Simplified Implementation of Energy-Aware Production Scheduling in Job Shops

Vom Fachbereich Maschinenbau
an der Technischen Universität Darmstadt
zur Erlangung des Grades eines Doktor-Ingenieurs (Dr.-Ing.)
genehmigte

DISSERTATION

vorgelegt von

Benedikt Emanuel Grosch, M.Sc.
aus Witten

Berichterstatter:	Prof. Dr.-Ing. Matthias Weigold
Mitberichterstatter:	Prof. Dr. Christoph Glock
Tag der Einreichung:	15. August 2023
Tag der mündlichen Prüfung:	21. November 2023

Darmstadt 2023
D17

Grosch, Benedikt Emanuel: Simplified Implementation of Energy-Aware Production Scheduling in Job Shops
Darmstadt, Technical University Darmstadt
Year of publication of the dissertation on TUprints: 2023
URN: urn:nbn:de:tuda-tuprints-264591
Day of the oral examination: 21.11.2023

Published under CC-BY-SA 4.0 International

Foreword of the Editor

The pressing challenge of climate change requires urgent global action to reduce greenhouse gas emissions and curb global warming. To tackle this challenge, the German Federal Government has set the ambitious goal of achieving climate neutrality by 2045. To achieve this goal, the government has begun implementing the Easter Package of 2022, which outlines a strategic plan to triple the expansion speed of renewable energy.

Industrial processes and production plants are critical for improving energy efficiency and flexibility. Tapping this potential can greatly speed up achieving set goals. The Industry 4.0 paradigm enables detailed planning and control of manufacturing processes, which can lead to improved energy efficiency and flexibility.

Benedikt Grosch's thesis is dedicated to improving energy efficiency and flexibility through technological innovation in the field of energy-aware production scheduling. The author proposes a scheduling system architecture supported by a practical implementation procedure to simplify the implementation of energy-aware production scheduling systems. This combination allows manufacturing companies to adopt energy-aware practices in their real-world production systems. The modular architecture of the energy-aware production scheduling system ensures adaptability across diverse production systems.

In conclusion, this work provides an approach for industries to meet the demands of a changing climate and energy scenario. The thesis not only contributes to the academic discourse but also offers practical solutions for industries to contribute to sustainable and energy-efficient practices.

Darmstadt, December 2023 Prof. Dr.-Ing Matthias Weigold

Acknowledgements

Writing this thesis has been a journey filled with challenges and growth. I owe a debt of gratitude to the many individuals and organizations who have played a crucial role in its completion.

I extend my profound appreciation to the Technical University of Darmstadt and, the Institute for Production Management, Technology, and Machine Tools (PTW). The academic environment provided by PTW and the research group Energy Technologies and Applications in Production (ETA) has been instrumental in shaping my research by fostering an atmosphere of intellectual curiosity. The experience working at ETA was transformative, contributing not only to my professional development but also to my personal growth.

I could not have completed this research without the guidance, expertise, and support of my supervisor, Prof. Dr.-Ing. Matthias Weigold. Additionally, I am grateful for the valuable insights provided by my co-supervisor, Prof. Dr. Christoph Glock.

This research was enabled by the funding provided by the research projects PHI-Factory and KI4ETA. These projects were supported by the German Federal Ministry for Economic Affairs and Climate Action (BMWK) and supervised by Projektträger Jülich (PtJ). I would like to express my sincere gratitude to all our industry and research partners for their valuable cooperation during these projects.

I want to express my appreciation to my colleagues in the ETA research group. I cannot thank Daniel Fuhrländer-Völker enough for always being available to have engaging discussions with me. Thomas Kohne has provided invaluable feedback and motivation. I am grateful to Niklas Panten for inspiring my research and for his continued guidance. Special thanks also go to Andreas Wächter, Ghada Elserafi, Fabian Borst, and Benjamin Arzt for their support and assistance during experiments and technical challenges. I would also like to express my gratitude to my student assistant, Jerome Stock, whose help and dedication have been invaluable almost since day one.

I would like to convey my gratitude to Stefan Seyfried and my mother, Heike Grosch, for their valuable feedback and proofreading, which significantly improved the quality of this thesis.

Finally, I extend my deepest thanks to my family for their unwavering support and belief in me, even during the most challenging times. Your encouragement has been a constant source of strength.

Darmstadt, December 2023 Benedikt Emanuel Grosch

Abstract

Climate change presents a pressing global challenge, necessitating urgent actions to reduce greenhouse gas emissions and limit global warming. As a significant energy consumer, the industrial sector plays a crucial role in advancing sustainable practices and mitigating the impact of climate change. In this context, this thesis focuses on developing an implementation procedure and an energy-aware production scheduling system architecture to optimize production schedules while considering production-related and energy-related objectives.

The research goal of this thesis is to simplify the implementation of energy-aware production scheduling systems in real production systems. To achieve this, the thesis addresses three key research areas: (1) to find whether the absence of standardized procedures and architectures hinders the implementation of energy-aware production scheduling systems for job shops, (2) to propose a standardized and partially automated implementation procedure for energy-aware production scheduling, and (3) to design an architecture supporting the implementation procedure.

The proposed implementation procedure includes a structured system configuration and deployment approach, ensuring alignment with stakeholder requirements. It comprises three phases: discovery and planning, development and configuration, and testing and deployment. The energy-aware production scheduling system architecture implements a cyber-physical production system with a virtual representation of the actual production system. The architecture incorporates the Non-Dominated Sorting Genetic Algorithm-II optimization algorithm with a graph-based solution encoding and the production system environment, which adapts to specific production system requirements. An energy model parameter estimation module supports the automatic configuration of production machine energy models.

Evaluation of the proposed concepts in the ETA Research Factory demonstrates the system's success in reducing energy consumption while maintaining production-related objectives. The energy-aware production scheduling system achieves average energy cost savings of 13 % and 18 % compared to traditional Shortest Processing Time dispatching rules while slightly improving or marginally decreasing production-related performance, respectively.

This thesis contributes to the field of energy-aware production scheduling by providing an implementation procedure and an adaptable architecture that fulfills the set requirements and success criteria. The proposed concepts offer practical solutions for adopting energy-aware production scheduling systems in industrial settings, promoting environmentally conscious and economically viable production practices. The thesis also identifies areas for improvement and future research, ensuring the continuous development of energy-efficient and sustainable manufacturing processes.

Keywords: Demand Response, Energy-Efficiency, Energy-Flexibility, Cyber-Physical Production System, Production Machine Energy Model, Implementation Procedure, Scheduling System Architecture

Kurzfassung

Der Klimawandel stellt eine globale Herausforderung dar, die dringende Maßnahmen zur Verringerung der Treibhausgasemissionen und zur Begrenzung der globalen Erwärmung erforderlich macht. Der Industriesektor spielt als bedeutender Energieverbraucher eine entscheidende Rolle bei der Umsetzung einer nachhaltigen Wirtschaft und der Minimierung der Auswirkungen des Klimawandels. In diesem Zusammenhang konzentriert sich diese Arbeit auf die Entwicklung eines Implementierungsverfahrens und einer Systemarchitektur für die energiebewusste Produktionsplanung unter Berücksichtigung von produktions- und energiebezogenen Zielfunktionen.

Das Forschungsziel dieser Arbeit ist, die Implementierung von energiebewussten Produktionsplanungssystemen in realen Produktionssystemen zu vereinfachen. Die Arbeit befasst sich mit drei Forschungsschwerpunkten, um (1) herauszufinden, ob das Fehlen standardisierter Verfahren und Architekturen die Implementierung energiebewusster Produktionsplanungssysteme in der Werkstattfertigung behindert, (2) ein standardisiertes und teilweise automatisiertes Implementierungsverfahren für die energiebewusste Produktionsplanung vorzuschlagen und (3) eine Systemarchitektur zu entwerfen, die das Implementierungsverfahren unterstützt.

Das vorgeschlagene Implementierungsverfahren stellt einen strukturierten Ansatz für die Konfiguration und -einführung eines Produktionsplanungssystems bereit, der eine Anpassung an die Anforderungen der Beteiligten gewährleistet. Das Verfahren besteht aus drei Phasen: Untersuchung und Planung, Entwicklung und Konfiguration sowie Test und Umsetzung. Die Architektur des energiebewussten Produktionsplanungssystems ermöglicht es, reale Produktionssysteme als cyber-physisches Produktionssystem abzubilden. Für die Optimierung nutzt das Produktionsplanungssystem den Non-Dominated Sorting Genetic Algorithm-II-Optimierungsalgorithmus mit einer graphenbasierten Codierung für die Lösungen. Die Produktionssystemumgebung, als zweiter Bestandteil der Architektur kann an die spezifischen Anforderungen des Produktionssystems angepasst werden. Weiterhin umfasst die Systemarchitektur ein Modul, das eine automatisierte Parameteridentifikation für Energiemodelle von Produktionsmaschinen ermöglicht.

Die Bewertung der vorgeschlagenen Konzepte in der ETA-Forschungsfabrik zeigt, dass das System die Energiekosten der Produktion unter Beibehaltung produktionsbezogener Ziele senken kann. Das Produktionsplanungssystem erzielt in den durchgeführten Experimenten durchschnittliche Kosteneinsparungen von 13 % beziehungsweise 18 % im Vergleich zur Shortest Processing Time Zuordnungsregel, während die produktionsbezogene Zielfunktion leicht verbessert oder geringfügig verschlechtert wird.

Diese Arbeit leistet einen Beitrag zur Forschung im Bereich der energiebewussten Produktionsplanung, indem sie ein Implementierungsverfahren und eine anpassungsfähige Systemarchitektur bereitstellt, die die festgelegten Anforderungen und Erfolgskriterien erfüllt. Die vorgeschlagenen Konzepte bieten Lösungen für die Einführung energiebewusster Produktionsplanungssysteme in der Industrie und fördern umweltbewusste und wirtschaftlich tragfähige Produktionsverfahren. Außerdem zeigt diese Arbeit Bereiche für zukünftige Forschung auf, um die Weiterentwicklung der energiebewussten Produktionsplanung voranzutreiben.

Stichwörter: Demand Response, Energieeffizienz, Energieflexibilität, Cyber-Physisches Produktionssystem, Energiemodell für Produktionsmaschinen, Implementierungsverfahren, Produktionsplanungs-Systemarchitektur

Contents

List of Figures

List of Tables

List of Symbols

Notation	Description	Symbol
Indices		
α	Index of an estimated variable in a multivariate regression model $\alpha \in 1, 2, \dots, \alpha_{\max}$	
f	Index of a pareto-optimal front $f \in \mathbb{N}$	
g	Index of a generation of solutions, $g \in \{1, 2, \dots, g_{\max}\}$	
h	Index of an event on one machine with $h \in \mathbb{N}$	
i	Index of a processing job with $i \in \{1, 2, \dots, i_{\max}\}$	
k	Index of a produced item for operation o of job i with $k \in \left\{1, 2, \dots, q_i^{\text{order}}\right\}$	
m	Index of a machine with $m \in \{1, 2, \dots, m_{\max}\}$	
o	Index of an operation with $o \in \{1, 2, \dots, o_{\max}\}$	
φ	Index of a regression parameter in a multiple regression model	
src	Index of an energy form with $src \in \mathscr{E}$	
w	Index of a solution to an optimization problem, $w \in \{1, 2, \dots, w_{\max}\}$	
Objective Criteria		
ERC	Energy related cost of a production schedule	€
$MKSP$	Makespan of a production schedule	s
Time and Events		
$c_{m,h}$	Ending time t of event h on machine m	s
e_m	Set of events on machine m	
$h_{m,\max}$	Total number of events on machine m	
h'	Index of current event h being processed	
h''	Index of next event h to be processed	
$s_{m,h}$	Starting time t of event h on machine m	s
t	Time	s
t_0	Starting time of the optimization	s
$t_{\max}$	Ending time of the optimization	s
y_m^*	Clock time until the next event starts on machine m	s
$y_{m,h}$	Clock time until an event h starts on machine m	s

Notation	Description	Symbol
Optimization Algorithm		
γ	Number of genes in a chromosome	
F_f	Set of solutions in the pareto-optimal front f	
$g_{\max}$	Number of generations before optimization terminates	
κ	Crossover rate in percent of the genes over the entire population	%
Λ_t	Reward or values of objective criteria for time t	
n^{genes}	Number of selected genes for crossover or mutation	
$n^{\text{solutions}}$	Number of selected solutions for crossover or mutation	
ω	Mutation rate in percent of the genes over the entire population	%
ψ	Distribution of mutation and crossover rate between the chromosomes	
S_g	Set of solutions in the generation g	
S_g^{offspr}	Set of offspring solutions of the generation g	
S_g^{parent}	Set of parent solutions of the generation g	
Θ_t	Actions/Inputs to the environment generated by the optimization algorithm for time t	
$w_{\max}$	Number of solutions in a generation (population)	
Ξ_t	State (observations) of the environment for time t	
Jobs, Operations and Machines		
$d_{i,o}^{\text{proc}}$	Duration of processing for operation o of job i	s
$d_{i,o}^{\text{setup}}$	Duration of setup time for operation o of job i	s
$i_{\max}$	Total number of jobs to be scheduled	
$I_{i,o,m,h}^{\text{op}}$	Binary matrix that is 1 if operation o of job i should be processed on machine m as event h	
$I_{m,h}^{\text{setup}}$	Binary variable which is 1 if setup is required before processing event h on machine m	
I_m^{unique}	Binary parameter which is 1 if machine m can only process a single type of job at once	
J	Set of $i_{\max}$ jobs	
M	Set of $m_{\max}$ machines in a factory	
$m_{\max}$	Total number of machines in the factory	
O_i	Set of $o_{\max}$ operations o in job i	

Notation	Description	Symbol
$o_{i,\max}$	Total number of operations to be scheduled for job i	
q_m^{capa}	Capacity (number of parts processed in parallel) of machine m	pcs
$q_{m,h}^{\text{concurrent}}$	Number of concurrent operations on machine m after the operation belonging to event h is added.	pcs
q_i^{order}	Number of customer orders for job i	pcs
$q_{i,o,t}^{\text{stored}}$	Number of stored parts for operation o of job i at time t	pcs
Energy		
a_t^{heater}	Boolean variable for time t, 1 if cleaning medium tank heater is active	
a_t^{op}	Boolean variable for time t, 1 if production resource is in operational state	
a_t^{st}	Boolean variable for time t, 1 if production resource is in standby state	
a_t^{wk}	Boolean variable for time t, 1 if production resource is in working state	
$\beta_{\text{M}}^{\text{c}}$	Regression parameter representing heat capacity of a machine	
β^{el}	Regression parameter for overall electric energy	
$\beta_{\text{op}}^{\text{el}}$	Regression parameter for electric power during operational state	W
$\beta_{\text{proc}}^{\text{el}}$	Regression parameter for electric power due to processing	W
$\beta_{\text{st}}^{\text{el}}$	Regression parameter for electric power during standby state	W
$\beta_{\text{cool}}^{\text{th}}$	Regression parameter representing the thermal transfer coefficient between machine and coolant	
$\beta_{\text{heater}}^{\text{th}}$	Regression parameter representing cleaning machine tank heater power	W
$\beta_{\text{env}}^{\text{th}}$	Regression parameter representing the thermal transfer coefficient between machine and environment	
$\beta_{\text{spray}}^{\text{th}}$	Regression parameter representing the thermal transfer coefficient between the cleaning medium and the environment	W
$\beta_{\text{wp}}^{\text{th}}$	Regression parameter representing the thermal transfer coefficient between the cleaning medium and the workpiece	W

Notation	Description	Symbol
$\beta^{\mathrm{el}}_{\mathrm{wk}}$	Regression parameter for electric power during working state	W
$\mathscr{E}$	Set of all energy forms	
$I^{\mathrm{tank}}_{\mathrm{lower}}$	Parameter setting the lower hysteresis temperature for the cleaning medium tank	
$I^{\mathrm{tank}}_{\mathrm{upper}}$	Parameter setting the upper hysteresis temperature for the cleaning medium tank	
P	Power	W
$P^{\mathrm{el}}_{m,t}$	Electric power of machine m at time t	W
P^{el}_{t}	Electric power at time t	W
$P^{\mathrm{el,CM}}_{t}$	Electric power of electric cleaning machine at time t	W
$P^{\mathrm{th,cool}}_{t}$	Thermal power removed by cooling system at time t	W
$P^{\mathrm{th,heater}}_{\mathrm{nominal}}$	Nominal thermal power of the cleaning machine tank heater	W
$P^{\mathrm{th,heater}}_{t}$	Thermal power generated by cleaning machine tank heater at time t	W
$P^{\mathrm{th,env}}_{t}$	Thermal power dissipated to the environment at time t	W
$P^{\mathrm{th,spray}}_{t}$	Thermal power dissipated to the environment due to cleaning medium movement at time t	W
$P^{\mathrm{th,wp}}_{t}$	Thermal power transferred to workpieces during a cleaning process at time t	W
T^{cool}_{t}	Average coolant temperature at time t	°C
T^{env}_{t}	Average environment temperature at time t	°C
T^{M}_{t}	Average machine temperature at time t	°C
U	Inner energy of a system	
z^{proc}_{src}	Generic machine and process dependent parameter for energy form *src*	

Others

α_{max}	Total number of estimated variables in a multivariate regression model	
$\beta_{\alpha,\varphi}$	Regression parameter φ for the estimated variable α	
$\epsilon_{\alpha,t}$	Error of a regression model for the estimated variable α at time t	
G	A very large number	

Notation	Description	Symbol
Γ	Set of estimated variables in a multivariate regression model	
Φ	Set of regression parameters in a multiple regression model	
$X_{\alpha,\varphi,t}$	Matrix of independent variables for a multivariate regression model at time t	
$y_{\alpha,t}$	Estimated variable α in the set of estimated variables Γ of a multivariate regression model at time t	

Acronyms

API	Application Programming Interface
CHP	Combined Heat and Power Unit
CNC	Computerized Numerical Control
CPLEX	IBM ILOG CPLEX Optimization Studio
DC	Direct Current
ENTSO-E	European Network of Transmission System Operators for Electricity
ERP	Enterprise Resource Planning
ETA Research Factory	Energy Technologies and Applications Research Factory
HVFA	High-Volume Fly Ash
IIC	Industry IoT Consortium
IICF	Industrial Internet of Things Connectivity Framework
IMF	International Monetary Fund
JSON	Javascript Object Notation
MAE	Mean Absolute Error
MES	Manufacturing Execution System
Modbus TCP	Modbus via TCP/IP
MRP	Material Requirement Planning
MRP II	Manufacturing Resource Planning
NSGA-II	Non-Dominated Sorting Genetic Algorithm-II
OPC UA	Open Plattform Communications Unified Automation
PLC	Programmable Logic Controller
RAMI 4.0	Reference Architecture Model Industrie 4.0
ReST	Representational State Transfer
RMSE	Root-Mean-Squared Error
SPT	Shortest Processing Time
SRU	Sachverständigenrat für Umweltfragen
URL	Unique Resource Locator

1. Introduction

Climate change is one of the most critical global challenges facing humankind. As Joe Biden, President of the United States of America, said in a speech at the *2021 Leaders Summit on Climate*:

> "To protect livelihoods and keep global warming at a maximum of 1.5 °C, we must get on the path now!"

The yearly *State of the Global Climate* report by the World Meteorological Organization shows that all major global warming indicators are increasing. The years between 2015 to 2022 were the eight warmest years on record. For example, in 2022, the global mean temperature was 1.15 °C above the 1850 – 1900 baseline (World Meteorological Organization, 2023, p. 7). This places 2022 as the fifth or sixth warmest year on record (World Meteorological Organization, 2023, p. 8). The International Monetary Fund (IMF) also sees climate change and the resulting exacerbation of natural disasters as a relevant risk factor for economic development, especially for smaller economies (International Monetary Fund, 2021, p. 15, p. 27).

Climate Change endangers our society, and urgent action is existential to reduce its impact on our living standards. The United Nations have set sustainable development goals to counteract this impact (United Nations, 2015); however, a significant and fast overall reduction of greenhouse gas emissions is required to reach the 1.5 °C goal (Intergovernmental Panel on Climate Change, 2018, p. 32). The German example further illustrates the urgent need for action: the Sachverständigenrat für Umweltfragen (SRU) states that the German CO_2 budget would be used up in 2038 assuming a linear reduction of emissions (Hornberg et al., 2020, p. 10). Therefore, the German government plans to reduce greenhouse gas emissions by 55 % by 2030 compared to 1990 (Bundesministerium für Umwelt, 2016, p. 28). In the United States of America, the government presented similar ideas with a planned reduction of greenhouse gas emissions by 50 % – 52 % below 2005 levels by 2030 (The White House, 2021). However, Kobiela et al. (2020) analyze the German targets and argue that these reductions still fall short of what is actually required.

There are two primary paths to reduce emissions and achieve the 1.5 °C target: reducing energy consumption through energy efficiency measures and replacing greenhouse gas-emitting primary energy sources. As Benndorf et al. (2014, p. 90) explain, renewable electric energy generation in combination with the electrification of fossil energy consumers can help to achieve the latter. Green synthetic fuels must be used instead of fossil fuels where electrification is impossible (Benndorf et al., 2014, p. 90).

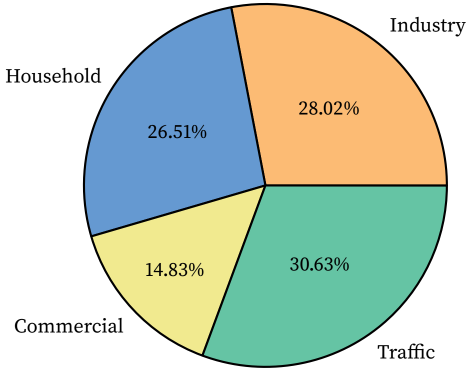


Figure 1.1: Share of industrial final energy consumption in Germany (own figure; Arbeitsgemeinschaft Energiebilanzen e.V., 2020).

As a very large final energy consumer, the industrial sector can bring vital contributions to both paths. The industrial share of final energy consumption in Germany in 2019 was approximately 28 %, as shown in Figure 1.1 (Arbeitsgemeinschaft Energiebilanzen e.V., 2020). The industrial sector can directly contribute to energy efficiency improvements by investing in energy efficiency measures. In Germany the government encourages green investments in the industrial sector through tax reform, the extension of emissions trading, and public subsidy programmes (Bundesministerium für Umwelt, 2019, p. 88). The *World Economic Outlook* mentions the importance of investment in green technologies to improve future development and suggests carbon pricing as a possible measure to incentivize investment by internalizing the cost of environmental damage from using fossil fuels (International Monetary Fund, 2021, p. 21). Energy efficiency technologies, such as waste heat reduction and substituting high-energy processes, are crucial to achieving the emission reduction targets in the industrial sector (Bundesministerium für Umwelt, 2019, p. 89).

Concurrently the industrial sector can also facilitate the replacement of greenhouse gas emitting primary energy sources by improving the integration of renewable energy sources into the power grid through demand-side integration (Bundesministerium für Umwelt, 2019, p. 90; Walther et al., 2022). Demand response is part of demand-side integration and describes measures like the adaptation of electricity demand to available generation to better handle the volatile nature of renewable energy sources (Walther et al., 2022) – this improves the integration of renewable energy sources into the power grid. The term demand-side integration covers the whole range from long-term improvements in energy efficiency, to short-term demand response measures (Walther et al., 2022). The necessity for demand response is explicitly illustrated by increasing re-dispatch costs (Bundesnetzagentur, 2022) to stabilize the electric power grid.

In addition to the trend toward energy efficiency and demand-side integration, many other trends influence developments in the industrial sector. Industry 4.0 and smart manufacturing are two of these trends (Kang et al., 2016; Lasi et al., 2014). Smart manufacturing describes manufacturing systems that can automatically adapt to complicated situations (Kang et al., 2016). In contrast, Industry 4.0 describes the paradigm shift toward digitalization and integrating smart devices into manufacturing processes (Lasi et al., 2014). Among other improvements, this leads to the more convenient collection and higher availability of data about production processes (Mohamed et al., 2019). As a result, Mohamed et al. (2019) identify many areas where energy efficiency could benefit from Industry 4.0. Other trends, such as shorter development periods, more individualized products, higher flexibility, and decentralization, accelerate the trend toward Industry 4.0 (Lasi et al., 2014). Recently direct applications of artificial intelligence are also gaining more traction (Groombridge, 2021). Researchers and practitioners agree that Industry 4.0 is an enabler for sustainable manufacturing (Bunse et al., 2011).

Considering climate change as a critical global challenge and acknowledging the aforementioned trends, this thesis aims to simplify the implementation of energy-aware production scheduling systems. Energy-aware production scheduling combines the goals of improving energy efficiency and providing energy flexibility for demand response (refer to definitions in Section 2.3.1) with conventional production-related objectives to create more sustainable production schedules. To achieve this, energy-aware production scheduling systems need high availability production system data and sometimes also connectivity to energy utilities, which is enabled by the trend toward Industry 4.0. The same trend also simplifies data gathering during the system's implementation phase.

1.1. Research Goal and Research Questions

While the desire to advance energy efficient and energy flexible production is the primary motivation for this thesis, all cited trends influence the goals and structure. As Garetti and Taisch (2012) mention, the manufacturing industry, which is the focus of this thesis, has a high remaining potential for energy efficiency. They also underpin the idea of energy-aware production scheduling as a relevant building block for energy efficient production (Garetti & Taisch, 2012). Industry 4.0 can contribute to implementing energy-aware production scheduling through data availability and by offering paths to control production machine operations. Systematic literature reviews reveal the increasing research efforts in the area of energy-aware production scheduling (Bänsch et al., 2021; Biel & Glock, 2016; Gahm et al., 2016); however, they identify the practical relevance and the understanding of scheduling-relevant energy characteristics as an area of future research potential (Gahm et al., 2016).

This thesis aims to address part of this potential by providing tooling and models for the implementation of energy-aware production scheduling systems:

Research Goal

This thesis aims to simplify the implementation of energy-aware production scheduling systems in real production systems by manufacturing companies using generalized modelling and tooling.

The following questions guide the research in this thesis by examining the research gaps necessitating this work and by providing ideas to fill this gap. To examine the research gap, this thesis needs to analyze existing research with real industrial use cases and study the implementations and guidance given regarding the implementation procedure:

Research Question 1: Implementation Shortfall

Can a lack of implementations be attributed to an absence of procedures and architectures for implementing energy-aware production scheduling systems in job shops?

If true, this thesis aspires to provide the missing components by developing a concept for an *implementation procedure* and an *energy-aware production scheduling* system architecture with energy model parameter estimation. These two parts should build upon each other; requirements for the energy-aware production scheduling result from the specifics of the implementation procedure:

Research Question 2: Implementation procedure

Can a standardized and partially automated implementation procedure for the adoption of energy-aware production scheduling systems be proposed such that an energy-aware production scheduling system can be more easily applied to real industrial use cases?

The implementation procedure will contain some steps that software tooling can support. The necessity to support these steps entails the requirements for the energy-aware production scheduling system architecture and the energy model parameter estimation:

Research Question 3: Energy-aware production scheduling system architecture

How should the architecture of an energy-aware production scheduling system be designed to support the implementation procedure, and which additional tooling is needed to reduce the implementation efforts?

These research-guiding questions are the basis for the research design and methodology outlined in the subsequent section.

1.2. Research Design and Outline

The research design of this thesis follows the Design Research Methodology described by Blessing and Chakrabarti (2009). This thesis focuses on the design of an implementation procedure and a corresponding energy-aware production scheduling system architecture. Thus, the Design Research Methodology, which explicitly addresses design research projects, is well suited to govern this research. It suggests four research stages that may be revisited iteratively (Blessing & Chakrabarti, 2009):

- *Research Clarification* as the first stage is essential to determine the project's aim, focus, and scope (Blessing & Chakrabarti, 2009, p. 29).
- In the *Descriptive Study-I* stage, undertaking a literature review extends the understanding gathered in the first stage (Blessing & Chakrabarti, 2009, p. 31). The results also facilitate elaborating this research's aim, focus, and scope. The result of this stage should include success criteria for the research project (Blessing & Chakrabarti, 2009, p. 31).

– The *Prescriptive Study* stage follows the first descriptive study and is the main focus area of this thesis. Developing the implementation procedure and the energy-aware production scheduling system architecture are two focal points of the prescriptive study phase.
– Finally, in the *Descriptive Study-II* stage, the developed concept and implementation procedure are evaluated (Blessing & Chakrabarti, 2009, p. 35).

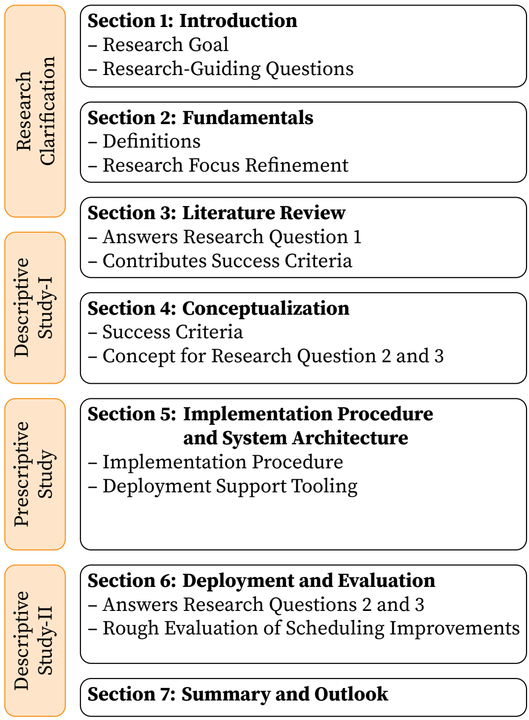


Figure 1.2: Outline of this thesis within the Design Research Methodology context (own figure).

The following paragraphs contextualize each stage of the Design Research Methodology with the research questions mentioned in Section 1.1 in mind and describe the organization and outline of this thesis document. Overall, the structure of this thesis adheres to the third type of design research project suggested by Blessing and Chakrabarti (2009, p. 60). In this type, the *research clarification* and the *descriptive study-I* are review-based. A comprehensive *prescriptive study* constitutes the project's core, while an initial *descriptive study-II* validates the results (Blessing & Chakrabarti, 2009, p. 61). Figure 1.2 also illustrates the outline of this thesis in the context of Design Research Methodology.

Sections 1 and 2 of this thesis document the *research clarification* stage and the research goals. Section 1.1 details the research-guiding questions generated from this initial literature review. Section 2 further clarifies the definitions this thesis relies on to create a shared understanding of the researched subjects and the specific research area.

Section 3 discusses the *descriptive study-I* comprising the literature review. The literature review clarifies the research gap (Section 3.3) and contributes to the requirements and success criteria (Section 4.1). Section 4 summarizes the concept resulting from the *research clarification* and the literature review. The results of the *descriptive study-I* should also answer the first research question.

The *prescriptive study* stage is at the core of this thesis. Section 4 describes the concept for the implementation procedure developed during this stage. Section 5 details the structure and components of the energy-aware production scheduling system architecture. Predominantly, the *prescriptive study* suggests answers for the second and third research questions. Thus, besides the energy-aware production scheduling system architecture, it also provides a concept to support the implementation procedure through energy model parameter estimation.

Finally, this thesis evaluates the implementation procedure and the energy-aware production scheduling system architecture by deploying it to an actual research production system. The experiences from this deployment constitute the initial *descriptive study-II*. Section 6 reveals the results of this deployment trial, which answer the second and third research questions. The section also provides a rough analysis of the achieved scheduling improvements, although that is not the focus of this thesis.

1.3. Highlights

The primary objective of this thesis is making the implementation of energy-aware production scheduling systems in job shops easier by proposing an implementation procedure with a corresponding system architecture supported by energy model parameter estimation. The literature review (Section 3) highlights that most research conducted on real industrial use cases lacks sufficient guidance on implementing the proposed solutions, which emphasizes the need for this work.

The implementation procedure derives from state-of-the-art procedures for realizing traditional production scheduling, Manufacturing Execution Systems (MESs), or Enterprise Resource Planning (ERP) systems. It expands on these procedures by including energy-awareness. The procedure identifies the areas where implementers need to gather additional information and the steps where automation could support the implementation process (refer to Section 4).

This thesis also proposes a modular energy-aware production scheduling system architecture with generalized energy models for machine tools and industrial aqueous cleaning machines. The system uses configuration options to select the relevant modules needed to represent a particular production system, and thus, it provides the ability to adapt to a variety of production systems. The configuration considers production machines, products (also: production jobs) and respective manufacturing operations (refer to Section 5). Models for other types of machines could also be added in the future (see Section 5). Additionally, energy model parameter estimation automates parts of the implementation procedure by estimating the parameters of generalized models for production machines.

Finally, this thesis successfully applies the proposed implementation procedure and energy-aware production scheduling system by deploying the system to an actual research production system. The system consists of multiple machines of each type; thus, it demonstrates that the generalized models are flexible enough to describe various machines (see Section 6). Additionally, this thesis describes a preliminary study of transferability to an actual industrial production system.

2. Fundamentals and Definitions

This section defines some fundamental terms that this thesis frequently uses, to ensure clarity and focus. There are three main focus points for this: production planning, digital connectivity in production systems, and energy considerations in production scheduling.

Regarding production planning in general, it is crucial to understand the terms *production planning* and *production scheduling*, as well as the various *machine environments* to which they apply. Additionally, this section provides an overview of the algorithms commonly used for production scheduling.

This thesis will use the terms *cyber-physical production system* and *Industrial Internet of Things* to refer to connectivity in production systems. Therefore, focusing on these concepts and the *connectivity frameworks* required to implement them in real production systems is vital.

Finally, since this thesis aims to simplify the implementation of energy-aware production scheduling systems, this section will look at energy considerations in production scheduling. In this area, a shared understanding of *energy efficiency*, *energy flexibility*, concepts for *demand-side integration*, and energy modelling of production machines is elemental.

