speed	Temperature	Sound	Vibration
1	100.00	100.00	80.00	60.00	120.00
2	100.19	98.89	80.01	60.07	120.40
3	100.55	96.62	80.02	61.33	121.00
4	100.68	96.46	80.19	63.28	121.60
5	102.03	95.76	80.38	65.14	123.29
6	102.57	94.51	80.69	65.18	124.04
7	102.61	92.81	80.97	65.81	124.36
8	103.02	92.15	81.10	67.38	125.10
9	103.19	91.93	81.18	76.39	126.16
10	103.56	90.06	81.45	77.11	129.57
11	103.60	88.87	81.51	78.44	129.96
12	100.00	100.00	80.00	60.00	120.00
13	100.63	99.53	80.52	61.31	121.28
14	100.74	99.29	80.83	61.36	121.40
15	100.83	99.04	81.02	62.52	123.45
16	101.68	97.77	81.10	62.82	124.03
17	101.79	97.23	81.21	63.09	129.65
18	102.02	96.03	81.56	64.65	131.48
19	102.30	95.11	81.62	66.14	132.87
20	102.46	93.80	81.67	68.46	133.45

In this work, we have to predict multiple features in parallel for multiple hours in the future. After predicting the values of the features for an hour using the trained model, we input these predicted values into the data. Then, the features of the next hours will be predicted based on the historical data with the addition of the previous predictions. The prediction is performed recursively. When we add a new forecast to the dataset, we exclude the oldest data from the series and use this new set to make the forecasts. Using this strategy can lead to an increase in prediction error accumulated over time given that predictions are made based on the earlier hours predictions. However, in our work, this strategy resulted in better forecasts when compared to predicting all the next 112 hours at once.

We tested several configurations for the neural network architecture and transformations in the input data to improve the predictions. The best results are obtained considering the following characteristics: We use 80% of the data for training the neural network (corresponding to 8602 data) and 20% for testing (2150 data). The training set consists of the dataset used to develop models, estimating their several parameters, whereas the validation or test set is the dataset used to evaluate the performance of the final set of models (Kuhn and Johnson, 2013). According to Ayvaz and Alpay (2021), normalizing the features eliminates potential problems

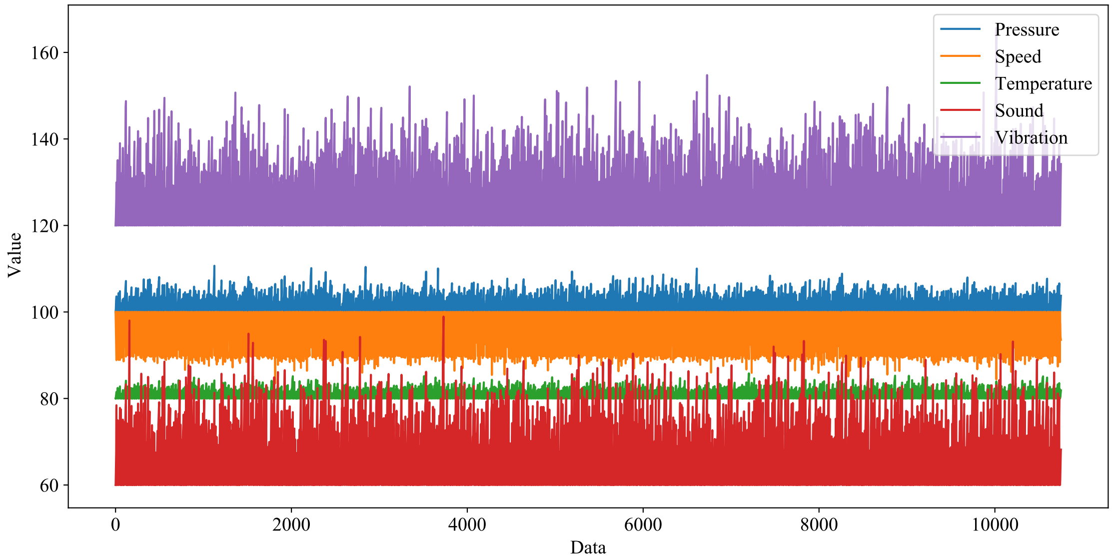


Figure 5.3: Behavior of the historical data - Machine 1.

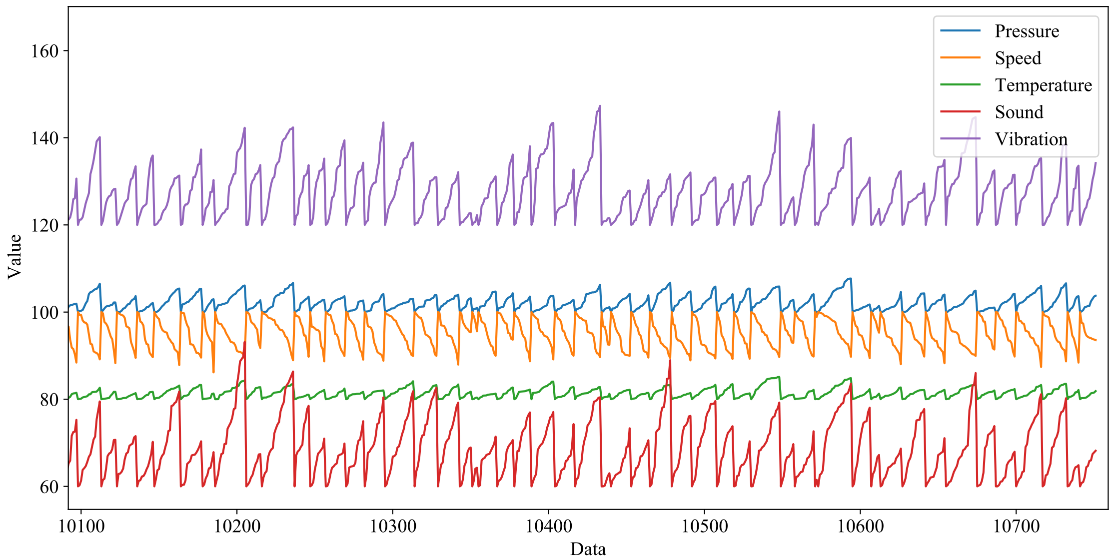


Figure 5.4: Behavior of a fraction of the historical data - Machine 1.

during data modeling, making it less sensitive to the scale of the data and improving the quality of the analysis. Therefore, we first converted the data into a scale (0,1) and transformed it into supervised data, using a lag size of 112 hours. In this work, we consider an environment with two parallel machines. Then, we trained the network for each machine individually with different configurations allowing a better generalization for each case.

Considering the architecture of the model for machine 1, we use 4 LSTM layers with 32 units and an L1 regularizer equal to 0.00001. Also, we added a last dense layer with 5 units. In the first 2 LSTM layers, the Leaky ReLU (Rectified Linear Unit) activation function is used with $\alpha = 0.5$. Considering machine 2, we use 4 LSTM layers with 64 units and an L1 regularizer equal to 0.00001, with a last dense layer with 5 units. In this case, we use the default activation function for all layers. Although the machines are identical, we train models to identify the specific behavior of failures that occurred in the past for each one. Therefore, the model parameters for each machine are different. For both machines, we chose the Root Mean Square Propagation (RMSprop) optimizer with a learning rate equal to 0.001, considering the Mean Absolute Error (MAE) as the loss. For model fit, we ran 100 epochs, with 1500 steps per epoch, batch size equal to 1, and shuffle defined as False.

The chosen models obtained a loss equal to 0.1240 at the end of the 100 runs for the test set for machine 1, and equal to 0.1187 for machine 2. We also had to make qualitative analyses to find the models that best generalize our data. In many cases, the models encountered small losses for the training and test data. Nonetheless, the resulting predictions were not applicable in practice, e.g., predicting the same value for the 112 future hours. For this reason, we chose to present the parameters and configurations only for the models that best generalize for each machine. Furthermore, the main objective of our work is to use the predictions to modify production planning problems, creating solutions robust to possible failures, and not to make a comparative analysis of different machine learning algorithms. We guide the interested reader to the work of Kuhn and Johnson (2013) for details on performance measures, regression and classification methods, data preprocessing, among other concepts of predictive modeling.

5.1.1.2 Classification of the failures

Based on the predictions of the features for the next 112 hours, we perform the classification. The model predicts for each of these hours if there will be a failure. We use the 10752 historical data, dividing them in the same way presented in the previous section to avoid biased solutions, 8602 data for training, and the remaining for model validation. For classification, we also use a neural network. In this case, the network has 2 dense layers, with 256 and 1 unit, respectively. We train the model considering different units for the first layer, resulting in better predictions using 256 units. The output of the neural network consists of an array with predictions for each hour of the period being currently scheduled. In the first layer, we use the ReLU activation function, while in the second, we use the sigmoid function since, in this case, we would like the output of our model to be in the range between 0 and 1, indicating or not failures for each hour. We use the RMSprop optimizer and the binary crossentropy loss function, given that our problem consists of a binary classification task. For model fit, we set 40 epochs and batch size equal to 10. The same architecture was used for both machines, resulting in a validation loss of 0.0612 for machine 1 and 0.0753 for machine 2. Since the data considered is imbalanced, we use the AUC as a metric, finding a validation AUC equal to 0.9558 and 0.8937 for machines 1 and 2, respectively.

We also performed computational tests using other models for classification. We tested the Logistic Regression, Support Vector Machine (SVM), Bagging (Bootstrap Aggregating), Random Forest, and Gradient Boosting Algorithm (GBM) methods. However, these models showed poor results for the validation data. Similar results were obtained when using resampling techniques, such as oversampling and downsampling. Cost-sensitive models considering the Logistic Regression and the SVM were also tested, but the same behavior was observed. We emphasize that we analyze imbalanced data. Machine 1 has only 7.57% of data consisting of positive cases. Machine 2, on the other hand, has 5.17% of positive events, making classification even more difficult. For more information on imbalanced datasets, see Fernández et al. (2018).

5.1.2 Proactive approaches integrating failures prediction with planning problems

We propose two proactive approaches to integrate production planning problems with the predictions obtained through machine learning models. Section 5.1.2.1 presents the Proactive approach considering Machine Learning predictions to modify the Instance capacity parameter (PMLI), while Section 5.1.2.2 introduces the Proactive approach considering Machine Learning predictions to modify the Algorithm (PMLA).

5.1.2.1 Proactive approach considering Machine Learning predictions to modify the Instance capacity parameter

In this approach, based on the output information of the learning algorithm, we modify the instance of the planning problem with information on the occurrence of failures. Initially, the production capacity for each machine m is set to $C_{mt'} = 112$ hours. With information about the predicted failures, we modify this parameter to $C^*_{mt'} = C_{mt'} - \alpha_{mt'}$, with $\alpha_{mt'}$ indicating the total duration of predicted failures on machine m in period t'. Then, we solve the short-term production planning problem considering the replacement of the parameter $C_{mt'}$ by $C^*_{mt'}$. The short-term production planning problem is solved with the mathematical model (3.1)-(3.14), while the scheduling is solved using the proposed heuristic presented in Section 3.4. Given the result found, we evaluate the failures considering the predictions and a simulated environment. Figure 5.5 presents a flowchart for the PMLI approach.

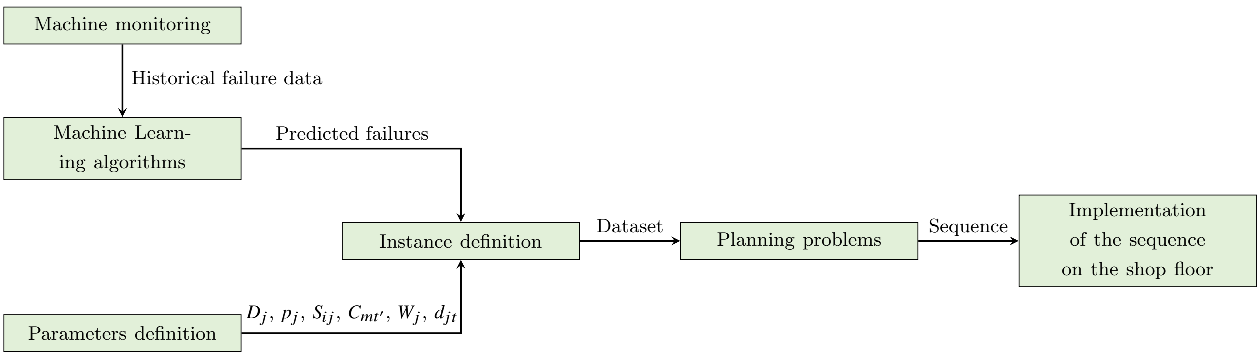
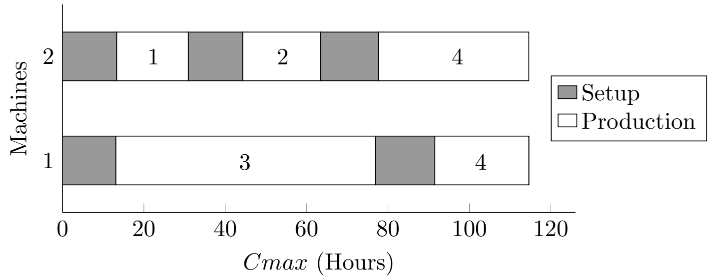


Figure 5.5: Flowchart representing the PMLI. In this case, we modify the instance of the problem based on the predictions through machine learning techniques.

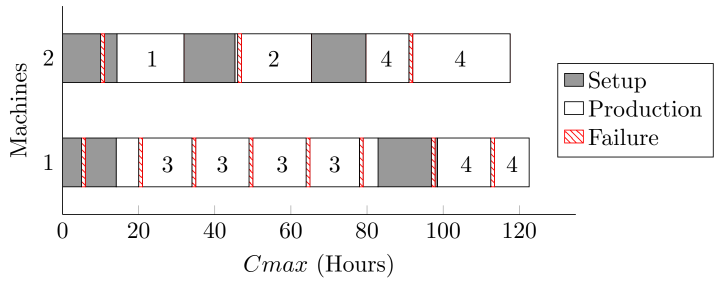
5.1.2.1.1 Example of the PMLI When solving an instance with 4 products for the first period, the short-term production planning and scheduling problems have as output the solution presented in Figure 5.6(a). In this case, the machine learning algorithms (Sections 5.1.1.1 and 5.1.1.2) predicted 8 and 3 failures for machines 1 and 2, respectively. Therefore, the production capacities used in the model (3.1)-(3.14) are equal to 104.00 hours for machine 1 and 109.00 hours for machine 2, considering an initial capacity of 112.00 hours. The makespans of both sequences are equal to 114.60 hours after the resolution of scheduling. Furthermore, a value of 271.40 units is found for the total weighted tardiness.

Figure 5.6(b) shows the solution considering the predictions made by the machine learning algorithms, while Figure 5.6(c) exhibits the impact on the initial solution when considering the real failures (reference). We observe that the machine learning model used for failures prediction of machine 2 does not make predictions as accurately as the model used for machine 1, predicting fewer failures than occur on the shop floor, which is related to its imbalanced dataset. In this case, the predictions made by the machine learning models were {5, 20, 34, 49, 64, 78, 93, 108} for machine 1 and {10, 46, 90} for machine 2. On the other hand, the actual failures occur at hours {7, 28, 37, 55, 66, 83, 97} for machine 1 and {7, 31, 55, 74, 92, 101} for machine 2.

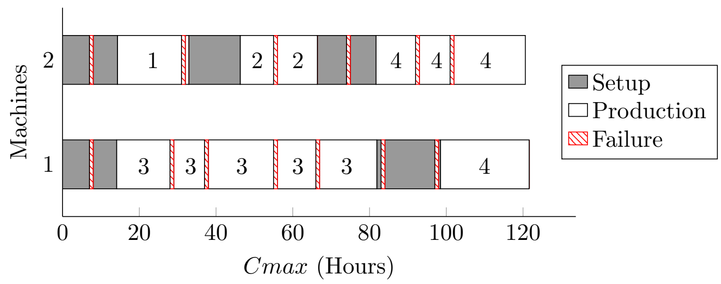
As we can see in Figure 5.6(a), the makespan for both machines is greater than the modified production capacity even before the occurrence of failures. When solving the short-term production planning model, we maximize the production quantities while respecting the capacity



(a) Solution before the failures.



(b) Solution considering the predictions of the failures.



(c) Solution considering the reference.

Figure 5.6: Example considering an instance with four products solved with the PMLI.

of each machine. However, as we hierarchically solve the short-term production planning and scheduling problems, the scheduling heuristic may reorder the products to minimize the total weighted tardiness, which may cause infeasibility regarding the production capacity. In these cases, we let the decision-maker decide how to deal with the infeasibility. They may employ the methods proposed in the article of Alves et al. (2021), may not deliver the quantities defined to be produced after the violation of production capacity, or use another production strategy to manufacture these quantities.

When failures occur, we notice in Figures 5.6(b) and 5.6(c) that capacity is violated in large quantities. We also observe that the completion times for the products produced in machine 1 are similar for both solutions (CT_3 = 82.90 and CT_4 = 122.6 considering the prediction and CT_3 = 81.90 and CT_4 = 121.6 considering the reference). However, the completion times for machine 2 have greater differences (CT_1 = 31.90, CT_2 = 65.40, and CT_4 = 117.60 for the prediction and CT_1 = 32.90, CT_2 = 66.40, and CT_4 = 120.60 for the reference). We hypothesize that this difference is due to the imbalanced class rate of machine 2, which has only 5.17% of positive events. Given the characteristics of the studied scenario, lower imbalanced rates make it even hard to predict the failures. We observe a better behavior when considering machine 1, which has a slightly higher rate. When we calculate the total weighted tardiness, we have a value equal to 338.40 and 334.40 for the prediction and the reference, respectively.

5.1.2.2 Proactive approach considering Machine Learning predictions to modify the Algorithm

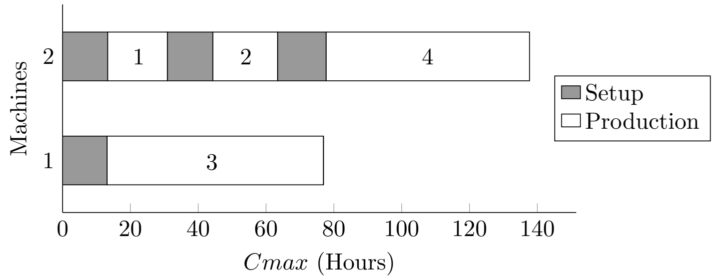
In the PMLA, we use the information returned by the machine learning method to alter the strategy of the planning algorithm along with the modification of the instance proposed in Section 5.1.2.1. The approach is based on simheuristics, where a simulation generates information that is passed to the heuristic and vice-versa. For more information about simheuristics, see the works of Juan et al. (2014a) and Chica et al. (2017). In the PMLA, we adapt the proposed scheduling heuristic to consider disruptions. Each deterministic solution found by the procedure is evaluated through a failure simulation based on the predictions made by the machine learning models. Thus, only the best stochastic solution is passed on to the decision-maker and implemented on the shop floor. Considering Figure 5.5, the only modification would be in the "Planning problems" step, in which the scheduling algorithm is altered according to the predicted failures.