2.1. Production Planning and Production Systems

To narrow down the focus of this thesis, discussing what production scheduling is and how it integrates into the overarching process of production planning and control is crucial. This section provides an understanding of the production planning and control process and the goals of production scheduling.

The formulation of production scheduling problems is strongly dependent on the machine environment present in a factory. The second part of this section introduces classification schemes for production systems and machine environments. The last part of this section deliberates various objectives for the production scheduling problem and algorithms to approximate optimal solutions.

2.1.1. Production Planning and Control

Production planning and control is the complex decision-making process governing production in industrial companies (Wiendahl & Wiendahl, 2019, p. 279):

Definition 2.1: Production Planning and Control

Production planning and control describes the multi-step planning of production in industrial companies (Grabner, 2017, p. 216).

Production planning and control covers the entire process, from determining the products to be produced and production sites or production machines to be used or built, down to the individual production operation performed to fulfil customer orders. There are many subtasks covering different aspects of production planning and control and many different taxonomies for differentiating them (e.g., Groover (2008, p. 753) and Silver et al. (1998, p. 539)). Most of these taxonomies use an hierarchical approach to structuring production planning and control dating back to Hax and Meal (1973). This approach classifies tasks by planning time horizon and level of aggregation. The Aachen Production Planning and Control Model is also often used to discuss these aspects (Grabner, 2017, p. 215). According to Schuh et al. (2015) it differentiates the tasks described in the subsequent paragraphs.

Production program planning determines the type and quantity of products (primary demand) to be produced, among other factors (Schuh, 2019). This is roughly equivalent to the *aggregate planning, demand management* and *master production scheduling* steps proposed by Silver et al. (1998, p. 539).

Production requirements planning estimates the demand for required resources by examining the bill of materials, production capacity, and the like (Schuh, 2019). Silver et al. (1998, p. 539) refer to this step as *material planning* and *capacity planning*.

In-plant production planning and control details the allocation of production quantities to the available resources, such as machines, and monitors the realization of production plans (Schuh, 2019). It includes the *short-range scheduling* and *capacity control* modules of the framework by Silver et al. (1998, p. 539).

Procurement planning and control defines the quantities and dates of products purchased from other manufacturers (Schuh, 2019).

Again, each of these tasks consists of multiple subtasks. Since the goal of this thesis is to analyze short-term planning, the subtasks of *in-plant production planning* *and control* are most relevant in this context. Detailed descriptions of other tasks can be found in Schuh et al. (2015, p. 29) and Grabner (2017, p. 214). The subtasks of *in-plant production planning and control* are defined as the following.

- *Lot-sizing* determines the size of production lots (Schuh et al., 2015, p. 50).
- *Fine scheduling* adjusts when each production order is produced (Schuh et al., 2015, p. 54).
- *Fine resource planning* considers the production resources and their capacity (Schuh et al., 2015, p. 55).
- *Sequencing* determines the sequence of production orders for each production resource (Schuh et al., 2015, p. 55).
- *Availability checks* ensure that production resources are available at production start (Schuh et al., 2015, p. 57).
- Finally, the *order release* confirms the start of production and generates all necessary documentation (Schuh et al., 2015, p. 57).

The segmentation of processes proposed by Schuh et al. (2015) is very detailed – in the context of this thesis, it is more suitable to aggregate the fine scheduling, fine resource planning, sequencing and availability check tasks. This work refers to this aggregate task as production scheduling:

Definition 2.2: Production Scheduling

"[Scheduling] deals with the allocation of resources to tasks over given time periods, and its goal is to optimize one or more objectives (Pinedo, 2016, p. 1)." Production scheduling is a sub-domain of scheduling.

Since production scheduling is concerned with allocating tasks to resources, it is essential to consider the characteristics of tasks and resources as well. In production scheduling the tasks are typically operations associated with jobs.

Definition 2.3: (Production) Job and Operation

A (production) job is the equivalent of a customer order within the production system. The result of a job is the production of one or more finished products. A job may consist of a sequence of operations necessary to produce the product.

An example of a job could be the assembly of a hydraulic cylinder from a piston, housing, and gaskets. One of the operations to complete this job could be pressing the gaskets into the cylinder housing.

2.1.2. Production Systems and their Structure

There are many different types and structures of production systems, each of which pose different demands on production scheduling due to variations in production resources, types of processes, and orders.

Definition 2.4: Production System

"A production system is a collection of people, equipment, and procedures organized to perform the manufacturing operations of a company." (Groover, 2008, p. 19)

Production systems consist of facilities and manufacturing support systems (Groover, 2008, p. 19). Facilities are the factory itself and the equipment used for production, while manufacturing support systems describe the procedures used to initiate and manage the production (Groover, 2008, p. 19). The facilities differ significantly depending on whether one studies continuous production or discrete production. This thesis is solely concerned with discrete production. In discrete production the output consists of discrete workpieces or products (Groover, 2008, p. 44).

There is no universal classification for production systems – instead, there are many views and classification schemes according to different aspects. Jodlbauer (2008, p. 1) provides an overview of typical aspects relevant to classifying production systems in discrete production. Groover (2008, p. 51) substantiates these aspects. They are:

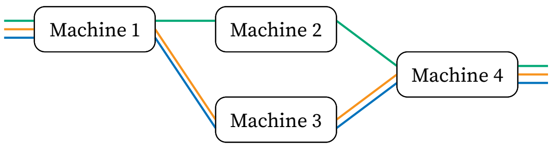
- order decoupling point,
- manufacturing structure,
- product variety and complexity,
- factory layout, and
- production quantity.

In contrast, Pinedo (2016) bases his classification of production systems on the specific requirements of production scheduling. Pinedo (2016, p. 14) refers to these as *machine environments* and differentiates between single-machine models, parallel machine models, flow shops, flexible flow shops, job shops, and open shops. Classifications similar to what Pinedo (2016) proposes are ubiquitous in the field of production scheduling.

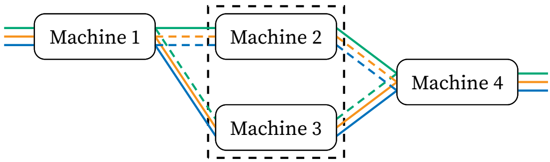
Definition 2.5: Machine Environment

The machine environment describes the structure of the facilities in a production system, considering the factory layout and flow of jobs through the factory.

Single-machine models are a special case of all other structures. As the name suggests, they enable scheduling for precisely one machine (Pinedo, 2016, p. 14). Parallel-machine models also consider only a single stage of production but allow for multiple identical or different machines in parallel (Pinedo, 2016, p. 14). The machines in such models may differ in terms of their capabilities, for instance, processing speed or the jobs they can perform.



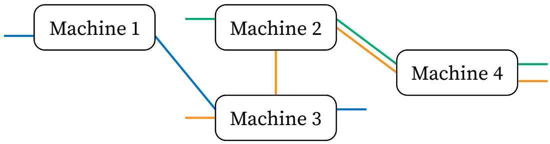
(a) Structure of a flow shop.



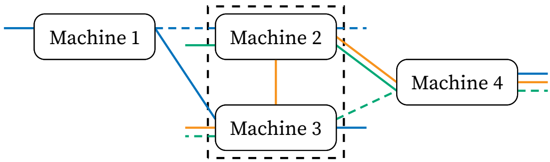
(b) Structure of a flexible flow shop.

Figure 2.1: Structure of flow shops. Dashed lines indicate alternative routes for the same job (own figure).

Flow shops are more complex than single-stage models. Flow shops can have multiple stages of production machines; however, jobs are constrained by having to pass through all stages in sequence (Pinedo, 2016, p. 15). Figure 2.1a illustrates the principle of a flow shop. A flexible flow shop, in comparison, can have multiple machines (work centers) for each stage, and any machine in a work center can process each job (Pinedo, 2016, p. 15). There is also an option for jobs to recirculate and be processed twice in the same work center (Pinedo, 2016, p. 15). Figure 2.1b shows the structure of flexible flow shops. The illustration shows that the same job can follow multiple routes through a work center.



(a) Structure of a job shop.



(b) Structure of a flexible job shop.

Figure 2.2: Structure of job shops. Dashed lines indicate alternative routes for the same job (own figure).

Job shops and flexible job shops feature individual routes for each job (Pinedo, 2016, p. 15). Machines in job shops are typically organized according to their capabilities (Jodlbauer, 2008, p. 10), and machines with similar production capabilities are located in the same factory area. Flexible job shops differentiate by the existence of work centers. Similar to flexible flow shops, any machine in a work center could process the same job, and production scheduling can choose the best suitable machine (Pinedo, 2016, p. 15). Figure 2.2 illustrates the structure of job shops and flexible job shops.

Since this thesis targets job shops, a specific definition for the term is essential. The definition of a job shop used in this thesis is based on Pinedo (2016) and focuses on the flexible routing aspect of job shops.

Definition 2.6: Job Shop

A job shop is a machine environment where each job follows its unique predetermined route (Pinedo, 2016, p. 15).

Through the flexible routing available in job shops this type of machine environment also provides high flexibility when reacting to new products or customer requirements (Jodlbauer, 2008). However, job shops pose one of the more complex problems for scheduling.

Open shops have even more relaxed requirements for the routing of jobs. In an open shop each job's routing through the factory can be chosen arbitrarily (Pinedo, 2016, p. 15). This complete flexibility means that open shops are the most general type of production scheduling problem.

2.1.3. Optimization of Production Schedules

The preceding sections characterize production planning and control in general and discuss production system structures or machine environments. Building on that information this section examines objective criteria and optimization methods for production scheduling. Literature often refers to objective criteria as cost factors or key values to be minimized. The objective criteria introduced in this section are *production-related objective criteria* - they are distinct from the *energy-related objective criteria* addressed in Section 2.3.1.

Pinedo (2016), Jodlbauer (2008) and Silver et al. (1998) provide some examples of production-related objectives of production scheduling. According to Jodlbauer (2008, p. 19), the key values used as objective criteria can be separated into inward-looking and outward-looking values, where inward-looking values describe a company's structure and outward-looking factors evaluate market potentials. However, no list of objective criteria can be exhaustive because there are specifics to every production system and many different perspectives on production scheduling performance.

Even the three named sources identify relevant cost factors from different viewpoints. Jodlbauer (2008, p. 19) concentrates on the performance of the production system and includes factors like utilization, inventory, costs for additional capacity, and lead time. On the other hand, Silver et al. (1998, p. 44) are more concerned with the cost of production for individual items and favour factors like the cost of carrying items in inventory, unit variable cost, or the ordering cost. Finally, the factors preferred by Pinedo (2016, p. 18) pertain more directly to the performance of production schedules. Since they are most relevant to the topic of this thesis, the following paragraphs introduce some of these factors in more detail. Additionally, Guzman et al. (2021) give a more detailed overview of objective criteria for production scheduling used in recent research. Valid objective criteria for production scheduling could also be formulated from many other factors.

Makespan is equivalent to the completion time of the last job to leave the production system (Pinedo, 2016, p. 18). It represents the total time passed while producing a defined set of jobs. Minimal makespan generally implies high utilization of production machines (Pinedo, 2016, p. 18). Simply put, makespan describes the total duration of a production schedule. It could be used, for example, to evaluate whether a production schedule can be completed within a single day.

Total weighted completion time represents the sum of completion times, often called flow time (Pinedo, 2016, p. 19). The flow time or completion time is the time taken from introducing a job into the production system until it leaves the production system (Dickmann, 2015, p. 395). It includes non-productive time like holding time and downtime of machines (Dickmann, 2015, p. 395). The weight in total weighted completion time depends on the job and could represent something like the importance of the job or associated storage costs (Pinedo, 2016, p. 14). This way, total weighted completion time may indicate the inventory costs caused by a schedule (Pinedo, 2016, p. 19).

Total weighted tardiness measures how many jobs are completed later than the due date and how much later they are completed (Pinedo, 2016, p. 19). The due date can be understood as the shipping date promised to a customer (Pinedo, 2016, p. 14). There is a differentiation between tardiness and lateness here, such that tardiness only refers to jobs completed after the due date, while lateness examines all jobs (Jodlbauer, 2008, p. 39).

While there are many more objective criteria for production scheduling, this short overview is sufficient for the purpose of this work. More detailed discussions of objective criteria and problem formulations for various machine environments are provided by Pinedo (2016).

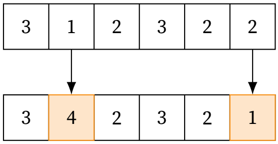
The second important aspect of optimizing production schedules are the optimization algorithms. They range from dispatching rules over local search procedures to machine learning algorithms. Similar to the objective criteria, there is a sheer infinite number of optimization methods. Pinedo (2016, p. 375) provides an overview of some methods, but since this is not the main focus of this thesis, the following focuses on the most pertinent broader categories of methods. Guzman et al. (2021) provide a much more detailed review of current research and algorithms regularly used in the field.

Dispatching rules are heuristics that are comparatively easy to implement and apply to various production systems (Pinedo, 2016, p. 376). They can lead to optimal results in specific situations and often achieve good results when optimizing for a single objective, such as makespan (Pinedo, 2016, p. 376). Composite dispatching rules extend this concept by combining multiple dispatching rules with scaling factors (Pinedo, 2016, p. 377). In addition, dispatching rules are constructive – they can be used to create new schedules without prior knowledge about starting points (Pinedo, 2016, p. 382).

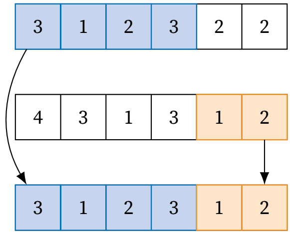
The *Shortest Processing Time (SPT)* rule, where the next operation for each production machine is determined to be the one with the shortest processing time on that machine is an example of a dispatching rule (Pinedo, 2016, p. 377). Another straightforward example is the *service in random order* rule, which simply selects a random operation to be processed next (Pinedo, 2016, p. 376). As a final example, the *shortest queue* rule assigns jobs to the production machine with the shortest queue of jobs waiting for the next operation (Pinedo, 2016, p. 376).

(Meta)heuristic algorithms are a very active field of research, and many researchers use heuristic algorithms to solve the energy-aware production scheduling problem. Pinedo (2016, p. 382) provides a series of examples for algorithms in this category: Simulated annealing and tabu search, for example, perform a neighbourhood search led by developer-specified rules, and genetic algorithms and ant colony optimization are more generalist procedures considering multiple schedules at the same time. Genetic algorithms are one of the most used solution techniques for the problem (refer to Section 3). All of these heuristic algorithms are improvement algorithms and cannot be used to generate new schedules (Pinedo, 2016, p. 382). Solutions for these algorithms can be initialized randomly or using dispatching rules. While many researchers use random initialization, we have also seen good results using the latter path (Grosch et al., 2021).

Again, there are many sub-types of each of these methods – the Non-Dominated Sorting Genetic Algorithm-II (NSGA-II) is an example of a specific genetic algorithm that will be of importance later on in this thesis. The NSGA-II was proposed by Deb et al. (2002) and is well suited to solving multi-objective problems. When discussing genetic algorithms, the term *population* refers to the number of solutions evaluated in parallel, *individual* refers to a single solution, and *generation* refers to all solutions in a single iteration (Pinedo, 2016, p. 389). Each individual has one



(a) Mutation operator for a genetic algorithm.



(b) Crossover operator for a genetic algorithm.

Figure 2.3: Schematic representation of mutation and crossover operators - the exact implementation often varies (own figure).

or more chromosomes to describe the values of the solution. Like other genetic algorithms, the NSGA-II uses mutation and crossover operators, as illustrated in Figure 2.3, to perform a local search. The mutation operator changes the values on the chromosomes of some individuals, while the crossover operator interchanges parts of the chromosomes of two individuals (Pinedo, 2016, p. 289). The NSGA-II is unique for its non-dominated sorting process to select the best individuals in a generation and its crowding-distance preservation mechanism (Deb et al., 2002). Section 5.2.1 provides a more detailed explanation of the algorithm and its implementation used in this thesis.

Mathematical programming generates exact solutions to the scheduling problem. For example, many scheduling problems can be formulated as mixed integer programs and solved with exact algorithms like branch-and-bound (Pinedo, 2016, p. 561). However, since many scheduling problems are NP-hard, as Pinedo (2016, p. 588) demonstrates, and exact solutions are often optional, there are also inexact solution algorithms for mathematical programming formulations. Besides the heuristics mentioned above these include constraint programming, where most requirements on the problem formulation are relaxed, and the algorithm only ensures that all constraints of the program are satisfied (Pinedo, 2016, p. 579). Beam search, which is similar to branch-and-bound but does not evaluate all nodes (Pinedo, 2016, p. 400), also belongs in this category.

Machine Learning algorithms have shown promising results in more recent research (e.g., Stricker et al., 2018; Waschneck et al., 2018). There are three categories

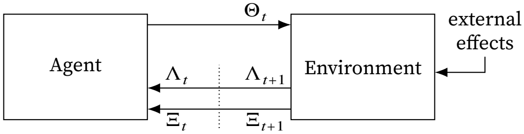


Figure 2.4: Markov decision process with an algorithm generating actions Θ_t and an environment that consumes the actions and provides the agent with its current state Ξ_t and a reward value Λ_t (own figure).

of machine learning approaches: supervised learning, unsupervised learning and reinforcement learning (Sutton & Barto, 2018, p. 2). As the two sources mentioned above exemplify, reinforcement learning appears to be particularly promising for scheduling use cases.

As shown in Figure 2.4, the Markov decision process is the basis of reinforcement learning (Sutton & Barto, 2018, p. 37). The Markov decision process formalizes the sequential decision-making process where an algorithm receives information about the state and a reward from an environment and returns a set of actions to the environment (Sutton & Barto, 2018, p. 37). The state represents the current condition of the environment and the reward measures the performance of the state with respect to a set objective (Sutton & Barto, 2018, p. 37). The actions are the instructions the algorithm gives to the environment to change its state.

2.2. Digital Connectivity in Production Systems

This thesis understands that future production systems will take the form of cyber-physical production systems. This section introduces some aspects of cyber-physical production systems and relevant manufacturing support services that provide production scheduling functionality. Since energy-aware production scheduling requires good access to data about the production systems, the second part of this section focuses on connectivity through the Industrial Internet of Things and the connectivity frameworks used to achieve it. The final part of this section considers implementation procedures for production scheduling systems in cyber-physical production systems.

2.2.1. Cyber-Physical Production Systems

The previous sections look at the definition of production scheduling in general, the variety of machine environments where scheduling problems arise, and some optimization algorithms. At the same time, the digitalization of production systems is also highly relevant to the implementation of production scheduling systems, and is quickly advancing – concepts like industry 4.0 and smart manufacturing are on the rise (see: Kang et al., 2016; Lasi et al., 2014; Pfeiffer, 2017). Both concepts focus on integrating cyber-physical systems along the complete value chain (Brauner et al., 2022).

Definition 2.7: Cyber-Physical System

"Cyber-physical systems are systems of collaborating computational entities which are in intensive connection with the surrounding physical world and its on-going processes, providing and using, at the same time, data-accessing and data-processing services available on the internet." (Monostori, 2019)

Although older, Groover (2008, p. 87) provides an intriguing overview of automation technologies in a production system. Groover (2008, p. 87) locates the manufacturing support systems at the enterprise level, and manufacturing systems as well as the automation, control and material handling technologies at the factory level. In combination with the interconnections between the production system components, the characterization proposed by Groover (2008, p. 87) can be interpreted as an early form of the cyber-physical production system.

Definition 2.8: Cyber-Physical Production System

In the context of production systems, cyber-physical systems are referred to as cyber-physical production systems (Napoleone et al., 2020).

VDI/VDE-Gesellschaft Mess- und Automatisierungstechnik (2013) and Monostori (2019) corroborate the departure from the typical automation pyramid with a move towards service-centric architectures. They foresee self-organizing networks of services in cyber-physical production systems. The technology necessary for implementing energy-aware production scheduling within cyber-physical production systems thus is a multitude of interconnected services communicating with the physical production machines to receive data and control production operations. This service-oriented architecture helps handle the pertinent technologies' complexity and heterogeneity (Schuh et al., 2007). According to International Organization for Standardization (2021) a service is a "distinct functionality that is provided by an entity through interfaces".

Many services and software systems can be involved in the production planning and control process. The services differ in the process stage where they are applied and the level of detail their output provides. Typical software systems offering such services include ERP, MES and Material Requirement Planning (MRP) (Grabner, 2017, p. 214).

ERP systems focus on general business functions such as accounting and personnel management besides production planning (Schuh, 2019). Therefore, ERP systems are also often used in other industries, such as retail and service (Grabner, 2017, p. 214). While ERP systems often have functionality for production planning, MESs additionally provide connectivity between the ERP system and process automation (Grabner, 2017, p. 215). MESs commonly include functions to collect machine and process data and to enable evaluations based on customer orders or production jobs (Grabner, 2017, p. 215). They also support the production scheduling process, considering machine utilization and production job due dates (Grabner, 2017, p. 215).

The terms MRP and Manufacturing Resource Planning (MRP II) refer to some of the earlier concepts in production planning and date back to the 1960s and 1990s respectively (Schuh, 2019). These concepts describe a gradual approach where required materials and production machines are scheduled subsequently (Schuh, 2019). Modern ERP systems integrate all of these concepts. Recent developments in ERP systems deviate from the traditional approach, using advanced, integrated planning techniques for all resources in a single run (Schuh, 2019). Additionally, ERP systems are generally company-wide systems, while MRP applications are usually more plant-centric (Groover, 2008, p. 779)

Production planning and control, as introduced in Section 2.1.1, can be offered as a module of one of the systems mentioned above or as a separate, self-contained service. As Grabner (2017, p. 215) mentions, production planning and control, ERP, and MES are frequently ambiguous in common parlance. This thesis defines the production planning and control service as an independent, self-contained part of the cyber-physical production system. In addition to, or as a replacement for, the production planning and control service, there can be an energy-aware production scheduling service.

2.2.2. Connectivity in the Industrial Internet of Things

Besides the services, connectivity between entities (services and physical assets) is the second cornerstone of cyber-physical production systems. The entities that are part of the cyber-physical production system must be able to communicate with each other and with external services via the Industrial Internet of Things.

Definition 2.9: Industrial Internet of Things

The Industrial Internet of Things is "a system comprising networked smart objects, cyber-physical assets, associated generic information technologies and optional cloud or edge computing platforms, which enable real-time, intelligent, and autonomous access, collection, analysis, communications, and exchange of process, product and/or service information, within the industrial environment, so as to optimize overall production value. This value may include; improving product or service delivery, boosting productivity, reducing labour costs, reducing energy consumption, and reducing the build-to-order cycle." (Boyes et al., 2018)

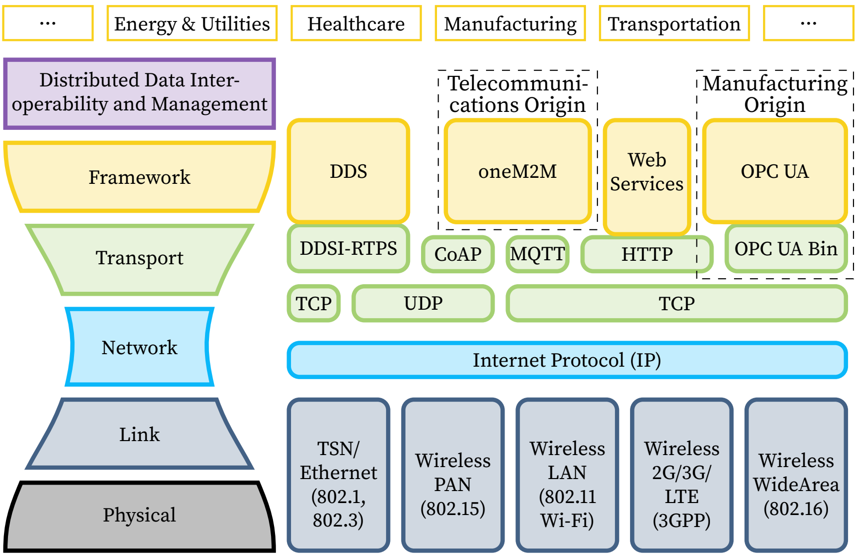


Figure 2.5: Connectivity stack proposed for the Industrial Internet Reference Architecture (Lin et al., 2017).

Currently, there are several national standards for Industrial Internet of Things connectivity architectures like the Industrial Internet Reference Architecture (Lin et al., 2022) and the Reference Architecture Model Industrie 4.0 (RAMI 4.0)

(Deutsches Institut für Normung e.V., 2016). A variety of different standardization organizations manage these standards; however, efforts for an alignment between the architecture models have been made (e.g., Lin et al., 2017; Gayko et al., 2018). The RAMI 4.0 strongly focuses on the manufacturing industry, while the Industrial Internet Reference Architecture is more generally applicable and caters to a variety of industrial sectors. The Industry IoT Consortium (IIC) assesses various connectivity technologies and identifies four technologies it suggests for use in the Industrial Internet of Things, as illustrated in Figure 2.5 (Joshi et al., 2022). The suggested core connectivity technologies are published as the Industrial Internet of Things Connectivity Framework (IICF) (Joshi et al., 2022).

The IICF focuses on the framework and transport layers, as shown in Figure 2.5. It differentiates connectivity technologies between the framework and transport layers by defining that a connectivity framework at least pairs a connectivity transport with a data type system (Joshi et al., 2022, p. 51).

Definition 2.10: Connectivity Framework

A connectivity framework provides syntactic interoperability independent of endpoint implementations (Joshi et al., 2022, p. 28).

In comparison, a connectivity transport only provides technical interoperability (Joshi et al., 2022, p. 39). Some connectivity frameworks provide mappings to multiple transports.

The IICF and the RAMI 4.0 agree that Open Plattform Communications Unified Automation (OPC UA) is well suited for the manufacturing industry and should be one of the core standards for the Industrial Internet of Things (Lin et al., 2017). The IICF also mentions the importance of Representational State Transfer (ReST) web services based on the Hypertext Transport Procotol. However, they are thought to be more suitable for manufacturing support systems than device-to-device communications within the factory (Joshi et al., 2022, p. 64). Finally, Fieldbus technologies like Modbus via TCP/IP (Modbus TCP) are relevant at the factory level. They should be integrated into the Industrial Internet of Things (Joshi et al., 2022, p. 63) because many applications, including energy-aware production scheduling systems, need access to information from them. The IICF does not list them as one of the core standards, though (Joshi et al., 2022).

Since this thesis uses connectivity frameworks and connectivity transports for similar purposes and abstracts their functionality, the term connectivity framework is applied more liberally and, for example, includes Modbus TCP, although it does not technically fulfil the requirement of having a data type system.

2.2.3. Implementation Procedures for Production Scheduling Services

Since this thesis strives to simplify the implementation of energy-aware production scheduling services, discussing existing procedures for implementing production planning and control services is crucial. Some steps of these procedures are similar for any software product, while other steps are specific to production planning and control services.

Schuh et al. (2015, p. 333) describe the implementation of production planning and control services and ERP systems in detail. Schuh et al. (2015, p. 333) differentiate two main phases during the implementation: selecting a suitable software system and implementing the system. The selection process is not discussed in detail here because it is irrelevant to this thesis; however, the implementation procedure is very significant for this work.

Definition 2.11: Implementation procedure

An *implementation procedure* is a multi-step process that companies can follow to introduce and implement a specific software system or service. The implementation procedure begins after a specific software product has been selected and ends once the company has successfully adopted the new system.

A review shows that software companies have developed specific implementation procedures which they follow. Schuh et al. (2015, p. 362) corroborate this finding. The Oracle Netsuite implementation plan (Schwarz, 2022) and SAP Activate (Musil, 2018) are examples of implementation procedures. In addition, Beeson (2022) from ERP Focus describes a manufacturer-independent process.

The procedure described by Musil (2018) has six steps: discover, prepare, explore, realize, deploy and run. The procedure in Schwarz (2022) also consists of six steps: discovery and planning, design, development, testing, deployment, and support. The concept outlined by Beeson (2022) focuses more on the planning and budgeting stages. It has seven steps: assemble a team, create a change management plan, estimate the implementation budget, begin data migration, train users, go live, and evaluate. The more theoretical approach by Kropik (2009, p. 366) suggests adapting software development processes. Schuh et al. (2015, p. 362) propose a six-step process with personnel qualification (for the implementation, not usage of the system), prototyping, detailed concept, development and configuration, data migration, and deployment.

Table 2.1: Correlation between the phases of various implementation procedures.

Step	Schuh et al. (2015)	ORACLE implementation plan
Discovery and planning	personnel qualification, prototyping, detailed concept	discovery and planning, design
Development and configuration	development and configuration, data migration	development
Testing and deployment	deployment	testing, deployment, support
Step	**SAP ACTIVATE**	**ERP FOCUS**
Discovery and planning	discover, prepare, explore	assemble a team, create change management plan, estimate implementation budget
Development and configuration	realize	begin data migration
Testing and deployment	deploy, run	train users, go-live, evaluate

When looking at the procedures in detail, many similarities become apparent, and the following typical phases can be identified: discovery and planning, development and configuration, testing and deployment. This thesis uses these three high-level phases for its implementation procedure. Table 2.1 shows a rough correlation between the different procedures introduced in the previous paragraphs and these phases. The following paragraphs outline the activities in each phase that have to be performed by a company wanting to implement a new production scheduling service.

Discovery and planning include the assembly of a project team for the implementation, the creation of a project plan, and an analysis of the business processes in the company to prepare the design phase. The team responsible for the implementation project must be adequately qualified (Schuh et al., 2015, p. 363) and great care must be taken to ensure that the budget allotted for the implementation project is sufficient (Beeson, 2022). The result of the discovery and planning phase is a concept for integrating the company's processes into the software system. Thus, the phase involves a detailed analysis of the company's processes and their juxtaposition with the software's capabilities and standardized processes. The integration might require transforming the company's processes and some software adaptation (Schuh et al., 2015, p. 339) which leads into the next phase.

Development and configuration of the system follow after the initial discovery. This phase encompasses adapting the system to the company's desired processes (Musil, 2018). The adaptation includes configuration where possible but might also require some software development effort (Schuh et al., 2015, p. 374). Once basic configuration is complete, data migration can begin. During the data migration, it is fundamental to ensure high data quality (Beeson, 2022).

Testing and Deployment ensure that the final system can perform the required processes without errors and that users can utilize the new system to its full potential (Schwarz, 2022). Real users must perform and test the required processes with actual data to minimize risk during the adoption (Schuh et al., 2015, p. 376). The deployment can comprise a one-off event or a staggered approach (Schwarz, 2022). Finally, it is essential not to forget continued support and improvements after the initial deployment (Schwarz, 2022).

It is obvious that many activities in the implementation procedure are more concerned with general project management than the actual data collection and migration. However, the authors seem to agree that it is critical to consider this entire general change process to ensure the successful implementation of ERP systems. Therefore, this thesis builds on the identified features of the implementation procedure to develop improvements to the implementation of energy-aware production scheduling services.

2.3. Energy Considerations in Production Scheduling

Besides production planning and connectivity in production systems, energy is the third important factor in the context of this thesis. The goal of energy-aware production scheduling services is to improve energy efficiency as well as provide energy flexibility. The first part of this section defines these two terms and gives some context related to production systems. The second part discusses monetization options specifically for energy flexibility with demand-side integration.

The final part of this section examines how to predict the energy consumption of production machines for given production schedules.

2.3.1. Energy Efficiency and Energy Flexibility

Energy-aware production scheduling can have an influence on energy efficiency and energy flexibility at the same time. Energy efficiency is the ratio of energy consumption to the output of a system. A macroeconomic measure for energy efficiency in the services and manufacturing sectors is energy intensity, which is energy per value-added or the ratio of total final energy consumption to gross domestic product (or purchasing power parity for comparisons between multiple countries) (International Energy Agency, 2020, p. 45). Specific energy consumption is another indicator of energy efficiency often used at the macroeconomic level (Phylipsen et al., 1997). The specific energy consumption is used instead of energy intensity when the resources used are measured in physical units, not in economic terms (Phylipsen et al., 1997). How to measure energy efficiency depends strongly on the use case and selected system boundaries (Hesselbach, 2012, p. 7). For production scheduling a more technical approach to measuring energy efficiency seems sensible. This thesis uses the definition from International Organization for Standardization (2017) to identify efficiency improvements through production scheduling.

Definition 2.12: Energy Efficiency

Energy efficiency is the "relationship between the result achieved and the resources used, where resources are limited to energy (International Organization for Standardization, 2017)."

There are many ways to improve energy efficiency in production systems – Hesselbach (2012, p. 14) proposes an onion model for identifying energy efficiency measures. The approach begins analysis at the process level, as illustrated in Figure 2.6. The model then suggests analyzing machines and building systems, and it finishes with identifying efficiency measures for decentralized energy suppliers (Hesselbach, 2012, p. 14). Energy-aware production scheduling would most likely influence energy efficiency on the machine level. For example, Blesl and Kessler (2021, p. 13) mention the reduction of standby energy consumption as one of the areas that has received little attention to date and could provide significant energy savings. Energy-aware production scheduling could contribute to savings in this

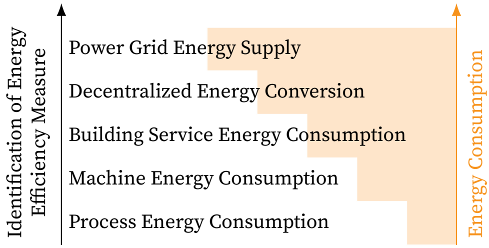


Figure 2.6: Bottom up identification of energy efficiency measures (own figure; Hesselbach, 2012, p. 15).

area by reducing standby energy consumption or by selecting the most efficient production machine for every process.

Energy flexibility is the second factor besides energy efficiency affected by energy-aware production scheduling. Energy flexibility measures are deliberate actions taken to change the state of a production system to alter its energy consumption over time (Verein Deutscher Ingenieure e.V., 2020). Energy flexibility typically entails a reaction to changes in energy markets or by power utilities.