5.1.2.2.1 Example of the PMLA In this section, we show an example of the PMLA for the instance presented in Section 5.1.2.1.1. When solving the short-term production planning with the PMLA, we find the same solution obtained when solving the problem with the PMLI, since the parameters used and the failure predictions are the same. Therefore, as in PMLI, products 1, 2, 3, and 4 are defined for production and passed to the scheduling heuristic along with their respective quantities. At the scheduling level, the first solution obtained by the heuristic is shown in Figure 5.7(a). This solution refers to steps 1 to 5 of the heuristic presented in Section 3.4. The makespan is equal to 76.90 hours for machine 1 and 137.70 hours for machine 2. The next steps of the heuristic will balance these makespan values.

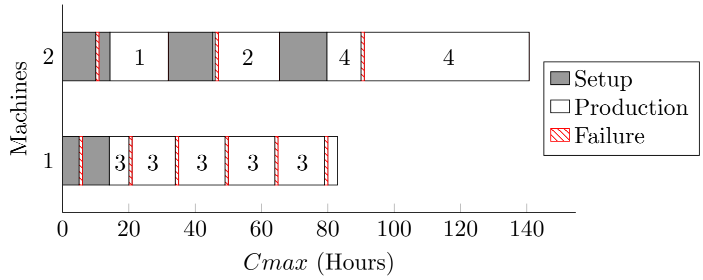
Based on the solution found in Figure 5.7(a), we perform a failure simulation considering the moments of disruptions predicted by the machine learning algorithms. The solution obtained is shown in Figure 5.7(b). The makespan values are equal to 82.90 and 140.70 for machines 1 and 2, respectively, resulting in total weighted tardiness of 410.80 units. Based on this solution, we perform Step 6 of the scheduling heuristic (see Section 3.4). The resulting sequences are exhibited in Figure 5.7(c). With the job splitting, the makespans of both machines are equal to 120.10 hours, resulting in a value of 328.40 units for the total weighted tardiness. After the job splitting, the scheduling heuristic executes the local searches, simulating each deterministic solution based on the predicted failures. For this specific instance, the local searches did not improve the solution presented in Figure 5.7(c).

We highlight that the products may change their initial positions after the failures simulation, as presented in Figure 5.8, consisting of the main difference between the PMLA regarding the PMLI, which did not consider the failures for alteration of the production sequence. In Figure 5.8(a), we have the first solution generated when simulating the failures for an instance with six products. In this case, the makespan of machine 1 is equal to 109.10 hours, while the makespan for machine 2 equals 114.30 hours. We found a value of 247.99 units for the total weighted tardiness for this solution. On the other hand, Figure 5.8(b) presents the resulting solution found by the PMLA. As we can observe, the products of the initial solution change their position based on the failures simulation, resulting in a solution with total weighted tardiness equal to 238.19 units and makespan values of 111.60 and 111.20 hours for machines 1 and 2, respectively.

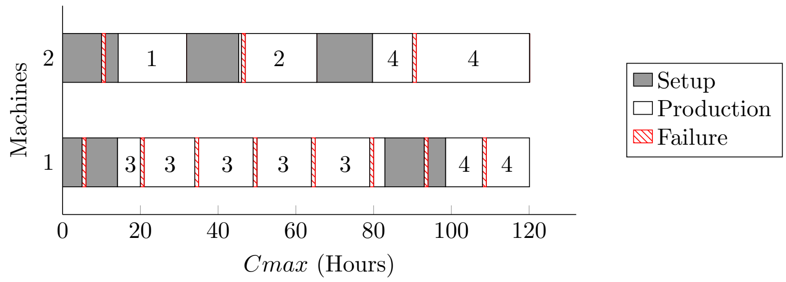
When solving the same instance with six products using the PMLI, we initially obtain the same solution presented in Figure 5.8(a), with products 3 and 5 allocated on machine 1 and products 6 and 2 produced on machine 2. However, PMLI does not consider the failures simulation for each deterministic solution found. Then, after performing the job splitting and the local searches, the output of PMLI is shown in Figure 5.9. The resulting solution has a makespan of 118.65 hours for machine 1 and 113.65 hours for machine 2, presenting total weighted tardiness of 261.05 units. Therefore, for this specific instance, the PMLA outperforms the PMLI, given that



(a) Solution before the failures.



(b) Solution considering the predictions of the failures.



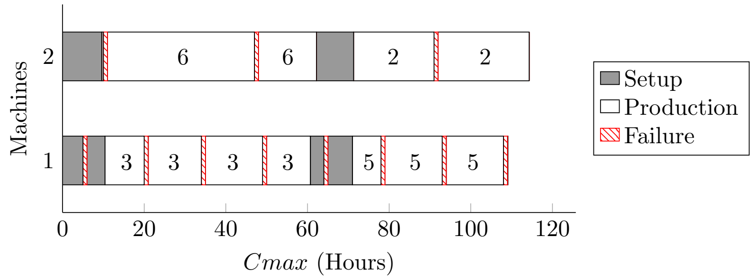
(c) Solution considering the job splitting.

Figure 5.7: Example considering an instance with four products solved with the PMLA.

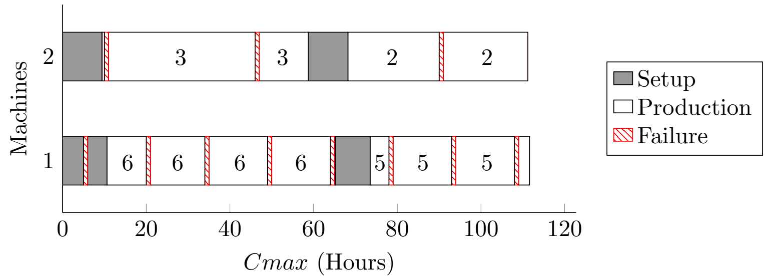
PMLA results in a lower total weighted tardiness and lower makespan values when considering the first planning period.

5.2 Proactive-online approach

In addition to the approaches presented in the previous section, another proposition of this thesis consists of a proactive-online approach. The proactive-online approach combines proactive and real-time decisions; we solve the problem as shown in Section 5.1.2 along with rescheduling actions based on real-time information. In this case, we first consider the short-term production planning modifying the capacity parameter according to the predictions made by the machine learning algorithms. Based on the results found, we solve the scheduling problem minimizing the total weighted tardiness. The scheduling resolution depends on the approach; if we are solving the problem with PMLA, we use the predictions to alter the scheduling solutions as explained in Section 5.1.2.2. Otherwise, if we use the PMLI to solve the problem, we modify only the production capacity at the short-term production problem. The resulting sequences are implemented in the factory. The proactive-online algorithm runs through the sequences until finding a failure. From there, it runs a rescheduling for the remaining products in the solution. The algorithm is free for changing the position of the products after the failure and can even



(a) First solution after the failures simulation.



(b) Resulting solution.

Figure 5.8: Example considering an instance with six products solved with the PMLA.

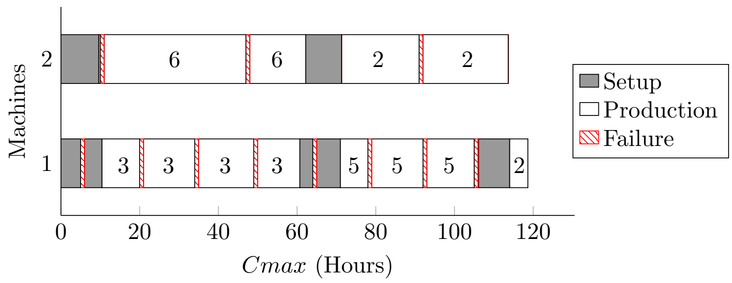


Figure 5.9: Example considering an instance with six products solved with the PMLI.

switch products between the machines if this action results in a lower total weighted tardiness. The goal of the proactive-online approach is to use the predictions to alter the planning problems and at the same time respond to real-time information about failures.

In this approach, for each hour of the period, we compare the predicted values for the parameters pressure, speed, temperature, sound, and vibration with its real values. If the predicted value is greater or lower than a certain margin of error compared to the actual value, we return the real information to the dataset used as input to the machine learning models. Based on the new dataset, the learning algorithms make the predictions again for the remaining hours, using this information in the scheduling heuristic when solving the problem with the PMLA. We propose this online strategy to reduce the forecast error for long future periods. Therefore, the algorithm adjusts the predictions as the data changes. Figure 5.10 exhibits the proactive-online approach. In the flowchart, the index h indicates the hours of the period being currently scheduled. The algorithm represented in Figure 5.10 runs until it reaches the maximum production capacity. Unlike Section 5.1.2, in the proactive-online approach, we do not compare the results obtained by the proposed method with the simulation since we already consider the simulated failure periods to reproduce a real environment. We use this data along with the proactive-online procedure to represent the real-time information.

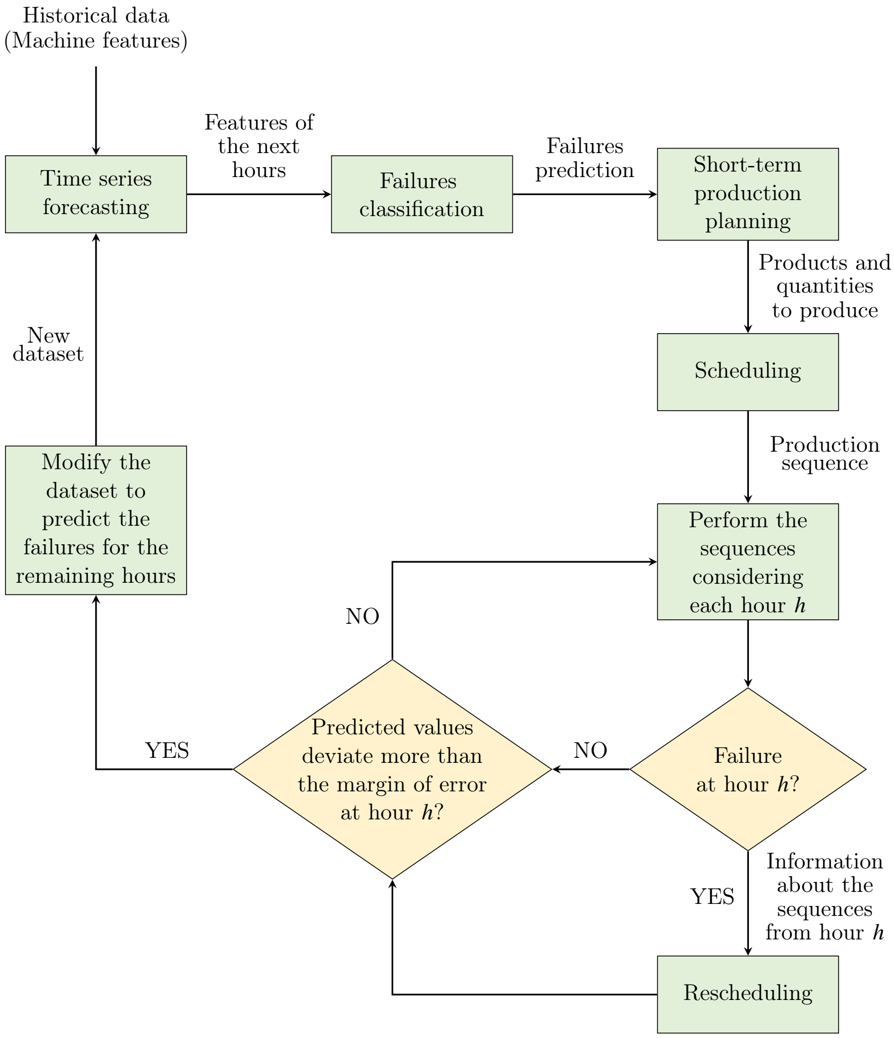
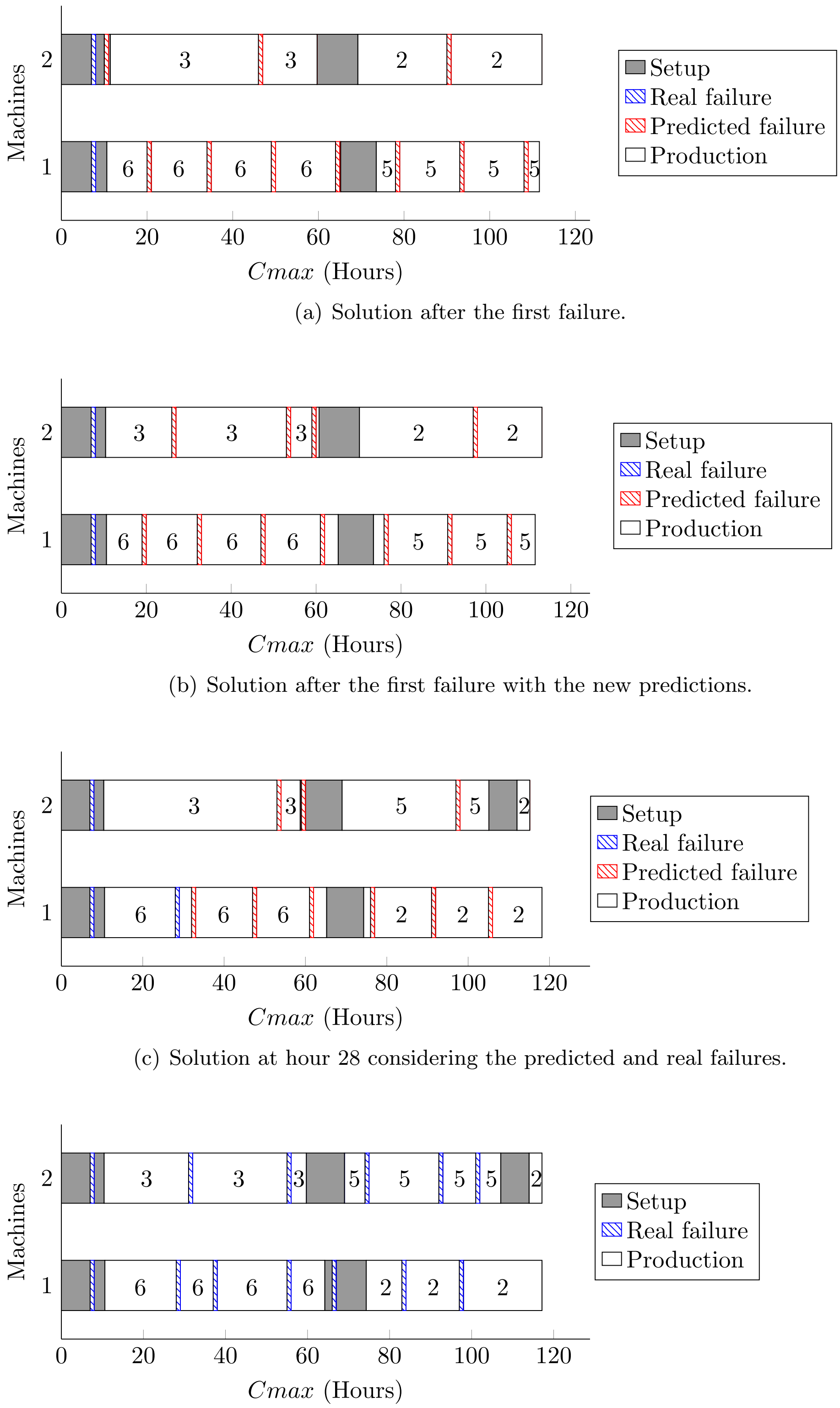


Figure 5.10: Flowchart representing the proactive-online approach. The algorithm runs until it reaches the production capacity of the period. When a failure occurs in hour h, the information passed to the rescheduling consists of the remaining production quantities, the remaining production sequences, and the products processed immediately before the failure in both machines. If the failure occurs during setup, information on the remaining setup time and the next product of the previous sequence is also passed to the rescheduling. Further, the parameters D_j, p_j, S_{ij}, $C_{mt'}$, W_j, and d_{jt} are also inputs to rescheduling.

The main advantage of using a proactive-online approach is related to online and real-time information from the shop floor. In this case, the predictions may be modified based on real information on the parameters pressure, temperature, speed, sound, and vibration. Therefore, we predict the failures more accurately, adjusting the predictions according to new data patterns. Furthermore, as we perform rescheduling after each failure, we may obtain lower total weighted tardiness than the proactive approaches.

When solving the same instance presented in Figure 5.8 with the proactive-online approach integrated with the PMLA, the procedure initially obtains the solution shown in Figure 5.8(b) after the resolution of the scheduling heuristic considering the predicted failures. Then, the sequences are implemented in the factory. For each hour, the method checks if a failure occurs. If this condition is true, a rescheduling is performed for the remaining sequences, allowing to alter the allocation of the products after the failure occurrence. The first failure of this example happens in hour 7 for both machines. The rescheduling outputs the solution presented in Figure 5.11(a). In this case, Figure 5.11(a) exhibits the real failure occurring in hour 7 in blue and the predicted disruptions using the color red.



(a) Solution after the first failure.

(b) Solution after the first failure with the new predictions.

(c) Solution at hour 28 considering the predicted and real failures.

(d) Final solution considering the real failures.

Figure 5.11: Example considering an instance with six products solved with the proactive-online approach.

In addition to checking whether there was a failure every hour, the proactive-online approach also verifies if the predicted values for machine features exceed a certain margin of error, in this example defined as 35.00%. In the first 7 hours of the tested instance, the predicted values were within the margin of error compared to the actual values measured on the shop floor. However, at hour 8, these values deviated more than the margin of error for machine 2. The predicted values for this machine were 108.40, 94.53, 92.80, 85.26, and 176.15 for pressure, speed, temperature, sound, and vibration, respectively. On the other hand, the real values are equal to 100.00, 100.00, 80.00, 60.00, and 120.00, indicating that a failure occurred in the previous hour. Then, the value predicted for the feature sound is more than 35.00%(60.00) and the prediction for the vibration is greater than 35.00%(120.00). In this case, the method makes new predictions, resulting in failure predictions for machine 1 at hours {19, 32, 47, 61, 76, 91, 105} and for machine 2 at hours {26, 53, 59, 97}. The new production plan is shown in Figure 5.11(b).

The next failure occurs at hour 28 in machine 1. The resulting solution is presented in Figure 5.11(c). In this case, we can observe that the approach modified the solution when rescheduling, producing products 6 and 2 on machine 1 and products 3, 5, and 2 on machine 2. The following failures occur at hours 31 (Machine 2), 37 (Machine 1), and 55 (Machines 1 and 2). Nonetheless, the sequences found by the scheduling heuristic were the same as shown in Figure 5.11(c). In hour 56, the predictions deviate more than the margin of error. The new failures predictions are {66, 79, 93, 107} for machine 1 and {76, 104, 109} for machine 2.