Definition 2.13: Energy Flexibility

Energy flexibility is the "ability of a production system to adapt quickly and in a process-efficient way to changes in the energy market (Verein Deutscher Ingenieure e.V., 2020)."

The previously introduced concepts of energy efficiency, energy flexibility and production scheduling (refer to Section 2.1.1) allow for a definition of energy-aware production scheduling, which combines these definitions into a single overarching concept.

Definition 2.14: Energy-Aware Production Scheduling

Energy-aware production scheduling combines conventional production-related objectives with the aim to improve the energy efficiency and energy flexibility of a factory through production scheduling.

As mentioned in Section 2.1.3, this thesis differentiates the energy-related objective criteria for production scheduling from the production-related objective criteria discussed earlier. Energy-related objective criteria usually aim to evaluate either the energy efficiency of production schedules or the energy flexibility they achieve. Biel and Glock (2016) find that *energy-related cost* and *total energy consumption* are the most used energy-related objective criteria for energy-aware production scheduling. As the names suggest, optimizing total energy consumption improves energy efficiency, while optimizing energy-related cost in combination with appropriate pricing schemes (refer to Section 2.3.2) advances energy flexibility. Section 2.3.2 discusses pricing schemes that enable evaluating energy-related cost. Some authors also utilize other objective criteria like *peak power consumption* or *energy-related greenhouse gas emissions* (Biel & Glock, 2016).

2.3.2. Demand Side Integration

Demand-side integration is the technical area devoted to realizing efficient and flexible use of electric energy. Besides the technical aspects of the concepts introduced in the previous section, demand-side integration also includes their commercialization. Demand-side integration includes short-term and long-term measures (Walther et al., 2022). Energy efficiency measures and on-site power generation are typically long-term measures requiring a planning phase before implementation (Walther et al., 2022). Other long-term measures include load-increasing measures like the electrification of processes that previously operated with different energy forms apart from electricity (Walther et al., 2022). In the short-term, there are energy flexibility measures which are usually referred to as demand response and may lead to an increase or decrease in load.

Definition 2.15: Demand Response

Demand response is a "change in electric usage by end-use customers from their normal consumption patterns in response to changes in the price of electricity over time, or to incentive payments designed to induce lower electricity use at times of high wholesale market prices or when system reliability is jeopardized (U.S. Department of Energy, 2006, p. ix)."

Verein Deutscher Ingenieure e.V. (2020) standardizes a classification scheme for demand response based on the corresponding automation hierarchy level and implementation time frame, as illustrated in Figure 2.7. Changes to shift times or shift break times must be made at the enterprise level because of their high impact on workers' schedules (Verein Deutscher Ingenieure e.V., 2020). On the

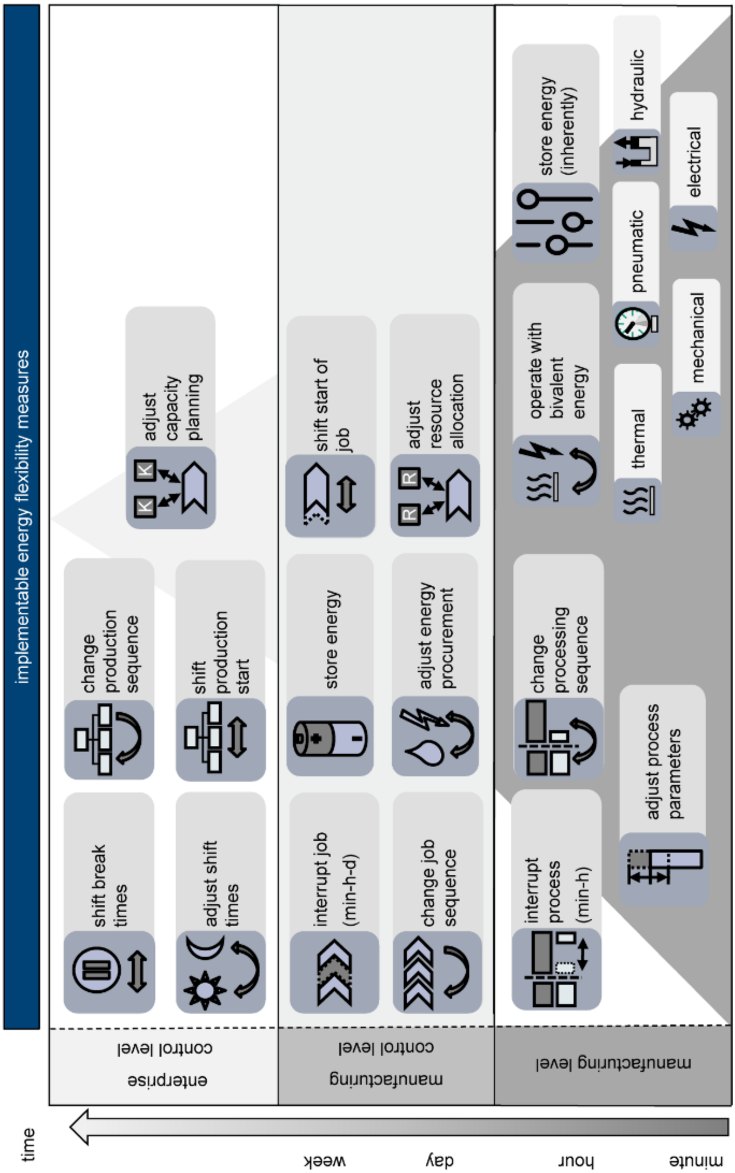


Figure 2.7: Classification of demand response based on the automation hierarchy level and implementation time frame (Verein Deutscher Ingenieure e.V. 2020; © Reproduced with permission of the Verein Deutscher Ingeniuere e.V.).

other hand, changes in the production sequence or the production start, as well as adjustments to capacity planning, have a long-term impact and, as such, are also classified at the enterprise level (Verein Deutscher Ingenieure e.V., 2020).

The classification locates measures with more time-constrained and localized impact at the manufacturing control level (Verein Deutscher Ingenieure e.V., 2020). These measures are most relevant for energy-aware production scheduling and are described in more detail below:

- *Interrupt job* refers to the interruption of a job with multiple operations, for example, to avoid high energy prices occurring during the production of the job (Verein Deutscher Ingenieure e.V., 2020).
- *Shift start of job* names a similar measure and moves the starting time of a job to comply with energy consumption requirements (Verein Deutscher Ingenieure e.V., 2020).
- *Change job sequence* means that jobs are re-sequenced based on their energy consumption – a job with lower consumption could be performed before another job with higher consumption, for example (Verein Deutscher Ingenieure e.V., 2020).
- *Adjust resource allocation* implies switching between production machines to produce a job on a machine with lower energy consumption (Verein Deutscher Ingenieure e.V., 2020).
- *Store energy* expresses the controlled energy storage for later use when there is higher demand or higher energy prices (Verein Deutscher Ingenieure e.V., 2020).
- *Adjust energy procurement* represents a change in the energy form used to power the process (Verein Deutscher Ingenieure e.V., 2020).

Furthermore, there are measures at the manufacturing level, including the interruption of already running processes, changes to the processing sequence, and adjustments to process parameters like the processing speed (Verein Deutscher Ingenieure e.V., 2020). Some production machines might also feature inherent storage or the option to operate with bivalent energy, meaning they are able to utilize two different energy forms (e.g., electricity and gas) (Verein Deutscher Ingenieure e.V., 2020).

Viable measures for energy-aware production scheduling are measures including changes to the process type or parameters, in addition to measures that change the sequencing and timing of jobs or the resource allocation. However, measures that lead to a change in the operating point of production machines might lead to negative interactions with efficiency (Verein Deutscher Ingenieure e.V., 2020). This problem affects direct changes to process parameters and process interruptions, whereas the interruption of jobs on the manufacturing control level usually does not have this specific problem.

Commercializing these demand response measures is quite complex and depends on many factors – Walther et al. (2022) propose a methodology that helps to find monetization options for specific demand response measures depending on the automation hierarchy level, factory lifecycle phase, achieved power difference, the planning horizon (how long in advance a company has to know about the requirements to be ready to implement a demand response measure within the required reaction duration), the reaction duration (how long it takes from a request to execution of the measure) and holding duration (how long the power difference is maintained). The monetization options can be differentiated between wholesale and retail markets (Walther et al., 2022). Two kinds of wholesale markets exist: integrated markets and exchange-based markets (Cramton, 2017). In an integrated market, the system operator optimizes the scheduling and dispatch of resources; in contrast, exchange-based markets let energy companies trade, which decentralizes resource optimization (Cramton, 2017). The products sold and the time frame between the trade and service delivery also differentiate wholesale energy markets (Cramton, 2017). For example, there are derivatives for long-term forward contracting of energy supply, day-ahead markets and intraday markets (Cramton, 2017). On the other hand, there are capacity markets to guarantee financing for new generating capacity before being built, and ancillary service markets to ensure power grid stability (Cramton, 2017).

There are many different wholesale market designs, depending on the region (Cramton, 2017). Paterakis et al. (2017) provide an overview of electricity markets and the commercialization of demand response in a multitude of regions across all continents. The European electricity market, as one example, is split into an energy-only market with futures to provide long-term hedging options and day-ahead and intraday markets, as well as real-time trading on a continuous intraday market (Bertsch et al., 2019). In addition to these energy-only markets, the European electricity market has ancillary service markets to provide primary and secondary balancing power and a minute-reserve (Bertsch et al., 2019). These are successively invoked when balancing power is required (Bertsch et al., 2019).

Entry into the wholesale electricity markets is limited by the smallest tradable units – the exact limit depends on the specific market. In Germany, for example, the smallest tradable units are 1 MW for the primary balancing power market or 5 MW for the secondary balancing power market and the minute reserve (Bertsch

Table 2.2: Incomplete overview of price-based demand response programs with planning horizon and holding duration (Walther et al., 2022).

Description	Planning Horizon	Holding Duration
Time-of-use pricing	> 12 h	1 h – 12 h
Critical peak pricing	Varying	1 h – 12 h
Real-time pricing	1 h – 24 h	5 min – 1 h

et al., 2019). Furthermore, to allow the commercialization of demand response measures, factors like the planning horizon and reaction duration must match the corresponding markets. Finally, regulatory and financial entry barriers can prevent participation in the wholesale markets (Walther et al., 2022).

Retail markets have lower barriers to entry and are more accessible for demand side market participants. Since there is no exact differentiation between wholesale and retail, some of the markets introduced above could also be open for retail customers if the entry barriers are sufficiently low. As Paterakis et al. (2017) discuss, this is the case in some regions. Where this is not the case, grid operators, like power utilities, act as retailers and potentially offer demand response programs (U.S. Department of Energy, 2006, p. x).

There are price-based and incentive-based programs for monetizing demand response on the retail market (U.S. Department of Energy, 2006, p. x). Incentive-based programs give customers incentives in addition to their regular electricity price (U.S. Department of Energy, 2006, p. xi) – overall lower prices could be an example of such an incentive. Direct control programs, where utilities can remotely control equipment, are an example of an incentive-based program (U.S. Department of Energy, 2006, p. xii). Curtailable load programs are similar; however, in this case, the power utility signals to the customer to curtail a load and does not have direct control over equipment (U.S. Department of Energy, 2006, p. xii). One more example is demand bidding, where the customer places bids for curtailments that the grid operator can accept or refuse (U.S. Department of Energy, 2006, p. xii).

Most research on energy-aware production scheduling assumes price-based programs, as introduced in Table 2.2. Time-of-use pricing schemes have fixed prices for predetermined daily time intervals (U.S. Department of Energy, 2006, p. ix) meant to shift demand from peak to off-peak periods (Lampropoulos et al., 2013). Critical peak pricing extends time-of-use pricing by a higher rate for specific trigger conditions (Lampropoulos et al., 2013). Trigger conditions for the peak price may include compromised system reliability or very high supply prices (U.S.

Department of Energy, 2006, p. ix). Finally, real-time pricing is the most flexible pricing scheme, with frequent adjustments of retail prices to reflect supply prices (Lampropoulos et al., 2013).

2.3.3. Energy Consumption of Production Machines

Models of the energy consumption of production machines are necessary to evaluate the performance of production schedules regarding energy considerations like energy efficiency and energy flexibility. There are multiple ways to account for energy consumption during production scheduling. For example, Sihn et al. (2018) differentiate two categories.

Simulation-based approaches employ simulation models with varying levels of detail to describe the production system and its energy consumption. Production schedules are then compiled by manual experimentation or using optimization techniques with the simulation model as an evaluation function (Sihn et al., 2018). Rager (2008, p. 106) uses a continuous-time simulation model for the energetic evaluation of production schedules generated by an evolutionary algorithm, for example. Thiede et al. (2016) describe a different approach and employ a multi-level simulation model combined with a manual, scenario-based approach for optimization.

Optimization model-based approaches integrate energy consumption models directly into an overarching optimization model. Such approaches typically use simplified energy consumption models (Sihn et al., 2018) such that the entire optimization model can be solved exactly (or optimal solutions can be approximated) using the methods introduced in Section 2.1.3. The disadvantage of these models is their much lower performance regarding the realistic description of system behaviour (Sihn et al., 2018).

The diverse approaches also implicitly bring differentiation in the level of detail incorporated by the models. Optimization models cannot be as detailed as simulation models because they must be directly solvable in a reasonable time frame. In simulation-based approaches the influence of model complexity on the solution time is much lower; thus, more complex models are possible (Sihn et al., 2018). This allows simulation-based approaches to take a variety of factors that are ordinarily not considered in optimization model-based approaches. As the literature review in Section 3 shows, most research on energy-aware production scheduling appears to be optimization model-based (see, e.g. Bänsch et al. (2021), who find that only about 4 % of articles use simulation-based approaches).

The optimization model-based approaches are structured by the energy demand aspects they attempt to reflect in their models. For example, Biel and Glock (2016) differentiate between processing power, power used during idle, base and holding states, setup power, power for tool changes, power used during shutdown, and power used during parts transport. Gahm et al. (2016) identify even more different energy demand aspects and propose disambiguating non-processing energy demand and processing energy demand. The different nomenclatures and considered aspects indicate a lack of generally applicable definitions of energy demand aspects for production scheduling.

This thesis uses the disambiguation between processing energy demand and non-processing energy demand, as proposed by Gahm et al. (2016). Non-processing energy demand aspects include energy consumed during storage (Gahm et al., 2016) and transportation (Biel & Glock, 2016). The technical specification VDMA 34179, which is specific to machine tools but can be applied to other types of production machines as well, provides a good starting point for differentiating processing energy demand aspects (Verein Deutscher Maschinen- und Anlagenbauer, 2019). VDMA 34179 discerns energy states and state transitions (Verein Deutscher Maschinen- und Anlagenbauer, 2019). Energy state transitions are characterized by a change in energy consumption determined by the prior energy state, posterior energy state, and by the duration of the transition (Verein Deutscher Maschinen- und Anlagenbauer, 2019). The specification asserts that the energy states differ between machines and processes; however, four universally applicable states can be identified (Verein Deutscher Maschinen- und Anlagenbauer, 2019):

- off
- standby
- operational
- working

Schrems (2014, p. 40) shows that the same approach can also represent other machine types. However, depending on the type of machine, the influence of particular process-independent components might be much more substantial than it is for machine tools (Schrems, 2014, p. 40). For example, the heat needed to control the temperature of the cleaning medium dominates the energy consumption of aqueous cleaning machines (Schrems, 2014, p. 41). Hence, besides the primary energy states, energy consumption also depends on the types of components within the machine.

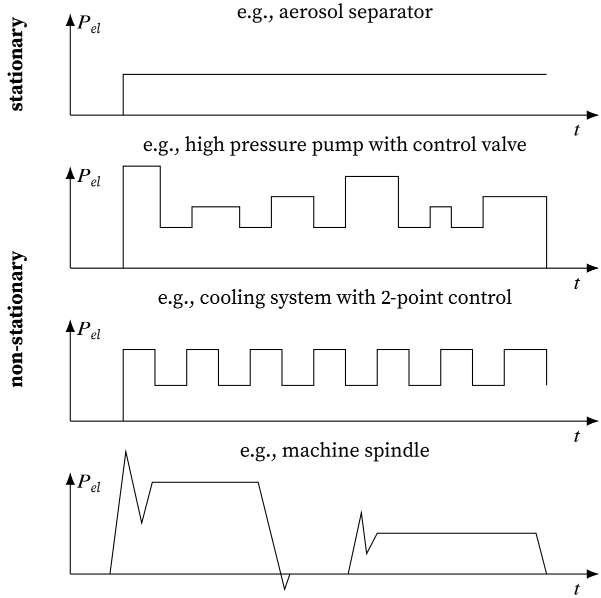


Figure 2.8: Electric power consumption of various production machine component types; translated from German (Kuhrke, 2011, p. 65).

The different types of components, according to Kuhrke (2011, p. 65), are illustrated in Figure 2.8. However, as Kuhrke (2011, p. 65) finds, the energy consumption characteristics of individual components can often be ignored when the goal is to model total energy consumption. To determine which components must be included in a model, Schrems (2014) analyzes each component individually to determine how relevant its characteristics are. Components with high average consumption in comparison to the total power consumption of the machine are the most relevant (Schrems, 2014). The takeaway for this thesis is that additional modelling effort on top of the energy states may be required depending on the machine and its components.

2.4. Research Focus Refinement

The fundamentals and definitions outlined in the preceding sections constitute the results of the research clarification phase within the Design Research Methodology.

The research clarification phase revealed areas needing focus refinement due to the abundance of paths to energy-aware production scheduling. Thus, it is crucial to look at all the topics covered above and determine how each topic should be handled throughout this thesis.

The production planning and control tasks are diverse – from production program planning to sequencing and order release, many subtasks may have some influence on the energy consumption of production. Therefore, this thesis aggregates some planning tasks and refers to them as *production scheduling*. Pinedo (2016), who integrates aspects of the mentioned subtasks in the proposed modelling scheme for production scheduling, supports this aggregate view.

This thesis will also focus on one machine environment, namely job shops, due to the varying requirements for production scheduling in different machine environments. Job shops present an attractive challenge for developing and implementing an energy-aware production scheduling system. However, although the focus is on job shops, the insights gained may also be applicable to other machine environments.

The previous sections also show the variety of production-related and energy-related objective criteria for production scheduling. Due to the amount and diversity of objective criteria and companies' varying needs, focusing this research on optimization algorithms that perform well with multiple objective criteria is sensible. The objective criteria chosen in this research should also be interchangeable to allow for adaptation to the requirements of different companies. This also affects the algorithm used for optimization – it should be applicable despite changes in the objective criteria without requiring complete re-creation of the entire production scheduling model. Since objective criteria should be interchangeable, a target of the implementation procedure should be choosing and implementing the objective criteria relevant to an individual company or factory.

Data availability and the ability to control the production processes performed within the production system are also essential requirements for implementing energy-aware production scheduling systems. While some of the demand response measures discussed in Section 2.3.2 might necessitate the involvement of workers, the additional work must be kept to a minimum to avoid decreasing process efficiency while energy efficiency and flexibility improve. This thesis focuses on utilizing the capabilities of cyber-physical production systems, and the energy-aware production scheduling system developed in this thesis is a service within the cyber-physical production system. Therefore, connectivity to manufacturing support systems and production machines is crucial and the solution must support modern connectivity frameworks.

Section 2.2.3 discusses various implementation procedures. This thesis uses the common steps identified in that section as a starting point for developing an implementation procedure for energy-aware production scheduling systems. The literature review in Section 3 will show whether there are other approaches proposing implementation procedures. If they exist, such approaches must be considered throughout this work. Apart from that, extensions to the identified common implementation procedures should be the result of this thesis, in addition to tools that aid in their execution. The *discovery and planning* as well as the *development and configuration* phases are most relevant for implementing energy-aware production scheduling systems. The *testing and deployment* phase likely does not have to be adjusted significantly to support energy-aware production scheduling.

Regarding energy considerations, both energy efficiency and energy flexibility are compelling goals for energy-aware production scheduling. A promising way to improve the energy efficiency of production is better management of power consumption during idle times. This path should be considered when developing the energy-aware production scheduling system. Looking at energy flexibility and demand response, Section 2.3.2 shows many of the relevant measures. While all of these measures can be relevant over the entire production planning and control process, measures on the manufacturing control level are most applicable to production scheduling. As the literature review in Section 3 will show, measures on the manufacturing level are also regularly considered in energy-aware production scheduling research. However, since this thesis looks to simplify the implementation of energy-aware production scheduling systems, it makes sense to exclude measures on the manufacturing level for now since their potential interactions with process parameters would complicate the implementation procedure unnecessarily.

If there is no economic relevance, nothing is likely to be implemented. Thus, monetizing energy-aware production scheduling is another issue to be aware of when proposing implementation procedures for energy-aware production scheduling systems. Since many manufacturing companies presumably do not fulfil the entry criteria for wholesale energy markets, the retail markets are most suitable for this thesis. In particular, this thesis focuses on real-time pricing since it offers a good compromise between an exact representation of the energy markets' flexibility needs and simplicity of implementation. Real-time pricing offers more control to companies performing production scheduling than, for example, curtailable load or time-of-use pricing schemes. It is also easier to understand, implement, and execute than, for example, a demand bidding scheme.

In conclusion, the research clarification corroborates the need to study implementation procedures for energy-aware production scheduling in job shops. The specific optimization algorithms and objective criteria, production- or energy-related, are not a focus; however, focus is placed on choosing an optimization algorithm that allows optimizing multiple objectives and quickly changing objective criteria without requiring extensive re-modelling. This work interprets the energy-aware production scheduling as a service within the cyber-physical production system and recognizes the need to support various connectivity frameworks. The goal is to improve energy efficiency and energy flexibility through demand response measures on the manufacturing control level with a real-time pricing scheme for monetization. The results of this thesis are the inclusion of energy awareness in the implementation procedure for production scheduling systems and the development of software tools supporting their deployment. This thesis aspires to expedite the dissemination of energy-aware production scheduling systems in industry.

3. Literature Review

Section 2 examines a host of factors that are generally relevant for the implementation and propagation of energy-aware production scheduling systems. Section 2.4 additionally identifies a set of research focus points. Combined with the research questions defined in Section 1.1, these focus points form the basis for this literature review. The literature review aims to identify other research in the energy-aware production scheduling field and the research gaps that should be addressed. In conjunction with the first research question set out at the beginning of this thesis, the literature review's purpose is to provide insights if and why there are few real implementations of energy-aware production scheduling.

Energy-aware production scheduling is a vast and active field of research, and many systematic literature reviews have been published over recent years. Therefore, this literature review begins by analyzing available literature reviews. These reviews provide a comprehensive overview of the research field. Based on this overview, the later parts of this section focus on actual implementations of energy-aware production scheduling. The last part of this section identifies and discusses the research gaps this thesis will address.

3.1. Prior Literature Reviews

Numerous literature reviews in the field of sustainable manufacturing acknowledge energy-aware production scheduling. Earlier reviews, like Duflou et al. (2012), Garetti and Taisch (2012), and Haapala et al. (2011), analyze the field with a broad focus and identify planning and scheduling as one of many relevant factors. Later reviews focus on more specific aspects of sustainable manufacturing because the number of published articles in the field increased significantly (Bänsch et al., 2021). For example, Narciso and Martins (2020) review the utilization of machine learning for energy efficient manufacturing in general, and Garwood et al. (2018) address the development of energy simulation tools. There are also several reviews with a more specific focus on energy-aware scheduling and scheduling for sustainable manufacturing. The following paragraphs take a more detailed look at some of these reviews because they constitute a good starting point for the literature search in this thesis.

Giret et al. (2015) is a narrative review and one of the first reviews with a focus on scheduling for sustainable manufacturing. The review states that at the time, there was a strong focus on the input factors that enable production, for example, the energy consumption of production processes, but little emphasis was given to the outputs (e.g. carbon emissions) (Giret et al., 2015). The authors argue that the outputs are even more relevant to achieve sustainability than the inputs because processes with relatively sustainable inputs but high carbon emissions exist (Giret et al., 2015). However, Giret et al. (2015) also find that developing global approaches to energy-aware production scheduling is a challenging task in itself. Additionally considering input and output factors while also allowing for rescheduling requires support by further paradigm shifts (Giret et al., 2015). In this regard, they identify cyber-physical production systems, intelligent products and new scheduling algorithms as some of the possible solutions (Giret et al., 2015). One of the most important conclusions from this review is that there is a need for realistic research production systems where the inputs and emissions from energy-aware production scheduling can be measured precisely (Giret et al., 2015). Such production systems could also provide the research community with benchmarks to test new algorithms and procedures and facilitate the effective development of energy-aware production scheduling (Giret et al., 2015).

The reviews by Biel and Glock (2016) and Gahm et al. (2016) are conducted only about one year later, in 2016. They both take the approach of a systematic literature review, which contrasts with the review by Giret et al. (2015). Biel and Glock (2016) and Gahm et al. (2016) use two sets of keywords to narrow the focus of their search. The first set of keywords is related to production planning and control, and the second set is related to energy. With this approach, they found 89 (Biel & Glock, 2016) and 87 articles (Gahm et al., 2016), respectively.

Gahm et al. (2016) differentiate three dimensions when they classify literature: the systems covered by the approach (for example, whether energy conversion is included), the energy supply (whether energy is supplied by internal infrastructure or from the grid), and the energy demands (whether additional factors apart from the processing energy demand are considered). Most of the articles Gahm et al. (2016) found concentrate on energy consumption within the production system and neglect energy conversion. Most articles that use grid power supply use a time-of-use pricing scheme for demand response; incentive-based programs are not widely considered (Gahm et al., 2016). Notably, many articles do not incorporate the production job in their energy consumption calculations (Gahm et al., 2016). However, articles usually account for differences in energy consumption resulting from using different machines (Gahm et al., 2016). Gahm et al. (2016) put some effort into evaluating the energy savings the reviewed articles report. Their evaluation proves that the energy-aware production scheduling approach is viable. In their conclusion, Gahm et al. (2016) highlight the fragmentation of energy-aware production scheduling research and assert that there is an urgent need for a common classification framework, which they provide. Bänsch et al. (2021) later reuse and extend the classification framework proposed by Gahm et al. (2016). Gahm et al. (2016) identify multi-objective models and more specialized solution algorithms as relevant areas for future research. They also reiterate the need for benchmark instances as well as a deeper understanding of the energy characteristics of involved systems and their relevance to the scheduling problem (Gahm et al., 2016).

In contrast to Gahm et al. (2016) and Giret et al. (2015), Biel and Glock (2016) put much more emphasis on the different planning horizons in the production planning and control hierarchy (refer to Section 2.1.1). In addition to energy-aware production scheduling, they include master production scheduling, capacity planning, and lot-sizing in their review (Biel & Glock, 2016). The authors differentiate between machine environments (which they call machine scheduling approaches), equivalent to the other reviews. The authors' quantitative analysis shows that most research in the field concentrates on energy-related cost and total energy consumption as objective criteria and uses processing and idle energy consumption in their models (Biel & Glock, 2016). Biel and Glock (2016) confirm the finding that energy-aware production scheduling is a viable option to improve sustainability in production processes but that, while there is no significant equipment investment required, the implementation requires a deep understanding of the underlying energy consumption characteristics and consultation with experts in the field is necessary to find satisfactory solutions. Similar to the other two reviews from the time, Biel and Glock (2016) identify a more global view on energy conversion and production scheduling as an area for further research. They suggest more research on integrating renewable energy sources and modelling CO_2 emissions (Biel & Glock, 2016). Broader consideration of the available demand response schemes is also on their agenda (Biel & Glock, 2016). Another contribution by Biel and Glock (2016) is the proposal of a standardized nomenclature for energy-aware production planning and control.

The three following reviews are more current than those previously introduced – they should also provide some insights into the progress in the field since these first reviews. Bänsch et al. (2021) expand explicitly upon the prior reviews by Biel and Glock (2016) and Gahm et al. (2016). With 192 articles, despite further narrowing the search and only looking at the four years between 2016 and 2020, Bänsch et al. (2021) found more than twice as many articles as the two previous reviews. Combined with the increasing trend in publications found by Bänsch et al. (2021), this reiterates the surge in research interest the field of energy-aware production scheduling received. Bänsch et al. (2021) narrow the search by adding a third set of keywords, limiting the results to mathematical optimization and heuristics. Gao et al. (2020) apply a similar strategy in their search but are not as rigorous in their documentation. These two reviews supplement each other since Bänsch et al. (2021) use the SCOPUS database, and Gao et al. (2020) use WEB OF SCIENCE.

Bänsch et al. (2021) propose the most comprehensive classification scheme and use ten criteria, including elements from the classification schemes by Biel and Glock (2016) and Gahm et al. (2016). The criteria include the energy supply with on-site generation and grid power, objective criteria (which they call planning objectives), the production planning and control hierarchy (they simplify this as planning horizon), machine environments (which they refer to as manufacturing model), and more (Bänsch et al., 2021). Interestingly Bänsch et al. (2021) have some findings that are similar to the earlier reviews. However, they also argue that many research areas identified by Biel and Glock (2016) and Gahm et al. (2016) have seen positive development.

In particular, Bänsch et al. (2021) identify the *system boundary*, the *conceptualization* and the *practical relevance* as the areas of future development found by the earlier reviews. Regarding the practical relevance of research, Bänsch et al. (2021) identify 48 articles that apply their findings to real-life case studies. Additionally, they find substantial progress in all development areas, with some papers even addressing all three (Bänsch et al., 2021). Bänsch et al. (2021) proceed to identify the primary research streams that have seen prominent advancements. These are *on-site generation environments, layout and process planning, assembly line balancing, dynamics and rescheduling, multiple forms of energy*, and the *integration of transportation processes*.

In Gao et al. (2020), the authors discover 90 publications – in contrast to Bänsch et al. (2021), they narrow their search to include only what they call *intelligent strategies* for optimization. They use the keywords *swarm intelligence, evolutionary algorithm*, and *meta-heuristic* to identify intelligent strategies (Gao et al., 2020); however, they do not rigorously document their methodology. Gao et al. (2020) use five criteria to classify papers, the machine environment (they refer to this as shop floor category), the model formulation (they call this problem model), the objective criteria (they call this scheduling objectives), the optimization algorithm and the energy demand aspects (the authors name these energy consumption aspects). The authors note that the percentage of genetic algorithms used for production schedule optimization is 52 %; the NSGA-II alone is used in 14 % of the analyzed articles (Gao et al., 2020). A significant outcome of Gao et al. (2020) is that few articles currently address practical implementations outside of special cases. They also observe a need for more unified frameworks and benchmarks for energy-aware production scheduling (Gao et al., 2020).

Dos Santos et al. (2023) take a more application-oriented view on energy-aware production scheduling research. The authors include the industrial sector to which the research applies in their classification scheme. They also use a more application-oriented approach to classify machine environments (they refer to this as manufacturing organization levels) by differentiating three levels: the *unit-process level*, *multi-machine level*, and *factory level* (Dos Santos et al., 2023). The systematic review by Dos Santos et al. (2023) adds value due to its unique approach to the classification and analysis of literature. In addition, the authors also analyze articles that apply to real practical scenarios and find that the implementation of demand response in industrial companies remains a considerable challenge (Dos Santos et al., 2023). They see some barriers to implementing energy-aware production scheduling as reasons for the scarcity of implementations. In particular, they mention lacking economic benefits and organizational obstacles (Dos Santos et al., 2023). On the other hand, they note that the Industrial Internet of Things will be convenient for deriving new energy models of factories by improved monitoring. Combined with more accurate multi-objective models, better monitoring could improve the profitability of investments in energy-aware production scheduling (Dos Santos et al., 2023).

The literature reviews introduced in this section provide a rough overview of the vast field of research concerning energy-aware production scheduling. As the articles found by each reviewer show, there has been copious research in energy-aware production scheduling, especially in recent years, where sustainability has become increasingly important. However, the reviews also show that agreement regarding the classification of research in the field has yet to be achieved. The classification scheme introduced by Gahm et al. (2016) and extended by Bänsch et al. (2021) is the most comprehensive currently available and provides categories for almost anything relevant. However, articles in the field have not widely adopted it, and authors sometimes leave the exact specifics of their research unclear due to inexact wording.

Giret et al. (2015) realize early on that there is a need to more accurately represent actual production systems by considering input and output factors while also allowing for rescheduling in case something goes wrong during the execution of production schedules. They identify cyber-physical production systems combined with realistic research production systems as a pathway for more effective development of energy-aware production scheduling approaches. The reviews agree that more accurate representations and a global view of energy consumption and conversion would help to improve the scheduling models. Bänsch et al. (2021) found positive developments in these areas, with more articles expanding the system boundaries and using real use cases to validate their models. However, Dos Santos et al. (2023) found that implementing demand response in industrial settings remains a significant challenge.

Overall, there still seems to be a lack of models that accurately represent production systems and are fit for implementation in real industrial use cases. Therefore, to answer the first research question of this thesis, an additional review of articles focusing on implementation in real use cases and considerations made concerning this is required.

3.2. Search Procedure, Criteria and Literature Analysis

Due to the apparent lack of implementations that the review of secondary literature above suggests, it makes sense to perform a more detailed literature review focusing on real practical implementations. This area can be approached from two angles: by looking at research that performs actual implementations or by looking at research that analyzes the barriers that prevent industrial implementations. The next part of this section introduces the search procedure and criteria used to select relevant articles. An analysis of the selected articles and a discussion of the results under the premises of the first research question posed in Section 1.1 follows afterwards.

3.2.1. Definition of Search Procedure and Criteria

The systematic literature review performed for this thesis adopts the procedure suggested by Hochrein and Glock (2012). Figure 3.1 illustrates the procedure, which ensures that appropriate care is taken when selecting search terms and inclusion or exclusion criteria. The method also emphasizes data evaluation and critical analysis of sampled articles.