New disruptions occur at hours 66 (Machine 1) and 74 (Machine 2), resulting in the same sequences presented in Figure 5.11(c). In hour 75, the algorithm makes new predictions due to the violation of the condition regarding the margin of error. The resulting predictions consist of failures occurring in hours {81, 93, 107} and {94} for machines 1 and 2, respectively. The following disruptions occur in hours 83 (Machine 1) and 92 (Machine 2), also resulting in the same sequences shown in Figure 5.11(c). The procedure makes new predictions in hour 93, resulting in failures {96, 105} for machine 1 and {110} for machine 2. In hour 97, we have another disruption in machine 1, and in hour 101, a failure happens in machine 2. The final output of the proactive-online approach consists of sequences with makespans of 117.15 hours for machines 1 and 2 and total weighted tardiness of 246.55 units for the first planning period, as exhibited in Figure 5.11(d).

5.3 Strategies used when solving the proposed approaches

This section presents the strategies used when solving the approaches shown in Sections 5.1 and 5.2 and introduces the corrective method. We test the PMLI and PMLA approaches (both proactive and proactive-online) considering three different modifications of the capacity parameter at the short-term production planning problem, named strategy Without Feedback, strategy with Feedback Type 1, and strategy with Feedback Type 2. Initially, we solve the studied problem for the first period with the PMLI and PMLA proactively or using the proactive-online approach. After implementing the sequence on the shop floor, we use real information on the failures to modify the next period planning. This modification is performed through the use of the three strategies presented in this section.

In the strategy Without Feedback, we modify the capacity parameter based on the predictions made for each planning period. We do not alter the predictions considering feedback information from the previous week. In the strategy with Feedback Type 1, information about real failures of the previous period is returned to the time series forecasting and used in the next planning period. The information consists of the deviation in the number of real failures compared to the predictions. We use this deviation to modify the number of failures considered in the next period, changing the available capacity. Therefore, the procedure adjusts the production capacity according to the deviations to capture possible changes in the data pattern. We emphasize that at the scheduling level we use the failure times provided by the machine learning techniques. We do not consider these deviations in the simulation.

In the strategy with Feedback Type 2, we alter the capacity parameter based on the FP and FN values, introduced in Chapter 2, which are obtained after the execution of the sequences in the previous period. We calculate the prediction error through the equation $Error = |(\frac{FP}{FN}) - 1|$, i.e., we consider the ratio between the incorrect predicted failures and the incorrect number of predicted normal cases. Based on the result, we have the confidence interval $(\alpha_{mt'} - FN(Error), \alpha_{mt'} + FN(Error))$ for the predictions. In this work, we consider a conservative approach, adjusting the capacity parameter based on the upper bound of this confidence interval.

For comparison purposes, we implemented a corrective method. In this method, we do not predict failures before the execution of the plan on the shop floor. We deal with disruptions when they occur; we stop the machine and repair it, continuing production following the original sequence defined at the beginning of the planning period. We implement this strategy since it is a common practice in industries. Therefore, we have a baseline for comparison with the approaches proposed in this work.

5.4 Framework

This section presents the framework proposed to dynamically predict the best approach for a given problem instance. The idea is to train a framework using the proactive, proactive-online, and corrective approaches, enabling the algorithm to select the method with the lowest total weighted tardiness for each instance set. Then, when receiving new instances, the framework predicts which approach provides production sequences robust to failures with the lowest total weighted tardiness. We highlight that we consider a specific scenario in this work, but the framework can be modified to other real situations, e.g., jobshop and flowshop scheduling or other types of uncertainties.

First, the framework receives as input the information on the instance, consisting of the number of products, average production time, average demand, and average setup time. The next step consists of a clustering algorithm to group the set of instances according to its characteristics. We calculate the optimal number of clusters using the Elbow method, see applications in the works of Lee et al. (2019) and Karamaziotis et al. (2020). Based on the result found, the framework runs the k-means algorithm (see the papers of Hartigan and Wong (1979) and Likas et al. (2003)), grouping the instances. Then, it randomly chooses five instances of each group for training. Only five instances are selected in order to avoid bias when training, given that we use information of the current period as input to the algorithm.

With the definition of the clusters, we run each proposed approach for the selected instances, saving the strategy that resulted in the lowest total weighted tardiness for each case. In total, we have 13 strategies, considering the PMLI and PMLA proactively with the three different feedbacks presented in Section 5.3, and the proactive-online approach executed for PMLI and PMLA for each of the three variations. Further, we run the corrective method. Then, we train a machine learning algorithm to classify which strategy enables the minimal total weighted tardiness when receiving information on new instances. In this case, we use the Random Forest model. The input for the Random Forest training consists of the initial characteristics of each instance with the addition of the best strategy. For training, we use oversampling to deal with imbalanced classes and to avoid overfitting since some strategies are chosen for a minority of the randomly selected instances.

When the framework receives new instances, it classifies which strategy will probably achieve the lowest total weighted tardiness. To improve classification, the framework initially runs a t-test to verify if there was a change in the data pattern. In the case of a p-value lower than the prespecified threshold (defined as 0.05), we conclude that the two samples (sample of the training and a sample considering the new instance) present different average values. Then, we retrain the framework by adding the new instance to the dataset. Otherwise, if the *p*-value is greater than the threshold, we assume that the new instance characteristics are represented by the initial dataset and proceed to the classification step. After classification, the new instance is solved using the predicted strategy and the resulting solution can be implemented on the shop floor. Figure 5.12 shows a flowchart exhibiting the proposed framework.

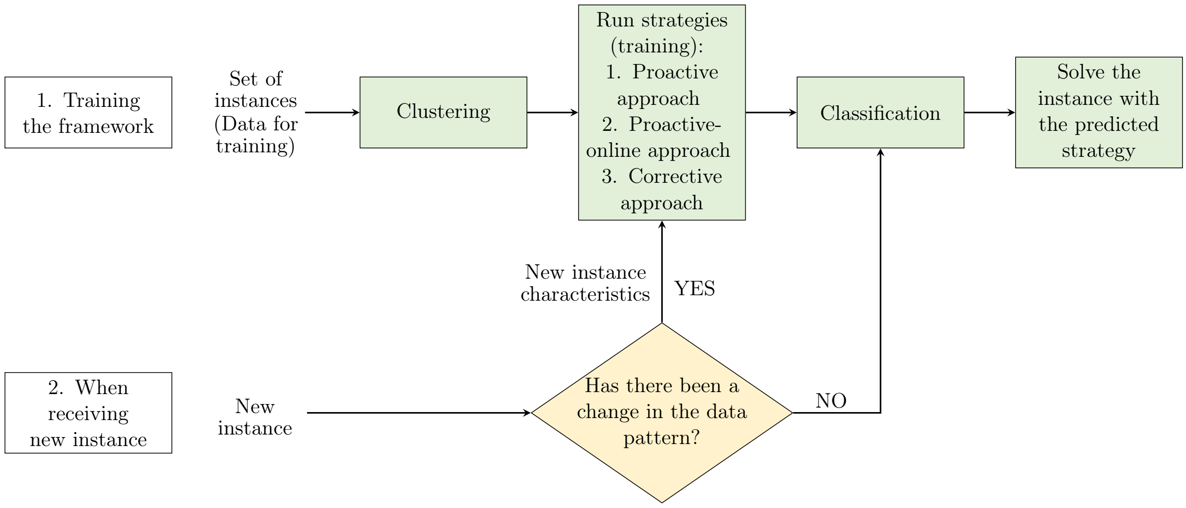


Figure 5.12: Flowchart representing the framework. In this case, we have two inputs for the framework, one when training it and another when receiving new instances to define the strategy to solve the problem.

In our work, we consider that the main objective of the scheduling heuristic is to minimize the total weighted tardiness in order to increase customer satisfaction. The secondary objective, minimizing the makespan, is an indirect strategy also to reduce tardiness. In this case, quantities of the final product allocated in a machine with violated capacity are transferred to the idle machine if it exists. Then, we perform job splitting on the two parallel machines, which reduces the completion time of this product and the makespan of both machines. We emphasize that this work aims to consider concepts of Industry 4.0 when planning the production. One of these concepts is the delivery of products on the exact due date defined by customers. Therefore, the minimization of the total weighted tardiness should be prioritized. For this reason, in the framework, we only consider this objective.

5.5 Computational Experiments

In this section, we discuss the results obtained with the proposed approaches and the framework. In Section 5.5.1, we show the distributions used to generate the failures and the remaining parameters, while in Section 5.5.2 we present the computational results. We run the tests in a computer with a Windows operating system with 8 GB of RAM and an Intel Core i5 processor. We use the AMPL Python API, with Python version 3.8 and CPLEX 20.1. The resolution time for the short-term production planning model was limited to 120 seconds.

5.5.1 Instances

Regarding the parameters of the production planning problems, we generate the data as shown in Table 4.6 for 4, 6, 8, 10, 12, 15, 20, 50, and 100 product quantities, generating 15 instances for each quantity considering an environment with two parallel machines ($|M| = 2$). We also test the approaches for two scenarios, named Case I and Case II. We examine the proposed approaches using both cases to verify if there is a trend in the use of the methods, e.g., a specific strategy may be better for a certain instance set, while another strategy works better for another set. We generate data for the features pressure, speed, temperature, sound, and vibration for each machine and each hour using the exponential distribution, as shown in Table 5.2. As mentioned in Section 5.1, the database consists of 10752 past information. Also, we generate data for the next future hours for comparison with the predictions, considering two periods for the planning horizon ($|T| = 2$) containing 112 hours each.

Table 5.2: Generation of features values.

Feature	Initial value	Machine 1		Machine 2	
		Increase rate	Decrease rate	Increase rate	Decrease rate
Pressure	100.00	3.00	-	3.00	-
Speed	100.00	-	1.20	-	4.00
Temperature	80.00	5.00	-	2.00	-
Sound	60.00	0.80	-	0.90	-
Vibration	120.00	0.80	-	0.40	-

We assume that failures occur according to the values of the features presented in Table 5.2. For data generation, we created rules that define the situations in which failures occur. For example, if the pressure is greater than 110, the speed is less than 95, and the temperature is greater than 125, we have a failure in machine 2. The other rules can be visualized in the files available on GitHub through the link `https://github.com/fernandafalves/PaperML`. Such rules are used to simulate a real factory environment. To the best of our knowledge, there are no instances in the literature that had the same characteristics as the generated instances. Usually, works in the literature considering failure data in production factories keep the data confidential. For this reason, we create our benchmark, which is available for future works through the GitHub link. In addition to the instances, we also make available all the codes used in this chapter.

The margin of error of the proactive-online strategy is defined as 0.35, indicating that if the difference between the actual value and the predicted value is greater than 0.35(Reference value), we return real information for the time series forecasting and classification. We check this for each hour of the period and each machine operation parameter.

5.5.2 Computational Results

Table 5.3 presents the average results for Case I and Case II. We calculate the averages based on the fifteen instances generated for each product quantity. Column 1 shows the strategy used to solve the problem, while Column 2 exhibits the number of products (n). Columns 3, 4, 5, and 6 indicate the results for Case I. Column 3 shows the average objective function of the short-term production planning problem, represented by f. In Columns 4 and 5, we present the average total weighted tardiness (f_S) for the prediction and reference, respectively. The predictions consist of the failures times predicted by the machine learning models used to alter planning problems. Otherwise, the reference is a simulated environment, considering that all failures are predicted correctly, also using this information to modify the planning problems. Lastly, Column 6 presents the average computational times. Columns 7, 8, 9, and 10 have the same meaning as columns 3, 4, 5, and 6. However, they indicate the average results for Case II.

When running the PMLINST and PMLALG proactively, we present the results for the prediction and reference. Nonetheless, when considering the proactive-online approach, we use only the reference to alter the scheduling, given that we assume a dynamic method that modifies the scheduling in real-time. The same occurs for the corrective approach, in which we adjust the resulting sequences based on real failures. For this reason, Table 5.3 does not present the averages for predictions of the proactive-online and corrective approaches.

Table 5.3: Results for the proposed strategies considering Case I and Case II.

Strategy	n	Case I				Case II			
		f	Prediction f_S	Reference f_S	Time (s)	f	Prediction f_S	Reference f_S	Time (s)
PMLI - WF	4	55.02	506.40	510.27	0.19	47.50	464.18	472.41	0.37
	6	88.01	802.26	817.37	0.55	68.95	585.03	597.55	0.22
	8	119.97	720.21	745.62	0.57	95.64	572.09	590.63	1.87
	10	149.33	700.73	724.15	16.27	118.14	538.21	558.62	1.03
	12	182.15	665.66	684.81	54.61	140.24	372.77	391.47	1.28
	15	228.36	726.48	756.29	66.10	180.27	603.86	626.92	1.95
	20	307.62	815.98	855.40	102.13	240.34	509.61	534.10	2.74
	50	806.69	1410.59	1505.68	123.04	608.87	380.65	395.94	11.69
	100	1663.63	943.42	1034.23	126.98	1196.27	148.67	150.07	33.06
PMLI - F1	4	54.69	512.00	516.33	0.15	47.41	458.87	466.63	0.12
	6	87.52	785.49	799.85	0.60	68.89	578.98	590.90	0.62
	8	119.41	734.09	758.50	0.58	95.58	551.09	568.03	2.02
	10	148.43	723.64	747.03	17.97	118.03	540.81	561.15	1.12
	12	181.23	638.06	657.42	52.17	140.15	366.89	385.23	1.33
	15	227.28	717.44	746.18	70.62	180.17	603.94	624.66	3.68
	20	306.31	871.72	911.42	106.90	240.19	502.96	526.28	6.47
	50	803.07	1255.54	1342.47	122.95	608.87	379.43	395.36	19.41
	100	1677.17	1051.80	1141.67	127.09	1168.53	140.91	141.98	52.04
PMLI - F2	4	54.42	501.54	506.67	0.14	47.34	466.68	474.63	2.74
	6	87.39	785.77	800.13	0.45	68.83	577.17	589.15	0.25
	8	118.85	721.70	745.54	0.66	95.52	550.46	567.39	0.41
	10	147.91	673.77	696.76	16.73	117.92	536.86	557.96	6.55
	12	180.59	631.26	650.46	50.27	140.07	366.25	384.29	1.18
	15	226.23	703.18	731.47	82.21	180.06	596.86	618.10	8.55
	20	305.04	869.27	906.73	108.10	239.99	497.41	520.91	2.81
	50	800.01	1279.97	1368.82	120.73	608.87	379.39	395.45	10.74
	100	1637.13	900.12	982.86	126.39	1196.27	148.20	149.77	31.19
PMLA - WF	4	55.02	486.78	461.26	0.13	47.50	490.42	488.93	0.11
	6	88.01	812.96	829.29	0.40	68.95	550.53	552.71	0.19
	8	119.97	739.50	767.56	0.70	95.64	548.87	574.10	0.54
	10	149.33	672.17	668.66	10.47	118.14	508.68	522.96	1.44
	12	182.15	628.85	603.01	59.69	140.24	354.51	373.17	1.82
	15	228.35	695.96	663.03	61.12	180.27	574.63	588.24	2.46
	20	307.61	797.84	850.22	118.47	240.34	498.37	523.34	6.03
	50	806.69	1396.17	1285.88	112.80	608.87	378.77	398.02	13.14
	100	1661.70	940.51	1079.99	132.06	1196.27	147.09	148.50	30.17
PMLA - F1	4	54.69	486.39	453.71	0.14	47.41	511.75	497.53	0.13
	6	87.52	749.75	773.98	0.52	68.89	541.75	549.78	0.24
	8	119.41	718.00	737.90	0.58	95.58	547.86	590.38	0.59
	10	148.43	688.99	647.04	14.93	118.03	517.78	537.44	1.69
	12	181.22	622.48	638.57	27.86	140.15	352.12	372.63	1.98
	15	227.28	728.25	696.61	87.01	180.17	571.76	587.07	3.18
	20	306.30	856.70	808.22	116.95	240.19	489.86	515.07	7.22
	50	803.07	1296.86	1278.23	116.68	608.87	378.14	398.14	13.33
	100	1676.77	1028.91	975.41	131.19	1196.27	148.24	147.91	30.68
PMLA - F2	4	54.42	479.78	447.96	0.14	47.34	486.95	486.48	0.10
	6	87.39	746.29	766.75	0.46	68.83	540.39	546.73	0.21
	8	118.85	741.41	732.24	0.79	95.52	544.47	570.27	0.59
	10	147.91	648.46	620.42	28.28	117.92	515.95	533.32	1.51
	12	180.59	610.56	615.23	19.35	140.07	351.32	378.98	1.75
	15	226.23	711.55	634.58	105.76	180.06	616.99	582.97	4.08
	20	305.05	868.29	853.45	99.14	239.99	485.63	512.07	4.92
	50	800.01	1267.40	1189.15	125.11	608.87	378.13	401.81	13.72
	100	1638.07	883.92	931.62	131.99	1196.27	147.19	148.91	30.28
PMLI - WF PO	4	55.02	-	510.79	0.78	47.50	-	468.22	0.43
	6	88.01	-	750.14	1.43	68.95	-	579.63	0.76
	8	119.97	-	696.76	1.80	95.64	-	563.27	1.83
	10	149.33	-	703.88	37.11	118.14	-	517.28	4.84
	12	182.15	-	665.32	84.81	140.24	-	342.52	7.48
	15	228.35	-	717.26	141.28	180.27	-	537.17	10.83
	20	307.64	-	809.48	226.10	240.34	-	456.04	16.40
	50	806.68	-	1375.22	276.54	608.87	-	344.17	60.98
	100	1659.60	-	1001.69	337.97	1196.27	-	145.49	134.04

Table 5.3: Results for the proposed strategies considering Case I and Case II.