This review will use the metadata databases Web of Science Core Collection and EBSCO Business Source Premier. The search is performed in April 2023 and is limited to articles published between 2016 and the search date. Due to the abundance of articles in the field, it is essential to find categories that limit the

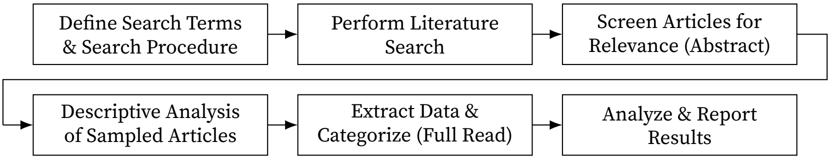


Figure 3.1: Procedure for a systematic literature review as used in this thesis (own figure; Hochrein & Glock, 2012).

sample to the most relevant research. The review should thus be constrained to energy-aware production scheduling in job shops. Most of the previously analyzed literature reviews use two keyword sets for energy-aware production scheduling: production planning-related keywords and energy-related keywords. Since this choice led to good results, this review also adheres to that strategy and extends it by requiring selected articles to be about job shops. This review uses a third category of implementation-related keywords to limit the sample to articles concentrating on actual implementations. Some of the implementation-related keywords match articles describing the implementation itself, and others match articles about implementation barriers. These deliberations yield the following list of keywords:

- job shop
- Production planning-related:
  - scheduling
  - planning
- Energy-related:
  - demand response
  - energy
  - sustainable
- Implementation-related:
  - use case
  - industrial case
  - case study
  - obstacle
  - barrier
  - challenge
  - experience

The search is limited to the article metadata available in the Web of Science Core Collection and Business Source Premier databases. Available fields include the title, abstract, keywords, and other fields like funding details and research area. The search logically combines the keywords in each category with the “OR”-operator and the categories with the “AND”-operator. The resulting search string requires that one keyword from each category is present:

```
  "job shop"
  AND ("scheduling" OR "planning")
  AND ("demand response" OR "energy" OR "sustainable")
  AND (
    "use case" OR "industrial case" OR "case study"
    OR "obstacle" OR "barrier"
    OR "challenge" OR "experience"
  )
```

The search produced 78 results from the Web of Science Core Collection and 20 results from Business Source Premier. Unfortunately, some of these results were duplicates, and the total number of unique articles found is 84. During the first screening, articles are included or excluded based on the title and abstract. Expressly, articles that fall into the following categories are excluded:

– Articles focusing on scheduling in other sectors like power grid maintenance, sensor networks, data centers, or quality control. For example, Aizpurua et al. (2021), Cao et al. (2018), Huang et al. (2021), Ibrahim et al. (2022), Orozco-Santos et al. (2021), and Z. Wang et al. (2022),
– Articles talking about demand response or scheduling from a supply-side point of view, such as Elliott et al. (2019), Shao et al. (2021), and Vahid-Ghavidel et al. (2021)
– Articles using scheduling as an example but focusing on something else, such as simulation. The articles in this category are Back et al. (2016) and Bai et al. (2022).
– Articles performing higher-level optimization of production systems like layout planning. Elahi (2021) falls into this group.
– Articles that do not consider energy in their scheduling approach, like Berti et al. (2021), Denkena et al. (2021), Kowalski et al. (2022), Ramya et al. (2019), and Q. Zeng et al. (2019)
– Articles carrying out scheduling for machine environments apart from job shops, such as Cheng et al. (2020), Rubaiee et al. (2019), Ujazdowski and Piotrowski (2022), and Z. Zeng et al. (2018).

- Articles mentioning in the abstract that their case study is only theoretical or numerical and thus not a real industrial use case, like Pereira and Lima (2018), Sun et al. (2020), and Zarte et al. (2021). The abstract often does not clearly state whether the included case study is a real industrial use case or a numerical or theoretical study. When in doubt, studies were included rather than excluded.
- Articles reviewing other articles without introducing new approaches (secondary literature) like Alvarez-Meaza et al. (2021) and He and Smith (2021).

The first screening excluded 52 articles according to the exclusion criteria. The 32 remaining articles constitute the sample for the detailed literature analysis. During the second scrutiny, which included a full read of the remaining sampled articles, the following articles were excluded because they met one of the exclusion criteria: Lotfi et al. (2021) (does not perform production scheduling) and Y. W. Zhang (2022) (deals with the evaluation of sustainability indicators, not production scheduling directly).

3.2.2. Descriptive Analysis and Classification Scheme

After the selection process, 30 articles are in the sample for a more thorough analysis. As illustrated in Figure 3.2, the Journal of Cleaner Production emerged as the leading journal in the field, with about 23 % of the sampled articles published there. The prior literature reviews introduced in Section 3.1 obtained similar results. Sustainability published three articles, while other journals only published one article each. Five articles were published in conference proceedings; however, all were presented at different conferences.

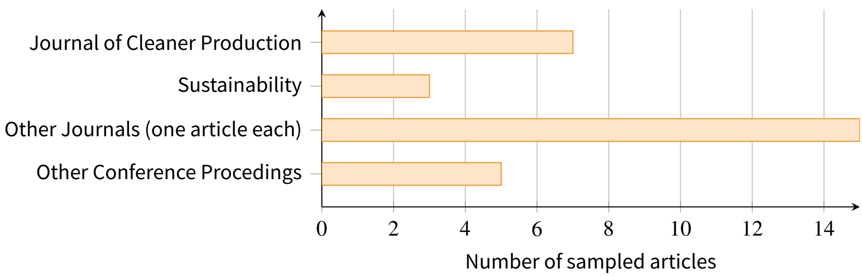


Figure 3.2: Number of sampled articles published per journal (own figure).

As Figure 3.3 shows, there has not been a significant increase in the number of articles published per year over the studied period. Two additional articles were published until April 2023; these are not shown in the figure to prevent misinterpretation because additional articles might still be published in 2023.

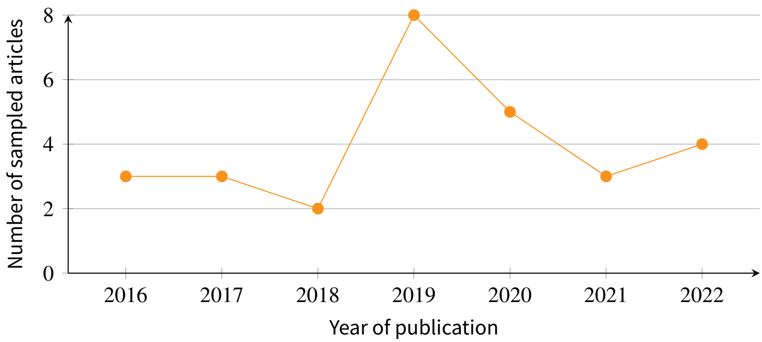


Figure 3.3: Number of sampled articles published per year until 2022 (own figure).

Bänsch et al. (2021) developed a comprehensive classification scheme for articles that incorporates their literature review findings presented above. This work adapts their literature classification by adding additional categories to evaluate the sampled articles regarding their focus on real industrial use cases and implementation. To conform to the nomenclature of this thesis, some of the categories Bänsch et al. (2021) propose are renamed here; however, the identifying letters remain the same. Bänsch et al. (2021) proposed the categories A to J, and the categories K and L were added to their scheme. Categories C *energy storage*, F *machine environment*, H *planning horizon* and I *model type* are omitted because they do not provide relevant insights. In particular, the machine environment is irrelevant because the search only included articles concerned with job shops. In the other omitted categories, there was scant variation within the sampled articles. Most articles do not include energy storage and perform short-term planning with mixed integer programming models. As a result, the following classification scheme is used to group the sampled articles:

- A Energy supply
- B Energy demand
- D Objective criterion
- E System of objectives

- G Mode characteristic
- J Solution algorithm
- K Use case
- L Implementation procedure

For more detailed descriptions of categories A to J, please refer to Bänsch et al. (2021). Since this review is more focused than the review by Bänsch et al. (2021), it does not include some of the sub-categories they used.

In the A *energy supply* category, there is some differentiation regarding how the energy is supplied to the production processes. Many articles do not specify the energy's origin and just assume it is available. These articles are classified as *electric grid (off-site)*. Some articles in this category consider energy prices, for example, using the time-of-use pricing scheme or other pricing schemes. The *other off-site* category is for articles that consider other energy forms apart from electricity. Some articles additionally utilize *non-adjustable on-site* electricity supply, for instance, from photovoltaic installations.

Within category B *energy demand*, all articles include energy consumed while machines are in the *working* energy state. Some articles also include energy consumed while machines are in the operational or standby states. Other articles specifically differentiate energy used during setup, tool change, or energy state transitions. This thesis calls these aspects *processing-related*. A few articles also model *non-processing* energy demand aspects like transportation or customer deliveries.

The most common D *objective criteria* set by the articles are *energy-related costs, total energy consumption,* and *makespan*. Some of the sampled articles set *other energy-related* and *other production-related* criteria like the difference between renewable power supply and power demand or the total workload of machines. Regarding the E *system of objectives,* differentiation between *single-objective* and *multi-objective* optimization is possible. Some articles also perform multi-objective optimization on multiple levels.

Category G *mode characteristic* differentiates between models that allow production jobs to be processed with different energy consumption depending on the machine or process parameters. For example, in a *multi-mode* model, the same production job could be processed with different speeds or on multiple machines with different energy consumption characteristics. *Single-mode* models exclude these degrees of freedom.

In category J *solution algorithm,* most articles use heuristics to solve their mathematical programming problems. *Genetic algorithms* are the most pervasive, but some articles also use various *other heuristics*. Finally, a few articles use exact solution algorithms.

Compared to the classification scheme proposed by Bänsch et al. (2021), the two added categories are K *use case* and L *implementation procedure*. Category K *use case* discerns articles using *theoretical* problem instances to evaluate their model and solution method and articles with *real industrial* use cases. To be classified as a real industrial use case, articles must provide some evidence that they implemented the scheduling approach in a real industrial setting. Articles in this category show some results obtained during the actual execution of their production schedules. There are also intermediate articles which use data from real industrial use cases but do not include an actual implementation of their schedules. Such articles are classified as *numerical* studies.

Within the articles with a numerical or a real industrial use case, category L *implementation procedure* identifies articles that detail what it takes to implement energy-aware production scheduling in industrial use cases. Relevant details could include managerial aspects as well as technical aspects. Some articles may provide *limited insights* into these aspects, for example, by outlining preconditions they considered when creating their model. Other authors describe the *full procedure* they used to develop and implement their approach.

Tables 3.1 and 3.2 represent the resulting classification of the sampled articles. An analysis of the results follows in Section 3.2.3.

3.2.3. Analysis and Classification of Sampled Articles

The following examines the various approaches mentioned in the sampled articles, focusing on each classification category. This analysis aims to offer further insights into the first research question introduced in Section 1.1: *Can a lack of implementations be attributed to an absence of procedures and tooling for implementing energy-aware production scheduling systems in job shops?*

A Energy supply considerations vary little between the articles – all sampled articles include an off-site electricity supply in their models. This result interconnects with the results in category D *objective criterion,* where most articles focus on total energy consumption as the energy-related objective. The focus on energy consumption instead of energy-related costs or other energy-related objective functions means that few articles consider pricing schemes to enable demand response. Notable exceptions to this rule are Ayyoubzadeh et al. (2021), Selmair et al. (2016), and Y. Wang et al. (2019). Ayyoubzadeh et al. (2021) follow the objective

Table 3.1: Classification of sampled articles
(*: multi-level approach. **: unclear classification, GA: genetic algorithm).

	Alotaibi et al. (2016)	Ayyoubzadeh et al. (2021)	Bokah and Maheri (2021)	Coca et al. (2019)	Dai et al. (2018)	Dai et al. (2019)	Feng et al. (2020)	Jiang et al. (2022)	Liao and Wang (2019)	Liu, Guo, and Wang (2019)	Liu, Guo, Wang, et al. (2019)	Liu et al. (2022)	Lv et al. (2022)	Mirahmadi and Taghipour (2019)	Qu et al. (2022)
A Energy supply															
non-adjustable on-site			x												
electric grid (off-site)	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
other off-site				x					x						
time-of-use pricing															
other pricing scheme		x													
B Energy demand															
working	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
processing-related	x	x	x	x	x	x		x				x	x	x	x
non-processing			x		x	x			x	x		x			x
D Objective criterion															
energy-related costs									x						
total energy consumption	x				x	x	x	x		x	x	x	x		x
other energy related		x	x	x											x
makespan	x		x	x	x	x	x	x		x	x	x	x	x	x
other production related		x		x			x	x	x		x				x
E System of objectives															
single objective														x	
multi-objective	x	x	x	x	x	x	x	x	x*	x	x	x	x		x
G Mode characteristic															
single-mode		x	x	x	x	x		x	x						x
multi-mode	x						x			x	x	x	x	x	
J Solution algorithm															
heuristic (GA)		x	x	x	x	x	x		x	x	x			x	
heuristic (other)	x							x				x	x		x
exact															
K Use case															
theoretical	x		x		x	x								x	
numerical		x					x	x	x	x		x	x		x
real industrial				x							x				
L Implementation procedure															
limited insights							x				x	x			
full procedure				x											

Table 3.2: Classification of sampled articles (Continued)
(*: multi-level approach. **: unclear classification, GA: genetic algorithm).

	Ramos et al. (2020)	Selmair et al. (2016)	Tian et al. (2023)	H. Wang et al. (2018)	J. Wang et al. (2021)	J. Wang et al. (2020)	J. Wang et al. (2019)	Y. Wang et al. (2019)	Xu and Wang (2017)	Yang et al. (2016)	Yin et al. (2017)	W. Zhang et al. (2023)	X. Zhang et al. (2020)	Y. Zhang et al. (2017)	Zhu et al. (2020)
A Energy supply															
non-adjustable on-site	x														
electric grid (off-site)	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
other off-site															
time-of-use pricing	x**	x						x							
other pricing scheme															
B Energy demand															
working	x**	x	x	x	x	x	x	x	x	x	x	x	x	x	x
processing-related		x	x	x	x	x	x		x		x		x	x	
non-processing			x						x						
D Objective criterion															
energy-related costs	x**	x						x							
total energy consumption			x	x	x	x	x		x	x	x	x	x	x	
other energy-related													x		x
makespan			x		x	x	x	x	x	x	x	x	x	x	
other production-related			x	x	x	x	x				x			x	x
E System of objectives															
single objective	x**	x							x						
multi-objective			x	x*	x*	x*	x	x		x	x	x	x	x*	x
G Mode characteristic															
single-mode					x			x	x			x		x**	
multi-mode	x**	x	x	x		x	x			x	x		x		x
J Solution algorithm															
heuristic (GA)	x							x	x	x	x		x		x
heuristic (other)			x	x	x	x	x					x		x	
exact		x													
K Use case															
theoretical		x				x	x		x				x	x	x
numerical	x		x	x	x					x	x	x			
real industrial								x							
L Implementation procedure															
limited insights				x	x	x	x	x							
full procedure															

of reducing the power consumed by the factory. To achieve this, they use taxes on surplus power consumption as an objective function in addition to lateness cost (Ayyoubzadeh et al., 2021). In contrast, Selmair et al. (2016) and Y. Wang et al. (2019) use a more typical time-of-use pricing scheme to determine and optimize energy-related costs.

A few articles also include energy supply other than the electricity grid. For example, Bokah and Maheri (2021) and Ramos et al. (2020) incorporate non-adjustable on-site generation. Ramos et al. (2020) limit this to photovoltaic power, while Bokah and Maheri (2021) consider power available from a more generic renewable energy system, including energy storage. The model proposed by Bokah and Maheri (2021) also focuses on predicting photovoltaic power generation using irradiance values from weather data. They use the knowledge about available renewable power to reduce the energy consumed from the electricity grid (Bokah & Maheri, 2021). Coca et al. (2019) go even further and include a variety of energy supply sources in their multi-objective model. They include other energy forms as well as the production of waste, and they also include optimization of social aspects of production (Coca et al., 2019).

B Energy demand aspects studied by the sampled articles frequently include energy consumption during the working state of production machines. A large share of the articles also integrate other processing-related energy demand aspects like the energy consumption during the setup of parts and while machines are in the operational and standby states. Bokah and Maheri (2021) and Qu et al. (2022) additionally include aspects like lighting, ventilation, or air conditioning. Others like Dai et al. (2018), Liu, Guo, and Wang (2019), and Tian et al. (2023) include transportation within the factory, and Liao and Wang (2019) incorporate customer delivery into their model.

D Objective criteria are diverse within the sampled set of articles. When it comes to production-related criteria, most articles use makespan as their primary criterion. Correspondingly, among the energy-related criteria, total energy consumption is the most commonly used within the sample. However, some articles like Selmair et al. (2016) and Y. Wang et al. (2019) incorporate pricing schemes in their energy supply models and focus on energy-related costs instead of total energy consumption. Ayyoubzadeh et al. (2021) have an objective criterion that depends on peak power pricing, while X. Zhang et al. (2020) optimize peak power consumption in addition to total energy consumption. Zhu et al. (2020) also have a unique approach where they optimize for parts per emission and profits per emission. These objective criteria are interesting as they combine production-related and energy-related aspects into a single objective criterion.

Coca et al. (2019) take a rare path by creating a model with three objectives: one each for economic, environmental and social performance. Each objective criterion includes multiple weighted factors such as makespan, CO_2 emissions, water consumption, sound intensity and weights manipulated by workers (Coca et al., 2019). Qu et al. (2022) also establish a model with many objectives – they optimize the total cost of production, CO_2 emissions, and the amount of power used by machines to process parts compared to the total power consumption. Other studies, including Feng et al. (2020), Jiang et al. (2022), Liu, Guo, Wang, et al. (2019), Tian et al. (2023), and H. Wang et al. (2018), also incorporate production-related cost indicators.

E Systems of objectives and G mode characteristics provide additional insights into the complexity of models presented in the sampled articles. Most articles perform multi-objective optimization, some of them using a multi-level approach. Only some articles propose single-objective models with either energy-related or production-related constraints. Xu and Wang (2017) use the intermediate technique and perform single objective optimization of makespan and total energy cost by assigning weights to both objectives. In contrast, the mode characteristics are distributed relatively evenly, with 17 articles studying multi-mode models and 13 articles studying single-mode models.

J Solution algorithms employed by the sampled articles are mostly heuristics. Of these, genetic algorithms are the most common, with the NSGA-II being one of the most frequently used. However, most researchers adapt their algorithms to suit their specific problem better. Even more than within the group of genetic algorithms, this is visible in the articles using other heuristics. The heuristics range from simple rule-based approaches, as used by Alotaibi et al. (2016), over bee colony optimization (e.g., Jiang et al., 2022; Tian et al., 2023) and game-based approaches (e.g., J. Wang et al., 2021; Y. Zhang et al., 2017) to heuristics simulating the behaviour of grey-wolf packs (see: W. Zhang et al., 2023). Finally, one article in the sample performs exact optimization (see: Selmair et al., 2016).

K Use cases and L implementation procedures are rarely described in detail in the sampled articles. Although numerical studies using data from real industrial use cases are common, most do not explain how the data was collected or how the results could be implemented in reality. However, some studies provide insights into important requirements for actual implementations, particularly regarding aspects of the optimization model. For instance, transportation is a significant constraint (e.g., Dai et al., 2018; Liu et al., 2022). Many articles also mention the stochastic behaviour of production systems due to timing inaccuracies, machine breakdowns or rush jobs. This is addressed through rescheduling (e.g., Lv et al., 2022; Xu & Wang, 2017; Zhu et al., 2020) or stochastic models (see: Yang et al., 2016). Some authors, for example, Feng et al. (2020), J. Wang et al. (2020), and J. Wang et al. (2021) suggest using edge computing to gather data from production systems and to control machines; however, all of them remain conceptual and do not show actual implementations. Y. Wang et al. (2019) is an exception – the article contains a detailed description of the energy measurements and preliminary analysis performed before the authors propose a solution through production scheduling.

Despite this, the sample only includes three articles describing real industrial use cases. Coca et al. (2019) worked with a company in the metal machining sector to create a many-objective model and to implement it in production. The various aspects of the model are developed and parameters are estimated in cooperation with the industrial partner (Coca et al., 2019). The authors describe their implementation procedure and the aspects that must be considered when implementing a production scheduling model with environmental and other aspects in a company in detail. The implementation procedure consists of 22 steps in four phases: planning, information collection, evaluation of the multi-objective solutions, and decision-making (Coca et al., 2019). The schedules generated in this research are optimized once, and priorities and products are assumed not to change over time; otherwise, the implementation procedure would have to be partially repeated (Coca et al., 2019). The last two phases of the procedure, which focus on identifying the best solution generated by the optimization, emphasize this need.

The authors Liu, Guo, Wang, et al. (2019) implement their production scheduling model at a cement equipment manufacturer and provide some insights into their experience. Like Coca et al. (2019) they calculate feasible production schedules in advance and allow for some managerial processes before implementation of a production schedule. In particular, they identify the problem that while the parent company wants to improve energy efficiency, some of its subsidiaries might reject production schedules with lower energy consumption because they result in higher overall production costs (Liu, Guo, Wang, et al., 2019). Liu, Guo, Wang, et al. (2019) determine production costs by considering material costs, labour costs, production costs and costs for delay claims by customers. They find that multi-objective optimization can achieve solutions with lower energy consumption and costs, facilitating adoption in their use case (Liu, Guo, Wang, et al., 2019).

Liu et al. (2022) build on the same use case as Liu, Guo, and Wang (2019) and include automated guided vehicles instead of cranes for transportation. The authors also add multiple machine speeds and energy states to the use case and use a newly developed heuristic optimization algorithm to solve the production scheduling problem. Liu et al. (2022) report energy savings of up to approximately 40 % with their approach compared to manual dispatching of production jobs. However, the article does not specify exactly how they conducted the experiment – it appears like their optimized schedule was executed by manual dispatchers.

3.3. Conclusion and Research Gap

For the conclusion, it is necessary to go back and examine the first research question: *Can a lack of implementations be attributed to an absence of procedures and tooling for implementing energy-aware production scheduling systems in job shops?* The literature review shows that few researchers have investigated the implementation of energy-aware production scheduling for job shops in real industrial use cases. On the other hand, many sampled articles have more or less thoroughly considered the requirements for an actual implementation when developing their models. Many authors have additionally cooperated with industrial partners to create numerical use cases which they use to test their models. However, in most cases, the details of how authors obtained data and which challenges might be incurred when trying to implement the optimization in the actual production system are not available. It is interesting to note how few of the numerical studies have led to actual implementations. There seem to be barriers to the implementation that research has so far failed to address fully.

Overall, the lack of industrial use cases in energy-aware production scheduling research equates to a lack of research into the actual implementation of energy-aware production scheduling in real production systems. Thus, there is likely a significant discrepancy between the approaches proposed in scientific literature and the requirements of actual implementations. Overcoming this discrepancy would mean identifying and surmounting the barriers preventing real implementations. The answer to the research question follows from the preceding analysis: a lack of research into real implementations led to an insufficient understanding of the barriers preventing implementations which entails a scarcity of procedures to overcome these barriers. When looking for actual industrial implementations of energy-aware production scheduling their scarcity underpins this conclusion.

Nevertheless, the literature review helps by gathering the requirements other authors have identified. For example, some studies include requirements for the scheduling model, the software and Industrial Internet of Things devices needed to automate control of production machines. Concerning the requirements for the scheduling approach itself, many authors agree on the following points:

- Scheduling models should include all factors with a meaningful impact on the execution of production schedules. This can include transportation, different machining speeds, or machine breakdowns; however, the factors relevant to a specific system differ and must be established prior to model development.
- In addition to ensuring all relevant factors are incorporated, it is also essential to confirm that changes in the real production system, such as machine breakdowns, are reflected in the production schedule and do not lead to unnecessary downtimes. Possible solutions to this include stochastic optimization and rescheduling.
- Multi-objective optimization is the technical basis for including multiple aspects deemed relevant by a company. It is unlikely that companies will completely set aside production-related objectives for energy-related objectives, making multi-objective optimization a prerequisite.
- Sufficient information about the state of the production system must be available: the scheduling system and the actual production system can form a cyber-physical production system to achieve this. Technologies enabling a cyber-physical production system mentioned in the sampled articles include edge devices, track-and-trace systems, and wireless Industrial Internet of Things communication.
- Considering requirements brought forward by any involved or affected stakeholders is instrumental. Failing to do so might lead to failure of the entire implementation.

Most articles provide valid solutions for some of the mentioned requirements; however, almost none comprehensively investigate the requirements and procedures necessary to achieve real implementations. Looking at the resulting research gaps, this literature review's results align with the prior reviews discussed in Section 3.1. The research gaps can be broken down into two main points:

1. There is a need for accurate representations of production systems with a high tolerance for deviations during the execution of production schedules, for example, using rescheduling approaches integrated into cyber-physical production systems.
2. Implementations in real production systems as complete cyber-physical production systems are necessary to evaluate any significant factors that might influence such implementations. Since this is extremely difficult to achieve in actual industrial systems, realistic research production systems might offer a solution.

There is abundant research on how to model and optimize production systems accurately. Thus, the first point focuses on tolerating some deviations while maintaining the required accuracy when following production schedules. The second point highlights the process of actually implementing these models, including developing and parameterizing production scheduling models. Unfortunately, while there are some articles addressing the first point, articles addressing the second point are extremely rare.

This thesis aims to address both points by introducing a production scheduling system architecture that provides energy model parameter estimation and an implementation procedure based on real-life production system requirements. Furthermore, this thesis evaluates both aspects effectively by testing this implementation procedure and energy model parameter estimation within an actual research production system.

4. Conceptualization and Implementation Procedure

This thesis aims to address the research gap identified in the literature review by answering the second and third research questions posed in Section 1.1. Answering the questions requires developing an implementation procedure and a corresponding energy-aware production scheduling system architecture. This section's goal is to aggregate the information gathered in Sections 2 and 3 to derive requirements and success criteria. The requirements motivate the concepts proposed in the subsequent sections. The following differentiates between the implementation team, users, and stakeholders. The implementation team is a team made up of users and representatives for other stakeholders. The team performs the implementation procedure. Stakeholders are people or departments involved in or affected by the implementation of the energy-aware production scheduling system, and users are those people who utilize the system once it is completely implemented.

The first part of this section aggregates the requirements and develops success criteria. The second part builds on the success criteria to establish an implementation procedure, and the third part of this section defines additional requirements for the energy model parameter estimation and energy-aware production scheduling system architecture arising from the implementation procedure.

4.1. Requirements and Success Criteria

This section aggregates the functional and technical requirements for the energy-aware production scheduling system. These requirements later inform the development of the implementation procedure and the system architecture.

Little (1970) understands decision-support systems as an extension of a manager's decision-making process, helping them find and implement their decisions. Thus, energy-aware production scheduling systems are decision-support systems, and the generic functional requirements for decision-support systems stipulated by Little (1970) apply. Additionally, requirements to ensure that the production scheduling systems can directly control production machines must be established. Like Little (1970), the following requirements for an energy-aware production scheduling system assume that the system should be geared towards managerial roles – explicitly production managers.

Simplicity. The system should be as easy to understand as possible – it should only include the required factors and exclude everything else (Little, 1970). This requirement influences, for example, the selection of objective criteria and the complexity of the scheduling model.

Robustness. Ideally, system users should be unable to produce incorrect answers when using the model independent of the input values (Little, 1970). While user inputs, as mentioned by Little (1970), are one possible source of disturbance, the production scheduling system must also be robust against partially incorrect measurements, machine breakdowns and inaccurate timing of production operations.

Controllability. The system should behave as the user expects; conversely, any changes the user makes should lead to the intended consequences (Little, 1970). Controllability impacts the objective criteria of the energy-aware production scheduling system and the handling of changes occurring during the execution of a production schedule.

Adaptivity. When new information becomes available, the models' parameters and structure should be updated (Little, 1970). The energy-aware production scheduling system must be adaptive regarding the production system model and production jobs. During execution it must adapt to machine breakdowns and inaccurate timing of production operations. Adaptivity also includes adapting the entire energy-aware production scheduling system to various uses and production systems.

Completeness. The system should incorporate all factors impacting the objective criteria (Little, 1970). This requirement conflicts with simplicity – the model should reflect a reasonable compromise between the two requirements (Little, 1970). The compromise between completeness and simplicity depends on the goals set by the implementation team. Therefore, the implementation procedure should aim to identify an acceptable compromise.

Interactivity. The system should enable users to easily alter input data and obtain output data (Little, 1970). The energy-aware production scheduling system should provide easy-to-use interfaces for inputting and visualizing output data. In contrast to the other requirements, since this thesis proposes a prototype system, this requirement addresses the needs of researchers instead of managers. Additional interactivity needed by managers can be added later when a final system is implemented.

Connectivity. In addition to the requirements set by Little (1970), providing connectivity between the energy-aware production scheduling system and the production machines is crucial. Connectivity partially alleviates the need for manual data entry during the system implementation and execution of production schedules. Thus, it supplements the robustness and adaptivity requirements.

The technical requirements derive from the functional requirements defined above. However, before defining the technical requirements, it is necessary to look back at the research focus refinement in Section 2.4 to unequivocally summarize the findings, decisions and limitations up to this point. In short, this thesis:

- performs *production scheduling* and does not consider longer-term optimization of production planning and control,
- only considers *discrete production* and applies to the *job shop* machine environment,
- should implement *multi-objective optimization* with energy and production-related objective criteria,
- strives to improve both *energy flexibility* and *energy efficiency*,
- endeavours to implement a *cyber-physical production system* with the production machines as physical components and manufacturing support systems as external entities,
- attempts to provide options for adapting the proposed system to divergent needs of varying users,
- assumes that data from the production system is available and production machines are controllable through *connectivity frameworks*, and
- expects that the production processes are stable enough to estimate processing and setup times with reasonable accuracy warranting automated scheduling.

With this in mind, the following technical requirements for the energy-aware production scheduling system architecture and the implementation procedure are defined:

Collection of Requirements. The implementation procedure should formalize the collection of requirements for the energy-aware production scheduling system. The literature review highlighted the necessity of considering requirements from all involved or affected stakeholders. The implementation procedure should ensure the integration of all relevant facets of the production system in the deployed production scheduling system based on the collected requirements.

Production System Configuration. Different production systems differ in many aspects, for example, the machine environment, the types of production machines, the production jobs and operations, and the Industrial Internet of Things devices. Consequently, the energy-aware production scheduling system architecture should have interfaces for configuring the aspects that could change when adapting the system to various production systems.

Standardized Models. The production machine energy models should provide a standardized interface for production machine models to simplify adapting the energy-aware production scheduling system to different production systems.

Automatic Model Parameter Estimation. The energy model parameter estimation should enable automatic model parameter estimation of the standardized models using data collected from the production system. Automatic model parameter estimation reduces the model creation effort and helps to increase completeness.

Customer Order Input. Users should be able to supply the energy-aware production scheduling system with information about customer orders. The system should use this input to create the production schedule.

Individualized Objective Criteria. As the literature review showed (refer to Section 3.3), the energy-aware production scheduling system should realize all requirements brought forward by the stakeholders. Such requirements likely concern the objective criteria; thus, the system should allow users to select individualized objective criteria during the implementation procedure, and the system architecture should be valid for whichever objective criteria they choose.

Multi-Objective Optimization. The literature review also revealed that successful implementations are most likely when the system can optimize production-related objective criteria simultaneously with energy-related criteria. Hence, the energy-aware production scheduling system should use multi-objective optimization.

Rescheduling. The requirements collected during the literature review include reacting to unexpected events. Thus, the scheduling system architecture should enable rescheduling. This way, production schedules can adapt to changes, for example, caused by machine breakdowns or changing energy prices. Data acquired from the production system and external services helps determine when rescheduling is necessary.

Production System Data Acquisition. According to the literature review, energy-aware production scheduling needs sufficient information about the production system state. The scheduling system should be able to acquire data from production machines, sensors and measuring equipment in the factory to gather this information. Accordingly, the system architecture should incorporate connectivity frameworks supported by production machines and measuring equipment. Optionally including fieldbus communication could provide connectivity to a broader range of machines and equipment.

Controlling Production Machines. The energy-aware production scheduling system should automatically execute production schedules by controlling machines via a connectivity framework. Optionally including fieldbus communication could extend connectivity to a wider range of machines.

Connectivity with External Services. External factors, like energy prices, also affect the production system state. Thus, the energy-aware production scheduling system architecture should facilitate connectivity with external services supplying this data.

Table 4.1: Relationship between the functional and technical requirements.

Technical Requirement	Addresses (Functional Requirement)
Collection of Requirements	Simplicity, completeness
Production System Configuration	Completeness, adaptivity, interactivity
Standardized Models	Simplicity, adaptivity
Automatic Model Parameter Estimation	Adaptivity, completeness
Customer Order Input	Interactivity
Individualized Objective Criteria	Controllability, completeness
Multi-Objective Optimization	Controllability, completeness
Rescheduling	Adaptivity
Production System Data Acquisition	Adaptivity, connectivity
Controlling Production Machines	Connectivity
Connectivity with External Services	Adaptivity, connectivity

Since each technical requirement relates to one or more functional requirements, Table 4.1 presents an overview of the relationships between the functional and technical requirements. Likewise, the technical requirements influence the design of the system's components – the implementation procedure, energy-aware production scheduling system architecture, and energy model parameter estimation. Table 4.2 illustrates which of these components each requirement affects.

Table 4.2: Relationship between the technical requirements and components of the energy-aware production scheduling system.

Technical Requirement	Affected Components
Collection of Requirements	Implementation procedure
Production System Configuration	Implementation procedure, system architecture
Standardized Models	Energy model parameter estimation, system architecture
Automatic Model Parameter Estimation	Energy model parameter estimation
Customer Order Input	System architecture
Individualized Objective Criteria	Implementation procedure, system architecture
Multi-Objective Optimization	System architecture
Rescheduling	System architecture
Production System Data Acquisition	System architecture
Controlling Production Machines	System architecture
Connectivity with External Services	System architecture

Finally, according to the Design Research Methodology (Blessing & Chakrabarti, 2009), this thesis should establish success criteria to be used to evaluate the success of the prescriptive study during the descriptive study-II. While success criteria can be qualitative, measurable success criteria are necessary to succinctly evaluate the results (Blessing & Chakrabarti, 2009). The following success criteria will provide measurable results at the end of this research.

Functional Requirements Fulfilled. The functional requirements ensure that the energy-aware production scheduling system suits its users and provides the necessary functions. Therefore, all functional requirements for the production scheduling system must be met. This criterion should be measured using the number of met requirements – some requirements may only be partially fulfilled.

Technical Requirements Fulfilled. The technical requirements delineate technical aspects deemed essential for a sound production scheduling system. Hence, all technical requirements also must be fulfilled. The number of met requirements is measured to check this criterion. Some requirements may only be partially fulfilled.

Accurate Representation of the Production System. The production scheduling system should represent the actual production system with sufficient accuracy. An accurate representation is especially relevant for the energy-related aspects of the production scheduling model. Appropriate model accuracy measures should be selected and used to evaluate this criterion.

Improved Energy-Awareness of Production Scheduling. Since this research is concerned with energy-aware production scheduling, the resulting scheduling approach should improve the energy-awareness of production scheduling. Energy-awareness can be measured through improved energy efficiency (reduced energy consumption for the same number of production jobs) and reduced energy cost as a measure of energy flexibility.

Sufficient Performance in Production-Related Objectives. While this thesis focuses on the energy-related aspects of production scheduling, the production-related objectives should not be significantly affected negatively by the improvements in energy-awareness. At the same time, small decreases in production-related objectives for larger increases in energy-related objectives can be acceptable. *MKSP* is a possible criterion for measuring the production-related performance of a schedule.

Transferable Scheduling System Architecture. This last success criterion is not directly measurable; however, the descriptive study-II phase of this research should prove that the developed energy-aware production scheduling system is transferable to various production systems. This transferability can be verified by applying the system to multiple machines and production systems and keeping track of the necessary adaptations to the proposed system.

Overall, this section introduced the functional requirements, technical requirements and success criteria. The functional requirements ensure the basic functionality of a decision support system for production scheduling. The technical requirements extend the functional requirements by inspecting the research focus refinement in Section 2.4 and the literature review in Section 3. Finally, the success criteria summarize the requirements and define measures that can be checked when evaluating the resulting energy-aware production scheduling system during the descriptive study-II phase of this thesis.

4.2. Implementation Procedure for Production Scheduling Systems

The literature review (refer to Section 3.3) revealed that additional research on implementing energy-aware production scheduling in real production systems is necessary. Section 2.2.3 discusses approaches to implementing ERP systems and MESs; however, while these procedures do pertain to energy-aware production scheduling systems, they do not sufficiently guide the treatment of energy-related aspects. Additionally, most concepts for energy-aware production scheduling found in the literature review also do not cover their implementation.

Overall, the literature documents many models and optimization algorithms for energy-aware production scheduling but few implementations in job shops. The need to develop complex, individualized models specific to a particular production system likely exacerbates the scarcity of implementations. Research showcasing pathways to more standardized energy-aware production scheduling system architectures and implementation procedures could improve the dissemination of energy-aware production scheduling.

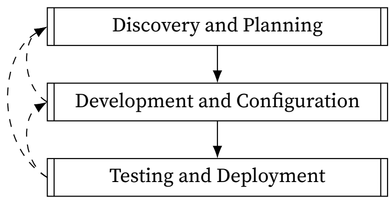


Figure 4.1: Three phases of the implementation procedure (own figure).

Therefore, this section introduces an implementation procedure with three phases based on the procedures studied in Section 2.2.3. The three phases identified in Section 2.2.3 are *discovery and planning, development and configuration,* and *testing and deployment,* as illustrated in Figure 4.1. The figure also shows that the procedure is recursive, and it is possible to return to prior phases as needed. The next three parts of this section detail the procedures performed during each phase and highlight the outputs and results of the phases. The implementation procedure is performed by an implementation team assembled at the beginning of the first phase.

4.2.1. Discovery and Planning Phase

The discovery and planning phase prepares the implementation process – it obliges users following the procedure to put necessary prerequisites in place, analyze their requirements, and create a concrete implementation plan. As mentioned in Section 2.2.3, this phase begins by assembling an implementation team and creating a project plan. Thus, an adequately qualified team (Schuh et al., 2015, p. 363) and sufficient budgeting (Beeson, 2022) for the project are two outputs of this phase. The implementation team should consist of representatives for all stakeholders. As defined in Section 4, stakeholders are people or departments involved in or affected by the implementation or use of the system.

After completing the first two steps, the third output of this phase is a concept for integrating the company's processes into the software system. A detailed analysis of the company's processes and their correlation with capabilities and processes provided by the software are the basis for creating this concept. Where the company's requirements and the software's capabilities diverge, the implementation team should evaluate whether transforming the company's processes or adapting the software is more appropriate (Schuh et al., 2015, p. 339).

Figure 4.2 illustrates the discovery and planning phase in detail. After the two steps of assembling a team and establishing the budget, discussed in the previous paragraphs, it is essential to set the project scope, goals and requirements. The project scope defines the production system boundaries included in the production scheduling system. It also provides information about the typical production jobs encountered within the boundary.

The goals and requirements determine the project's focus and ensure it fulfills all stakeholders' expectations. Besides ensuring that the implementation achieves improvements compared to an existing production scheduling system, the goals should include commercializing the enhanced energy efficiency and flexibility. Ultimately, the requirements should determine all aspects that must be included in the energy-aware production scheduling system to fulfil the company's needs. At the end of this section there is a list of exemplary requirements. Note the difference between the functional and technical requirements for the production scheduling system as a whole, set in Section 4.1, and the requirements mentioned here, which are set by the implementation team to guide the implementation procedure. The literature mentions many methods to determine scope, goals and requirements. For example, the design thinking framework offers many options (e.g., Lewrick et al., 2018). Accordingly, this thesis suggests using an established method to set scope, goals and requirements.

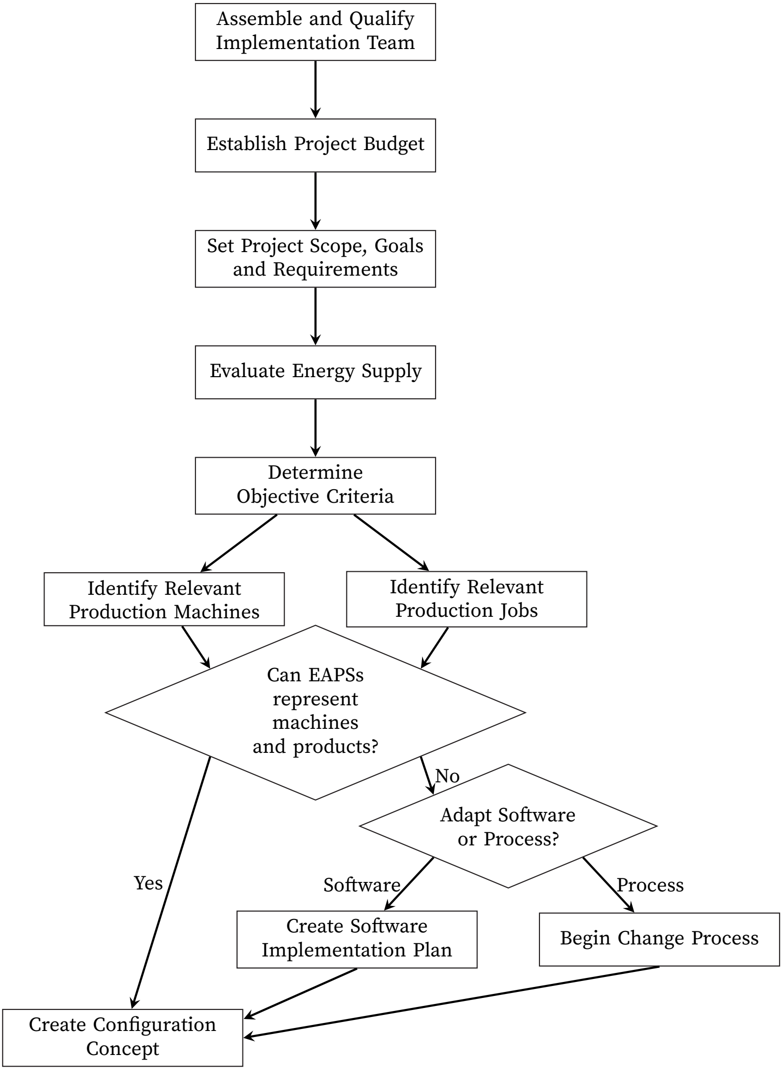


Figure 4.2: Detailed procedure of the discovery and planning phase (own figure). (EAPSs: energy-aware production scheduling system)

The scope, goals and requirements are the basis for determining the objective criteria that the energy-aware production scheduling system should optimize. The objective criteria are distinct for each company, and the implementation team should ensure that the objective criteria are suitable to fulfil the goals and requirements. Since this implementation procedure aims to implement an energy-aware production scheduling system, there should be at least one production-related objective criterion and one energy-related criterion. Some examples of production-related and energy-related objective criteria were introduced in Sections 2.1.1 and 3.2.3 respectively.

Knowledge of the previously determined factors allows identifying the relevant production machines as well as relevant production jobs. As Figure 4.2 shows, these two steps can occur in parallel. Information about the relevant production machines and production jobs is needed to determine whether the energy-aware production scheduling system can represent them. This question must be answered for all machines and production jobs individually, considering the following factors:

- Does the production scheduling system offer configuration options describing the operations required by each production job?
- Are the energy models included in the energy-aware production scheduling system sufficient to represent the machine types present in the production system?
- Do the models incorporate all relevant factors to evaluate the chosen objective criteria?

As Figure 4.2 illustrates, there are three possible answers to the above questions. If the energy-aware production scheduling system already incorporates all necessary aspects, creating a concept for the configuration is the next step. The configuration concept includes a summary of the required data and a plan for gathering data. For example, for configuring a machine tool, required data might include the energy states, power consumption, and products produced on the machine (see also Section 5.2.5). The data-gathering plan should be devised according to DIN ISO 50015 (Deutsches Institut für Normung e. V., 2018). The standard requires planning the data-gathering process by selecting goals before choosing corresponding data values and appropriate data-gathering methods (Deutsches Institut für Normung e. V., 2018). Information needed to create a data-gathering plan according the the standard includes among others (Deutsches Institut für Normung e. V., 2018):

- names of variables,
- necessary data quality (i.e., measurement accuracy, and validation), and
- measuring equipment or technique.

The data-gathering plan could comprise measuring power consumption with external measuring equipment and logging the energy states and products determined by the machine's Programmable Logic Controller (PLC). Other data, for example, regarding production jobs, could be collected manually or from other systems like ERP systems.

If the energy-aware production scheduling system cannot represent a machine or production job, the implementation team can adapt the software or change the production processes. This work focuses on adapting the software since changes to production processes often require specialized knowledge about the products and processes themselves. Changes to the software might become necessary when energy models for specific machine types are not available in the software or when configuration options for relevant aspects of the production processes are unavailable. As an example, this could be the case if transportation times are significant for a specific company but are not considered in the software yet. The implementation team should create a software implementation plan if software adaptations are necessary. The implementation plan determines which parts of the software need to be changed and how they should be changed. For example, this could include implementing mathematical models for additional production machine types. This path also ends in creating a configuration concept (see Figure 4.2 above) for the newly implemented features.

The following list summarizes the outcomes of the discovery and planning phase. Overall, the outcomes should include everything needed for the development and configuration phase. The outcomes of this phase provide the necessary information to begin the next steps in the development and configuration phase, whether they are additional software development or gathering data for configuration. The outcomes also include the criteria needed to evaluate the success of the implementation procedure in the final testing and deployment phase:

- Scope and goals:
    - Production system boundaries
    - Typical production jobs
    - Pricing schemes for energy efficiency and energy flexibility (refer to Section 2.3.2)
    - Goals for improvements compared to an existing system

- Requirements, for example:
  - Planning horizon (refer to Section 2.3.2)
  - Number of jobs and products
  - Types of machines
  - Additional production-related aspects to be considered (e.g., due dates)
  - Energy supply of the company (e.g., available pricing schemes)
- Objective criteria:
  - Production-related criterion
  - Energy-related criterion
- Configuration concepts:
  - Summary of required data
  - Data collection plan
- Software implementation plan:
  - List of features to be implemented
  - Configuration concepts for additional configuration options for new features

The implementation team should ensure that all goals and requirements can be achieved. The team should also verify that all necessary information is available and correct. If the team discovers any shortcomings in these aspects, they have the ability to return to a previous step and make the necessary adjustments. This approach ensures that the implementation process is thorough and effective.

4.2.2. Development and Configuration Phase

After completing the initial discovery and planning phase, the development and configuration phase begins. This phase aims to adapt the system to the company's desires and to fulfil the goals and requirements set in the previous phase (Musil, 2018). Adapting the system includes configuration and software development as planned in the discovery and planning phase (Schuh et al., 2015, p. 374). Successful configuration requires prior data-gathering and assurance of data quality (Beeson, 2022).

Figure 4.3 illustrates the development and configuration phase's procedure. This phase is not necessarily sequential – it is possible to re-sequence steps and execute some steps in parallel if deemed necessary by the implementation team. The phase begins with implementing the objective criteria selected in the previous phase. Most likely, each implementation team will choose slightly different objective criteria meaning that this step will always require some implementation effort (see Section 5.2.4). While or after implementing the objective criteria, the execution of the other software implementation plans begins. The newly implemented features should provide the additional functionality and configuration options set in the implementation plan.

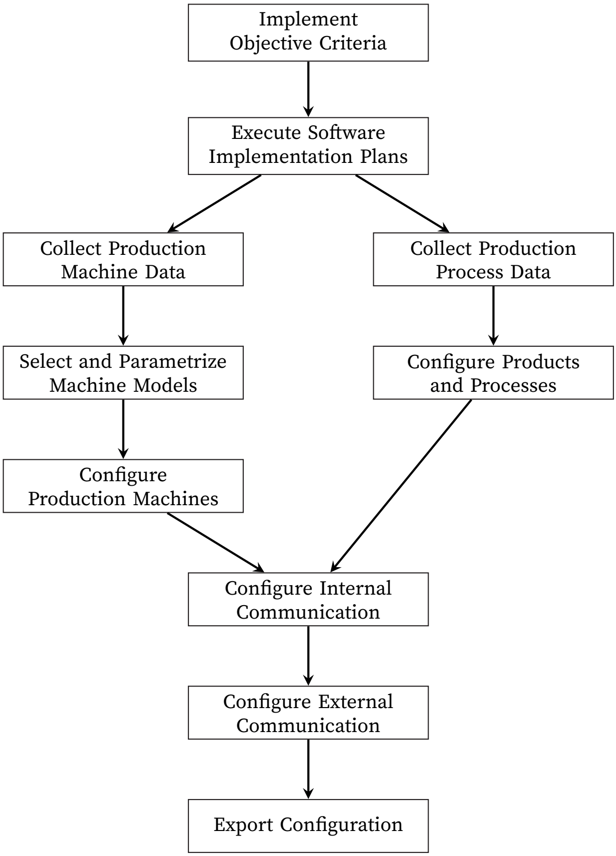


Figure 4.3: Steps of the development and configuration phase procedure (own figure).

Often, newly implemented features will be energy models of production machine types which are not provided by default. Implementations of energy models should closely examine the needed level of detail. While precisely calculating the power consumption of production machines might require very complex physical models, this level of detail is usually unnecessary for production scheduling. Instead, predicting the energy consumption of the machine's energy states can be sufficient. Furthermore, the influence of different production operations can be represented using characteristic factors like the material removal rate. Since the factors influencing energy consumption during each energy mode and the characteristic factors to evaluate the energy consumption of production operations can be challenging to determine manually, this thesis proposes using regression models for this purpose. Regression models also enable the creation of generalized models for types of similar machines.

Once the energy-aware production scheduling system contains all essential features, the implementation team starts the configuration process. As Figure 4.3 shows, the configuration requires prior data-gathering. Some data can be collected manually, for example, from ERP systems, while other data can be measured. The configuration concepts and associated data-gathering plans established in the discovery and planning phase provide insights into the collection of different data values. Since all manufacturing companies need some kind of production planning and control system, data about production jobs and operations should be readily available. Any data that is not readily available must be measured, estimated or collected – the exact values to be measured or determined depend on the data-gathering plans for the configuration parameters of the energy-aware production scheduling system. When data-gathering is complete, parameters for the energy models of production machines can be estimated using regression, as proposed above. The result are fitted models which should represent the actual energy consumption of machines well; however, the testing and deployment phase validates their performance to make sure.

Apart from configuring the production machines, production jobs and operations, the implementation team must also configure connectivity between the energy-aware production scheduling system and internal or external entities. Information needed to configure connectivity includes the addresses of external services and services publishing data from internal measuring equipment and production machines. Additionally, some information about the data available from these services is required to identify individual data elements fully.

Overall, the outcomes of this phase consist of the following items which have been implemented and configured throughout this phase:

- Implemented software functionality:

    - Objective criteria
    - Missing energy models of production machines
    - If needed, additional software features
  - Collected data about:
    - Production jobs
    - Production operations for each job
    - Production machine data (energy and otherwise)
  - Fitted production machine models
  - Configurations for system components:
    - Production jobs and operations
    - Production machines
    - Internal and external connectivity

Once the software implementation is complete, data is collected, and all system components are correctly configured, the implementation team must export the resulting data and configuration in a reusable format. The exported data is a prerequisite for the testing and deployment phase.

4.2.3. Testing and Deployment Phase

The testing and deployment phase is the implementation procedure's concluding phase. It should ensure that users can utilize the system's full potential and that the system can perform all required processes (Schwarz, 2022). As Schuh et al. (2015, p. 376) suggest, real users must perform final testing with actual data to minimize the remaining risks. Since many shortcomings only become visible after the deployment, continued support and improvements are also essential (Schwarz, 2022). When the tests detect issues or when other problems arise, the implementation team can always decide to return to previous phases of the implementation procedure recursively.

Figure 4.4 illustrates the steps of the testing and deployment phase. User training commences at the beginning of the phase and is performed in parallel with tests of the energy-aware production scheduling system. Since user training methodology does not directly pertain to the central goals of this thesis, the implementation team should select an appropriate methodology.

On the other hand, testing the quality of the implemented system is fundamental to the energy-aware production scheduling system's operation. Similar to many other steps of the implementation procedure, the testing steps are well suited for

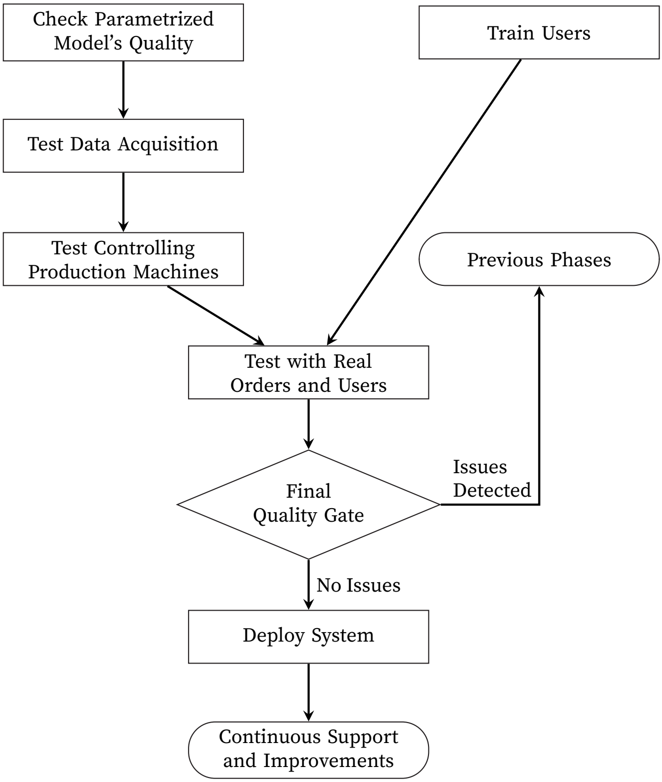


Figure 4.4: Overview of the testing and deployment phase steps (own figure).

re-sequencing or parallelization. In Figure 4.4 testing begins by checking the fitted energy model's performance. The implementation team should select appropriate criteria to evaluate the performance. For example, energy models provide predictions that must be accurate over relatively long time intervals compared to the measured data. Hence, an energy model of a production machine's power consumption should accurately predict the machine's energy consumption over a few minutes; however, power consumption is measured every second. Due to the disparate time intervals, accuracy measures applying rolling averages or integrated values are fitting (see also Section 6.2.3).

Since connectivity with the production system and external services is a requirement for the energy-aware production scheduling system, testing the data acquisition and controlling production machines are separate steps of the implementation procedure. When testing data acquisition, evaluating all machines and services configured for connectivity is essential. Once acquiring data from machines and services is stable, testing controlling the production machines follows as the next step. The implementation team should ensure that most errors are corrected before connecting the system to actual production machines.

After the implementation team completes the elementary model quality and connectivity tests, the trained users can begin quality assurance of the entire system. These tests should be performed with actual customer orders to ensure realism (Schuh et al., 2015, p. 376). As Schuh et al. (2015, p. 376) point out, realistic tests minimize the risks during system deployment. The tests end in the final quality gate, the last checkpoint before the system's deployment. If any remaining issues could not be corrected up to this point, returning to a corresponding step in one of the previous phases is crucial. The implementation team should only advance to the next step and deploy the system if no relevant issues remain, as illustrated in Figure 4.4.

Summarizing the outcomes of the testing and deployment phase, similar to the previous phases, results in the following list:

- Tested system components and processes:
    - Quality of fitted models
    - Connectivity
    - Full scheduling process
- Trained users
- Corrected any detected problems
- Deployed system

Finally, once the entire implementation procedure is complete and the energy-aware production scheduling system is deployed, continuous support and improvements are vital to driving users' support and adoption of the new system (Schwarz, 2022).

4.3. Summary and Implications for the Scheduling System Architecture

Overall, the implementation procedure proposed in Section 4.2 has many steps valid for any production scheduling system, not just energy-aware production scheduling systems. However, the procedure also illustrates the need to scrutinize the implications of energy-aware production scheduling during the implementation process because the implementation team must take additional steps to consider energy properly. For example, the implementation team should evaluate the energy supply of the factory during the discovery and planning phase, and it must create appropriate energy models for production machines during the development and configuration phase.

Energy-aware production scheduling also requires connectivity with additional external services and internal equipment. Collecting energy pricing information and data about production machines' energy consumption are examples of additional communication. Due to the high relevance of information and data, configuring internal and external connectivity becomes an integral part of the implementation procedure for energy-aware production scheduling systems.

The requirements set and the implementation procedure proposed in this section form the basis for developing the energy-aware production scheduling system architecture in the next section. There is a close interrelation between some steps of the implementation procedure and the architecture of the energy-aware production scheduling system, for example, because the system must enable the required configuration and connectivity. The implementation procedure proposed in Section 4.2 reinforces many of the requirements set in Section 4.1.

An important takeaway from the implementation procedure is the need for standardization and modularity. Standardization and modularity go hand in hand because modularity is achieved by standardizing modules to fit many use cases by adjusting their parameters. Thus, the energy-aware production scheduling system should provide configuration options to select and adapt standardized modules to fit a particular production system. The availability of configuration options can also guide the formulation of data-gathering plans. The following section proposes an energy-aware production scheduling system architecture, which supports the implementation procedure proposed in this section.

5. Scheduling System Design

Section 4 discussed the requirements that the energy-aware production scheduling system proposed in this section must fulfil. The section also contemplated the implementation procedure and established prerequisites for the energy-aware production scheduling system architecture. This section introduces a system architecture with energy model parameter estimation to support the implementation of energy-aware production scheduling systems and to facilitate related research. The system architecture is based on the mathematical production scheduling model introduced in Section 5.1.

The result of this section is a concept for a cyber-physical production system with energy-aware production scheduling – the scheduling system implements the production scheduling model and connects to the production system to control its operation. The first part of this section describes the energy-aware production scheduling model formulation used for the optimization in this thesis. The second part introduces the energy-aware production scheduling system architecture with its provisions for configuration and adaptation to various production systems, and the third part summarizes the findings. All of this leads to testing and deployment of the system in the next section.

5.1. Energy-Aware Production Scheduling Model

A general understanding of the production scheduling problem used in this thesis is essential to follow some of the decisions made in the subsequent sections. The production scheduling problem in this thesis builds on the formulations published in Grosch et al. (2019) and Grosch et al. (2021) – it represents an energy-aware job shop scheduling problem. It is imperative to point out that the optimization algorithm described in Sections 5.2.1, 5.2.2, and 5.2.3 solves an equivalent problem due to the specified encoding and decoding strategies. It does not directly solve the mathematical model specified in this section; however, writing the underlying model down helps to clarify the scheduling approach.

This section only provides the general model structure and does not establish particular objective criteria since the requirements prescribe that the implementation team chooses individualized objective criteria during the implementation procedure (refer to Section 4.1). Specific objective criteria are selected in Section 6 when configuring the system for a use case. Sections 2.1.1 and 3.2.3 discuss examples of possible objective criteria. The basic model introduced here makes the following simplifications:

- Machines are always available
- Transportation times between production machines and storage are negligible
- Storage space is unlimited and there is no cost associated with it
- Jobs are not prioritized

In practice, these limitations should not have a significant impact on the scheduling results' applicability. The simplifications regarding transportation time and machine availability can be directly counteracted by implementing rescheduling. Job prioritization does not have practical relevance because the production scheduling system proposed here usually only considers one or a few days. In such short time-frames, job prioritization is usually not necessary. The same is true for storage limitations – since the scheduled time-frame is short, storage space and storage cost are less relevant. However, future research should evaluate how storage space and cost could be integrated into the proposed scheduling system.

5.1.1. Jobs, Operations, Storage, and Machines

For the general structure of the production scheduling model, there is a set of $i_{\max}$ jobs i (see equation (5.1)), and each job has an ordered sequence O_i of $o_{i,\max}$ operations as equation (5.2) states. There is also a set of $m_{\max}$ machines m to perform the operations (equation (5.3)).

$$J = \{1, 2, \ldots, i_{\max}\} \tag{5.1}$$

$$O_i = \left\{1, 2, \ldots, o_{i,\max}\right\} \qquad \forall i \in J \tag{5.2}$$

$$M = \{1, 2, \ldots, m_{\max}\} \tag{5.3}$$

Each job has an associated order quantity q_i^{order} but some partially (or fully) finished parts can be in storage. The parameter $q_{i,o,t}^{\text{stored}}$ denotes the number of parts in storage for each operation of a job. Thus, it includes information about the processing state of the stored parts. Each operation also has a specified processing time $d_{i,o}^{\text{proc}}$ and a setup time $d_{i,o}^{\text{setup}}$.

Additionally, each machine has a defined capacity q_m^{capa} and a parameter to determine if this capacity can be used by multiple different jobs or only a single unique job I_m^{unique}.

5.1.2. Time, Events, and Decision Variables

The following describes a discrete-event approach for production-related optimization. In contrast, the energy models of production machines examined in Section 5.1.5 are discrete-time models following a simulation-based approach. The discrete-event production scheduling model and the discrete-time energy models are connected through the machine's energy states. The discrete-event modelling for the production-related models has advantages due to the encoding necessary for the NSGA-II used for the optimization (refer to Sections 5.2.1 and 5.2.2).

There is a set of events e_m on each machine comprised of events for every operation in every job processed on that machine and according to the demand for each operation $h_{m,\max}$.

$$e_m = \left\{1, 2, \dots, h_{m,\max}\right\} \qquad \forall m \in M. \tag{5.4}$$

Each event has an associated job and operation specified by the matrix $I^{\text{op}}_{i,o,m,h}$ in equation (5.5). The number of events equals the demand for each operation given by equation (5.6). The demand depends on the number of customer orders for the job, q_i^{order} and the number of stored workpieces for that job and operation $q^{\text{stored}}_{i,o,t_0}$ with t_0 being the optimization starting time.

$$I^{\text{op}}_{i,o,m,h} = \begin{cases} 1 & \text{if } i, o \text{ processed on } m \text{ as } h \\ 0 & \text{otherwise} \end{cases} \qquad \forall i \in J, o \in O_i, m \in M, h \in e_m \tag{5.5}$$

$$\sum_{m \in M} \sum_{h \in e_m} I^{\text{op}}_{i,o,m,h} = q_i^{\text{order}} - q^{\text{stored}}_{i,o,t_0} \qquad \forall i \in J, o \in O_i \tag{5.6}$$

The optimization determines the optimal starting time of each event $s_{m,h}$ resulting in a clock structure, which fully specifies the solution; however, an ordered sequence of events is required to specify the constraints of the production scheduling problem. According to Cassandras and Lafortune (2008, p. 271), the following clock structure recursively identifies the next event to be processed on each machine h'' from the current event h'. In equation (5.7), the current time t, which is independent of any event, determines the remaining clock time $y_{m,h}$ until each event is scheduled. The minimum remaining clock time y^*_m over all events follows from equation (5.8), and equation (5.9) determines the next event $e_{m,h''}$.

$$y_{m,h}(t) = s_{m,h} - t \qquad \forall m \in M, h \in e_m \tag{5.7}$$

$$y^*_m(t) = \min_{h \in e_m} \left(y_{m,h}(t)\right) \qquad \forall m \in M \tag{5.8}$$

$$h'' = \arg\min_{h \in e_m} \left(y_{m,h}(t) \right) \qquad \forall m \in M \tag{5.9}$$

Finally, the time step when the next event starts is

$$t_{m,h'} = t + y_m^*(t) \qquad \forall m \in M. \tag{5.10}$$

5.1.3. Constraints

The production schedule optimization is subject to the constraints discussed in this section. The constraints ensure the optimized production schedules are feasible and can be executed in a factory.

Equation (5.11) calculates the ending time $c_{m,h''}$ of the next event. In addition to the starting time, the ending time depends on the setup time $d_{i,o}^{\text{setup}}$ if setup is required $I_{m,h}^{\text{setup}}$ and the processing time $d_{i,o}^{\text{proc}}$ of the job and operation. However, if the starting time of the next event is smaller than the starting time of the current event $c_{m,h'}$ the ending time is set equal to the ending time of the previous event for all $m \in M, h \in e_m$:

$$c_{m,h''} = \begin{cases} c_{m,h'} & \text{if } s_{m,h''} < c_{m,h'} \\ s_{m,h''} + \left(I_{m,h''}^{\text{setup}} d_{i,o}^{\text{setup}} + d_{i,o}^{\text{proc}} \right) I_{i,o,m,h''}^{\text{op}} & \text{otherwise.} \end{cases} \tag{5.11}$$

Equation (5.12) ensures that the current event h' can start on the machine. Depending on the capacity constraint in equation (5.14) the previous event either has to be complete ($\geq c_{m,h'}$), or not. In combination with the previous equation this equation sets the ending times of the current and next events equal if available capacity of the machine m is used.

$$s_{m,h''} \geq \begin{cases} c_{m,h'} & \text{if } q_{m,h''}^{\text{concurrent}} > q_m^{\text{capa}} \\ s_{m,h'} & \text{if } q_{m,h''}^{\text{concurrent}} \leq q_m^{\text{capa}} \end{cases} \qquad \forall m \in M \tag{5.12}$$

If the machine has a capacity larger than one, multiple events can be processed concurrently ($q_{m,h''}^{\text{concurrent}}$), limited by equation (5.13). The equation checks if the machine requires all concurrent operations to belong to the same job as specified by the parameter I_m^{unique}. Adding the very large number G in case the machine requires a unique job and the next event is not for the same job as the current event violates the capacity constraint from equation (5.14). The capacity constraint in equation (5.14) avoids exceeding the machine capacity.

$$q^{\text{concurrent}}_{m,h''} = \begin{cases} q^{\text{concurrent}}_{m,h'} + G & \text{if } I^{\text{unique}}_{m}, s_{m,h''} = s_{m,h'}, I^{\text{op}}_{i,o,m,h''} \neq I^{\text{op}}_{i,o,m,h'} \\ q^{\text{concurrent}}_{m,h'} + 1 & \text{if } I^{\text{unique}}_{m}, s_{m,h''} = s_{m,h''} \\ 0 & \text{otherwise} \end{cases} \tag{5.13}$$

$$q^{\text{concurrent}}_{m,h''} \leq q^{\text{capa}}_{m} \qquad \forall m \in M \tag{5.14}$$

Additionally, setup time can be required if the product changes between events. The variable $I^{\text{setup}}_{m,h''}$ defined in equation (5.15) determines whether setup time is needed.

$$I^{\text{setup}}_{m,h''} = \begin{cases} 1 & \text{if } I^{\text{op}}_{i,o,m,h''} \neq I^{\text{op}}_{i,o,m,h'} \\ 0 & \text{otherwise} \end{cases} \qquad \forall m \in M \tag{5.15}$$

Finally, there must be enough stored workpieces from prior operations to process the subsequent operations. Therefore, equations (5.16) and (5.17) determine the number of workpieces in storage $q^{\text{stored}}_{i,o,h''}$ for each operation of each job after completing event h''. The event adds workpieces to the storage for the current operation $q^{\text{stored}}_{i,o,h''}$ and removes workpieces for the previous operation $q^{\text{stored}}_{i,o-1,h''}$. Finally, equation (5.18) ensures that storage does not become negative.

$$q^{\text{stored}}_{i,o,h''} = q^{\text{stored}}_{i,o,h'} + \sum_{m \in M} I^{\text{op}}_{i,o,m,h'} \qquad \forall i \in J, o \in O_i \tag{5.16}$$

$$q^{\text{stored}}_{i,o-1,h''} = q^{\text{stored}}_{i,o-1,h'} - \sum_{m \in M} I^{\text{op}}_{i,o,m,h'} \qquad \forall i \in J, o \in O_i \tag{5.17}$$

$$q^{\text{stored}}_{i,o-1,h'} \geq 0 \qquad \forall i \in J, o \in O_i \tag{5.18}$$

5.1.4. Graph Representation of the Scheduling Problem

The jobs, operations and events defined in the previous sections can be interpreted and represented as graphs. The graph representation helps define the solution encoding and decoding strategies discussed in Sections 5.2.2 and 5.2.3. There are two separate graphs representing the scheduling problem. The following refers to them as the *product graph* and *machine graph*. Both graphs are directed acyclic graphs, and the relationships between them are given by the matrix $I^{\text{op}}_{i,o,m,h}$, as detailed in equation (5.5) and Section 5.1.2. In directed graphs every edge can only be traversed in one direction. Acyclic graphs additionally do not have cycles:

When starting to traverse the graph from any node it is impossible to reach that same node again.