Strategy	n	Case I f	Case I Prediction f_S	Case I Reference f_S	Case I Time (s)	Case II f	Case II Prediction f_S	Case II Reference f_S	Case II Time (s)
PMLI - F1 PO	4	53.51	-	492.76	0.27	47.10	-	450.71	0.23
	6	86.37	-	689.53	0.49	68.51	-	562.53	0.44
	8	116.90	-	637.46	1.19	95.26	-	545.08	0.97
	10	145.88	-	726.01	14.09	117.62	-	511.97	2.22
	12	178.06	-	595.15	17.17	139.77	-	331.90	4.23
	15	222.78	-	634.22	92.09	179.39	-	515.39	6.78
	20	300.80	-	730.11	79.33	239.38	-	431.55	14.50
	50	788.25	-	1093.14	139.56	608.87	-	343.85	35.66
	100	1649.87	-	802.28	172.97	1196.27	-	148.30	74.38
PMLI - F2 PO	4	54.42	-	500.37	0.24	47.34	-	456.68	0.23
	6	87.39	-	747.57	0.51	68.83	-	576.65	0.45
	8	118.85	-	691.70	0.96	95.52	-	548.92	0.92
	10	147.91	-	683.91	16.95	117.92	-	514.73	2.68
	12	180.59	-	627.66	50.05	140.07	-	340.83	4.31
	15	226.24	-	697.72	83.89	180.06	-	530.11	13.80
	20	305.04	-	794.89	113.92	239.99	-	444.11	10.33
	50	800.01	-	1226.55	142.45	608.87	-	343.06	33.65
	100	1634.53	-	932.03	172.05	1196.27	-	147.27	74.70
PMLA - WF PO	4	55.02	-	486.97	0.25	47.50	-	483.54	0.29
	6	88.01	-	733.21	0.57	68.95	-	543.38	0.42
	8	119.97	-	706.55	0.92	95.64	-	536.07	1.08
	10	149.33	-	657.00	17.01	118.14	-	487.62	3.46
	12	182.15	-	613.20	57.36	140.24	-	326.85	5.68
	15	228.36	-	654.18	70.46	180.27	-	521.17	8.24
	20	307.65	-	717.33	109.17	240.34	-	439.63	13.08
	50	806.69	-	1228.01	146.04	608.87	-	336.50	41.81
	100	1652.50	-	843.68	182.35	1196.27	-	146.35	104.10
PMLA - F1 PO	4	53.70	-	490.44	0.39	47.14	-	473.21	0.25
	6	86.77	-	706.13	0.60	68.66	-	529.20	0.52
	8	117.81	-	667.24	1.10	95.41	-	524.18	1.27
	10	147.05	-	653.37	21.25	117.90	-	483.49	3.86
	12	179.55	-	568.76	27.97	139.79	-	314.76	6.18
	15	225.04	-	620.70	90.92	179.72	-	509.79	9.67
	20	303.82	-	706.55	108.69	239.77	-	432.75	22.23
	50	796.69	-	1030.04	147.60	608.87	-	336.28	44.27
	100	1663.30	-	784.17	188.44	1196.27	-	146.05	102.91
PMLA - F2 PO	4	54.42	-	475.19	0.28	47.34	-	480.40	0.27
	6	87.39	-	723.60	0.63	68.83	-	533.26	0.50
	8	118.85	-	689.34	0.94	95.52	-	530.67	1.22
	10	147.91	-	640.20	17.53	117.92	-	480.75	4.08
	12	180.59	-	581.87	51.97	140.07	-	316.52	6.34
	15	226.23	-	635.89	84.85	180.06	-	515.31	16.37
	20	305.03	-	722.49	113.40	239.99	-	430.16	13.82
	50	800.01	-	1108.76	142.78	608.87	-	334.40	43.48
	100	1634.47	-	791.68	178.06	1196.27	-	145.93	112.57
Corrective	4	58.11	-	636.96	0.17	48.26	-	505.77	0.10
	6	91.48	-	903.20	0.77	69.47	-	608.70	0.21
	8	125.57	-	943.60	0.72	96.01	-	589.76	0.45
	10	156.67	-	930.56	8.57	119.09	-	596.22	1.04
	12	190.38	-	846.99	41.21	140.70	-	402.91	1.15
	15	238.46	-	961.32	81.79	181.14	-	645.62	1.60
	20	321.59	-	1197.62	99.45	241.09	-	551.78	2.43
	50	843.55	-	2313.39	123.09	608.87	-	399.45	10.68
	100	1769.27	-	2275.13	129.09	1196.27	-	148.03	31.46

In Table 5.3 and following figures, the acronym "PO" indicates the proactive-online approach. The values obtained for f_S should have small differences when comparing the prediction and the reference, showing that the sequences considering the predictions are similar to the scheduling after the real failures. The number of failures predicted by the machine learning model for machine 2 could be a justification for the differences between the f_S values for the prediction and the reference shown in Table 5.3. In this case, the model underestimates the number of failures, distancing the prediction and the reference solutions for machine 2, increasing the deviation between the total weighted tardiness values obtained for each case.

Figure 5.13 presents the average results for the objective function of the short-term production planning problem for Case I. The average considers the fifteen variations for each product quantity. As we can observe, there is no significant difference between the results of the proposed approaches. The same behavior is observed considering Case II. For this reason, at the framework, we evaluate the best strategies only based on the results of the scheduling problem. We can also observe an increase in the objective function when the number of products grows, which may be due to the decrease in processing and setup times for larger product quantities. Since demand is not dependent on the product quantity, we observe this increase in the objective function.

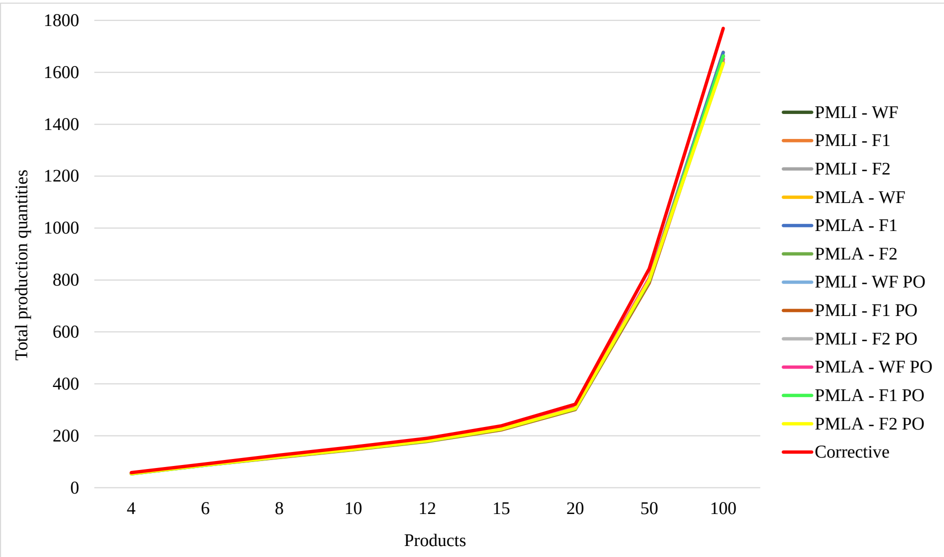


Figure 5.13: Comparison of the proposed approaches and strategies considering the objective function of the short-term production problem - Case I.

A larger difference in the objective function of the short-term production planning problem is observed for 100 products with the corrective approach. In this case, we do not use the predictions to modify the capacity parameter, resulting in more available hours for production, and consequently, more products can be produced, increasing the objective function. However, this approach does not consider the failures that will occur on the shop floor *a priori*. If we considered the failures at the moment they occur, it would result in higher total weighted tardiness, as can be seen in Figure 5.14.

On the other hand, when analyzing the results of the scheduling problem considering Case I, see Figure 5.14, we observe larger differences between the average total weighted tardiness found by the proposed approaches. Since the corrective method presents the greatest average total weighted tardiness for all the product quantities, it was the worst strategy compared to the proactive and proactive-online approaches. As we increase the number of products, the difference between the total weighted tardiness found by the remaining strategies compared to the corrective method becomes bigger. Since the corrective approach is the most used in practice, our proposed strategies would benefit the company considering the minimization of the total weighted tardiness and the characteristics of Case I.

Analyzing Figure 5.14, we observe an increase in the average total weighted tardiness for 50 products followed by a decrease for 100 products. This behavior may be due to the small variation in the distribution intervals when considering 100 products, as shown in Table 4.6. When solving

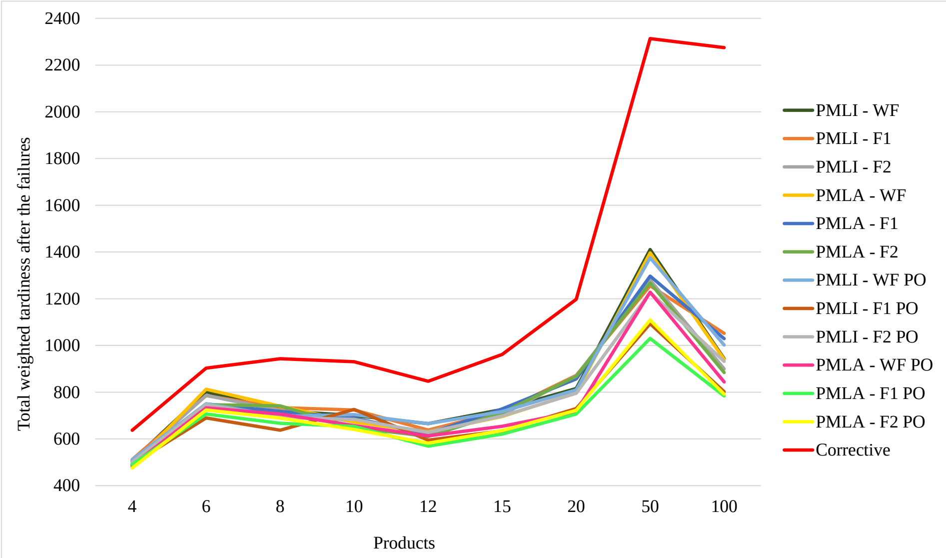


Figure 5.14: Comparison of the proposed approaches and strategies considering the total weighted tardiness - Case I.

the short-term production planning problem for 100 products, the factory can produce more quantities compared to 50 products; see the growth trend in Figure 5.13. Nonetheless, as the variation of processing and setup values are small for each of the 100 products, the scheduling heuristic allocates them with less impact on the total weighted tardiness compared to 50 products.

Figure 5.15 shows the results for the scheduling level after the failures considering Case II. As we can see, in this case, the corrective approach resulted in average total weighted tardiness closer to the values found by the proposed strategies. Therefore, when setup times are lower than processing times, the corrective approach could be an alternative for implementation on the shop floor, especially when we consider 100 products. Nonetheless, for the remaining product quantities, the corrective method still results in greater average total weighted tardiness compared to the proactive and proactive-online approaches.

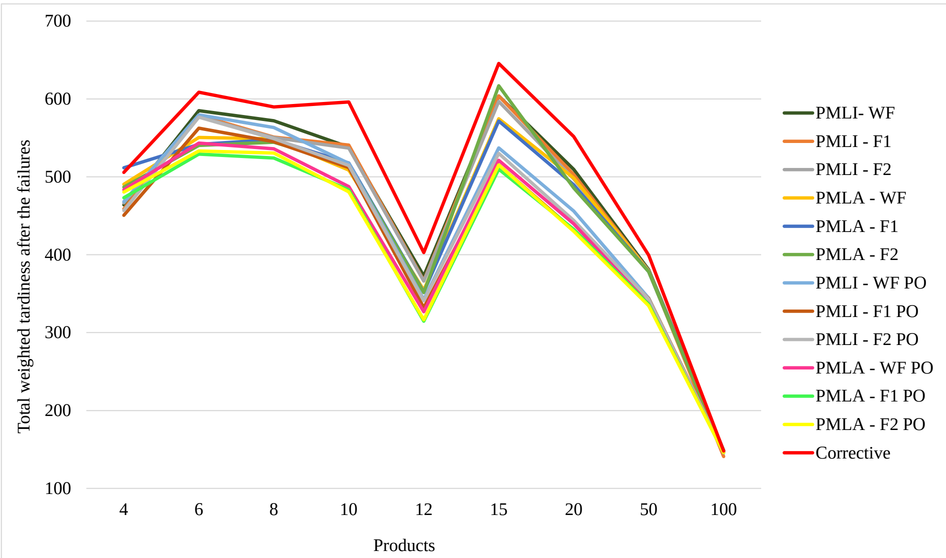


Figure 5.15: Comparison of the proposed approaches and strategies when considering the total weighted tardiness - Case II.

When considering Case I, the processing times are lower than Case II. Then, the factory can produce more products considering the specified capacity, increasing the objective function of the short-term production planning problem. At the scheduling level, with more production quantities, we observe a growth in completion times and, hence, an increase in the tardiness of the products. For this reason, the average total weighted tardiness are higher for Case I. Since the corrective method does not consider slacks at the capacity constraint like the other proposed strategies, the resulting solutions are more affected by the failures.

In Case II, we observe a decreasing trend when we increase the number of products. As in Case I, this behavior may be due to the small variation between the processing and setup times for each product when increasing the product quantities. In addition, we have a greater variation in the demand values, resulting in high values for the total weighted tardiness for some instances and an opposite behavior for other instances, generating average values with large variations. For example, see the change of behavior on the averages for 12 and 15 products.

Table 5.4 presents the results obtained with the proposed framework. In Column 1, we show the product quantities (n), while Column 2 exhibits the instance variation (v). Columns 3, 4, 5, and 6 present the results for Case I. Column 3 indicates the strategy predicted by the framework, whereas Column 4 presents the respective total weighted tardiness. Column 5 shows the average total weighted tardiness considering all the proposed strategies. In Column 6, we calculate the deviation (Dev) of the value obtained with the predicted strategy related to the average total weighted tardiness. In this case, $Dev = \frac{\sum W_j T_j - Average}{\sum W_j T_j}$. Columns 7, 8, 9, and 10 have the same meaning as Columns 3, 4, 5, and 6. However, they refer to Case II.

As we can observe in Table 5.4, considering Case I (Column 3), some strategies were predicted only for low product quantities, such as the PMLA-F2, used for 4 and 6 products. Other strategies were predicted only for medium quantities of products (PMLA-WF, for ten products), while others were used for large product quantities (PMLA-F1, PMLI-WF, PMLI-F1, and PMLA-WF PO, for 50 and 100 products). It is noteworthy that some of these strategies were used to solve only one instance, such as the PMLA-WF PO. On the other hand, some strategies were predicted for most instance sets, like the PMLA-F1 PO. We also highlight the use of PMLI-F1 PO for low and medium quantities of products and PMLA-F2 PO for large quantities.

Considering Case II (Column 7), we observe the use of PMLA-F1 PO for low product quantities (until ten products). Strategy PMLI-F1 PO was predicted only for four products, while PMLA-WF was chosen for several instance sets (6, 8, 15, and 100 products). Strategy PMLA-WF PO was also used for different product quantities (8, 10, 12, 15, 20, and 50 products). For medium product quantities, strategies PMLA-F2 PO and PMLI-WF PO were predicted for implementation on the shop floor (15 and 20 products). Lastly, for 50 and 100 products, we observe the prediction of PMLI-WF, PMLI-F1, PMLA-F2, and PMLI-WF PO.

Strategies PMLI-F2, PMLI-WF PO, PMLI-F2 PO, and the corrective method are not predicted for Case I, while strategies PMLI-F2, PMLA-F1, PMLI-F2 PO, and the corrective method are not chosen for Case II. Here, we emphasize the importance of using the proposed framework. With the training of all strategies, the method predicts the most suitable strategy for a specific instance based on its characteristics without having to solve and compare all the strategies each time. The use of several methods to solve the studied problem is a strategic decision, given that some procedures will behave better for a given instance set, whereas other methods result in better solutions for other production characteristics, justifying the proposition of the framework. However, we emphasize that for cases where all strategies output close values for the total weighted tardiness, it would not be necessary to use the proposed framework, e.g., as shown in Figure 5.15 when considering 100 products.

Table 5.4: Results obtained with the framework for Case I and Case II.

n	v	Case I Strategy	$\sum W_j T_j$	Average	Dev (%)	Case II Strategy	$\sum W_j T_j$	Average	Dev (%)
4	1	PMLA-F2	629.30	710.01	-12.82%	PMLI-F1 PO	464.35	556.44	-19.83%
	2	PMLA-F2	570.30	655.33	-14.91%	PMLA-F1 PO	832.30	807.52	2.98%
	3	PMLA-F2	218.55	256.23	-17.24%	PMLA-F1 PO	538.00	602.60	-12.01%
	4	PMLA-F2	734.60	571.92	22.14%	PMLA-F1 PO	197.30	196.99	0.16%
	5	PMLA-F1 PO	231.35	278.52	-20.39%	PMLI-F1 PO	489.60	489.61	0.00%
	6	PMLA-F2	534.70	495.68	7.30%	PMLA-F1 PO	820.40	865.02	-5.44%
	7	PMLI-F1 PO	614.80	691.12	-12.41%	PMLA-F1 PO	213.40	249.79	-17.05%
	8	PMLA-F2	296.20	244.35	17.51%	PMLA-F1 PO	305.80	355.78	-16.34%
	9	PMLA-F2	324.20	359.85	-11.00%	PMLA-F1 PO	1117.25	1139.42	-1.98%
	10	PMLA-F2	488.65	535.89	-9.67%	PMLA-F1 PO	219.00	264.55	-20.80%
	11	PMLA-F2	453.75	520.65	-14.74%	PMLA-F1 PO	355.20	350.68	1.27%
	12	PMLA-F2	483.65	504.19	-4.25%	PMLA-F1 PO	360.10	374.35	-3.96%
	13	PMLI-F1 PO	303.70	353.47	-16.39%	PMLA-F1 PO	121.60	124.98	-2.78%
	14	PMLA-F2	661.50	823.02	-24.42%	PMLI-F1 PO	337.30	444.72	-31.85%
	15	PMLA-F2	563.10	576.35	-2.35%	PMLI-F1 PO	254.75	328.39	-28.91%
6	1	PMLA-F2	1112.60	1096.03	1.49%	PMLA-WF	412.10	430.55	-4.48%
	2	PMLA-F2	1276.90	1093.02	14.40%	PMLA-F1 PO	1152.50	1183.56	-2.70%
	3	PMLA-F2	566.05	642.04	-13.42%	PMLA-WF	738.90	721.03	2.42%
	4	PMLA-F1 PO	1077.35	1018.18	5.49%	PMLA-WF	247.35	263.87	-6.68%
	5	PMLA-F2	434.35	435.20	-0.20%	PMLA-WF	714.30	749.87	-4.98%
	6	PMLA-F1 PO	909.40	1012.19	-11.30%	PMLA-WF	257.65	293.38	-13.87%
	7	PMLA-F2	549.90	604.50	-9.93%	PMLA-WF	615.20	548.06	10.91%
	8	PMLA-F2	1055.60	1148.75	-8.82%	PMLA-WF	685.05	683.19	0.27%
	9	PMLA-F2	399.65	494.20	-23.66%	PMLA-WF	259.80	276.10	-6.27%
	10	PMLA-F2	659.55	602.19	8.70%	PMLA-WF	652.30	602.49	7.64%
	11	PMLA-F2	547.60	580.90	-6.08%	PMLA-WF	74.20	93.87	-26.51%
	12	PMLA-F2	585.90	588.62	-0.46%	PMLA-WF	195.60	203.82	-4.20%
	13	PMLA-F2	442.30	585.59	-32.40%	PMLA-WF	244.75	275.10	-12.40%
	14	PMLA-F2	1215.90	1246.61	-2.53%	PMLA-WF	1586.40	1668.01	-5.14%
	15	PMLA-F2	278.55	316.47	-13.61%	PMLA-WF	426.80	438.49	-2.74%
8	1	PMLI-F1 PO	354.55	398.45	-12.38%	PMLA-WF PO	406.80	433.57	-6.58%
	2	PMLA-F1 PO	454.25	541.47	-19.20%	PMLA-F1 PO	590.70	590.77	-0.01%
	3	PMLI-F1 PO	319.15	354.65	-11.12%	PMLA-WF	589.95	524.49	11.10%
	4	PMLA-F1 PO	567.90	613.36	-8.01%	PMLA-WF PO	949.80	964.25	-1.52%
	5	PMLA-F1 PO	826.90	893.42	-8.04%	PMLA-WF PO	158.10	168.48	-6.57%
	6	PMLI-F1 PO	509.20	621.66	-22.09%	PMLA-WF PO	467.05	473.37	-1.35%
	7	PMLI-F1 PO	1050.40	1331.91	-26.80%	PMLA-F1 PO	1297.00	1426.00	-9.95%
	8	PMLI-F1 PO	904.00	975.45	-7.90%	PMLA-WF PO	550.00	582.76	-5.96%
	9	PMLI-F1 PO	706.05	787.64	-11.56%	PMLA-WF PO	20.40	20.40	0.00%
	10	PMLI-F1 PO	455.10	559.48	-22.94%	PMLA-WF PO	35.00	50.03	-42.95%
	11	PMLA-F1 PO	349.25	452.92	-29.68%	PMLA-F1 PO	935.10	998.27	-6.76%
	12	PMLA-F1 PO	526.45	704.91	-33.90%	PMLA-WF PO	629.60	633.39	-0.60%
	13	PMLI-F1 PO	1187.90	1192.26	-0.37%	PMLA-F1 PO	879.40	897.04	-2.01%
	14	PMLI-F1 PO	601.00	683.48	-13.72%	PMLA-WF PO	246.70	262.48	-6.40%
	15	PMLA-F1 PO	765.50	743.82	2.83%	PMLA-WF	217.10	227.92	-4.98%
10	1	PMLA-F1 PO	334.40	368.42	-10.17%	PMLA-WF PO	285.90	369.47	-29.23%
	2	PMLA-WF	838.30	824.31	1.67%	PMLA-WF PO	656.20	767.01	-16.89%
	3	PMLA-WF	966.20	979.64	-1.39%	PMLA-WF PO	567.10	596.63	-5.21%
	4	PMLA-F1 PO	684.30	557.14	18.58%	PMLA-WF PO	1050.20	1116.16	-6.28%
	5	PMLA-F1 PO	465.05	637.22	-37.02%	PMLA-WF PO	91.10	105.06	-15.33%
	6	PMLA-F1 PO	1224.40	1039.08	15.14%	PMLA-WF PO	179.10	179.97	-0.49%
	7	PMLA-F1 PO	725.50	809.63	-11.60%	PMLA-WF PO	333.35	358.58	-7.57%
	8	PMLA-WF	691.10	712.78	-3.14%	PMLA-WF PO	448.10	457.02	-1.99%
	9	PMLA-F1 PO	302.60	330.82	-9.32%	PMLA-WF PO	37.50	68.86	-83.62%
	10	PMLA-F1 PO	645.00	833.71	-29.26%	PMLA-WF PO	582.60	592.77	-1.75%
	11	PMLA-WF	181.30	245.01	-35.14%	PMLA-WF PO	633.80	650.48	-2.63%
	12	PMLA-WF	868.65	1055.74	-21.54%	PMLA-WF PO	673.10	650.26	3.39%
	13	PMLA-WF	785.00	804.76	-2.52%	PMLA-F1 PO	966.45	1018.62	-5.40%
	14	PMLA-WF	547.10	671.47	-22.73%	PMLA-F1 PO	428.60	499.96	-16.65%
	15	PMLA-F1 PO	627.45	633.38	-0.94%	PMLA-WF PO	388.70	358.01	7.90%

Table 5.4: Results obtained with the framework for Case I and Case II.

n	v	Case I Strategy	Case I $\sum W_j T_j$	Case I Average	Case I Dev (%)	Case II Strategy	Case II $\sum W_j T_j$	Case II Average	Case II Dev (%)
12	1	PMLI-F1 PO	508.42	553.11	-8.79%	PMLA-WF PO	273.30	296.46	-8.47%
	2	PMLI-F1 PO	125.85	172.82	-37.32%	PMLA-WF PO	164.70	149.61	9.16%
	3	PMLI-F1 PO	391.15	528.52	-35.12%	PMLA-WF PO	267.40	291.87	-9.15%
	4	PMLI-F1 PO	290.05	265.97	8.30%	PMLA-WF PO	44.40	42.92	3.33%
	5	PMLA-F1 PO	471.80	542.81	-15.05%	PMLA-WF PO	435.70	467.30	-7.25%
	6	PMLI-F1 PO	254.35	306.35	-20.44%	PMLA-WF PO	118.10	118.31	-0.18%
	7	PMLI-F1 PO	1219.90	1310.25	-7.41%	PMLA-WF PO	521.20	551.71	-5.85%
	8	PMLI-F1 PO	562.40	657.61	-16.93%	PMLA-WF PO	130.30	130.45	-0.12%
	9	PMLI-F1 PO	998.12	1014.24	-1.61%	PMLA-WF PO	69.90	79.38	-13.56%
	10	PMLI-F1 PO	556.42	417.13	25.03%	PMLA-WF PO	641.45	704.04	-9.76%
	11	PMLI-F1 PO	611.55	581.93	4.84%	PMLA-WF PO	376.65	444.26	-17.95%
	12	PMLA-F1 PO	1178.20	1328.53	-12.76%	PMLA-WF PO	181.20	192.99	-6.51%
	13	PMLI-F1 PO	496.90	672.98	-35.44%	PMLA-WF PO	762.30	768.41	-0.80%
	14	PMLI-F1 PO	336.00	389.36	-15.88%	PMLA-WF PO	664.40	699.66	-5.31%
	15	PMLI-F1 PO	727.30	830.50	-14.19%	PMLA-WF PO	251.80	301.23	-19.63%
15	1	PMLI-F1 PO	766.75	815.27	-6.33%	PMLA-WF PO	1207.50	1260.82	-4.42%
	2	PMLI-F1 PO	299.35	450.51	-50.50%	PMLA-WF PO	1014.60	1117.12	-10.10%
	3	PMLI-F1 PO	582.40	712.30	-22.30%	PMLA-WF PO	20.60	36.94	-79.31%
	4	PMLI-F1 PO	270.95	316.98	-16.99%	PMLI-WF PO	1575.05	1721.32	-9.29%
	5	PMLI-F1 PO	674.25	737.67	-9.41%	PMLA-F2 PO	741.30	840.96	-13.44%
	6	PMLI-F1 PO	269.45	349.22	-29.61%	PMLI-WF PO	170.30	181.95	-6.84%
	7	PMLI-F1 PO	1194.95	1212.76	-1.49%	PMLA-WF PO	121.60	121.52	0.07%
	8	PMLI-F1 PO	1207.45	1388.54	-15.00%	PMLI-WF PO	479.45	472.04	1.55%
	9	PMLA-F1 PO	367.65	393.64	-7.07%	PMLA-F2 PO	749.35	944.41	-26.03%
	10	PMLI-F1 PO	239.35	282.10	-17.86%	PMLA-WF PO	254.60	275.86	-8.35%
	11	PMLI-F1 PO	410.80	528.65	-28.69%	PMLI-WF PO	512.40	570.92	-11.42%
	12	PMLI-F1 PO	760.70	804.05	-5.70%	PMLI-WF PO	193.80	213.09	-9.95%
	13	PMLI-F1 PO	811.20	873.45	-7.67%	PMLA-WF	83.80	90.64	-8.16%
	14	PMLI-F1 PO	1190.70	1302.91	-9.42%	PMLI-WF PO	236.85	219.92	7.15%
	15	PMLI-F1 PO	423.05	452.10	-6.87%	PMLI-WF PO	359.00	404.74	-12.74%
20	1	PMLA-F2 PO	176.20	190.60	-8.17%	PMLA-WF PO	514.40	599.60	-16.56%
	2	PMLA-F1 PO	732.80	917.63	-25.22%	PMLA-F2 PO	431.10	491.48	-14.01%
	3	PMLA-F1 PO	541.10	667.60	-23.38%	PMLA-WF PO	457.70	479.77	-4.82%
	4	PMLA-F2 PO	450.40	524.25	-16.40%	PMLA-WF PO	942.30	996.36	-5.74%
	5	PMLI-F1 PO	172.95	358.38	-107.22%	PMLA-WF PO	335.00	420.80	-25.61%
	6	PMLA-F2 PO	539.80	736.54	-36.45%	PMLI-WF PO	353.30	360.43	-2.02%
	7	PMLI-F1 PO	1500.30	1778.11	-18.52%	PMLA-WF PO	67.20	87.28	-29.88%
	8	PMLA-F2 PO	409.90	524.14	-27.87%	PMLA-WF PO	1517.10	1615.88	-6.51%
	9	PMLI-F1 PO	691.10	738.12	-6.80%	PMLA-WF PO	185.00	216.28	-16.91%
	10	PMLA-F2 PO	1566.50	1584.99	-1.18%	PMLA-WF PO	254.10	255.62	-0.60%
	11	PMLI-F1 PO	625.85	736.28	-17.65%	PMLI-WF PO	175.25	202.13	-15.34%
	12	PMLI-F1 PO	330.25	475.50	-43.98%	PMLI-WF PO	903.30	814.30	9.85%
	13	PMLI-F1 PO	639.52	910.29	-42.34%	PMLA-WF PO	249.00	253.43	-1.78%
	14	PMLA-F2 PO	1558.20	1668.34	-7.07%	PMLA-WF PO	101.10	112.32	-11.09%
	15	PMLA-F2 PO	504.25	602.62	-19.51%	PMLA-WF PO	208.30	213.37	-2.43%
50	1	PMLA-F1 PO	728.75	1126.35	-54.56%	PMLI-WF	1150.20	1056.27	8.17%
	2	PMLA-F2 PO	1023.55	1278.46	-24.90%	PMLA-F2	471.40	478.32	-1.47%
	3	PMLA-F2 PO	544.65	839.13	-54.07%	PMLA-WF PO	558.69	568.53	-1.76%
	4	PMLA-F2 PO	646.49	727.95	-12.60%	PMLA-WF PO	669.80	770.88	-15.09%
	5	PMLA-F1	1299.80	1266.74	2.54%	PMLA-WF PO	88.10	74.70	15.21%
	6	PMLA-F1 PO	1488.55	2036.45	-36.81%	PMLA-F2	256.40	272.54	-6.29%
	7	PMLA-F2 PO	1352.90	1523.43	-12.60%	PMLI-F1	391.20	338.99	13.35%
	8	PMLA-F2 PO	1744.70	2198.00	-25.98%	PMLA-WF PO	609.49	640.87	-5.15%
	9	PMLA-F2 PO	1882.80	2162.21	-14.84%	PMLI-WF	15.10	16.61	-9.98%
	10	PMLA-F2 PO	733.30	929.73	-26.79%	PMLA-F2	22.50	17.02	24.38%
	11	PMLA-F1 PO	574.20	1002.52	-74.59%	PMLA-WF PO	103.30	114.24	-10.59%
	12	PMLA-F1	897.59	1311.95	-46.16%	PMLA-F2	697.99	658.88	5.60%
	13	PMLA-F2 PO	406.80	619.82	-52.36%	PMLI-F1	63.50	73.51	-15.76%
	14	PMLA-F1 PO	131.80	369.01	-179.98%	PMLI-WF	54.10	50.48	6.68%
	15	PMLA-F2 PO	2332.40	2548.59	-9.27%	PMLI-WF	324.40	305.34	5.87%

Table 5.4: Results obtained with the framework for Case I and Case II.

n	v	Case I				Case II			
		Strategy	$\sum W_j T_j$	Average	Dev (%)	Strategy	$\sum W_j T_j$	Average	Dev (%)
100	1	PMLA-F2 PO	3005.50	3169.05	-5.44%	PMLI-WF PO	101.60	107.92	-6.22%
	2	PMLA-F2 PO	926.10	1605.64	-73.38%	PMLA-WF	465.20	445.30	4.28%
	3	PMLI-F1	524.65	403.84	23.03%	PMLI-WF PO	19.20	23.05	-20.07%
	4	PMLI-WF	1207.49	1160.46	3.89%	PMLA-WF	13.00	14.62	-12.43%
	5	PMLI-F1	861.59	967.71	-12.32%	PMLI-WF PO	0.00	1.05	0.00%
	6	PMLI-F1	89.20	400.79	-349.32%	PMLI-WF PO	350.80	369.13	-5.23%
	7	PMLI-WF	208.10	872.91	-319.47%	PMLI-WF	212.10	215.40	-1.56%
	8	PMLI-F1	119.80	579.70	-383.89%	PMLA-WF	209.70	227.82	-8.64%
	9	PMLI-WF	2019.09	1905.86	5.61%	PMLI-WF PO	2.00	2.00	0.00%
	10	PMLA-F2 PO	698.30	934.66	-33.85%	PMLA-WF	402.40	415.99	-3.38%
	11	PMLA-WF PO	802.85	1068.48	-33.09%	PMLA-WF	18.00	17.82	1.03%
	12	PMLA-F2 PO	588.50	672.53	-14.28%	PMLI-WF PO	235.30	241.51	-2.64%
	13	PMLI-F1	658.49	771.17	-17.11%	PMLA-WF	32.20	33.36	-3.61%
	14	PMLA-F2 PO	266.10	473.76	-78.04%	PMLI-WF	0.40	0.40	0.00%
	15	PMLI-F1	155.90	220.37	-41.35%	PMLI-WF PO	87.20	85.85	1.55%

Figure 5.16 shows the average deviation (Dev) obtained for each product quantity for Cases I and II. The more negative the deviation, the better the strategy provided by the framework. Considering Case I, the average deviations tend to decrease when we increase the product quantities. Then, the greater n, the greater the effectiveness of the proposed framework to find solutions smaller than the average obtained by all strategies. For Case II, we observe a smaller variability between the average deviation considering the different product quantities. Furthermore, larger average deviations are observed when compared with the deviations obtained when solving Case I. Therefore, we can conclude that the framework allows smaller average deviations for most instances for Case I, i.e., when setups are longer than production times. This behavior may happen due to the lower variability between the results found by the different strategies for Case II, as shown in Figure 5.15.

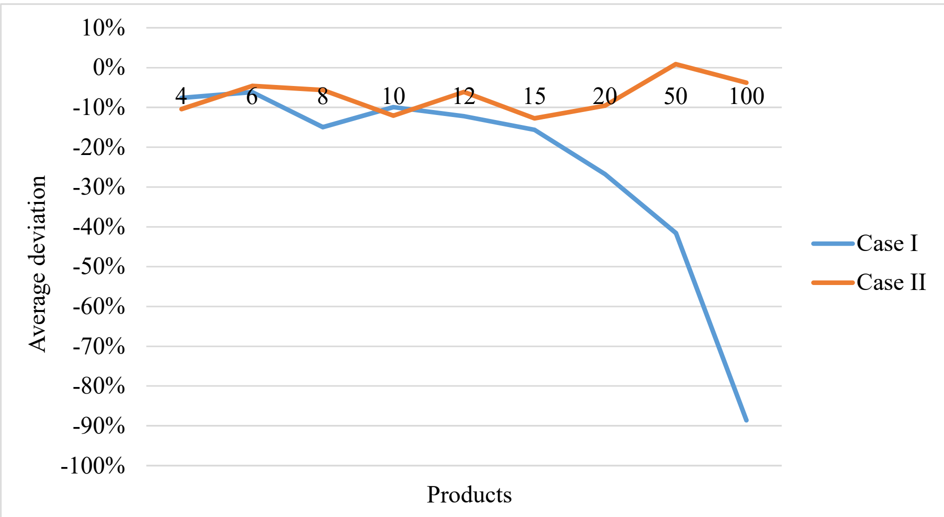


Figure 5.16: Average deviation considering Case I and Case II.

Regarding the results obtained by the strategies predicted by the framework, we can observe in Table 5.4, columns 6 and 10, that negative deviations were found for most instances considering Case I and II. This result shows that the strategies provided by the framework resulted in total weighted tardiness lower than the average of the strategies for most cases, which would bring substantial gains to the company. Furthermore, the strategies proved to be efficient in practice, with low average computational times (less than 300 seconds for large instances in Case I and less than 150 seconds for Case II). The longest computational times are obtained with the proactive-online approach since, in this case, the algorithm makes new predictions for the machine features if the forecast values deviate from the real values more than a specified margin of error.

The study of the adaptations to be made in the formulations and premisses of production planning, which was introduced in this thesis, consider technological changes that are already widely used in other fields, e.g., the increasing use of artificial intelligence in the healthcare sector. With the application of the proposed approaches and framework in real cases, the decision-maker would be considering the new available tools used in an Industry 4.0 environment, such as machine learning techniques for predicting production process variables, which would improve traditional production planning. Furthermore, using strategies that define production planning based on real-time information and that adapt the predictions according to new patterns of historical data would allow decisions to be made with greater certainty. With the increasing use of data for decision-making through sensors in the production environment, the decision-maker could easily collect the necessary information and use it as input for the algorithms presented in this work.

Chapter 6

Conclusions and future works

In this work, we study production planning problems subject to failures considering characteristics of an Industry 4.0 scenario. In Chapter 4, we introduce two proactive approaches aiming to find low probabilities of infeasibility when dealing with uncertainties. We verify if both approaches present similar results and test several robustness parameters to deal with failures at the short-term production planning level. The PA-I considers simulations to alter the production capacity constraint of the planning problem, whereas the PA-II analyzes the failures probability distribution to define the information inserted in the cuts. In Chapter 5, we propose proactive and proactive-online approaches to deal with disruptions. Based on failures predictions, we solve short-term production planning and scheduling problems through the proposed PMLI and PMLA approaches. Three feedback variations are used to alter the predictions, along with a corrective approach for comparison purposes. We also introduce a framework that predicts which approach results in the lowest total weighted tardiness based on the instance set. Further, we propose a benchmark to deal with failures considering a parallel machine environment.