The product graph represents the jobs and operations (see equations (5.1) and (5.2)), and has disconnected components for each job. Each component contains nodes for each operation of the corresponding job, and its edges are directed from the first operation of the job to the next. The order quantity q_i^{order} determines the number of nodes for each operation of the related job. To be explicit, each job i has an associated number of orders and its operations o occur multiple times, each occurrence being a single workpiece. Therefore, a node in the product graph is denoted as $O_{i,o,k}$ with the job i, operation o and the workpiece k. Equation (5.19) calculates the number of nodes for job i and operation o.

$$\sum O_{i,o,k} = q_i^{\text{order}} \qquad \forall i \in J, o \in O_i, k \in \left\{1, 2, \dots q_i^{\text{order}} - q_{i,o,t_0}^{\text{stored}}\right\} \tag{5.19}$$

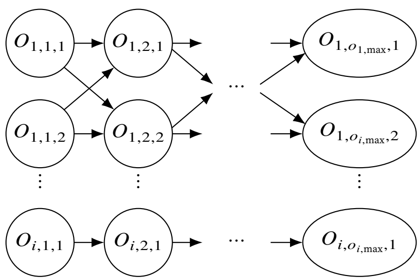


Figure 5.1: Exemplary illustration of a product graph, the last job is $i_{\max}$ (own figure).

Figure 5.1 illustrates an exemplary product graph with $i_{\max}$ jobs and $o_{i,\max}$ operations in each job. In the example, job $i = 1$ has an order quantity of $q_1^{\text{order}} = 2$, and the order quantity for job $i_{\max}$ is $q_{i_{\max}}^{\text{order}} = 1$. The figure shows that each operation and all possible successors are fully connected – it is irrelevant whether node $O_{1,2,1}$ or $O_{1,2,2}$ follows either node $O_{1,1,k}$. At the same time, different jobs have separate, disconnected components in the graph. In Figure 5.1, there are no connections between the operations $O_{1,o,k}$ and the operations $O_{i_{\max},o,k}$.

Figure 5.2 illustrates the machine graph, which represents the order of operations on a machine by determining the order of the set of events e_m (see equation (5.4)). While the product graph is constant for an entire scheduling problem as long as the sets of jobs and operations are constant, the machine graph is a distinct solution to the scheduling problem. The constraints established in Section 5.1.3 decide whether a machine graph is a valid solution.

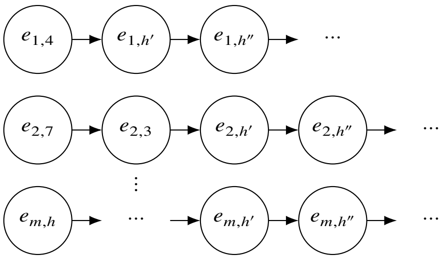


Figure 5.2: Exemplary illustration of a machine graph (own figure).

Figure 5.2 shows that the machine graph has a node for each event in the set of events e_m. It is built recursively according to equations (5.7) to (5.9). While the product graph contains multiple nodes for each operation according to equation (5.19), the machine graph only contains nodes for events that actually need to be processed, bearing the stored quantity $q_{i,o,t}^{\text{stored}}$ in mind. Thus, equation (5.6) gives the number of nodes per machine in the machine graph. Figure 5.2 also shows that the events in e_m do not have to be ordered by occurrence h. For example, event $e_{1,4}$ might come before event $e_{1,1}$.

5.1.5. Production Machine Energy Models

In addition to the regular production scheduling model presented in the previous sections, the energy-aware production scheduling model also includes energy models of the production machines. The energy models must consider that machines consume energy in different forms within the set of energy forms $\mathscr{E}$. The models in this thesis focus on electric and thermal energy; however, other energy forms, like compressed air, could be added to the models during the implementation procedure. This section first looks at electric energy consumption and later continues with thermal energy models. The energy models are discrete-time models with the time t as a connection to the discrete-event production-related scheduling models. To introduce some diversity in the modelled machine types and to prepare implementation for the use case introduced in Section 6.1, this thesis proposes models for machine tools and industrial aqueous parts cleaning machines.

Electric energy consumption is often well represented by a machine's energy states (refer to Section 2.3.3) (Verein Deutscher Maschinen- und Anlagenbauer, 2019); however, this assumption is only valid if the machine has no singular components significantly influencing power consumption patterns (see Figure 2.8).

As Section 6 will show, the implementation procedure might reveal a need to extend the models to integrate such components. This thesis assumes that the relevant energy states for most machines in the context of energy-aware production scheduling are:

- off (no power consumption)
- standby, (a^{st})
- operational, (a^{op})
- working, (a^{wk})

The models proposed in this thesis do not consider the state *off* since it only occurs when the main switch of a machine is turned off (Verein Deutscher Maschinen- und Anlagenbauer, 2019). The models also do not consider transitions between energy states as state transitions are typically very short and become irrelevant over longer production scheduling time frames. To improve model accuracy over shorter time periods, state transitions may be an area for future framework extensions; however, the tradeoff between accuracy improvement and additional data collection efforts must be considered.

The generalized model for a machine's electric power P_t^{el} in equation (5.20) is based on the energy states determined by the binary variables a_t^{st}, a_t^{op}, and a_t^{wk} (see above), which are one if the corresponding energy state is active.

$$\begin{aligned} P_t^{\text{el}} = {} & \left(a_t^{\text{st}} + a_t^{\text{op}} + a_t^{\text{wk}}\right) \cdot \beta_{\text{st}}^{\text{el}} + \left(a_t^{\text{op}} + a_t^{\text{wk}}\right) \cdot \beta_{\text{op}}^{\text{el}} \\ & + a_t^{\text{wk}} \cdot \left(z_{src}^{\text{proc}} \cdot \beta_{\text{proc}}^{\text{el}} + \beta_{\text{wk}}^{\text{el}}\right) \end{aligned} \tag{5.20}$$

Since only one of the binary variables is active at a time, equation (5.20) sums the binary variables such that the regression parameters $\beta_{\text{st}}^{\text{el}}$ and $\beta_{\text{op}}^{\text{el}}$ represent the respective power increase of an energy state over lower energy states. For example, $\beta_{\text{op}}^{\text{el}}$ describes the electric power increase between the standby and operational state. In the working state when a^{wk} is one, the equation expects a generally higher load compared to the operational state. $\beta_{\text{wk}}^{\text{el}}$ represents this general load increase which could, for example, be caused by activating additional machine components or by positioning movements of the machine's axes.

In addition to the general load increase during the working state, there are process-dependent factors influencing the machine's power consumption in the working state. Process-dependent factors could include the number of workpieces processed at once in a batch production machine or the material removal rate for a machine tool. Therefore, the machine energy models need a process-dependent parameter z_{src}^{proc}. This parameter depends on the type of machine and the process performed by the machine. In conclusion, the regression parameters $\beta_{\text{st}}^{\text{el}}$ and $\beta_{\text{op}}^{\text{el}}$ represent the machine's power during the respective energy states, and the energy consumption in the working state combines a generally higher load during the working state $\beta_{\text{wk}}^{\text{el}}$ with the influence of the workpiece $\beta_{\text{proc}}^{\text{el}}$.

Since the machine energy models are based on the machine's energy states, the connection between the production scheduling model and the machine energy models is also based on the machine's energy states as given by equations (5.21) to (5.23).

$$a_t^{\text{st}} = \begin{cases} 1 & \text{if } s_{m,h''} - c_{m,h'} > 180\,\text{s and } c_{m,h'} < t < s_{m,h''} \\ 0 & \text{otherwise} \end{cases} \tag{5.21}$$

$$a_t^{\text{op}} = \begin{cases} 1 & \text{if } a_t^{\text{st}} = 0 \text{ and } c_{m,h'} < t < s_{m,h''} \\ 0 & \text{otherwise} \end{cases} \tag{5.22}$$

$$a_t^{\text{wk}} = \begin{cases} 1 & \text{if } a_t^{\text{st}} = 0 \text{ and } a_t^{\text{op}} = 0 \\ 0 & \text{otherwise} \end{cases} \tag{5.23}$$

Generalizing models for the thermal power of production machines is more diverse, as illustrated by the selection of machine tools and cleaning machines as the two primary machine types in this thesis. Machine tools mainly consume electric energy, and their thermal power is dominated by the need for cooling to dissipate heat converted from electric power consumption. On the other hand, for cleaning machines the thermal power is most relevant (Bayerisches Landesamt für Umwelt, 2006, p. 17). Bayerisches Landesamt für Umwelt (2006, p. 16) analyze the thermal power consumption of spray cleaning machines and determine that heat dissipation to the environment through spraying and to cleaned workpieces dominate the power consumption.

Figure 5.3 illustrates the machine tool as a thermal system with electric power as an input and thermal power as an output. As Denkena et al. (2020) state, the machine emits part of the input power P^{el} to the environment as waste heat $P^{\text{th,env}}$, and the remaining thermal power is usable heat $P^{\text{th,cool}}$ collected by a fluid-bound machine cooling system.

Using the first law of thermodynamics (Lauth & Kowalczyk, 2022, p. 74) on the system shown in Figure 5.3 yields equation (5.24) for the change in the inner energy of the system $\mathrm{d}U$.

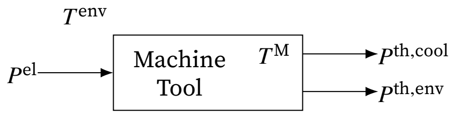


Figure 5.3: Machine tool system with electric and thermal power (own figure).

$$\mathrm{d}U = P^{\text{el}} - P^{\text{th,cool}} - P^{\text{th,env}} \tag{5.24}$$

The change in inner energy is isentropic and thus equivalent to a change of the average machine temperature T^{M} multiplied by a regression parameter for the machine's heat capacity $\beta_{\text{M}}^{\text{c}}$ as stated by equation (5.25) (Lauth & Kowalczyk, 2022, p. 76).

$$\beta_{\text{M}}^{\text{c}} \cdot \mathrm{d}T^{\text{M}} = P^{\text{el}} - P^{\text{th,cool}} - P^{\text{th,env}} \tag{5.25}$$

For simplicity, the proposed models assume that the exact mechanism of heat transfer to the environment and the cooling medium is unknown and that regression parameters for the thermal transfer coefficients $\beta_{\text{cool}}^{\text{th}}$ and $\beta_{\text{env}}^{\text{th}}$ can estimate the resulting heat transfer. This assumption yields equations (5.26) and (5.27) to describe the heat transfer from the machine to the environment and the coolant discretized in time t. In the equation, T_t^{env} is the average temperature of the factory environment, and T_t^{cool} is the coolant temperature.

$$P_t^{\text{th,env}} = \beta_{\text{env}}^{\text{th}} \left(T_t^{\text{M}} - T_t^{\text{env}}\right) \qquad \forall t \tag{5.26}$$
$$P_t^{\text{th,cool}} = \beta_{\text{cool}}^{\text{th}} \left(T_t^{\text{M}} - T_t^{\text{cool}}\right) \qquad \forall t \tag{5.27}$$

Substituting $P^{\text{th,cool}}$ and $P^{\text{th,env}}$ in equation (5.25) with these two equations and solving for the machine temperature at the next time step T_t^{M} results in equation (5.28).

$$\begin{aligned} T_t^{\text{M}} = {} & \frac{P_t^{\text{el}}}{\beta_{\text{M}}^{\text{c}}} - \frac{\beta_{\text{M}}^{\text{c}} - \beta_{\text{env}}^{\text{th}} - \beta_{\text{cool}}^{\text{th}}}{\beta_{\text{M}}^{\text{c}}} \cdot T_{t-1}^{\text{M}} \\ & + \frac{\beta_{\text{cool}}^{\text{th}}}{\beta_{\text{M}}^{\text{c}}} \cdot T_t^{\text{cool}} + \frac{\beta_{\text{env}}^{\text{th}}}{\beta_{\text{M}}^{\text{c}}} \cdot T_t^{\text{env}} \qquad \forall t > t_0 \end{aligned} \tag{5.28}$$

The system of equations given by equations (5.20), (5.26), (5.27), and (5.28) fully determines the machine tool's electric and thermal power.

In contrast to machine tools, cleaning machines often have a large tank for cleaning medium which must remain at a specified temperature. Thus, the thermal power of a cleaning machine mostly depends on the heat losses from the cleaning medium tank. In Fuhrländer-Völker et al. (2023), we identify three main components of heat loss affecting the cleaning machine: heat loss to the environment $P^{\text{th,env}}$, heat loss to the cleaned workpieces $P^{\text{th,wp}}$, and heat loss to the environment due to cleaning medium circulation during the spray cleaning process $P^{\text{th,spray}}$. Figure 5.4 illustrates the cleaning machine as a thermal system with these heat losses. The thermal input power $P^{\text{th,heater}}$ is provided by a heating element in the cleaning medium tank or by a heat exchanger.

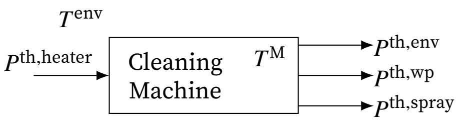


Figure 5.4: Cleaning machine system with electric and thermal power (own figure).

The thermal behaviour of the cleaning machine depends on its operating state. For example, the heat losses to workpieces and due to spray cleaning only occur while the machine is working ($a^{\text{wk}} = 1$). Additionally, the cleaning machine's heating element is only active when the temperature within the cleaning medium tank requires it – the 2-point controller defined by equation (5.29) controls its operation a_t^{heater} with the upper temperature limit $I_{\text{upper}}^{\text{tank}}$ and lower temperature limit $I_{\text{lower}}^{\text{tank}}$. T_t^{M} describes the average temperature in the cleaning medium tank.

$$a_t^{\text{heater}} = \begin{cases} a_{t-1}^{\text{heater}} & \text{if } I_{\text{lower}}^{\text{tank}} \leq T_{t-1}^{\text{M}} \leq I_{\text{upper}}^{\text{tank}} \\ 1 & \text{if } T_t^{\text{M}} < I_{\text{lower}}^{\text{tank}} \\ 0 & \text{if } T_t^{\text{M}} > I_{\text{lower}}^{\text{tank}} \end{cases} \tag{5.29}$$

Equations (5.30) and (5.31) result using the first law of thermodynamics around the cleaning machine system and considering the operating states defined above.

$$\mathrm{d}U = a^{\text{heater}} \cdot P^{\text{el}} - P^{\text{th,env}} - a^{\text{wk}} \cdot \left(P^{\text{th,wp}} - P^{\text{th,spray}}\right) \tag{5.30}$$

$$\beta_{\text{M}}^{\text{c}} \cdot \mathrm{d}T^{\text{M}} = a^{\text{heater}} \cdot P^{\text{th,heater}} - P^{\text{th,env}} - a^{\text{wk}} \cdot \left(P^{\text{th,wp}} - P^{\text{th,spray}}\right) \tag{5.31}$$

Similar to equations (5.26) and (5.27), the next equation represents the heat transfer to the environment and workpieces with regression parameters for the thermal transfer coefficients $\beta^{\text{th}}_{\text{env}}$, $\beta^{\text{th}}_{\text{wp}}$, and $\beta^{\text{th}}_{\text{spray}}$. The regression parameter $\beta^{\text{th}}_{\text{heater}}$ represents the nominal power input by the cleaning medium tank heater. Substituting this into equation (5.31) and solving for T^{M}_t yields equation (5.32).

$$\begin{aligned} T^{\text{M}}_t = {} & a^{\text{heater}}_t \cdot \frac{\beta^{\text{th}}_{\text{heater}}}{\beta^{\text{c}}_{\text{M}}} - \frac{\beta^{\text{th}}_{\text{env}}}{\beta^{\text{c}}_{\text{M}}} \cdot \left(T^{\text{M}}_{t-1} - T^{\text{env}}_t\right) \\ & - a^{\text{wk}}_t \cdot \frac{\beta^{\text{th}}_{\text{spray}} + z^{\text{proc}}_{src} \cdot \beta^{\text{th}}_{\text{wp}}}{\beta^{\text{c}}_{\text{M}}} \cdot \left(T^{\text{M}}_{t-1} - T^{\text{env}}_t\right) + T^{\text{M}}_{t-1} \end{aligned} \tag{5.32}$$

Finally, the cleaning medium tank heater's power is determined by its operating state a^{heater}_t and its estimated nominal power $\beta^{\text{th}}_{\text{heater}}$ according to equation (5.33).

$$P^{\text{th,heater}}_t = a^{\text{heater}}_t \cdot \beta^{\text{th}}_{\text{heater}} \tag{5.33}$$

The equations derived in this section represent the energy consumption of production machines (see also Section 2.3.3). The energy models are included in the production machine models discussed in Section 5.2.5. The structure of these models defines a reusable interface used by the energy model parameter estimation proposed in Section 5.2.6.

5.2. Energy-Aware Production Scheduling System Architecture

The architecture of the energy-aware production scheduling system must fulfil all previously determined requirements as specified in Section 4.1. The requirements include standardization, configuration, and connectivity with internal and external entities. The energy-aware production scheduling system architecture should also support the implementation procedure using energy model parameter estimation. The following sections describe the components of the energy-aware production scheduling system architecture.

Figure 5.5 illustrates the energy-aware production scheduling system architecture as a cyber-physical production system. The real world is at the bottom of the figure and consists of the actual production system and external entities. The actual production system has some production machines to produce products; additionally, some parts might be in storage. The upper part of the figure illustrates the virtual parts of the cyber-physical production system, whose central components are the *production system environment* (see Section 5.2.4) and the *optimization*

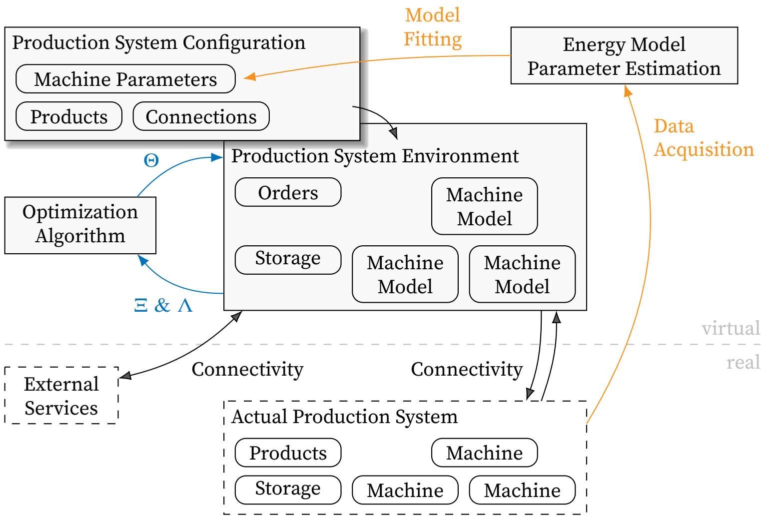


Figure 5.5: Energy-aware production scheduling system architecture (own figure).

algorithm (see Section 5.2.1). The production system environment has *production machine models* (see Section 5.2.5) representing the actual machines. It also knows about workpieces in storage and information about orders to be processed (see Section 5.2.8).

The *production system configuration* is closely connected to the production system environment because it contains the necessary information about the structure of the production system to instantiate the production system environment. The production system configuration results from the implementation procedure – it contains estimated parameters for the production machine models and data about products (which are types of jobs) and their associated production operations. The energy model parameter estimation supports the creation of the production system configuration by estimating the parameters for the energy models.

The production system environment also provides connectivity to the actual production system and external entities using *connectors*. The connectors are the interface between the virtual and real parts of the cyber-physical production system. The production system configuration also contains data to configure the connectors.

The large number of components and variety of requirements regarding the system architecture justify the need for a framework serving as a basis for more specific additions. Hence, the architecture proposed in this section utilizes the *eta_utility* framework – Figure 5.6 depicts the framework's modules (Grosch, Ranzau, et al., 2022).

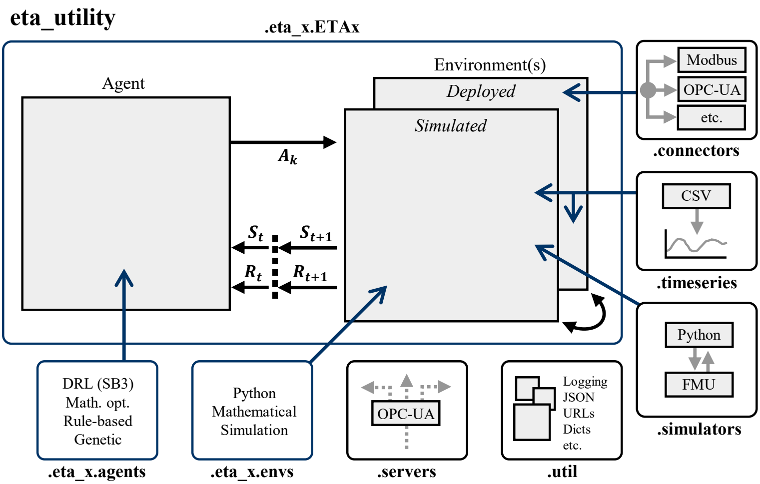


Figure 5.6: Interaction between the environment, agent (optimization algorithm), and other modules of the optimization framework. The blue arrows indicate modules which compose the environments and agent (Grosch, Ranzau, et al., 2022).

The framework provides an abstract structure for rescheduling using an *environment* class as a virtual representation of the system to be optimized. The virtual (the framework refers to this as simulated) environment may communicate with a real (called deployed in the figure) environment, as indicated by the black arrow in the bottom right corner of the figure. The *environment* interacts with the real system through *connectors*. Due to its roots in reinforcement learning, the framework refers to *optimization algorithms* as *agents*. The *eta_x* module of the framework manages the interaction between the real and virtual environments and their interface with the optimization algorithm (Grosch, Ranzau, et al., 2022). The interaction between the environment and optimization is based on the interfaces and structures implemented in the GYM (Brockman et al., 2016) and STABLE_BASELINES3 (Raffin et al., 2021) frameworks. These frameworks implement the Markov Decision Process introduced in Figure 2.4 of Section 2.1.3.

The subsequent parts of this section describe the components of the energy-aware production scheduling system architecture from Figure 5.5 in more detail (see references at the beginning of this section). The following parts base their explanations on the production scheduling model outlined in Section 5.1.

5.2.1. Optimization Algorithm

The optimization algorithm interacts with the virtual production system environment to evaluate and execute production schedules – each production schedule is a solution to the optimization problem. The NSGA-II was initially proposed by Deb et al. (2002), and as the literature review in Section 3 expresses, it is widely used for energy-aware production scheduling. Moreover, the algorithm is well suited to multi-objective optimization (Deb et al., 2002), one of the requirements from Section 4.1. Thus, this thesis uses the NSGA-II for optimization, explicitly building on the hybrid approach published in Grosch et al. (2021). The hybrid algorithm uses dispatching rules for initialization and the NSGA-II for local optimization.

Section 2.1.3 introduces general aspects concerning genetic algorithms and discusses the language used to describe genetic algorithm-based optimization algorithms. This section details adjustments to the algorithm and its integration into the energy-aware production scheduling system by encoding and decoding solutions. Note that the modular *eta_utility* framework and the system architecture allow interchanging optimization algorithms – other algorithms could also be utilized for energy-aware production scheduling. Interchanging algorithms is limited by the problem structure – it would not be possible to replace the heuristic algorithm presented here directly with, for example, a reinforcement learning algorithm.

According to Deb et al. (2002), who proposed the NSGA-II, there are three main advantages over similar algorithms. It reduces the computational complexity of the non-dominated sorting, supports elitism to preserve good solutions through generations, and features a diversity preservation method without additional parameters. Figure 5.7 shows the entire process of the NSGA-II. The algorithm starts by initializing the *parent generation* of solutions S_g^{parent} (Deb et al., 2002). Since this thesis uses the hybrid approach proposed by Grosch et al. (2021), it implements dispatching rules for the initialization. The dispatching rule used in this thesis is the SPT rule, as introduced in Section 2.1.3; however, other rules may also be applicable depending on the objective criteria of the optimization.

As Figure 5.7 illustrates, the algorithm modifies the parent generation with mutation and crossover operators (refer to Section 2.1.3) to create the offspring solutions S_1^{offspr} (Deb et al., 2002). Afterwards, the algorithm passes the new solutions to

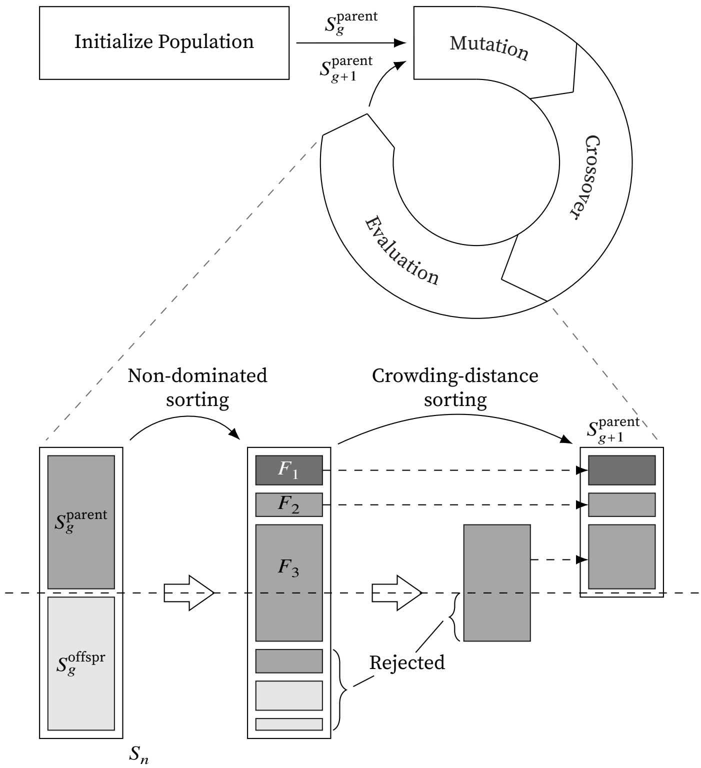


Figure 5.7: Illustration of the NSGA-II procedure's steps (own figure; Deb et al., 2002) (lower portion © IEEE, 2002).

the production system environment for evaluation and the environment returns values for the objective criteria. Then the evaluation procedure shown in the bottom half of Figure 5.7 takes the parent and offspring solutions and passes them through the non-dominated sorting and the crowding distance sorting steps (Deb et al., 2002). The non-dominated sorting step compares the objective criteria of the solutions, taking multiple objectives into account (Deb et al., 2002). One solution dominates another if all objective criteria are equal and at least one criterion of the dominating solution is smaller than that of the other solution. In pseudocode, this is written as follows:

```
all(sol1.objectives <= sol2.objectives)
    && any(sol1.objectives < sol2.objectives)
```

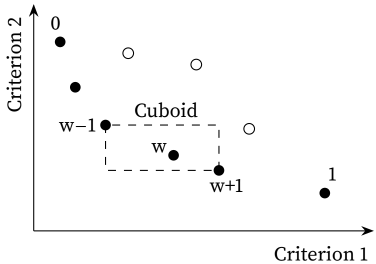


Figure 5.8: Crowding distance between solutions w with multiple objective criteria. Black points are solutions of the same pareto front – the cuboid indicates the distance between three adjacent solutions (Deb et al. 2002; © IEEE).

Solutions are placed in pareto fronts F_f depending on the number of other solutions they dominate (Deb et al., 2002). The algorithm rejects all fronts exceeding the population size after the non-dominated sorting step (Deb et al., 2002). If the last remaining front (F_3 in Figure 5.7) still exceeds the population size, the crowding-distance sorting operator decides which solutions to include in the next parent generation S_{n+1}^{parent} (Deb et al., 2002). Crowding-distance sorting keeps the solutions with the largest distance from other solutions and rejects solutions with close neighbours (Deb et al., 2002). Figure 5.8 illustrates the crowding distance calculation proposed by Deb et al. (2002). Solutions on the border of the solution space are assigned infinite crowding distance (Deb et al., 2002).

The following parameters control the algorithm's procedure. The next section discusses the mutations and crossovers parameters in more detail.

- The *mutations* parameter ω specifies the percentage of each solution that is mutated.
- The *crossovers* parameter κ specifies the percentage of solutions that incur crossover into another solution.
- The *population* parameter $w_{\max}$ controls the total number of solutions per generation.
- The *generations* parameter $g_{\max}$ contains the algorithm termination condition. The optimization will end after it reaches the set number of generations.

Refer to Deb et al. (2002) for more details regarding and performance evaluations of the algorithm. The following sections of this thesis will more specifically discuss the encoding and decoding of solutions and corresponding adaptations to the crossover and mutation operators.

5.2.2. Solution Encoding and Algorithm Implementation

The solution encoding and algorithm implementation are crucial for the optimization's performance. The implementation of the crossover and mutation operators depends on the solution encoding. Therefore, this section discusses the solution encoding and then examines the crossover and mutation operator's implementations.

The solution encoding maps the production scheduling problem's events to the genes on the chromosomes of the genetic algorithm's solutions. The encoding used in this thesis is based on the machine graph introduced in Section 5.1.4. Figure 5.2 shows the machine number m and event number h for each node in the machine graph. The solution encoding also uses these unique machine and event number combinations to represent the sequencing of events on machines.

In addition to the sequencing problem, the optimization must solve the start time scheduling problem. Since these are separate sub-problems, the encoding proposed here consists of two separate chromosomes. The *sequence chromosome* encodes the unique event numbers, and the *time chromosome* governs the pauses between operations on each machine.

Section 5.2.3 explains the decoding of solutions and the interpretation of both chromosomes in more detail. For now, it is sufficient to note the distinct concepts the two chromosomes use to encode their part of the solution. The sequence chromosome conveys information through the sequence of genes in the chromosome. Thus, the information in the sequence chromosome is contained in its sorting order. A single gene on the chromosome does not provide any information by itself. In contrast, the information in the time chromosome consists of time values representing the pauses between two events. Thus, each gene in the time chromosome contains information independent of the genes next to it. The mutation and crossover operators must consider these divergent concepts.

Section 2.1.3 examines the general concepts of mutation and crossover. The implementation proposed here builds on this default but also deals with the differences between the sequence and time chromosomes. The mutation and crossover operators for the time chromosome are equivalent to the process shown in Figures 2.3a and 2.3b. However, the operators for the sequence chromosome differ from these because they directly manipulate the sequence of the chromosome.

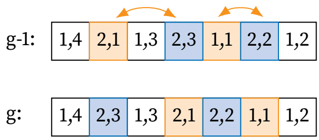


Figure 5.9: Mutation of the sequence chromosome with events specified by the machine number m and event number h as m, h (own figure).

The mutation operator for the sequence chromosome works by reordering parts of the solution, as illustrated in Figure 5.9. The mutation operator selects a number of genes based on the algorithm's mutation parameter to achieve the reordering. The operator then randomly assigns each of these genes to another gene on the chromosome and interchanges their positions, as illustrated in Figure 5.9.

The mutation of the sequence chromosome also behaves differently with respect to the mutation rate because it always affects two genes simultaneously. Looking at the extreme case of 100 % mutations, it is evident that the mutation rate for the sequence chromosome must be halved. Otherwise, the mutation operator would select all events in the sequence chromosome and try to find partners on the same chromosome to switch with them, which is impossible. Therefore, equation (5.34) determines the number of solutions to be mutated $n^{\text{solutions}}$ such that there are at least two mutations per solution. In addition to the algorithm parameters discussed previously, the equation introduces the number of genes per chromosome γ. Equation (5.35) ensures that the mutation rate used for each selected solution is equivalent to the rate that would have been applied if all solutions were mutated.

$$n^{\text{solutions}} = \frac{1}{2}\left(\frac{w_{\max}\cdot\gamma\cdot\omega}{2} + w_{\max} - \left|w_{\max}\cdot\gamma\cdot\omega - w_{\max}\right|\right) \tag{5.34}$$

$$\omega' = \frac{w_{\max}\cdot\gamma\cdot\omega}{n^{\text{solutions}}\cdot\gamma} \tag{5.35}$$

Figure 5.10 illustrates the crossover operator for the sequence chromosome. Since the crossover operator combines two chromosomes and the sequence chromosome encodes information in the sequence of genes, direct replacement of the crossover sequence would lead to duplicate or missing genes on the resulting chromosome. Thus, the following describes the process of keeping the chromosome structure intact and ensuring each event occurs only once.

The crossover operator first selects two random chromosomes from the parent

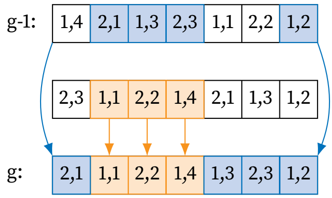


Figure 5.10: Crossover of the sequence chromosome with two chromosomes from the parent generation $g-1$ and the resulting offspring chromosome from generation g (own figure).

generation of solutions ($g-1$). The subsequent description of the crossover process, refers to the topmost chromosome in Figure 5.10 as the *primary chromosome* and to the chromosome in the second row as the *secondary chromosome*. The resulting chromosome for the offspring generation g is the *offspring chromosome*.

After choosing the primary and secondary chromosomes from the parent generation, the crossover operator selects a random sequence of genes from the secondary chromosome. The crossover parameter of the optimization algorithm determines the length of this sequence. For example, the selected sequence in Figure 5.10 is $(1,1)$, $(2,2)$, $(1,4)$. It is marked orange in the figure. The sequence begins at the second and ends at the fourth gene in the chromosome – the crossover operator conserves this placement when creating the offspring chromosome.