Based on the experiments performed in Chapter 4, we conclude that the proposed approaches present similar results for most instance sets when comparing the averages and standard deviations for each variation and scenario tested. The greatest difference is obtained when considering Variation 3 for all the scenarios solved based on the data of Case I. For the remaining instance sets, the decision-maker could implement any of the proposed proactive approaches. Both propositions presented greater average deviations (***Dev***) when increasing the duration of the failures (Scenario 1 to Scenario 3), with ***Dev*** decreasing as the product quantities grow. The average total weighted tardiness increases with the augment in the duration of the failures. Furthermore, high averages were obtained for the tardiness values, which resulted in the proposition of new approaches in Chapter 5. The idle times increase for Scenarios 2 and 3, achieving high values, especially for Case II. Lastly, the probabilities of infeasibility did not present a pattern with the modification in the scenario and variation, which may be due to the use of an iterative strategy to solve the problem. Therefore, for implementation in a real environment, we recommend that the decision-maker analyze the results based on the product quantities, capturing differences between both approaches.

With the progress of Industry 4.0, decision-makers have the opportunity to integrate production planning problems with failures prediction to ensure accurate due dates for customers through data analysis and real-time information. Thus, in Chapter 5, we propose such integration using machine learning techniques. Regarding the objective function of the short-term production planning problem, we did not observe significant differences when comparing the results of the proposed approaches and strategies for Cases I and II. Therefore, in a real case, if the factory only wants to maximize the production quantities, the decision-maker could use any proposed approach and strategy. When minimizing the total weighted tardiness at the scheduling level, the computational results showed larger differences for Case I when comparing the values found for each approach considering the several strategies tested. The corrective method presented the maximum objective function. For Case II, the corrective method showed average results closer to the values obtained by the proposed approaches when considering 100 products. Nonetheless, our propositions resulted in lower averages for the total weighted tardiness for the remaining product quantities. Then, for Cases I and II, the decision-maker could use the proposed framework to predict which approach and strategy will probably result in the lowest objective function.

We conclude that the proactive and proactive-online strategies resulted in total weighted tardiness values much lower when compared to the corrective approach, which is the method usually used in industries for repair. Furthermore, the proposed framework predicted strategies that resulted in total weighted tardiness lower than the average values found by all the strategies for most instances.

From a managerial perspective, we are in a unique position where we can sense and monitor the whole factory shop in real-time with great precision. This massive amount of data allows new possibilities for planning strategies. At the same time, the need to quickly answer changes in the market and deliver products at a swift pace creates a challenge for the development of new algorithms. Our methods allow decision-makers to consider historical data to predict future events and use real-time information. The proposed framework automatically predicts which method will likely result in the lowest total weighted tardiness, enabling the planning process's automation, reducing cost, and thus, improving customer service. We strongly believe that solutions in this line of reasoning will be essential for succeeding companies. The managerial role is changing; instead of being a specialist on one type of problem, managers will need to focus on creating different strategies that can be inserted into frameworks like the one proposed here.

We emphasize that the proposed approaches could be applied in several real-world cases, such as: electronics (Dolgui et al. (2005)), project scheduling (Lambrechts et al. (2011)), manufacturing process of X-ray film (Aghezzaf et al. (2011)), automotive industry (Hu and Hu (2016)), stainless steel industry (Ruiz-Sarmiento et al. (2020)), personal care goods (Ayvaz and Alpay (2021)), among others. In these scenarios, the decision-maker should perform proper modifications in the formulations of the planning problems and analyze the specific uncertainties to define the information to be predicted and used to alter the production plan.

For future works, we intend to investigate the changes in the strategic decision-making process according to the different algorithms used, focusing on the human-algorithm connection. We will consider several plants of a supply chain, analyzing the types of information used for decision-making and how algorithms like machine learning techniques can improve this task.

Bibliography

Abbasimehr, H., Shabani, M., and Yousefi, M. (2020). An optimized model using LSTM network for demand forecasting. *Computers & Industrial Engineering*, 143:106435.

Aghezzaf, E.-H., Sitompul, C., and Van den Broecke, F. (2011). A robust hierarchical production planning for a capacitated two-stage production system. *Computers & Industrial Engineering*, 60(2):361–372.

Allahverdi, A. (2015). The third comprehensive survey on scheduling problems with setup times/costs. *European Journal of Operational Research*, 246(2):345–378.

Allahverdi, A. (2016). A survey of scheduling problems with no-wait in process. *European Journal of Operational Research*, 255(3):665–686.

Aloulou, M. A., Dolgui, A., and Kovalyov, M. Y. (2014). A bibliography of non-deterministic lot-sizing models. *International Journal of Production Research*, 52(8):2293–2310.

Alves, F. F., Nogueira, T. H., de Souza, M. C., and Ravetti, M. G. (2021). Approaches for the joint resolution of lot-sizing and scheduling with infeasibilities occurrences. *Computers & Industrial Engineering*, 155:107176.

Aytug, H., Lawley, M. A., McKay, K., Mohan, S., and Uzsoy, R. (2005). Executing production schedules in the face of uncertainties: A review and some future directions. *European Journal of Operational Research*, 161(1):86–110.

Ayvaz, S. and Alpay, K. (2021). Predictive maintenance system for production lines in manufacturing: A machine learning approach using IoT data in real-time. *Expert Systems with Applications*, 173:114598.

Azizoglu, M. and Alagöz, O. (2005). Parallel-machine rescheduling with machine disruptions. *IIE Transactions*, 37(12):1113–1118.

Bertsimas, D., Brown, D. B., and Caramanis, C. (2011). Theory and applications of robust optimization. *SIAM review*, 53(3):464–501.

Bertsimas, D. and Sim, M. (2004). The price of robustness. *Operations Research*, 52(1):35–53.

Blazewicz, J., Ecker, K., Pesch, E., Schmidt, G., and Weglarz, J. (2019). *Handbook on scheduling*. Springer.

Brahimi, N., Absi, N., Dauzère-Pérès, S., and Nordli, A. (2017). Single-item dynamic lot-sizing problems: An updated survey. *European Journal of Operational Research*, 263(3):838–863.

Branco, P., Torgo, L., and Ribeiro, R. (2016). A survey of predictive modeling under imbalanced distributions. *ACM Computing Surveys*, 49(2):1–31.

Brčić, M., Katić, M., and Hlupić, N. (2019). Planning horizons based proactive rescheduling for stochastic resource-constrained project scheduling problems. *European Journal of Operational Research*, 273(1):58–66.

Bueno, A. F., Godinho Filho, M., and Frank, A. G. (2020). Smart production planning and control in the Industry 4.0 context: A systematic literature review. *Computers & Industrial Engineering*, 149:106774.

Burez, J. and Van den Poel, D. (2009). Handling class imbalance in customer churn prediction. *Expert Systems with Applications*, 36(3):4626–4636.

Cadavid, J. P. U., Lamouri, S., Grabot, B., Pellerin, R., and Fortin, A. (2020). Machine learning applied in production planning and control: A state-of-the-art in the era of Industry 4.0. *Journal of Intelligent Manufacturing*, 31:1531–1558.

Carvalho, T. P., Soares, F. A., Vita, R., Francisco, R. d. P., Basto, J. P., and Alcalá, S. G. (2019). A systematic literature review of machine learning methods applied to predictive maintenance. *Computers & Industrial Engineering*, 137:106024.

Chica, M., Juan Pérez, A. A., Cordon, O., and Kelton, D. (2017). Why simheuristics? Benefits, limitations, and best practices when combining metaheuristics with simulation. *SSRN*.

Chrétienne, P. (2020). Maximizing the number of jobs scheduled at their baseline starting times in case of machine failures. *Journal of Scheduling*, 23(1):135–143.

Crowder, M. and Lawless, J. (2007). On a scheme for predictive maintenance. *European Journal of Operational Research*, 176(3):1713–1722.

Cui, W., Lu, Z., Li, C., and Han, X. (2018). A proactive approach to solve integrated production scheduling and maintenance planning problem in flow shops. *Computers & Industrial Engineering*, 115:342–353.

de Jonge, B. and Scarf, P. A. (2020). A review on maintenance optimization. *European Journal of Operational Research*, 285(3):805–824.

Dias, L. S. and Ierapetritou, M. G. (2016). Integration of scheduling and control under uncertainties: Review and challenges. *Chemical Engineering Research and Design*, 116:98–113.

Dolgui, A., Kovalyov, M. Y., and Shchamialiova, K. (2011). Multi-product lot-sizing and sequencing on a single imperfect machine. *Computational Optimization and Applications*, 50(3):465–482.

Dolgui, A., Levin, G., and Louly, M.-A. (2005). Decomposition approach for a problem of lot-sizing and sequencing under uncertainties. *International Journal of Computer Integrated Manufacturing*, 18(5):376–385.

Fan, B., Li, S., Zhou, L., and Zhang, L. (2011). Scheduling resumable deteriorating jobs on a single machine with non-availability constraints. *Theoretical Computer Science*, 412(4-5):275–280.

Fazayeli, M., Aleagha, M.-R., Bashirzadeh, R., and Shafaei, R. (2016). A hybrid meta-heuristic algorithm for flowshop robust scheduling under machine breakdown uncertainty. *International Journal of Computer Integrated Manufacturing*, 29(7):709–719.

Feng, W., Zheng, L., and Li, J. (2012). The robustness of scheduling policies in multi-product manufacturing systems with sequence-dependent setup times and finite buffers. *Computers & Industrial Engineering*, 63(4):1145–1153.

Fernández, A., García, S., Galar, M., Prati, R. C., Krawczyk, B., and Herrera, F. (2018). *Learning from imbalanced data sets*. Springer.

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

Ghaleb, M., Zolfagharinia, H., and Taghipour, S. (2020). Real-time production scheduling in the Industry-4.0 context: Addressing uncertainties in job arrivals and machine breakdowns. *Computers & Operations Research*, 123:105031.

Ghezail, F., Pierreval, H., and Hajri-Gabouj, S. (2010). Analysis of robustness in proactive scheduling: A graphical approach. *Computers & Industrial Engineering*, 58(2):193–198.

Gonzalez-Neira, E. M., Ferone, D., Hatami, S., and Juan, A. A. (2017). A biased-randomized simheuristic for the distributed assembly permutation flowshop problem with stochastic processing times. *Simulation Modelling Practice and Theory*, 79:23–36.

Goren, S. and Sabuncuoglu, I. (2009). Optimization of schedule robustness and stability under random machine breakdowns and processing time variability. *IIE Transactions*, 42(3):203–220.

Gruson, M., Cordeau, J.-F., and Jans, R. (2018). The impact of service level constraints in deterministic lot sizing with backlogging. *Omega*, 79:91–103.

Gürel, S., Körpeoğlu, E., and Aktürk, M. S. (2010). An anticipative scheduling approach with controllable processing times. *Computers & Operations Research*, 37(6):1002–1013.

Hartigan, J. A. and Wong, M. A. (1979). Algorithm AS 136: A k-means clustering algorithm. *Journal of the royal statistical society. Series C (Applied Statistics)*, 28(1):100–108.

Hatami, S., Calvet, L., Fernández-Viagas, V., Framiñán, J. M., and Juan, A. A. (2018). A simheuristic algorithm to set up starting times in the stochastic parallel flowshop problem. *Simulation Modelling Practice and Theory*, 86:55–71.

Hochreiter, S. and Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

Hofmann, E. and Rüsch, M. (2017). Industry 4.0 and the current status as well as future prospects on logistics. *Computers in Industry*, 89:23–34.

Hu, Z. and Hu, G. (2016). A two-stage stochastic programming model for lot-sizing and scheduling under uncertainty. *International Journal of Production Economics*, 180:198–207.

James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). *An introduction to statistical learning*, volume 112. Springer.

Jans, R. and Degraeve, Z. (2008). Modeling industrial lot sizing problems: A review. *International Journal of Production Research*, 46(6):1619–1643.

Juan, A. A., Barrios, B. B., Vallada, E., Riera, D., and Jorba, J. (2014a). A simheuristic algorithm for solving the permutation flow shop problem with stochastic processing times. *Simulation Modelling Practice and Theory*, 46:101–117.

Juan, A. A., Faulin, J., Grasman, S. E., Rabe, M., and Figueira, G. (2015). A review of simheuristics: Extending metaheuristics to deal with stochastic combinatorial optimization problems. *Operations Research Perspectives*, 2:62–72.

Juan, A. A., Grasman, S. E., Caceres-Cruz, J., and Bektaş, T. (2014b). A simheuristic algorithm for the single-period stochastic inventory-routing problem with stock-outs. *Simulation Modelling Practice and Theory*, 46:40–52.

Kang, Z., Catal, C., and Tekinerdogan, B. (2021). Remaining useful life (RUL) prediction of equipment in production lines using artificial neural networks. *Sensors*, 21(3):932.

Karamaziotis, P. I., Raptis, A., Nikolopoulos, K., Litsiou, K., and Assimakopoulos, V. (2020). An empirical investigation of water consumption forecasting methods. *International Journal of Forecasting*, 36(2):588–606.

Keha, A. B., Khowala, K., and Fowler, J. W. (2009). Mixed integer programming formulations for single machine scheduling problems. *Computers & Industrial Engineering*, 56(1):357–367.

Kim, J.-G., Song, S., and Jeong, B. (2020). Minimising total tardiness for the identical parallel machine scheduling problem with splitting jobs and sequence-dependent setup times. *International Journal of Production Research*, 58(6):1628–1643.

Kim, K.-J. (2003). Financial time series forecasting using Support Vector Machines. *Neurocomputing*, 55(1-2):307–319.

Kim, T.-Y. and Cho, S.-B. (2019). Predicting residential energy consumption using CNN-LSTM neural networks. *Energy*, 182:72–81.

Kim, Y.-H. and Kim, R.-S. (2020). Insertion of new idle time for unrelated parallel machine scheduling with job splitting and machine breakdowns. *Computers & Industrial Engineering*, 147:106630.

Kim, Y.-I. and Kim, H.-J. (2021). Rescheduling of unrelated parallel machines with job-dependent setup times under forecasted machine breakdown. *International Journal of Production Research*, 59(17):5236–5258.

King, W. R. and Teo, T. S. (2000). Assessing the impact of proactive versus reactive modes of strategic information systems planning. *Omega*, 28(6):667–679.

Kleijnen, J. P. and Gaury, E. (2003). Short-term robustness of production management systems: A case study. *European Journal of Operational Research*, 148(2):452–465.

Kuhn, M. and Johnson, K. (2013). *Applied predictive modeling*, volume 26. Springer.

Lambrechts, O., Demeulemeester, E., and Herroelen, W. (2011). Time slack-based techniques for robust project scheduling subject to resource uncertainty. *Annals of Operations Research*, 186(1):443–464.

Lee, C.-Y. and Yu, G. (2008). Parallel-machine scheduling under potential disruption. *Optimization Letters*, 2(1):27–37.

Lee, J., Kim, C., and Choi, G. (2019). Exploring data envelopment analysis for measuring collaborated innovation efficiency of small and medium-sized enterprises in Korea. *European Journal of Operational Research*, 278(2):533–545.

Leukel, J., González, J., and Riekert, M. (2021). Adoption of machine learning technology for failure prediction in industrial maintenance: A systematic review. *Journal of Manufacturing Systems*, 61:87–96.

Li, Z. and Floudas, C. A. (2014). A comparative theoretical and computational study on robust counterpart optimization: III. Improving the quality of robust solutions. *Industrial & Engineering Chemistry Research*, 53(33):13112–13124.

Likas, A., Vlassis, N., and Verbeek, J. J. (2003). The global k-means clustering algorithm. *Pattern recognition*, 36(2):451–461.

Lindberg, P. (1990). Strategic manufacturing management: A proactive approach. *International Journal of Operations & Production Management*, 10(2):94–106.

Liu, L. and Zhou, H. (2013). On the identical parallel-machine rescheduling with job rework disruption. *Computers & Industrial Engineering*, 66(1):186–198.

Liu, Z. and Ro, Y. K. (2014). Rescheduling for machine disruption to minimize makespan and maximum lateness. *Journal of Scheduling*, 17(4):339–352.

Mehta, S. V. and Uzsoy, R. M. (1998). Predictable scheduling of a job shop subject to breakdowns. *IEEE Transactions on Robotics and Automation*, 14(3):365–378.

Miller, C. E., Tucker, A. W., and Zemlin, R. A. (1960). Integer programming formulation of Traveling Salesman Problems. *Journal of the ACM (JACM)*, 7(4):326–329.

Mokotoff, E. (2004). An exact algorithm for the identical parallel machine scheduling problem. *European Journal of Operational Research*, 152(3):758–769.

Mourtzis, D. and Vlachou, E. (2018). A cloud-based cyber-physical system for adaptive shop-floor scheduling and condition-based maintenance. *Journal of Manufacturing Systems*, 47:179–198.

Nourelfath, M. (2011). Service level robustness in stochastic production planning under random machine breakdowns. *European Journal of Operational Research*, 212(1):81–88.

Ozlen, M. and Azizoğlu, M. (2011). Rescheduling unrelated parallel machines with total flow time and total disruption cost criteria. *Journal of the Operational Research Society*, 62(1):152–164.

Park, T., Lee, T., and Kim, C. O. (2012). Due-date scheduling on parallel machines with job splitting and sequence-dependent major/minor setup times. *The International Journal of Advanced Manufacturing Technology*, 59(1-4):325–333.

Parker, R., Deane, R., and Holmes, R. (1977). On the use of a vehicle routing algorithm for the parallel processor problem with sequence dependent changeover costs. *AIIE Transactions*, 9(2):155–160.

Peng, H. and van Houtum, G.-J. (2016). Joint optimization of condition-based maintenance and production lot-sizing. *European Journal of Operational Research*, 253(1):94–107.

Pochet, Y. (2001). Mathematical programming models and formulations for deterministic production planning problems. In *Computational Combinatorial Optimization*, pages 57–111. Springer.

Pochet, Y. and Wolsey, L. A. (2006). *Production Planning by Mixed Integer Programming*. Springer Science & Business Media.

Priore, P., de la Fuente, D., Puente, J., and Parreño, J. (2006). A comparison of machine-learning algorithms for dynamic scheduling of flexible manufacturing systems. *Engineering Applications of Artificial Intelligence*, 19(3):247–255.

Priore, P., Ponte, B., Puente, J., and Gómez, A. (2018). Learning-based scheduling of flexible manufacturing systems using ensemble methods. *Computers & Industrial Engineering*, 126:282–291.

Rangsaritratsamee, R., Ferrell Jr, W. G., and Kurz, M. B. (2004). Dynamic rescheduling that simultaneously considers efficiency and stability. *Computers & Industrial Engineering*, 46(1):1–15.

Riotteau, G., Scaloni, O., and Nèron, E. (2001). Ordonnancement sur des machines identiques avec splitting et temps de prèparation dèpendant de la sequence. In *MOSIM'01: conference francophone de modelisation et simulation*.

Rocco, C. D. and Morabito, R. (2016). Robust optimisation approach applied to the analysis of production/logistics and crop planning in the tomato processing industry. *International Journal of Production Research*, 54(19):5842–5861.

Rossit, D. A., Tohmé, F., and Frutos, M. (2019). Industry 4.0: Smart Scheduling. *International Journal of Production Research*, 57(12):3802–3813.

Ruiz-Sarmiento, J.-R., Monroy, J., Moreno, F.-A., Galindo, C., Bonelo, J.-M., and Gonzalez-Jimenez, J. (2020). A predictive model for the maintenance of industrial machinery in the context of Industry 4.0. *Engineering Applications of Artificial Intelligence*, 87:103289.

Sagheer, A. and Kotb, M. (2019). Time series forecasting of petroleum production using deep LSTM recurrent networks. *Neurocomputing*, 323:203–213.

Santos, P., Maudes, J., and Bustillo, A. (2018). Identifying maximum imbalance in datasets for fault diagnosis of gearboxes. *Journal of Intelligent Manufacturing*, 29(2):333–351.

Schmidt, G. (2000). Scheduling with limited machine availability. *European Journal of Operational Research*, 121(1):1–15.

Shiue, Y.-R., Guh, R.-S., and Lee, K.-C. (2012). Development of machine learning-based real time scheduling systems: Using ensemble based on wrapper feature selection approach. *International Journal of Production Research*, 50(20):5887–5905.

Soyster, A. L. (1973). Convex programming with set-inclusive constraints and applications to inexact linear programming. *Operations Research*, 21(5):1154–1157.

Sun, Y., Wong, A. K., and Kamel, M. S. (2009). Classification of imbalanced data: A review. *International Journal of Pattern Recognition and Artificial Intelligence*, 23(04):687–719.

Susto, G. A., Schirru, A., Pampuri, S., McLoone, S., and Beghi, A. (2015). Machine learning for predictive maintenance: A multiple classifier approach. *IEEE Transactions on Industrial Informatics*, 11(3):812–820.

Taieb, S. B., Sorjamaa, A., and Bontempi, G. (2010). Multiple-output modeling for multi-step-ahead time series forecasting. *Neurocomputing*, 73(10-12):1950–1957.

Talay, I. and Özdemir-Akyıldırım, Ö. (2019). Optimal procurement and production planning for multi-product multi-stage production under yield uncertainty. *European Journal of Operational Research*, 275(2):536–551.

Tang, L. and Zhang, Y. (2009). Parallel machine scheduling under the disruption of machine breakdown. *Industrial & Engineering Chemistry Research*, 48(14):6660–6667.

Topan, E. and van der Heijden, M. C. (2020). Operational level planning of a multi-item two-echelon spare parts inventory system with reactive and proactive interventions. *European Journal of Operational Research*, 284(1):164–175.

Trunk, A., Birkel, H., and Hartmann, E. (2020). On the current state of combining human and artificial intelligence for strategic organizational decision making. *Business Research*, 13(3):875–919.

Vallada, E. and Ruiz, R. (2011). A genetic algorithm for the unrelated parallel machine scheduling problem with sequence dependent setup times. *European Journal of Operational Research*, 211(3):612–622.

Verderame, P. M., Elia, J. A., Li, J., and Floudas, C. A. (2010). Planning and scheduling under uncertainty: A review across multiple sectors. *Industrial & Engineering Chemistry Research*, 49(9):3993–4017.

Vidal, A. and Kristjanpoller, W. (2020). Gold volatility prediction using a CNN-LSTM approach. *Expert Systems with Applications*, 157:113481.

Vieira, G. E., Herrmann, J. W., and Lin, E. (2000). Predicting the performance of rescheduling strategies for parallel machine systems. *Journal of Manufacturing Systems*, 19(4):256–266.

Vieira, G. E., Herrmann, J. W., and Lin, E. (2003). Rescheduling manufacturing systems: A framework of strategies, policies, and methods. *Journal of Scheduling*, 6(1):39–62.

Wang, D.-J., Liu, F., Wang, Y.-Z., and Jin, Y. (2015). A knowledge-based evolutionary proactive scheduling approach in the presence of machine breakdown and deterioration effect. *Knowledge-Based Systems*, 90:70–80.

Yin, Y., Cheng, T., and Wang, D.-J. (2016). Rescheduling on identical parallel machines with machine disruptions to minimize total completion time. *European Journal of Operational Research*, 252(3):737–749.

Yu, Y., Si, X., Hu, C., and Zhang, J. (2019). A review of recurrent neural networks: LSTM cells and network architectures. *Neural Computation*, 31(7):1235–1270.

Zhang, X., Liu, S.-C., Lin, W.-C., and Wu, C.-C. (2020). Parallel-machine scheduling with linear deteriorating jobs and preventive maintenance activities under a potential machine disruption. *Computers & Industrial Engineering*, 145:106482.

Appendix A

Performance of the scheduling heuristic

Table A.1 presents the results for Case I obtained with the proposed scheduling heuristic comparing its results with a mathematical model, presented in (A.1)-(A.8), which is based on the works of Parker et al. (1977), Riotteau et al. (2001), and Keha et al. (2009). Columns 1 and 2 present, respectively, the number of product quantities (n) and the fifteen variations generated for each product quantity (v), explained in Section 4.5.1. Column 3 shows the objective function found by the heuristic (f_S), while Column 4 exhibits the resolution time ($\boldsymbol{Time}$ (s)). Columns 5, 6, and 7 show, respectively, the objective function found by the mathematical model (A.1)-(A.8), the optimality gap ($\boldsymbol{Gap}$ (%)), and the resolution time. Lastly, Column 8 presents the deviation of the value found with the proposed heuristic related to the value found by the mathematical model ($\boldsymbol{Dev}$ (%)). In the formulation, Γ indicates the set of products defined for production on machines 1 and 2. The resolution time of the short-term production planning and the scheduling models was limited to 120 seconds. We emphasize that columns 4 and 7 present the total resolution time, including the time to solve the short-term production planning problem. In this study of the performance of the proposed heuristic, we consider only one planning period.

$$\text{Min} \quad \sum_{j \in \Gamma} W_j T_j \tag{A.1}$$

$$\text{s.t.} \quad (3.4) - (3.9), (3.11) - (3.14), \tag{A.2}$$

$$\sum_{m \in M} Q_{jm} = p_j \sum_{m \in M} q_{jm}, \qquad \forall j \in \Gamma, \tag{A.3}$$

$$Q_{jm}/C_{m1} \le y_{jm} \le C_{m1} Q_{jm}, \qquad \forall j \in \Gamma, \forall m \in M, \tag{A.4}$$

$$r_{jm} \ge r_{im} + Q_{im} + S_{ij} y_{im} - \phi(1 - x_{ijm}), \qquad \forall i \in \Gamma \cup \{0\}, \forall j \in \Gamma : i \ne j, \forall m \in M, \tag{A.5}$$

$$T_j \ge r_{jm} + Q_{jm} - d_{j1}, \qquad \forall j \in \Gamma, \forall m \in M, \tag{A.6}$$

$$r_{jm} \ge 0, \qquad \forall j \in \Gamma \cup \{0\}, \forall m \in M, \tag{A.7}$$

$$T_j \ge 0, \qquad \forall j \in \Gamma. \tag{A.8}$$

The scheduling formulation minimizes the total weighted tardiness, as presented in Equation (A.1). In (A.2), we have the Constraints (3.4)-(3.9) and (3.11)-(3.14), which were introduced in Section 3.3. Equations (A.3) guarantee the production of the quantities defined by the short-term production planning problem for each product, with variable q_{jm} consisting of an output of this problem formulation. We highlight that the scheduling model allows job splitting, i.e., products may be split for production into both machines. Constraints (A.4) ensure that, if product j is produced on machine m, variable y_{jm} assumes a value equal to one. Otherwise, the variable is equal to zero. In Constraints (A.5), we calculate the release time for each product j of machine

m. The parameter ϕ indicates a large value defined as $trunc(\sum_{k \in J} p_k D_k + \sum_{k \in J} max\{l \in J\} S_{lj}, 1)$. The tardiness values are calculated in Constraints (A.6), with $r_{jm} + Q_{jm}$ indicating the completion time of product j on machine m. Lastly, Constraints (A.7) and (A.8) present the domain of the decision variables.

Table A.1: Results obtained with the proposed scheduling heuristic compared with a mathematical model solved with a commercial solver - Case I.

n	v	Heuristic		Exact			$Dev(\%)$
		f_S	$Time(s)$	f_S	$Gap(\%)$	$Time(s)$	
4	1	166.50	0.20	109.50	0.00%	1.16	34.23%
	2	525.50	0.41	490.85	0.00%	0.52	6.59%
	3	484.50	0.20	350.75	0.00%	0.28	27.61%
	4	314.15	0.27	240.00	0.00%	0.58	23.60%
	5	40.65	0.09	40.65	0.00%	0.52	0.00%
	6	47.50	0.22	47.50	0.00%	0.80	0.00%
	7	252.30	0.30	232.20	0.00%	0.48	7.97%
	8	45.00	0.14	45.00	0.00%	0.48	0.00%
	9	15.75	0.22	15.75	0.00%	0.78	0.00%
	10	46.80	0.28	46.80	0.00%	0.41	0.00%
	11	306.25	0.27	298.25	0.00%	0.39	2.61%
	12	508.00	0.33	411.90	0.00%	0.44	18.92%
	13	274.00	0.28	235.50	0.00%	1.64	14.05%
	14	229.80	0.23	229.80	0.00%	0.66	0.00%
	15	105.20	0.22	57.65	0.00%	1.61	45.20%
6	1	367.40	1.03	297.55	0.00%	21.62	19.01%
	2	575.00	0.58	530.20	0.00%	63.30	7.79%
	3	278.65	0.48	200.90	0.00%	41.44	27.90%
	4	312.40	0.84	285.65	0.00%	28.80	8.56%
	5	266.40	0.48	241.20	0.00%	71.88	9.46%
	6	482.95	0.88	464.00	0.00%	40.59	3.92%
	7	339.50	0.27	294.30	0.00%	2.59	13.31%
	8	368.05	0.59	332.60	0.00%	70.86	9.63%
	9	28.65	0.56	28.05	0.00%	45.72	2.09%
	10	614.50	0.56	538.45	0.00%	2.81	12.38%
	11	323.05	0.48	244.10	0.00%	1.95	24.44%
	12	367.60	0.17	289.00	0.00%	2.22	21.38%
	13	254.35	0.91	154.20	0.00%	38.11	39.37%
	14	573.00	0.45	502.20	0.00%	59.17	12.36%
	15	142.95	0.89	81.80	0.00%	30.48	42.78%
8	1	141.00	0.98	115.20	100.00%	121.17	18.30%
	2	288.25	0.91	267.75	82.00%	121.28	7.11%
	3	317.80	0.28	298.35	90.00%	120.41	6.12%
	4	477.50	0.33	447.00	100.00%	120.50	6.39%
	5	558.70	0.58	543.60	95.00%	122.23	2.70%
	6	134.25	0.22	88.25	100.00%	120.53	34.26%
	7	395.90	0.92	415.90	78.00%	121.66	-5.05%
	8	441.55	1.12	422.25	87.00%	121.50	4.37%
	9	234.35	0.23	225.60	100.00%	120.34	3.73%
	10	40.30	0.44	36.60	100.00%	120.58	9.18%
	11	366.35	0.41	335.70	89.00%	120.48	8.37%
	12	635.50	0.31	595.30	94.00%	120.36	6.33%
	13	775.75	0.73	704.15	88.00%	120.66	9.23%
	14	115.05	0.95	95.25	100.00%	120.88	17.21%
	15	175.85	0.45	155.45	100.00%	120.47	11.60%
10	1	238.60	2.02	227.40	97.00%	121.31	4.69%
	2	338.40	1.16	368.45	91.00%	121.00	-8.88%
	3	358.10	1.36	365.90	86.00%	121.19	-2.18%
	4	347.20	2.12	331.45	98.00%	122.05	4.54%
	5	121.75	2.64	121.50	100.00%	122.16	0.21%
	6	517.80	24.67	499.55	93.00%	150.78	3.52%
	7	451.90	1.72	426.50	100.00%	121.70	5.62%
	8	311.90	1.28	319.65	100.00%	121.08	-2.48%
	9	77.55	2.81	40.85	100.00%	122.55	47.32%
	10	402.40	1.47	383.15	100.00%	121.27	4.78%
	11	367.00	2.06	360.75	100.00%	121.47	1.70%
	12	646.35	1.44	632.30	99.00%	121.16	2.17%
	13	149.15	25.09	145.05	100.00%	152.28	2.75%
	14	50.55	2.50	60.60	100.00%	122.92	-19.88%
	15	191.90	1.72	252.00	100.00%	124.75	-31.32%

Table A.1: Results obtained with the proposed scheduling heuristic compared with a mathematical model solved with a commercial solver - Case I.

n	v	Heuristic		Exact			$Dev(\%)$
		f_S	$Time(s)$	f_S	$Gap(\%)$	$Time(s)$	
12	1	148.35	120.86	124.85	76.00%	240.28	15.84%
	2	38.50	2.62	55.20	100.00%	122.62	-43.38%
	3	213.50	4.47	258.70	100.00%	124.95	-21.17%
	4	237.00	2.70	210.20	95.00%	123.41	11.31%
	5	314.70	37.53	234.75	100.00%	162.31	25.41%
	6	27.00	120.98	63.14	100.00%	240.28	-133.85%
	7	613.20	27.81	658.20	98.00%	156.30	-7.34%
	8	107.55	5.34	92.25	100.00%	126.34	14.23%
	9	240.80	8.81	275.80	100.00%	129.53	-14.53%
	10	440.10	66.97	479.30	100.00%	194.91	-8.91%
	11	343.85	101.27	343.30	100.00%	226.38	0.16%
	12	773.40	48.53	728.50	99.00%	171.78	5.81%
	13	170.50	6.48	207.65	98.00%	126.86	-21.79%
	14	39.90	120.77	65.70	100.00%	240.27	-64.66%
	15	363.05	3.80	407.90	100.00%	124.06	-12.35%
15	1	137.25	11.02	270.50	100.00%	128.16	-97.09%
	2	110.60	12.84	183.64	94.00%	131.22	-66.04%
	3	560.10	81.70	686.71	100.00%	204.78	-22.60%
	4	286.60	120.78	508.93	96.00%	240.19	-77.58%
	5	712.15	120.92	722.30	100.00%	240.30	-1.43%
	6	111.80	43.89	127.55	100.00%	164.22	-14.09%
	7	588.90	120.78	509.40	96.00%	240.23	13.50%
	8	276.85	8.48	321.10	100.00%	128.02	-15.98%
	9	524.20	93.83	540.92	96.00%	219.39	-3.19%
	10	13.10	20.05	83.85	100.00%	141.39	-540.08%
	11	142.55	120.84	216.35	100.00%	240.22	-51.77%
	12	188.95	20.84	264.23	100.00%	141.84	-39.84%
	13	87.50	121.25	326.30	100.00%	240.23	-272.91%
	14	755.95	4.48	726.81	98.00%	124.12	3.85%
	15	383.55	12.61	458.75	100.00%	132.58	-19.61%
20	1	100.95	121.41	383.73	100.00%	240.33	-280.12%
	2	1007.55	121.62	1720.97	100.00%	240.27	-70.81%
	3	77.50	121.39	257.84	100.00%	240.30	-232.70%
	4	157.00	121.09	414.71	100.00%	240.36	-164.15%
	5	172.10	121.55	594.28	100.00%	240.33	-245.31%
	6	148.65	120.64	660.98	100.00%	240.20	-344.66%
	7	1037.85	55.30	1053.23	99.00%	184.95	-1.48%
	8	339.10	114.02	719.80	100.00%	240.28	-112.27%
	9	413.95	120.75	1032.25	100.00%	240.33	-149.37%
	10	334.90	121.53	846.11	100.00%	240.33	-152.65%
	11	93.85	121.88	212.69	100.00%	240.39	-126.63%
	12	239.90	121.53	778.30	100.00%	240.33	-224.43%
	13	129.40	121.58	476.02	100.00%	240.36	-267.87%
	14	401.00	121.92	944.18	99.00%	240.31	-135.46%
	15	79.95	121.64	272.11	100.00%	240.28	-240.35%
50	1	162.50	124.42	2493.11	100.00%	240.34	-1434.22%
	2	1799.90	125.75	5328.69	100.00%	240.55	-196.05%
	3	118.70	123.62	2718.94	100.00%	240.47	-2190.60%
	4	329.60	124.11	2869.72	100.00%	240.33	-770.67%
	5	548.20	123.80	3333.89	100.00%	240.39	-508.15%
	6	998.90	124.20	4055.48	100.00%	240.47	-305.99%
	7	752.30	123.95	4092.95	100.00%	240.31	-444.06%
	8	1198.20	124.00	3934.11	100.00%	240.42	-228.34%
	9	1147.20	123.91	4069.15	100.00%	240.52	-254.70%
	10	263.10	123.94	3222.74	100.00%	240.28	-1124.91%
	11	198.30	123.89	3104.51	100.00%	240.41	-1465.56%
	12	484.10	125.59	2912.81	100.00%	240.45	-501.70%
	13	168.00	123.77	3338.64	100.00%	240.48	-1887.29%
	14	161.40	124.19	2342.30	100.00%	240.41	-1351.24%
	15	1594.64	55.56	5029.70	100.00%	171.98	-215.41%

Table A.1: Results obtained with the proposed scheduling heuristic compared with a mathematical model solved with a commercial solver - Case I.

n	v	Heuristic		Exact			$Dev(\%)$
		f_S	$Time(s)$	f_S	$Gap(\%)$	$Time(s)$	
100	1	679.34	134.78	7621.90	100.00%	245.58	-1021.96%
	2	695.75	131.94	7984.50	100.00%	242.50	-1047.61%
	3	40.10	130.78	7999.70	100.00%	263.19	-19849.38%
	4	135.70	133.41	7637.30	100.00%	242.09	-5528.08%
	5	126.40	130.39	8697.10	100.00%	240.98	-6780.62%
	6	261.80	137.66	8456.68	100.00%	245.77	-3130.21%
	7	205.40	130.03	8147.39	100.00%	240.80	-3866.60%
	8	16.00	137.47	7207.05	100.00%	241.05	-44944.06%
	9	684.55	135.83	8478.55	100.00%	242.86	-1138.56%
	10	334.90	132.47	8902.80	100.00%	242.14	-2558.35%
	11	477.29	133.72	8087.44	100.00%	247.36	-1594.45%
	12	249.40	130.56	8767.36	100.00%	241.72	-3415.38%
	13	4.40	129.45	5801.95	100.00%	241.00	-131762.50%
	14	378.55	144.03	8294.28	100.00%	245.14	-2091.07%
	15	41.60	129.83	8244.82	100.00%	241.23	-19719.28%

The results presented in Table A.1 were obtained considering an environment without disruptions with setups and processing times varying according to Case I. Considering the deviation of the objective function (***Dev***) in Column 8, we observe that the proposed heuristic found higher total weighted tardiness than the values found by the mathematical model for most instances until 8 products, which is indicated by a positive ***Dev***. Therefore, in these cases, we recommend the mathematical model for resolution. From 10 to 15 products, some instances present lower ***Dev*** while others present high values. We can observe that some results for ***Dev*** are negative, indicating that the objective function found by the proposed heuristic is lower than the value found by the mathematical model. The scheduling heuristic may find lower total weighted tardiness since, for medium to large quantities of products, the mathematical model reaches the time limit with a high optimality gap, as indicated by Column 6. From 20 products, all the ***Dev*** values are negative. In this context, for medium and large product quantities, we indicate the proposed scheduling heuristic for solving the problem.