The crossover operator constructs the offspring chromosome for generation g by iterating over the primary chromosome and checking whether the current gene occurs in the crossover sequence from the secondary chromosome. If the gene does not occur in the crossover sequence, it is used for the offspring and otherwise it is discarded. For example, in Figure 5.10, the operator would start by taking event $(1,4)$ from the primary chromosome. However, this gene also occurs in the crossover sequence from the secondary chromosome. Therefore, the crossover operator does not copy it to the offspring. The next gene in the primary chromosome is $(2,1)$, which does not occur in the crossover sequence and is accordingly placed as the first gene in the offspring. The crossover sequence begins at the second gene, and thus, the operator copies it from the secondary to the offspring chromosome after placing the first gene. Then, the process continues iterating through the primary chromosome to fill the remaining places in the offspring. Gene $(1,3)$ is the next gene in the primary chromosome. It is copied to the fifth position in the offspring because it does not occur in the crossover sequence. Figure 5.10 marks genes copied from the primary chromosome in blue and genes copied from the secondary chromosome in orange.

Having two chromosomes requires distributing the mutation and crossover rates specified by the corresponding algorithm parameters. The algorithm determines the distribution ψ of mutation and crossover rates between the time and sequence chromosomes randomly. However, precautions must be taken because a rate affecting less than one event on either of the chromosomes is undefined. Looking at the example of mutation for a solution with $\gamma = 20$ genes and a mutation rate of $\omega = 0.05$ makes this more clear. If the example had a single chromosome this would mean a single mutation; however, when multiplying the mutation rate by a distribution of $\psi = 0.5$, for example, each chromosome would be left with a half mutation, which is impossible.

To check whether the selected crossover and mutation rate distributions are possible, equations (5.36) and (5.37) determine the number of affected genes per chromosome $n^{\text{genes}}_{\text{time}}$ and $n^{\text{genes}}_{\text{sequence}}$. Both equations are written for the example of a mutation but similarly apply to crossover rates. The interchange of genes on the sequence chromosome implicitly doubles its mutation rate. Therefore equation (5.36) doubles the mutation rate for the time chromosome to ensure that both chromosomes are affected equally.

$$n^{\text{genes}}_{\text{time}} = \gamma_{\text{time}} \cdot (1 - \psi) \cdot 2\omega' \tag{5.36}$$

$$n^{\text{genes}}_{\text{sequence}} = \gamma_{\text{sequence}} \cdot \psi \cdot \omega' \tag{5.37}$$

The algorithm then uses the number of affected genes to determine if adjustments to the crossover or mutation rates are necessary. The following pseudocode illustrates these adjustments for the mutation rate.

```
if ψ ≤ 0.5 && n^genes_sequence < 1:
    ψ = 1 / (γ_sequence · ω')
else if ψ > 0.5 && n^genes_time < 1:
    ψ = 1 - (1 / (2 γ_time · ω'))

ω'_sequence = max(ψ · ω', 0)
ω'_time = max((1 − ψ) ·2ω', 0)
```

The algorithm checks whether the chromosome with the smaller portion of the distribution has at least one complete affected gene, and if that is not the case, it will set the distribution such that it does. If both chromosomes have at least one affected gene, the rate is set by multiplying the mutation rate with the distribution or one minus the distribution, respectively.

Results generated with similar encoding, mutation, and crossover operators also using two separate chromosomes are published in Grosch et al. (2019) and Grosch et al. (2021); however, these publications used a different strategy to generate and decode the sequence chromosome. The strategy was not yet based on the graph-based representation of the optimization problem, making the decoding process less efficient. Grosch et al. (2019) also used a different strategy for calculating each chromosome's mutation and crossover rates.

5.2.3. Solution Decoding and Building Schedules

The environment must decode the solutions generated by the optimization algorithm before creating schedules that can be evaluated and executed. This section explains the decoding algorithm and examines how it derives schedules from the time and sequence chromosomes. The goal of building schedules is to map the processing events to corresponding starting times $s_{m,h}$. The algorithm also determines whether setup time is required ($I_{m,h}^{\text{setup}}$) before the production of an event starts.

The machine graph (refer to Section 5.1.4) can hold all information needed to sequence the operations correctly. Since the graph representation is efficient when decoding solutions, the decoding process first generates the machine graph from the encoded information on the chromosomes. The decoding assumes that each event has a unique identifier, and all identifiers are part of an increasing sequence of integers starting from one, so the `event` in the following pseudocode is an integer. The same is true for machines which also have a unique index starting from one.

```
machine_graph = DiGraph(length(sequence_chromosome))
prior_event = zeros(length(machines))
starting_event = zeros(length(machines))

for event in sequence_chromosome:
    # Determine the machine from the event
    machine = get_machine(event)

    if prior_event[machine] == 0
        starting_event[machine] = event
    else
        add_edge(machine_graph, prior_event[machine], event)
```

```
        end
        prior_event[machine] = event + 1
    end
```

The actual implementation is written in the Julia programming language; thus, the pseudocode uses a syntax similar to Julia. The preceding pseudocode describes the creation of the machine graph – it first initializes the machine graph as a directed graph and two arrays. The graph implementations use the JuliaGraphs/Graphs.jl package by Fairbanks et al. (2021). The `prior_event` array stores the event previously scheduled on the machine, and the `starting_event` array stores the first event scheduled on the machine. The decoding procedure initializes both arrays with zeros as special values meaning no prior events are scheduled on the machine.

The for loop then iterates through the sequence chromosome to create the machine graph. After getting the machine on which the event should be scheduled (`get_machine()`), it checks whether there are prior events on that machine. If there are no prior events, it will set the current event as the starting event. Otherwise, it will add an edge to the machine graph from the prior event to the current event. When the for loop is complete, the result is the machine graph with disconnected components for each machine in the production system.

The scheduling algorithm then uses the machine and product graphs to determine a starting time for each node in the machine graph. The implementation proposed in the following checks many of the constraints introduced in Section 5.1.3 and ensures they are always fulfilled.

Before creating the schedule, the decoding procedure must introduce and initialize some variables as shown in the following pseudocode. Since the following code iterates over machine and product graphs multiple times, it refers to the nodes the code has seen before as *visited nodes*. The `visited_from` array keeps track of cycles in the graphs – if a node is visited twice from the same node, a cycle in the graphs makes the solution invalid. The `scheduled_nodes` and `scheduled_nodes_bits` arrays are necessary to avoid duplicate scheduling of nodes and to check if a node can be used as a stored item by subsequent operations. The set of `used_nodes` ensures that nodes cannot be used again once subsequent operations have used them. In reality, this is equivalent to using a stored workpiece, and `scheduled_nodes` and `used_nodes` can be interpreted as keeping track of parts in storage. For example, if the `scheduled_nodes` array contains some nodes, these can be interpreted as workpieces in storage. Tracking of stored parts is an important enabler for rescheduling.

```
# Keep track of nodes that have been visited before
visited_from = falses(length(events), length(events + 1))

# Keep track of nodes that have already been scheduled
scheduled_nodes = Int[]
scheduled_nodes_bits = falses(length(events))

# Keep track of nodes already used by subsequent nodes
used_nodes = Set{Int}()
```

Finally, there are two arrays to store the in-degree of every node: `indegrees_product_graph` stores the in-degrees of the product graph's nodes, and `indegrees_machine_graph` store the in-degrees for the machine graph. Since the in-degrees are often needed in the scheduling loop, storing them in advance speeds up the iteration.

```
# Store indegrees for product and machine graphs
indegrees_product_graph = indegree(product_graph)
indegrees_machine_graph = indegree(machine_graph)
```

Obtusely iterating over nodes of the machine and product graphs would be inefficient since the schedule can only start from very few nodes. More specifically, the search can only start from nodes in the machine graph with an in-degree of zero, meaning they have no incoming edges. These are the same nodes previously stored in `starting_event`; thus, `starting_event = starting_nodes`. After this preparation, the actual iteration over the two graphs begins.

The while loop uses an additional array `starters` which stores the preceding node and the starting nodes to enable the cycle check using the `visited_from` array. The `starters` array starts equal to `starting_nodes` and is refilled with all out-neighbours of scheduled nodes from the product and machine graphs. The search strategy means that some nodes can be visited multiple times (checked with `scheduled_nodes_bits`). In this case, the loop directly continues to the next iteration.

```
while length(starters) > 0:
    node, from_node = popfirst(starters)

    if visited_from[node, from_node]:
        # The graph contains a cycle.
        return "Solution is invalid."
```

```
    else:
        visited_from[node, from_node] = true

        # Skip already scheduled nodes
        if scheduled_nodes_bits[node]:
            continue
        end
    end
```

The subsequent checks within the loop determine whether the node can be scheduled, for example, by checking whether stored workpieces and the machine are available. A node can always be scheduled on a machine if the in-degree of that node on the machine graph is zero. Otherwise, if the preceding node on the machine graph is not also in `scheduled_nodes`, the loop continues because the current node cannot be scheduled yet. If the preceding node is in `scheduled_nodes` the loop stores it as the `required_machine_node`.

```
    if indegrees_machine_graph[node] == 0
        required_machine_node = 0
    else
        required_machine_nodes = (
            scheduled_nodes ∩ inneighbors(
                machine_graph, node
            )
        )
        if length(required_machine_nodes) > 0
            required_machine_node = first(
                required_machine_nodes
            )
        else
            continue
        end
    end
```

The second check is the storage check, which ensures that previous operations required by the production job are scheduled before subsequent operations. There are no prior operations if the in-degree on the product graph is zero. Otherwise, the algorithm uses the set difference between in-neighbours on the product graph and the `used_nodes` and takes the intersection of the result with the `scheduled_nodes`. The algorithm stores the result in `required_product_nodes`. Any node within this array would be a valid predecessor for scheduling the current node – the algorithm always chooses the first possible node.

```
    if indegrees_product_graph[node] == 0
        required_product_node = 0
    else
        required_product_nodes = (
            scheduled_nodes ∩ (
                inneighbors(product_graph, node) - usednodes
            )
        )
        if length(required_product_nodes) > 0
            required_product_node = first(
                required_product_nodes
            )
        else
            continue
        end
    end
```

After these two checks ensure a node can be scheduled based on the machine and product graphs, the algorithm stores the selected nodes in `scheduled_nodes`, `scheduled_nodes_bits` and `used_nodes`. It also determines the next possible starting nodes by looking at the out-neighbours on both graphs and adds them to the `starters`.

Finally, the algorithm performs the actual scheduling and determines the starting time of the processing event identified by the node. The current node's earliest possible starting time (`earliest_start`) usually equals the latest ending time between the `required_product_node` and the `required_machine_node` selected above; however, if the machine has available capacity according to equation (5.14), the earliest starting time has to consider the starting time of the `required_machine_node`. In the latter case, the earliest starting time must not be between the starting and ending time of the last processing operation on the machine. If it is, the starting time will be set to begin after completing the prior operation.

```
if machine_capacity_available(required_machine_node):
    machine_endtime = starting_time(required_machine_node)
else:
    machine_endtime = ending_time(required_machine_node)
end

```

```
# The earliest starting time depends on the latest
# ending time of the preceding events.
earliest_start = max(
    machine_endtime, ending_time(required_product_node)
)

# If events are coinciding reuse values of the
# coinciding event.
if (
    machine_capacity_available(required_machine_node)
    && earliest_start != ending_time(required_product_node)
):
    setup = setup_time(required_machine_node)
    pause_time = pause_time(required_machine_node)
    starting_time = starting_time(required_machine_node)
else:
    # Get the preceding pause from the variables
    pause_time = time_chromosome(event)
    # Calculate the starting time
    starting_time = earliest_start + pausetime

    # Check if setup is necessary
    if equal_operation(required_machine_node):
        setup = 0
    else:
        setup = setup_time(event)
    end
end
```

The actual `starting_time` depends on the pause determined by the time chromosome. The time chromosome's genes determining pause times before the operation can start must be allocated to a specific machine to avoid unstable scheduling. Failing to assign the time chromosome's genes to specific machines impedes optimization convergence because the same pause time could be assigned to different machines depending on the sequence chromosome, which can lead to vastly different results. Therefore, the algorithm uses the mapping array `machine_genes` to ensure that each machine always uses the same genes from the time chromosome in the same order.

As a last step, the scheduling algorithm checks whether setup of the machine is necessary ($I^{\text{setup}}_{m,h}$). Setup time is needed if the previous event on the same machine belongs to a different job than the current event. When the algorithm reaches the end of the while loop and has not visited all nodes on the machine graph, it discards the entire solution as invalid. Otherwise the solution is valid, completely decoded, and ready to evaluate the objective criteria – this evaluation is discussed in more detail in Section 5.2.4 because it depends on the objectives selected during the implementation procedure.

5.2.4. Production System Environment

Looking back at Figure 5.5 in Section 5.2 on page 93, the optimization algorithm examined in Section 5.2.1 generates solutions to the optimization problem. A solution consists of the encoded sequence and time chromosomes. Conversely, the production system environment is the virtual representation of the actual production system. It implements the environment class described in Grosch, Ranzau, et al. (2022) and Section 5.2.

The solution encoding and decoding processes described in Sections 5.2.2 and 5.2.3 connect the environment and the optimization algorithm. The encoding and decoding processes are both part of the environment class. According to Figure 5.5, the environment also returns information about its state Ξ to indicate completed production operations, and it determines the value of production schedules using the objective criteria Λ.

As shown in Figure 5.5, the virtual environment can interact with the actual production system using connectors implementing different connectivity frameworks. Section 5.2.7 describes this interaction in more detail. Since some optimization algorithms have separate training and execution phases, the interaction is only allowed to occur during the execution phase. Hence, the virtual environment must be separate from the actual production system during training. The separation between the virtual and real environments provided by the *eta_utility* framework, as illustrated in Figure 5.6, allows for this differentiation between the training and execution phase. When the interaction between both environments is enabled, the production scheduling system will communicate with the actual production system. When the interaction between the environments is disabled, no communication will take place.

We have previously published results generated with some parts of the architecture in Grosch et al. (2019) and Grosch et al. (2021); however, the implementations used then are not completely integrated with the *eta_utility* framework. They also do not include any ability to communicate with the production system, they do not implement the graph-based encoding and decoding strategies proposed in

Sections 5.2.2 and 5.2.3, and they use different energy models. Furthermore, both of these papers use the Python programming language for their entire implementation, which leads to significant problems with the execution speed. Since the algorithms introduced in the previous sections contain tight loops, the problem is primarily due to how Python handles data types in loops. This thesis reimplements the entire optimization algorithm and production system environment in the Julia programming language. The reimplementation led to significant improvements in the algorithm's run time.

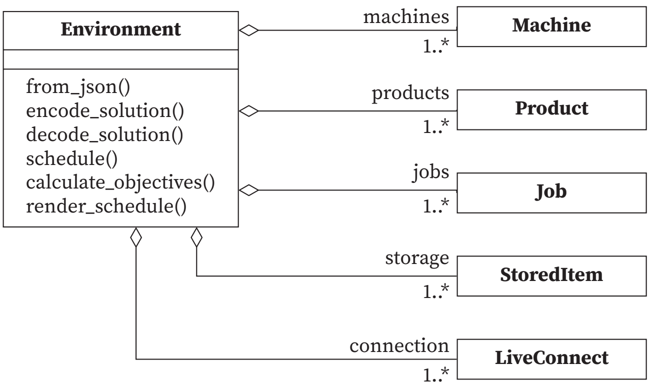


Figure 5.11: Simplified diagram of the production system environment and its associated components (own figure).

Figure 5.11 illustrates that the functions discussed in Section 5.2.2 and Section 5.2.3 are part of the `Environment` class as `encode_solution()`, `decode_solution()`, and `schedule()`. There are also functions for reading configuration files (`from_json()`), calculating objective values (`calculate_objectives()`) and rendering the results of production scheduling (`render_schedule()`). The `calculate_objectives()` function always has to be implemented during the implementation procedure according the the chosen individual objective criteria. It takes a production schedule and uses that information to calculate the objective criteria values. Additionally, the environment contains the following modules:

- Generalized *production machine models* (refer to Section 5.2.5) simulate production machines' behaviour and energy consumption. They are contained in `Machine` objects.

– The *product, order and storage configuration* (refer to Section 5.2.8) defines the products, their operations, and the jobs scheduled by production scheduling. This information is contained in the `Product`, `Job`, and `StoredItem` objects.
– The *connectors* (refer to Section 5.2.7) connect to and communicate with the production machines and potentially external entities. In conjunction with the other environment modules and the actual production system, they form the cyber-physical production system. The `LiveConnect` class implements connectivity utilizing the connectors.

The production system environment's modules are configurable through the *production system configuration*, which contains all information necessary to create the virtual representation of the actual production system. In addition, the configuration contains the information necessary to estimate the production machine model parameters and instantiate the connectors. The production system configuration also forms the interface between the energy model parameter estimation introduced in Section 5.2.6. The ability to configure different production systems provided by the production system configuration is existential to support the implementation procedure discussed in Section 4.2.

The production system configuration consists of multiple parts for production machine models and connectors. The section describing each module also details its configuration. Additionally, there is the product and production job configuration (refer to Section 5.2.8). The product configuration contains general information about the products manufactured in the factory. In contrast, the production job configuration configures a specific optimization run with information about customer orders for products and stored workpieces. Since this thesis creates a prototype implementation, the configuration consists of Javascript Object Notation (JSON) files exported by the energy model parameter estimation or created manually. Final implementations should have a graphical user interface to create and change the configuration. Until then, the `eta_x` module configuration described by Grosch, Ranzau, et al. (2022) is the primary configuration file and determines the other relevant configuration files. The production system environment reads and processes the configuration files to instantiate the relevant modules as objects.

5.2.5. Production Machine Models

The production machine models are a fundamental building block of the production system environment because they enable simulating the energy consumption of production schedules. Separating creating schedules as described in Section 5.2.3 from energy simulation improves the modularity of the energy-aware production scheduling system architecture. The production machine models also accumulate other information about production machines, including capacity and capabilities.

The information stored by the production machine models comprises production- and energy-related data. The models do not limit the information in each category, but they define the following attributes as a minimum:

- Production-related data
    - Human-readable machine name
    - Unique identifier for the machine m
    - Capacity q_m^{capa}
    - A flag indicating the machine can only process a single job at once I_m^{unique}
- Energy-related data
    - Energy consumption model

The production-related data fulfils the specification of the production scheduling problem introduced in Section 5.1. Additional data may be added to the production machine models to represent more complex systems. The production scheduling system assigns jobs to production machines based on the unique machine identifier m. The capacity q_m^{capa} limits how many workpieces the machine can process at once – it differentiates batch production machines and determines the respective batch size. When the capacity is greater than one, the unique job flag I_m^{unique} additionally constrains whether multiple different jobs can be processed simultaneously. Finally, the machine name is necessary to present human-readable output.

Regarding the energy data, this thesis proposes that each production machine model should have an energy model to calculate the energy for each final energy form consumed by the production machine. The model should be formed such that its parameters can be estimated automatically with data collected from the operation of the production machine. Automatic parameter estimation is achieved by formulating the energy models as regression models (see Sections 5.1.5 and 5.2.6).

The Unified Modelling Language diagram in Figure 5.12 shows the overall structure of the production machine models. The structure consists of the `Machine` class containing the production-related information and a generalized energy

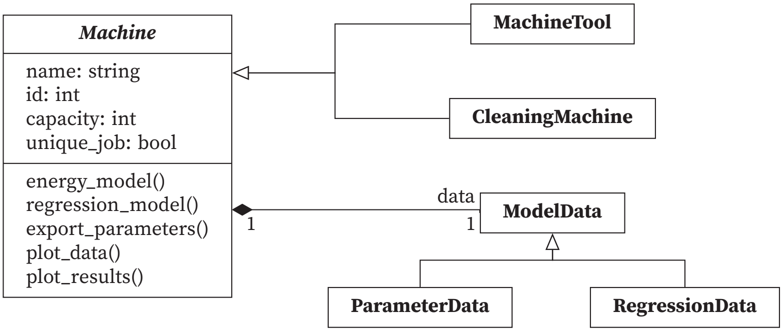


Figure 5.12: Simplified illustration of the production machine model components with the `Machine` class, a `ModelData` class and the energy model functions (own figure).

model as described above. The `Machine` class has a `ModelData` object containing the parameters for the energy model. The `ModelData` object can also contain the parameters for the regression model (`RegressionData`) before the parameter estimation for the energy model is finished. Once the parameter estimation is complete, the estimated parameters are stored in the `ParameterData` object. The `Machine` object cannot be used for scheduling until it has a `ParameterData` object.

Figure 5.12 also shows that the `Machine` can be a `MachineTool` or `Cleaning-Machine` object. These are two exemplary types – other machine types are also feasible (the use case in Section 6.1 explains why these machine types were selected as an example). Various machine types may have additional production-related attributes, but they primarily differ in the `energy_model` and the corresponding `ParameterData`. Figure 5.12 does not illustrate the various subclasses of `ParameterData` and `RegressionData`. In addition to the `energy_model()` and `regression_model()` functions which return the corresponding models, the `Machine` class also has some functions for exporting the model parameters determined during regression (`export_parameters()`) and for plotting (`plot_data()` and `plot_results()`).

The interface between the energy models within the `Machine` objects and the production system environment is an integral part of the specification; it enables combinations of multiple `Machine` objects to embody a complete production system. Section 5.1.5 examines two exemplary energy models for production machines. The interface between the production system environment and the energy models is based on the production machine's energy states and the process-dependent parameter introduced in that section. Overall, the interface includes the following parameters:

- Machine energy state (a^{st}, a^{op}, a^{wk}).
- Process-dependent parameter for the workpiece z_{src}^{proc}.
- Forecast of temperatures in the factory T_t^{env} over time.

The energy models should return the power for all relevant energy forms based on these parameters. For example, the models introduced in Section 5.1.5 would return thermal and electric power forecasts for the production schedule defined via the aforementioned interface.

5.2.6. Energy Model Parameter Estimation

During the implementation procedure, the production machine model's parameters must be estimated to fit the models to the behaviour of the actual machines in the production system. While the implementation team would typically configure the production-related part of the production scheduling problem manually, the production machine energy models proposed in Section 5.1.5 need many parameters that cannot easily be determined precisely.

The energy model parameter estimation therefore supports the fitting process for the production machine energy models – it provides data-based parameter identification using linear regression models. The regression models are implemented using the JuMP package in Julia (Lubin et al., 2023) and solved with the IBM ILOG CPLEX Optimization Studio (CPLEX) solver for quadratic optimization problems (IBM Corporation, 2022). The regression models estimate the parameters for the models from Section 5.1.5, and according to Rencher and Christensen (2012, p. 354), they are of the form

$$\min \sum_{t=t_0}^{t_{\max}} \sum_{\alpha \in \Gamma} {\epsilon_{\alpha,t}}^2 \tag{5.38}$$

$$\text{subject to } y_{\alpha,t} = X_{\alpha,\varphi,t} \beta_{\alpha,\varphi} + \epsilon_{\alpha,t} \qquad \forall \alpha \in \Gamma, \varphi \in \Phi, t. \tag{5.39}$$

Equation (5.39) describes a multivariate multiple regression model, where multivariate means there is a set of dependent variables Γ, and multiple refers to the set of independent variables Φ for each dependent variable (Rencher & Christensen, 2012). Each dependent variable is denoted by y_α, the independent variables are

$X_{\alpha,\varphi}$, and the regression parameters are $\beta_{\alpha,\varphi}$. The regression inspects multiple measured sets of independent and dependent variables at different times t. It uses the least squares estimator in equation (5.38) to minimize the regression error $\epsilon_{\alpha,t}$ for each measured data set and dependent variable. Equation (5.39) shows the relation between the dependent and independent variables, the regression parameters and the regression error for each measured data set over time t. The equations from Section 5.1.5 must be rearranged, conforming to equation (5.39). The rearrangement mainly consists of adding the regression error ϵ_{α}.

To utilize the energy model parameter estimation, the implementation team must collect the data sets required for the regression. For example, the machine tool model presented in Section 5.1.5 has the following dependent and independent variables.

– Dependent variables are:
  - P^{el}, the electric power consumed by the machine,
  - T^{M}, the machine's average temperature,
  - $P^{\mathrm{th,env}}$, the heat loss to the environment, and
  - $P^{\mathrm{th,cool}}$, the heat captured by the machine cooling system.
– Independent variables are:
  - a^{st}, a^{op}, a^{wk}, the machine's energy state,
  - z_{src}^{proc}, the process-dependent parameter,
  - T^{cool}, the coolant temperature, and
  - T^{env}, the temperature within the factory building.

In contrast, the cleaning machine model from Section 5.1.5 has the following dependent and independent variables.

– The dependent variables are:
  - P^{el}, the electric power consumed by the machine,
  - T^{M}, the machine's cleaning medium tank temperature,
  - a^{heater}, the heating element's state, and
  - $P^{\mathrm{th,heater}}$, the heating elements power.
– Independent variables are:
  - a^{st}, a^{op}, a^{wk}, the machine's energy state,
  - z_{src}^{proc}, the process-dependent parameter,
  - $l_{\mathrm{lower}}^{\mathrm{tank}}$, $l_{\mathrm{upper}}^{\mathrm{tank}}$, the 2-point controller's temperature limits, and
  - T^{env}, the temperature within the factory building.

The energy model parameter estimation needs data sets for all dependent and independent variables. It imports these data sets, performs the regression and exports the results as JSON files that are later imported by the production system environment to instantiate the `Machine` objects scrutinized in Section 5.2.5. The production machine models provide the `RegressionData` and `ParameterData` objects to contain the data sets needed for regression and the resulting regression parameters needed while executing the energy-aware production scheduling system.

The implementation team must identify the required models with their inputs and outputs before using the energy model parameter estimation. The complexity of creating models for various machine types and energy forms demonstrates the importance of creating a library of energy models that can be fitted using the energy model parameter estimation. Such a library would alleviate the need to create specific models for each implementation of energy-aware production scheduling systems. In addition, the parameter estimation functionality, in combination with flexible, standardized energy model integration and model export, facilitates different real machines employing the same model structure, as implemented in the `Machine` class.

5.2.7. Connectors

The production system environment must connect to and communicate with the production machines and other entities. For example, to gather information about the production system's state and external factors like energy prices. Therefore, it must have connectors to facilitate and standardize this connectivity. The connectors also enable controlling the production machine's operation and allow data collection for the energy model parameter estimation.

Since there are multiple relevant connectivity frameworks and other standards for connectivity (refer to Section 2.2.2), the connectors must support various connectivity frameworks. ReST-based web services are particularly problematic to support in a standardized way. Since each Application Programming Interface (API) they expose can be different and the specification is completely up to the API's vendor, the connectors do not attempt to abstract the APIs completely – instead they propose a generalized structure which is fully extensible to support API-specific functionality. The connectors' proposed generalized structure ensures that common functions such as reading and writing values are equivalent between different connectivity frameworks (Grosch, Ranzau, et al., 2022).

Figure 5.13 illustrates the structure of the connectors. The `LiveConnect` class is the central component and forms the interface to the production system environment. The `LiveConnect` class provides the `from_json()` function to read JSON configuration files and initialize all connections (Grosch, Ranzau, et al., 2022). The connectivity configuration can contain multiple `system` objects, each containing multiple `server` objects (Grosch, Ranzau, et al., 2022). This structure accommodates production machines that have multiple Industrial Internet of Things devices associated with themselves, as might be the case with a machine having a PLC and some sensors. For example, some of the machines introduced in Section 6.1 have a PLC communicating via OPC UA and energy meters communicating via Modbus TCP.

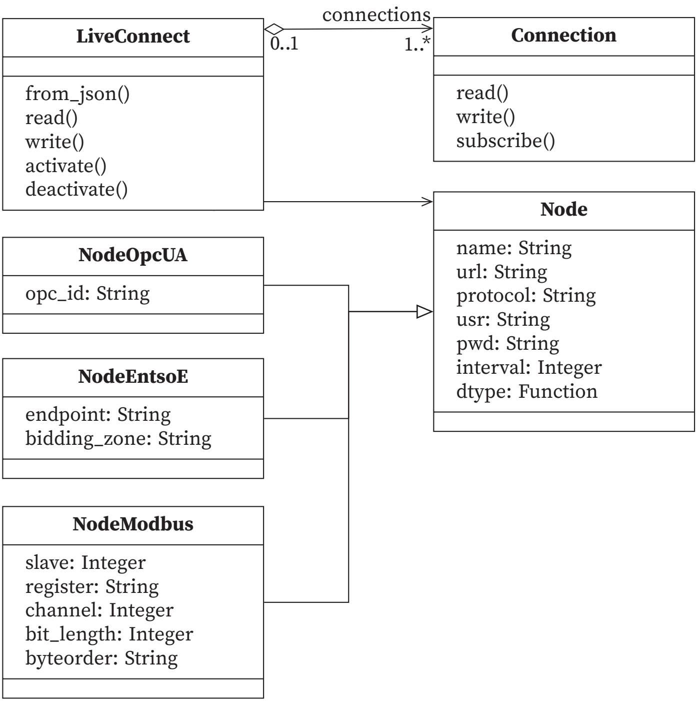


Figure 5.13: Simplified structure of the connectors, made up of the `LiveConnect` object and its interaction with the `Connection` and `Node` objects in the *eta_utility* framework (own figure; Grosch, Ranzau, et al., 2022).

```
{
  "system": [
    {
      "name": "MachineTool1",
      "servers": {
        "glt": {
          "url": "127.0.0.1:4840",
          "protocol": "opcua",
          "usr": "admin",
          "pwd": "0"
        }
      },
      "nodes": [{
        "name": "power_elek",
        "server": "glt",
        "opc_id": "ns=6;s=.MachineTool.PowerElek",
        "dtype": "float"
      }],
    }
  ]
}
```

The configuration also specifies a list of `nodes` containing the information needed to initialize a `Node` object, as illustrated in Figure 5.13 (Grosch, Ranzau, et al., 2022). A `Node` object represents the endpoint of a connectivity framework (Grosch, Ranzau, et al., 2022). The figure shows three exemplary types of `Node` objects, `NodeOpcUa` which is an OPC UA node, `NodeEntsoE` which represents an endpoint for the European Network of Transmission System Operators for Electricity (ENTSO-E) transparency platform, and `NodeModbus` which identifies a value using the Modbus TCP fieldbus protocol. The `Node` class specifies some common attributes like the name and Unique Resource Locator (URL) of a node; however, each connectivity framework also has unique attributes like the id for an OPC UA node or multiple attributes to identify Modbus TCP data (Grosch, Ranzau, et al., 2022).

Each `Node` type is specific to one connectivity framework, and the same is true of implementations of the `Connection` interface. An implementation of the `Connection` interface contains the implementation of the connectivity framework or API. The `Connection` interface specifies basic functionality, such as the ability to read data from and write data to specified nodes (Grosch, Ranzau, et al., 2022).

Classes for any connectivity framework may also contain additional functionality specific to that framework (Grosch, Ranzau, et al., 2022). As Figure 5.13 shows, the basic interface of a `Connection` includes the following methods (Grosch, Ranzau, et al., 2022):

- `read()`, to get a variable
- `write()`, to change a variable
- `subscribe()`, to continuously and asynchronously read multiple variables

The configuration also maps configured external nodes to internal variables (Grosch, Ranzau, et al., 2022). Using this mapping, the `LiveConnect` class further abstracts the `Connection` interface. For example, its `read()` and `write()` functions use the internal variable names to identify the corresponding `Node` and `Connection` objects (Grosch, Ranzau, et al., 2022). Furthermore, in the mapping each configured `system` object is referenced by its name attribute and contains a list of nodes referenced by their respective names (Grosch, Ranzau, et al., 2022). Overall, the structure provided by the connectors enables configuration during the implementation procedure and provides an interface enabling connectivity between the production system environment and actual machines in the factory.

5.2.8. Product, Storage and Order Configuration

Besides the structure of the actual production system, the production system environment also needs information about its state and the products and orders to be produced. As Figure 5.11 shows, three classes define the state of the workpiece storage (`StoredItem`), the produced products (`Product`) and the production jobs (`Job`). The configuration for each of these classes is defined in a JSON file and read by the production system environment during its instantiation.

The products define how a production job should be produced by enumerating the corresponding production operations. Each operation is associated with a distinct machine which must have a production machine model (see Section 5.2.5). The operation also contains information about the processing time $d_{i,o}^{\text{proc}}$ on the given machine and the setup time $d_{i,o}^{\text{setup}}$ if the previous operation on the same machine was different. Finally, the operation configuration defines the process-dependent parameter z_{src}^{proc}. In summary, the product configuration contains the following information:

- Human-readable product name
- Unique identifier of the product

- Ordered sequence of operations:
  - Unique identifier of the associated machine executing the operation m
  - Processing time $d_{i,o}^{\text{proc}}$
  - Setup time $d_{i,o}^{\text{setup}}$
  - Process-dependent energy consumption parameter z_{src}^{proc}

Another configuration file provides the current state of the factory's storage when starting the optimization. Stored workpieces might be in an intermediate state – they do not have to be completely finished. Accordingly, the storage configuration identifies the stored workpieces using their last completed operation. For example, when a workpiece for the third operation of a job is in storage, the fourth operation for that product is the next to be performed. The storage configuration for a workpiece contains the following information:

- Unique identifier of the product
- Unique identifier of the last operation completed on the workpiece
- Number of workpieces in storage $q_{i,o,t}^{\text{stored}}$

Finally, the production system environment should create a production schedule to fulfil customer orders. Similar to the stored workpieces, the configuration of customer orders is based on the previously defined products. It contains information about the number of ordered workpieces and provides a basis for future extension, for example, by including order due dates. The production order configuration contains the following information:

- Human-readable order identification
- Unique identifier of the product
- Number of workpieces to be produced q_i^{order}

The product configuration allows the environment to correctly decode solutions and check whether the created schedules fulfil all constraints defined in Section 5.1.3. In combination with the production order configuration and the storage configuration, the production system environment can create the solution encoding to communicate with the optimization algorithm.