Considering the optimality gap when solving the scheduling with the mathematical model (Column 6), until 6 products the problem was solved with a gap of 0.00%. The remaining instances present high values for the optimality gap, justifying the use of the proposed heuristic. Until 6 products, we do not observe significant differences in the computational times considering the heuristic and the mathematical model. From 8 products, the heuristic is solved in lower computational times compared to the exact method. The maximum resolution time for the heuristic was 144.03 seconds for an instance considering 100 products, while the maximum resolution time for the mathematical model was 263.19 seconds. We emphasize that this total time was calculated considering the resolution of the short-term production planning model and the scheduling algorithm.

As we can observe in Table A.1, until 6 products the use of the mathematical model is recommended when compared with the proposed heuristic since it finds optimal solutions in low computational times for most instances. Considering the remaining product quantities, the model reaches the computational time limit without finding the optimal solution, whereas the proposed heuristic finds acceptable solutions in low computational times for most instances. In many cases, the proposed heuristic finds better results than the solution obtained by the model, which is represented by the negative deviations. Considering large product quantities, we strongly recommend the use of the proposed heuristic since it finds lower results than the mathematical model in lower computational times.

When solving Case II, see Table A.2, we observe that the mathematical model was able to solve all the instances optimally only for 4 products. From 6 products, we see high optimality gaps, as shown in Column 6, achieving the time limit for most cases. Analyzing the ***Dev*** values for Case II, see Column 8, we have higher negative deviations than Case I, indicating that the proposed scheduling heuristic finds lower total weighted tardiness for scenarios in which the setup times are lower than processing times. We highlight that the instances indicating a deviation of 100.00% are obtained because the mathematical model results in an objective function equal to zero. We recommend the scheduling heuristic from 12 products given that from this product quantity, negative deviations or deviations equal to zero are observed. In the table, "-" indicates the instances with zero total weighted tardiness found by the scheduling heuristic, whereas a positive value was found by the model, making the calculation of ***Dev*** impossible.

Table A.2: Results obtained with the proposed scheduling heuristic compared with a mathematical model solved with a commercial solver - Case II.

n	v	Heuristic		Exact			$Dev(\%)$
		f_S	$Time(s)$	f_S	$Gap(\%)$	$Time(s)$	
4	1	107.25	0.25	40.00	0.00%	1.94	62.70%
	2	603.50	0.12	560.50	0.00%	3.97	7.13%
	3	526.35	0.20	314.75	0.00%	1.02	40.20%
	4	249.00	0.09	230.60	0.00%	3.94	7.39%
	5	156.00	0.11	119.00	0.00%	2.00	23.72%
	6	193.00	0.09	111.45	0.00%	1.00	42.25%
	7	0.00	0.14	0.00	0.00%	0.14	0.00%
	8	66.60	0.08	18.40	0.00%	1.06	72.37%
	9	666.00	0.08	580.50	0.00%	3.06	12.84%
	10	28.50	0.11	11.00	0.00%	1.03	61.40%
	11	25.00	0.17	6.75	0.00%	1.45	73.00%
	12	156.30	0.08	106.80	0.00%	0.89	31.67%
	13	126.70	0.09	91.60	0.00%	0.94	27.70%
	14	271.70	0.11	193.10	0.00%	2.03	28.93%
	15	224.50	0.11	190.20	0.00%	1.67	15.28%
6	1	0.00	0.08	0.00	0.00%	0.31	0.00%
	2	565.60	0.16	483.50	86.00%	120.31	14.52%
	3	595.60	0.31	390.65	80.00%	120.39	34.41%
	4	256.40	0.14	207.10	100.00%	120.33	19.23%
	5	620.85	0.30	454.80	71.00%	120.33	26.75%
	6	0.00	0.20	0.00	0.00%	0.39	0.00%
	7	277.55	0.36	162.65	73.00%	120.36	41.40%
	8	281.80	0.23	203.50	86.00%	120.45	27.79%
	9	94.50	0.42	36.75	0.00%	112.25	61.11%
	10	538.90	0.27	426.20	91.00%	120.23	20.91%
	11	48.10	0.17	0.00	0.00%	12.16	100.00%
	12	29.40	0.11	0.00	0.00%	2.94	100.00%
	13	2.80	0.14	0.00	0.00%	1.58	100.00%
	14	594.00	0.12	475.95	71.00%	120.23	19.87%
	15	18.50	0.28	0.00	0.00%	9.00	100.00%
8	1	330.80	0.75	298.55	91.00%	120.25	9.75%
	2	221.10	0.33	201.30	98.00%	120.38	8.96%
	3	135.00	0.84	66.60	100.00%	120.23	50.67%
	4	262.90	0.44	199.60	100.00%	120.38	24.08%
	5	0.00	0.52	0.00	0.00%	1.06	0.00%
	6	128.80	0.38	69.15	100.00%	120.20	46.31%
	7	810.90	0.73	722.10	88.00%	120.50	10.95%
	8	426.45	0.69	385.20	100.00%	120.38	9.67%
	9	70.80	0.44	6.80	100.00%	120.39	90.40%
	10	27.60	0.36	0.00	0.00%	14.52	100.00%
	11	333.50	0.52	249.75	100.00%	120.27	25.11%
	12	151.50	0.36	106.50	85.00%	120.20	29.70%
	13	284.80	1.09	246.80	100.00%	120.36	13.34%
	14	30.80	0.53	0.00	0.00%	2.66	100.00%
	15	0.00	0.69	0.00	0.00%	0.39	0.00%

Table A.2: Results obtained with the proposed scheduling heuristic compared with a mathematical model solved with a commercial solver - Case II.

n	v	Heuristic		Exact			$Dev(\%)$
		f_S	$Time(s)$	f_S	$Gap(\%)$	$Time(s)$	
10	1	0.00	1.16	0.00	0.00%	19.45	0.00%
	2	161.70	1.08	107.90	99.00%	120.39	33.27%
	3	129.20	1.23	119.59	100.00%	120.31	7.44%
	4	561.60	1.69	525.45	98.00%	120.38	6.44%
	5	178.10	1.47	126.50	100.00%	120.20	28.97%
	6	219.50	1.33	163.00	72.00%	120.44	25.74%
	7	35.40	1.23	0.00	0.00%	87.33	100.00%
	8	188.90	0.89	178.25	81.00%	120.30	5.64%
	9	0.00	0.73	0.00	0.00%	1.66	0.00%
	10	616.75	1.14	441.22	100.00%	120.38	28.46%
	11	71.50	0.75	19.00	18.00%	120.41	73.43%
	12	70.80	2.45	74.00	88.00%	122.02	-4.52%
	13	594.65	0.94	499.20	94.00%	120.31	16.05%
	14	319.90	1.06	230.70	100.00%	120.50	27.88%
	15	0.00	1.06	0.00	0.00%	1.38	0.00%
12	1	266.70	1.27	222.78	100.00%	120.36	16.47%
	2	30.65	1.42	33.60	100.00%	120.42	-9.62%
	3	64.90	0.84	33.15	100.00%	120.47	48.92%
	4	0.00	0.83	0.00	0.00%	1.53	0.00%
	5	138.00	1.23	108.35	100.00%	120.55	21.49%
	6	0.00	1.03	0.00	0.00%	12.55	0.00%
	7	618.00	1.28	562.23	95.00%	120.42	9.02%
	8	0.00	1.11	0.00	0.00%	5.20	0.00%
	9	0.00	1.16	0.00	0.00%	12.38	0.00%
	10	213.30	1.12	274.05	91.00%	120.38	-28.48%
	11	102.70	1.12	72.82	92.00%	120.47	29.09%
	12	66.30	1.11	57.49	100.00%	120.31	13.29%
	13	245.50	1.14	289.18	100.00%	120.42	-17.79%
	14	124.20	0.89	114.78	100.00%	120.31	7.58%
	15	0.00	1.22	0.00	0.00%	102.80	0.00%
15	1	191.90	1.42	396.00	100.00%	120.41	-106.36%
	2	332.70	1.66	464.23	100.00%	120.39	-39.53%
	3	0.00	1.55	5.03	100.00%	120.58	-
	4	837.80	1.62	1058.04	99.00%	120.78	-26.29%
	5	133.60	1.38	326.48	100.00%	120.52	-144.37%
	6	83.30	1.41	129.33	99.00%	120.52	-55.26%
	7	37.60	1.45	17.97	100.00%	120.38	52.21%
	8	245.70	1.81	296.59	92.00%	120.53	-20.71%
	9	528.55	3.88	781.36	100.00%	122.84	-47.83%
	10	12.30	1.22	22.40	100.00%	120.36	-82.11%
	11	240.80	1.52	299.65	100.00%	120.48	-24.44%
	12	0.00	1.52	0.00	0.00%	101.88	0.00%
	13	37.80	1.31	42.00	77.00%	120.47	-11.11%
	14	0.00	1.42	8.70	100.00%	120.50	-
	15	7.80	1.50	8.20	100.00%	120.48	-5.13%
20	1	59.50	2.17	389.52	100.00%	120.58	-554.66%
	2	108.30	2.09	515.09	100.00%	120.77	-375.61%
	3	317.30	1.73	754.18	100.00%	120.70	-137.69%
	4	113.10	1.77	588.91	99.00%	120.64	-420.70%
	5	143.80	1.55	585.54	100.00%	120.69	-307.19%
	6	160.10	1.53	478.81	97.00%	120.91	-199.07%
	7	59.20	1.59	377.67	100.00%	120.83	-537.96%
	8	1107.00	1.59	1381.13	100.00%	120.73	-24.76%
	9	0.00	1.44	94.95	100.00%	120.61	-
	10	20.10	1.47	59.53	95.00%	120.62	-196.17%
	11	174.00	1.58	499.88	100.00%	120.78	-187.29%
	12	211.20	1.78	587.21	99.00%	120.72	-178.04%
	13	12.80	1.58	19.24	100.00%	120.70	-50.31%
	14	68.00	1.44	162.75	100.00%	120.64	-139.34%
	15	0.00	1.52	67.50	100.00%	120.72	-

Table A.2: Results obtained with the proposed scheduling heuristic compared with a mathematical model solved with a commercial solver - Case II.

n	v	Heuristic		Exact			$Dev(\%)$
		f_S	$Time(s)$	f_S	$Gap(\%)$	$Time(s)$	
50	1	404.20	8.34	3580.59	100.00%	126.47	-785.85%
	2	425.40	8.41	3016.69	100.00%	126.19	-609.14%
	3	0.00	9.75	2326.94	100.00%	127.89	-
	4	301.60	9.64	2968.97	100.00%	126.94	-884.41%
	5	25.10	10.52	1100.42	100.00%	127.95	-4284.14%
	6	6.80	10.66	3327.74	100.00%	127.34	-48837.35%
	7	453.10	8.00	2984.01	100.00%	124.30	-558.58%
	8	51.90	11.88	2254.42	100.00%	126.55	-4243.78%
	9	0.00	11.69	2246.48	100.00%	126.94	-
	10	54.10	10.97	2510.40	100.00%	125.83	-4540.30%
	11	3.90	9.89	2348.40	100.00%	125.45	-60115.38%
	12	354.00	10.95	2828.97	100.00%	126.12	-699.14%
	13	3.50	12.06	2196.26	100.00%	127.45	-62650.29%
	14	39.80	12.92	1379.68	100.00%	126.47	-3366.53%
	15	16.80	11.75	1817.51	100.00%	126.67	-10718.51%
100	1	8.00	27.64	3767.45	100.00%	136.73	-46993.13%
	2	201.20	30.20	6549.15	100.00%	133.78	-3155.04%
	3	15.40	25.88	3850.15	100.00%	135.22	-24900.97%
	4	13.50	23.00	4694.00	100.00%	129.97	-34670.37%
	5	0.00	26.61	5022.10	100.00%	142.23	-
	6	87.90	29.48	3956.35	100.00%	146.83	-4400.97%
	7	23.20	30.12	5172.25	100.00%	136.72	-22194.18%
	8	7.50	25.05	4608.00	100.00%	135.31	-61340.00%
	9	0.00	25.28	3417.45	100.00%	133.06	-
	10	169.30	28.19	5880.70	100.00%	135.36	-3373.54%
	11	9.30	28.34	6033.90	100.00%	137.05	-64780.65%
	12	8.00	25.08	4561.15	100.00%	132.97	-56914.38%
	13	0.00	26.03	4790.10	100.00%	132.81	-
	14	0.00	26.78	3810.00	100.00%	134.83	-
	15	26.30	24.58	4828.40	100.00%	132.98	-18258.94%

Appendix B

Results of the hierarchical approach

Table B.1 shows the average results obtained with the hierarchical approach for Cases I and II. In this context, the short-term production planning and scheduling problems are solved without considering disruptions. Based on the solution, we perform simulations of the failures. In this work, we used the hierarchical approach to compare the strategies proposed in Chapter 4, Section 4.5.2. The first column exhibits the scenario indicating the duration of the failures in the simulation, while Column 2 presents the number of products considered. Columns 3, 4, 5, and 6 refer to Case I. Column 3 shows the expected total weighted tardiness, represented by f_S. Columns 4 and 5 show the probabilities of infeasibility for machines 1 and 2, respectively. Finally, Column 6 shows the average idle hours for each instance set ($\boldsymbol{Idle}$). Columns 7, 8, 9, and 10 have the same meaning as columns 3, 4, 5, and 6, but refer to Case II.

As shown in Table B.1, we observe an increase in the f_S values for Cases I and II when going from Scenario 1 to Scenario 3, indicating that as the duration of the failures grows, we expect an increase in the total weighted tardiness. Regarding the probabilities of infeasibility, we observe high values for Case I. When solving the problem using the hierarchical approach, we did not insert slacks at the capacity constraint of the short-term production planning problem. Then, the final solution is not prepared for disruptions, occupying all the available capacity with production and setup times. When simulating the failures, the sequences present high makespan values because of the failures durations and, consequently, higher probabilities of infeasibility. For Case II, when processing times are greater than setup times, the probabilities of infeasibility are lower when compared with Case I, achieving values near to 0.00% for large product quantities.

Regarding the idle times, we observe high values for Case I when considering 4 products for all the scenarios analyzed, indicating that, for some instances, the processing of the demands did not use all the available capacity. For the remaining product quantities, low average idle times were obtained. When analyzing Case II, we note in Column 10 that the hierarchical approach results in high idle times for all the instance sets and scenarios, especially when analyzing 100 products. In Case II, we have several instances presenting makespans lower than the production capacity while producing all the demand. Therefore, we expect that, even with disruptions, Case II will result in low probabilities of infeasibility due to the failures and greater idle times.

Table B.1: Average results obtained with the hierarchical approach considering Case I and Case II.

Scenario	n	Case I				Case II			
		f_S	P_1	P_2	$Idle$	f_S	P_1	P_2	$Idle$
1	4	318.08	60.00%	56.67%	21.74	277.70	27.47%	25.73%	29.81
	6	424.40	67.07%	91.47%	2.97	311.36	24.47%	11.07%	39.53
	8	406.49	83.40%	90.20%	0.65	257.05	15.73%	24.07%	26.95
	10	372.29	87.20%	85.07%	0.53	255.59	19.27%	14.67%	29.21
	12	350.43	89.80%	91.60%	0.32	162.56	6.53%	6.87%	31.23
	15	409.81	90.33%	88.07%	0.44	222.93	12.07%	13.00%	31.97
	20	415.29	92.33%	92.87%	0.05	216.25	6.93%	10.60%	31.44
	50	904.59	90.93%	92.20%	0.01	209.14	0.00%	0.07%	35.09
	100	643.41	72.93%	70.60%	2.15	67.76	0.00%	0.00%	56.03
Average		471.64	81.56%	84.30%	3.21	220.04	12.50%	11.79%	34.58
SD		186.60	11.90%	12.46%	7.02	71.53	10.02%	9.07%	8.82
2	4	357.81	61.67%	58.33%	19.14	320.14	41.60%	34.27%	27.16
	6	487.13	75.27%	92.47%	2.14	353.68	27.80%	14.40%	34.64
	8	478.13	86.93%	91.00%	0.36	325.45	31.93%	32.53%	22.69
	10	457.98	89.47%	87.47%	0.29	321.38	29.13%	33.20%	25.09
	12	448.50	90.33%	90.40%	0.18	229.21	28.40%	14.73%	26.03
	15	530.65	90.13%	89.33%	0.23	298.50	20.73%	19.87%	26.35
	20	577.23	90.27%	90.60%	0.03	320.32	18.00%	22.47%	25.75
	50	1304.78	88.27%	89.27%	0.01	403.43	6.07%	9.13%	28.16
	100	1364.65	78.33%	77.87%	1.45	220.45	0.07%	0.07%	48.87
Average		667.43	83.41%	85.19%	2.65	310.29	22.64%	20.07%	29.42
SD		383.27	9.86%	10.94%	6.23	56.89	13.03%	11.81%	7.99
3	4	387.95	62.93%	59.80%	17.61	357.05	45.33%	40.80%	26.51
	6	523.77	80.13%	89.67%	1.75	389.30	33.87%	19.60%	32.15
	8	529.24	87.27%	89.93%	0.29	377.56	39.33%	38.93%	20.37
	10	520.11	89.27%	87.13%	0.23	375.55	35.33%	40.67%	22.63
	12	522.17	89.73%	89.33%	0.13	281.52	36.00%	23.60%	23.24
	15	617.23	89.87%	88.27%	0.17	366.13	29.00%	28.33%	23.31
	20	692.10	89.40%	89.33%	0.01	411.25	25.80%	32.40%	22.84
	50	1593.38	87.73%	88.07%	0.00	598.67	17.40%	20.87%	24.74
	100	1902.31	80.73%	79.53%	1.19	434.65	2.47%	2.33%	44.60
Average		809.81	84.12%	84.56%	2.38	399.08	29.39%	27.50%	26.71
SD		543.56	8.79%	9.82%	5.75	85.90	12.91%	12.55%	7.49