5.3. Summary

The proposed energy-aware production scheduling system design is a cyber-physical production system with an energy-aware production scheduling system architecture that implements the energy-aware production scheduling model discussed in Section 5.1. The production scheduling model has two parts: the production-related scheduling part and the production machine energy models. The production-related part of the scheduling model uses a discrete-event formulation to facilitate a graph representation, while the energy models are discrete-time regression models with a simulation-based approach.

The energy-aware production scheduling system architecture allows for the configuration, parameterization, and solving of the production scheduling model. It is based on the *eta_utility* framework and consists of the optimization algorithm and the primary modules of the production system environment. Together with the actual production system and external entities it forms a cyber-physical production system, as illustrated in Figure 5.5.

The optimization algorithm used is the NSGA-II, originally proposed by Deb et al. (2002), with adapted mutation and crossover operators to support the solution encoding used in this thesis. The solution encoding (see Section 5.2.2) uses a graph representation of the production scheduling model to create a sequence chromosome and a time chromosome. These chromosomes each represent part of the solution and are combined to create complete production schedules using the graph-based decoding algorithm described in Section 5.2.3.

Solving the model requires a fully configured production system environment that includes multiple attributes describing the actual production system, such as production machine models, the product configuration, storage configuration, and production order configuration. These attributes are configured using JSON files. The production machine models are a fundamental component of the production system environment because, besides some production related-information, they contain the energy models describing the electric and thermal power. The energy model parameter estimation integrates with the production machine models and helps to estimate their parameters using measured data.

The production system environment also maintains connectors that enable it to connect to and communicate with the actual production system and other entities. It uses these components to gather information about the production system's state, control the production machines, and obtain additional information such as energy prices.

The energy-aware production scheduling system architecture is built to address the technical requirements established in Section 4.1 and provides a standardized approach to implementing the production machine models. It enables connectivity and rescheduling and performs multi-objective optimization while allowing for individualized objective criteria independent of the production-related and energy-related models. The architecture is also fully configurable, which is another crucial requirement from Section 4.1.

6. Deployment and Evaluation

The second and third research questions (refer to Section 1.1) focus on evaluating the implementation procedure and energy-aware production scheduling system architecture. The second research question asks whether "a standardized and partially automated implementation procedure for the adoption of energy-aware production scheduling systems [can] be proposed such that an energy-aware production scheduling system can be more easily applied to real industrial use cases." The third research question scrutinizes the system architecture: "How should the architecture of an energy-aware production scheduling system be designed to support the implementation procedure, and which additional tooling is needed to reduce the implementation efforts?"

Sections 4 and 5 propose a concept for an implementation procedure and energy-aware production scheduling system architecture which can answer the research questions. Since the research questions ask about suitability for actual implementations, which the literature review in Section 3 also identified as a critical factor, this section aims to follow the proposed implementation procedure (refer to Section 4.2) for a research factory. Thus, this section constitutes the initial descriptive study-II stage of the Design Research Methodology following the research design established in Section 1.2.

The first part of this section introduces the Energy Technologies and Applications Research Factory (ETA Research Factory) (Abele et al., 2018) and its production system, which this thesis uses to validate the proposed approach. The second part follows the implementation procedure and evaluates how the procedure and the energy-aware production scheduling system architecture work for a real production system. Finally, the third part presents preliminary results from transferring the proposed implementation procedure and energy-aware production scheduling system architecture to an industrial company, and the fourth part evaluates the requirements and success criteria from Section 4.1.

6.1. Use Case ETA Research Factory

The ETA Research Factory at the Technical University of Darmstadt is well suited to evaluate the production scheduling system architecture because it provides a full-scale production process with actual production machines. The factory also has an established Industrial Internet of Things architecture with installed sensors measuring the power consumption of production machines and other equipment (Abele et al., 2018). The product produced in the ETA Research Factory is a control plate for a hydraulic pump needing milling, cleaning, hardening, and grinding operations (Abele et al., 2018). The factory building is also equipped with elaborate Technical Building Services, tightly integrating the factory building and the production system (Abele et al., 2018).

Figure 6.1 illustrates the energy networks and Technical Building Services in the ETA Research Factory (Abele et al., 2018). The production system shown in the center consists of two machine tools, two cleaning machines and a tempering furnace (Abele et al., 2018). These production machines are connected to Technical Building Services for energy and media supply (Abele et al., 2018). For example, the machine tools need cooling and compressed air, and the second cleaning machine needs a heat supply (Abele et al., 2018). The bottom of the figure shows the Technical Building Services present to provide energy and other media for the production machines (Abele et al., 2018). For example, Combined Heat and Power Unit (CHP) units generate heat and electricity from natural gas, and an air compressor provides compressed air to the production machines (Abele et al., 2018).

There are multiple energy storage facilities to operate the system most efficiently during every season of the year (Abele et al., 2018). Underground High-Volume Fly Ash (HVFA) insulated storage tanks store water at three different temperature levels (Abele et al., 2018). Each temperature level has a heating or cooling network connected to the production machines and other systems (Abele et al., 2018).

As Figure 6.1 highlights, the energy supply systems of the ETA Research Factory are complex and tightly interconnected with the production machines. The production machines themselves are typical machines found in industrial applications. The first machine tool is an Emag VLC100 Y vertical Computerized Numerical Control (CNC) lathe with driven tools for milling and drilling operations. The second machine tool is an Emag VLC100 GT CNC grinding machine. Both machine tools have automatic workpiece pick-up systems allowing automated operation. The two cleaning machines are operated manually and allow for cleaning batches of up to 42 parts at once. Cleaning machine 1 in Figure 6.1 is a Mafac KEA machine with an electric heater, and cleaning machine 2 is a Mafac JAVA heated by the hot water network. The heat pump between the cleaning machine and machine tools from Figure 6.1 has recently been removed. Lastly, an IVA tempering furnace is used to nitride the workpieces (surface hardening). The furnace is connected to the hot and warm water networks and acts as a heat supplier to both networks.

Overall, the production system of the ETA Research Factory is a relevant use case and allows for a first evaluation of transferability due to its various production

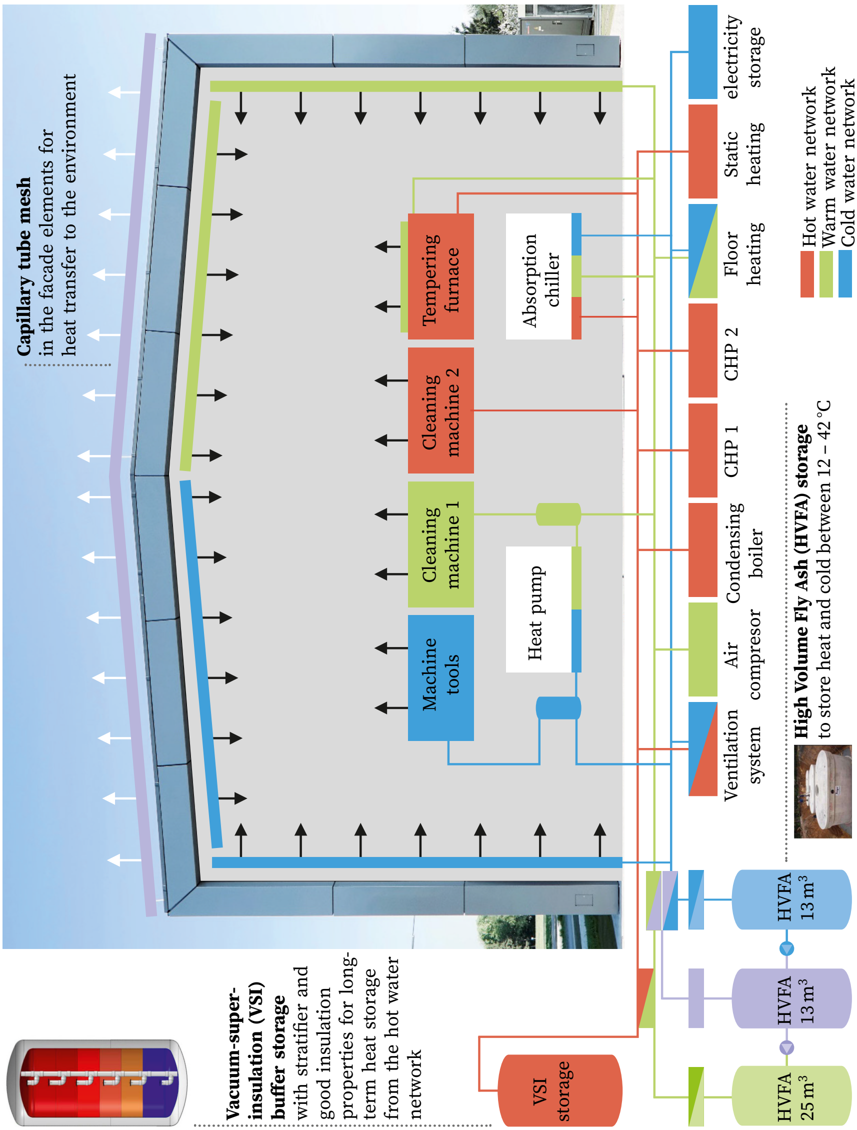


Figure 6.1: Energy systems in the ETA Research Factory (Abele et al., 2018).

machines. It also poses modelling challenges due to the tight integration of Technical Building Services and production machines. In the future, the production scheduling problem could incorporate additional complexity by increasing the number of products produced in the factory

6.1.1. Production Process

As previously mentioned, the ETA Research Factory produces a control plate for a hydraulic pump. The production process derives from a real industrial process in one of BOSCH REXROTH's factories and includes equivalent production operations.

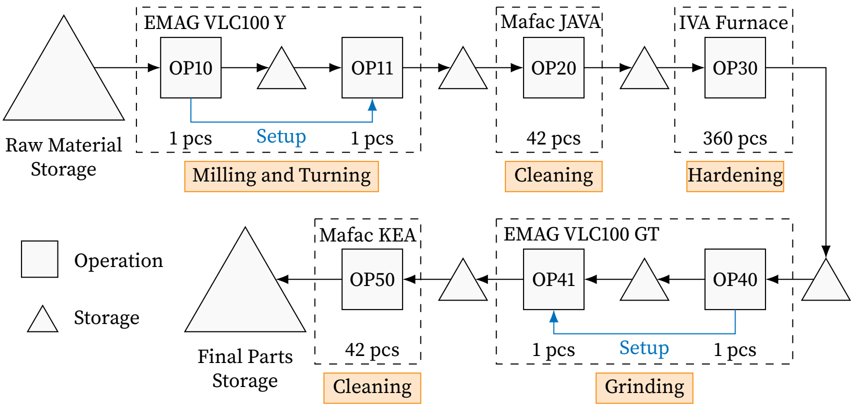


Figure 6.2: Production process for the control plate in the ETA Research Factory (Abele et al., 2019).

Figure 6.2 illustrates the entire production process. The raw material consists of forgings close to the final geometry. The machine tools feature equipment for automated handling of these forgings, and the process begins with turning and milling operations on the front (OP 10) and backside (OP 11). These two operations are responsible for the most material removal. The process then continues with cleaning the parts using the MAFAC JAVA aqueous cleaning machine (OP 20) before a heat treatment operation (OP 30) in the tempering furnace. The heat treatment process is a nitriding surface hardening process. Two grinding operations (OP 40 and OP 41) create the required surface finishes on the workpiece's front and backside, respectively. The production process concludes with a final cleaning operation (OP 50) before moving the finished workpieces to the final parts storage.

Figure 6.2 also indicates the capacity of the machines for each production operation. The machine tools have a capacity of one piece for each operation at a time, while the cleaning machines can handle 42 pieces per batch and the tempering furnace processes up to 360 pieces in a single batch. The figure also shows the need for setup time between operations OP 10 and OP 11 and between operations OP 40 and 41. The capacity and distribution of operations on the machine tools lead to the following limitations from a production scheduling point of view:

- The production machine capacity is not level over the entire production process. Hence, the cleaning machines have a much higher capacity and correspondingly lower utilization than the machine tools.
- The machine tools have to go through a setup process when switching between processing the front and backside of the workpieces. This dependency entails relatively large numbers of stored workpieces and decreases the production process flexibility because it increases the interdependence of production operations.

The setup times for changing operations on the machine tools depend on the operator's experience. They are approximately 15 min for the EMAG VLC100 Y and 5 min for the EMAG VLC100 GT for operators with intermediate experience. The processing times for the two cleaning operations (OP 20 and OP 50) are known to be 12 min, and the processing time for heat treatment is 36 h (Abele et al., 2019). The other processing times are unavailable before the case study begins and are determined in Section 6.2.2.

A third limitation, closely related to the first limitation, is the high capacity and very long duration of the hardening process in the furnace. In terms of energy consumption, it usually only makes sense to operate the oven at full capacity; however, producing enough parts to fill the furnace would take multiple days of production by the machine tools. Therefore, the following implementation follows the simplified production process illustrated in Figure 6.3. The tempering oven, which is excluded in the simplified process, should only operate when its entire capacity can be utilized – anything else would waste too much energy. Thus, the oven does not benefit from additional scheduling optimization and the simplified production process does not have relevant drawbacks for the purpose of evaluating the energy-aware production scheduling system's implementation.

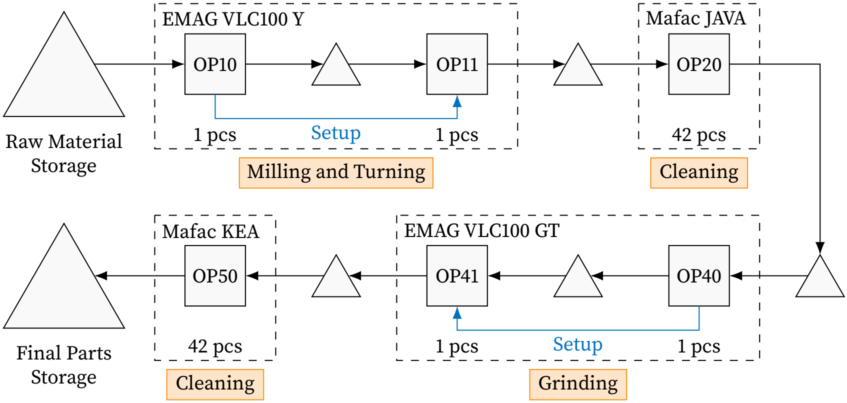


Figure 6.3: Simplified production process for the control plate in the ETA Research Factory.

6.2. Energy-Aware Production Scheduling System Implementation

The following sections build on the basic knowledge about the production system detailed in Section 6.1 and utilize the implementation procedure proposed in Section 4.2 to implement the energy-aware production scheduling system for the ETA Research Factory. The implementation begins with the *discovery and planning* phase, continues with the *development and configuration* phase, and finishes with the *testing and deployment* phase. Refer to Figure 4.1 on page 68 for an overview of the phases.

The discovery and planning phase consists of a detailed evaluation of the scope and goals to identify requirements and objective criteria and to create configuration concepts and software implementation plans. The configuration concepts and software implementation plans are used during the development and configuration phase to implement missing features and to create the configuration files introduced in Section 5.2. Finally, during the testing and deployment phase, the implemented system is tested with sets of orders and stored parts to create production schedules. The last phase also tests the communication between the scheduling system and the production machines.

6.2.1. Discovery and Planning Phase

Section 4.2.1 introduces the steps of the discovery and planning phase – refer to Figure 4.2 on page 70 for an overview of the steps. The phase begins with assembling and qualifying an implementation team and establishing the project budget. Together with other stakeholders in the implementation process, the implementation team discusses the project scope, goals and requirements. After establishing this basic framework, the team should evaluate the factory's energy supply and determine suitable objective criteria for the energy-aware optimization. Then the team identifies relevant production machines and jobs and checks whether the existing energy-aware production scheduling system architecture can represent them and which adaptations to the software and processes might be necessary. The discovery and planning phase concludes with creating a software implementation plan and configuration concepts. The following describes the results of each discovery and planning phase step for the ETA Research Factory use case.

Assemble and Qualify Implementation Team. Since this is a research project in a research environment, the implementation team mainly consists of the author of this thesis. Some student assistants and other stakeholders are involved in the project at various points, for example, when performing experiments. It is evident that a research use case is unsuitable for identifying shortcomings regarding organizational aspects of the proposed implementation procedure; however, this aspect still needs to be considered for industrial implementations.

Establish Project Budget. This step also does not directly apply to a research environment and cannot be validated well with the use case. In the research context, the budget could be considered limited by the time available to the implementation team. The project budget is somewhat limited since one person primarily performs the implementation with support from student assistants with limited available working hours. The limited budget demands some simplifications regarding the configuration, parameter estimation and testing.

Set Project Scope, Goals and Requirements. Section 6.1 describes the production system, which is this project's scope. As a simplification, this implementation will only consider the production machines and neglect a detailed analysis of the Technical Building Services. Nevertheless, the production machine energy models must specify the machine's thermal behaviour to identify their total electric power consumption. This case study's conversion between thermal and electric power uses a constant efficiency factor for simplicity. In the future, the scheduling system could integrate more detailed models of the Technical Building Services.

Implementing the energy-aware production scheduling system in the ETA Research Factory aims to have equivalent performance regarding the makespan as an SPT dispatching rule. At the same time, the optimization should reduce energy consumption or energy cost. In addition to these goals, there are the following requirements:

- The energy-aware production scheduling system should schedule a full day in advance in a reasonable time frame of approximately 15 min.
- As explained in Section 6.1, there is one product and four machines; however, the energy-aware production scheduling system should be extensible to multiple products.
- The system should consider the overall electricity cost of the proposed schedules.

Evaluate Energy Supply. The production machines in the ETA Research Factory mainly operate using electric energy. The MAFAC JAVA additionally utilizes the hot water network to heat the cleaning medium tank, and the EMAG VLC100 GT uses cold water from the cold water network for machine cooling. The two machine tools have a combined Direct Current (DC) electricity supply for their motors. The warm water network cools the DC converters. All machines also use compressed air supplied by an electric central air compressor connected to the warm water network for waste heat recovery.

A central compression chiller provides cold water for the cold water network, and central CHP units heat the water in the hot water network. Thus, the thermal energy supply is only partially electric; however, since this thesis focuses on evaluating the energy-aware production scheduling, not scheduling for Technical Building Services, this thesis assumes the thermal energy supply is fully electric. This thesis also assumes that the EMAG VLC100 Y is connected to the cold water network while it really has a separate compression chiller.

The ETA Research Factory building additionally provides options for cooling or heating the factory hall using a ventilation system and capillary tube mats in the walls and ceiling. The waste heat from the production machines directly influences the energy consumption of this heating, ventilation and air conditioning system; however, this thesis neglects this aspect to reduce the model's complexity.

Since the following considers all systems as using electric energy, the question of electric energy supply remains. The demand for heat affects the electric energy supply, which in turn affects the CHP units' operation and electric energy output. This thesis also neglects these effects and assumes that the electric power grid provides all the needed electric energy. This thesis also assumes that the electric utility company supplying the factory's electricity offers time-of-use pricing. Time-of-use pricing should indirectly decrease the carbon emissions of electricity generation because there is a correlation between electricity prices and carbon emissions (see: Grosch et al., 2021). This thesis does not directly consider the carbon emissions from using electricity.

Determine Objective Criteria. Since the goals for this implementation include optimizing a full day in advance, makespan is well-suited as an optimization criterion. Makespan is an evaluation of the total duration of production for the entire production schedule (refer to Section 2.1.3); thus, it ensures that production schedules do not exceed the available time, and it allows users to experiment with the number of products the factory produces during the available time. Additionally, makespan is easy to calculate using available data. Using it as an objective could provide similar results to optimization with the SPT dispatching rule defined as the reference case above. Equation (6.1) calculates the makespan *MKSP*.

$$MKSP = \max_{m \in M, h \in e_m} c_{m,h} \tag{6.1}$$

Since the energy supply systems in the ETA Research Factory are very complex and will not be fully modelled, using factors like the production's carbon emissions does not seem sensible. Instead, since one of the requirements is considering the energy cost of production schedules, using electricity cost as the secondary objective is reasonable. Cost-based objective criteria also have the advantage that cost reductions can strongly incentivize companies to implement new systems. Since this case only considers electric energy, electricity cost is equivalent to the total energy-related cost given by equation (6.2).

$$ERC = \sum_{t_0}^{t_{\max}} \sum_{m \in M} P^{\mathrm{el}}_{m,t} \tag{6.2}$$

Identify Relevant Production Machines and Production Jobs. Section 6.1.1 describes the production process performed in the ETA Research Factory. Since this process is straightforward enough, including the entire process and all machines makes sense. Section 6.1.1 also explains that the production scheduling system excludes the tempering furnace due to its high capacity and very long process times. The furnace should operate at full capacity whenever possible, thus not requiring more complex optimization.

Check whether the scheduling system can represent the machines and jobs. The proposed energy-aware production scheduling system includes models of machine tools with separate cooling supply. Using the regression parameters, these models should appropriately represent the machine tools present in the ETA Research Factory. The cleaning machine model separates the thermal power of the cleaning medium tank heater and the electric power for the machine's operation. This model holds for the MAFAC JAVA, which has a heat exchanger connected to the hot water network of the factory. On the other hand, the assumptions made in the cleaning machine model only partially hold for the MAFAC KEA because it uses an electric heating element. Thus, the cleaning machine model must be adapted to correctly represent the electric energy consumption of this type of machine.

The proposed energy-aware production scheduling system architecture can represent the production process of the ETA Research Factory fully; however, preliminary evaluations showed that the solution encoding and decoding strategies proposed in Sections 5.2.2 and 5.2.3 might lead to problems if a single product has multiple operations on the same machine. The problems likely arise because this requires exactly the correct number of events for the first operation to occur before the events for the second operation on the same machine. Products where separate machines perform each operation do not have this limitation. In this case suboptimal ordering of operations still leads to a long makespan but does not make the entire solution infeasible. The hybrid initialization strategy introduced in Section 5.2.1 helps improve this problem by providing the algorithm with a better starting point and avoiding generating too many invalid solutions at the start of the optimization.

Create Software Implementation Plan or Begin Change Process. Since the ETA Research Factory informed the energy-aware production scheduling system's creation, no significant changes are necessary. The most important adaptation is the inclusion of electric heaters for the cleaning medium in the MAFAC KEA. Looking back at equations (5.20) on page 88 and (5.33) on page 92, the thermal and electric power of the cleaning machine are already known. Since the electric heating element can be assumed to have a conversion efficiency from electricity to heat of approximately one (see: Fuhrländer-Völker et al., 2023), the thermal power can be added to the electric power resulting in equation (6.3) for the total electric power $P_t^{\mathrm{el,CM}}$ of an electric cleaning machine.

$$P_t^{\mathrm{el,CM}} = P_t^{\mathrm{el}} + P_t^{\mathrm{th,heater}} \tag{6.3}$$

Since no new parameters exist in this equation, the extended cleaning machine model does not require additional configuration options on top of those already introduced in Section 5.2.5. There only needs to be a differentiation between electrically heated cleaning machines like the MAFAC KEA and cleaning machines with a heat exchanger like the MAFAC JAVA.

Create Configuration Concept. The final outcome of this phase is the configuration concept, and a significant part of the configuration concept is the data-gathering plan. The data-gathering plan according to DIN ISO 50015 (Deutsches Institut für Normung e. V., 2018) prescribes a very detailed description of every data element to be gathered. Some of this information is irrelevant to this case study due to its research nature, for example, the access to measurement points, the responsibility for each data element, and operating constraints; thus, the following data-gathering plans do not include this information.

For a more succinct description of the data-gathering plans, the following is true for every data element if not specifically noted:

- Where possible, data will be collected every second.
- Outliers are eliminated from the data set.
- The measurements are instantaneous spot measurements.

The following provides examples of data-gathering plans for the different types of machines. Additionally, there is a data-gathering plan for general information. Manual data-gathering only occurs once, and the other variables are measured for a few days of production to gather enough data for accurate model parameter estimation. Data-gathering plans for the machines not shown here are in Appendix A. Refer to Section 6.1 for an overview of machines and sensors in the factory.

Table 6.1 is the data-gathering plan for generally applicable values, including Technical Building Services, temperatures and products. Since this is a general test of the production scheduling system and implementation procedure and the goal is not to accurately represent the factory's energy consumption, the heating and cooling efficiency are roughly estimated. The information about products is necessary to create the product configuration for the energy-aware production scheduling – it contains the product name in addition to information about each of the production operations needed to manufacture the product. The information

Table 6.1: Data-gathering plan for general information, products and operations.

Variable Name	Data Source	Type of Sensor	Unit
Temperature in factory	building automation PLC		°C
Heating efficiency	estimation		%
Cooling efficiency	estimation		%
Products	manual		-
Product name	manual	documentation	-
Operation IDs	manual	documentation	-
Processing times	manual	stopwatch	s
Setup times	manual	stopwatch	s

about products is collected manually. The processing times are measured for approximately five workpieces or batches.

Table 6.2 details the data-gathering process for the EMAG VLC100 Y machine tool. Since the machine tools are connected to a unified DC supply, the electric power is calculated as the sum of Alternating Current power, measured within the machine itself and DC power measured at the DC supply. The cooling power for the water-glycol mixture in the machine's cooling system can be calculated from the temperature difference between the feed and return and the volume flow. The machine's PLC defines the energy states depending on the machine's state. Finally, since the material removal rate for each operation is constant, manually measuring the weight before and after processing the workpiece and dividing it by the processing time is sufficient to calculate it. The material removal rate should be measured for multiple workpieces to avoid measuring errors.

Table 6.3 shows the data-gathering plan for the MAFAC JAVA cleaning machine. The machine has a built-in sensor measuring its electric power consumption. The MAFAC JAVA has two separate cleaning medium tanks, which it uses interchangeably throughout the cleaning process. For simplicity, the model averages the heating power and temperatures in the two cleaning medium tanks. The heat exchanger for the tank heating system has built-in sensors evaluating its performance: temperature probes in the feed and return lines and volume flow sensors.

This machine does not directly provide the energy states; a workaround is to use the machine's state indicator lights. All lights are off when the machine is operational, the green light lights up when the machine is working, and the yellow light lights up when the machine is in the standby state. Lastly, the tank heater operation can be calculated by checking whether there is a relevant amount of heating power. This case study uses 1000 watts as a minimal value.

Table 6.2: Data-gathering plan for EMAG VLC100 Y machine tool.

Variable Name	Data Source	Type of Sensor	Unit
Electric power	calculated (sum)		W
Machine (Alternating Current)	Janitza UMG 801	current transformer	W
Drives (DC)	QI-POWER-485-300	current transformer	W
Cooling power	calculated		W
Feed temperature	cooler PLC	HYDAC ETS 4146-A-050-000	°C
Return temperature	cooler PLC	HYDAC ETS 4146-A-050-000	°C
Volume flow	cooler PLC	HYDAC EVS 3116-A-0300-000	L/min
Energy states			
standby	machine PLC	software	binary
operational	machine PLC	software	binary
working	machine PLC	software	binary
Material removal rate	calculated		g/s
Weight before	manual	KERN 824	g
Weight after	manual	KERN 824	g
Processing time	manual	stopwatch	s
Product	machine PLC	software	-
Machine Temperature	machine PLC	internal sensor	°C

Table 6.3: Data-gathering plan for MAFAC JAVA cleaning machine.

Variable Name	Data Source	Type of Sensor	Unit
Electric power machine	Janitza UMG 96 RM	current transformer	W
Heating power	calculated (mean)		W
Feed temp. (tank 1)	machine PLC	PT 100	°C
Return temp. (tank 1)	machine PLC	PT 100	°C
Volume flow (tank 1)	machine PLC	internal (unknown)	L/min
Feed temp. (tank 2)	machine PLC	PT 100	°C
Return temp. (tank 2)	machine PLC	PT 100	°C
Volume flow (tank 2)	machine PLC	internal (unknown)	L/min
Medium tank temp.	calculated (mean)		°C
Tank 1 temperature	machine PLC	PT 100	°C
Tank 2 temperature	machine PLC	PT 100	°C
(Energy) states			
standby (yellow light)	machine PLC	software	binary
operational (no light)	machine PLC	software	binary
working (green light)	machine PLC	software	binary
heater operation	calculated	heating power > 1000 W	binary
Batch size	manual	counted	pieces
Capacity	manual	documentation	pieces
lower temperature limit	manual	documentation	°C
upper temperature limit	manual	documentation	°C

The data elements described by the data-gathering plans that are not marked with *manual* can be collected over the network via OPC UA or Modbus TCP. The data collection uses the connectors introduced in Section 5.2.7 and stores collected data in comma-separated value files.

Results. This first phase of the implementation procedure set the scope and goals for implementing the energy-aware production scheduling system in the ETA Research Factory. The detailed analysis of the production system shows that some simplifying assumptions are necessary to keep the modelling effort reasonable. These assumptions include a fully electric energy supply, including the hot and cold water networks. The simplifications also extend to more detailed aspects, like combining both tanks of the MAFAC JAVA cleaning machine.

The optimization will use the makespan *MKSP* and energy-related cost as objective functions and optimize the entire production system except for the tempering furnace. The configuration concept includes the relevant variables for configuring the production system environment with its machines and products, and the software implementation plan delineates the implementation of an additional cleaning machine model.

6.2.2. Development and Configuration Phase

The development and configuration phase utilizes the discovery and planning phase's results to implement changes to the energy-aware production scheduling system and adapt it to the actual production system. Section 4.2.2 introduces the steps of the development and configuration phase – refer to Figure 4.3 on page 74 for an overview of the steps. The development and configuration phase begins by executing the previous phase's software implementation plans and implementing the selected objective criteria. Afterwards, the production machine energy model parameters must be estimated, meaning that data must be collected according to the data-gathering plans, which are part of the configuration concepts. Besides the automatic parameter estimation, some data is also needed to manually configure the production machine models and production processes. The development and configuration phase also includes the configuration of the connectors, and it concludes with exporting the configured data such that subsequent steps can use it. The following discusses the results of the implementation procedure for the ETA Research Factory use case in more detail.

Implement Objective Criteria. Due to the energy-aware production scheduling system's modularity, implementing the makespan calculation is quite simple – each item in the schedule has an associated ending time (see also Section 5.2.3), and the latest ending time is equal to the makespan *MKSP*.

In contrast, the energy-related cost results from the machine's electric energy consumption and the energy prices. Thus, besides the production machine's energy models, the production scheduling system needs information about energy prices. The *eta_utility* framework provides a connector to the ENTSO-E transparency platform, which could read this information and transfer it to the energy-aware production scheduling system; however, to enable scheduling for future production days, the energy-aware production scheduling system would need price forecasts which are not readily available. This case study instead uses intraday energy prices for a fixed day, 29 November 2021.

Execute Software Implementation Plans. The software implementation plan only asks for a relatively minor adjustment. The equations for calculating the electric power of an electrically heated cleaning machine are included in a new class `ElectricCleaningMachine` which uses similar equations as the cleaning machine with the addition discussed in the previous phase (refer to Section 6.2.1).

Collect Production Machine Data. Throughout this thesis, multiple days of experiments were performed before coming to a point where collected data was usable. A major problem during these trials was the research production system's lack of stability. Since the system only operates for experiments and previously often operated without actually producing new workpieces, multiple days of production were spent identifying and solving problems that arose. The experiments began on 25 July 2022; however, since this was the first time in a few years that the machines had to produce actual workpieces for an extended amount of time, multiple malfunctions were identified. The MAFAC JAVA had problems with its heat supply, the MAFAC KEA had a malfunctioning motor, the EMAG VLC100 Y had incorrectly measured tools, and the EMAG VLC100 GT had a software problem intermittently blocking the process operation.

After the manufacturer's technicians fixed most of the problems identified in July, new experiments were performed in October 2022; however, problems with data collection via OPC UA and additional reliability problems with the machines prevented successful data-gathering. Additionally, the supply of forged raw parts ran out, and the supplier could not deliver more in a reasonable time frame.

Finally, the experiments were successfully conducted in December 2022, when the machines operated reliably. However, due to the previous problems with the production system the total amount of collected and usable data remains relatively small, containing only three production shifts of approximately eight hours each. Due to the lack of forged raw parts, the experiments used turned raw parts designed as a replacement with a different material (42CrMo4 instead of 8CrMo16). Measurements during the ETA-Fabrik project showed that the EMAG VLC100 Y has a higher power consumption when processing 42CrMo4 instead of 8CrMo16 (Abele et al., 2019). This difference means that the results produced during the experiments in December cannot be directly compared to the results of previous experiments.

The following two figures show data collected from the experiments on 13 December 2022. More experiments took place on 14 December and 21 December 2022. Refer to Appendix B for data from all machines and days not displayed here. Figure 6.4 displays data from the EMAG VLC100 GT machine tool, and Figure 6.5 displays data from the MAFAC KEA cleaning machine.

Figure 6.4 plots the electrically heated cleaning machine's electric power consumption, energy states, number of workpieces and various temperatures from top to bottom. The plots begin with a short setup phase until about 9:30. After the initial startup phase, the machine enters its warm-up phase, which it indicates as an *operational* state. Production begins after the warm-up phase and runs until approximately 10:30, when it is interrupted by a fault. Between 11:00 and 11:30, production resumes before it stops for a break due to a lack of raw parts. This lack is due to the slower production rate of the EMAG VLC100 Y machine tool, which performs prior operations. The plot confirms that the data-gathering process for the machine performs correctly.

Figure 6.5 plots the electrically heated cleaning machine's electric power consumption, energy states, number of workpieces and various temperatures from top to bottom. This plot begins at the same time as the previous plot; however, it includes part of the pre-heating phase before the machine reaches its operating temperature of approximately 60 °C. The figure also illustrates that the tank heating system is off while the machine is in the standby state between 11:00 and 14:00. Before the next cleaning operation can begin after this, some pre-heating is required.

Select Machine Models and Estimate Parameters. In the case of the ETA Research Factory, selecting the machine models and estimating the parameters is

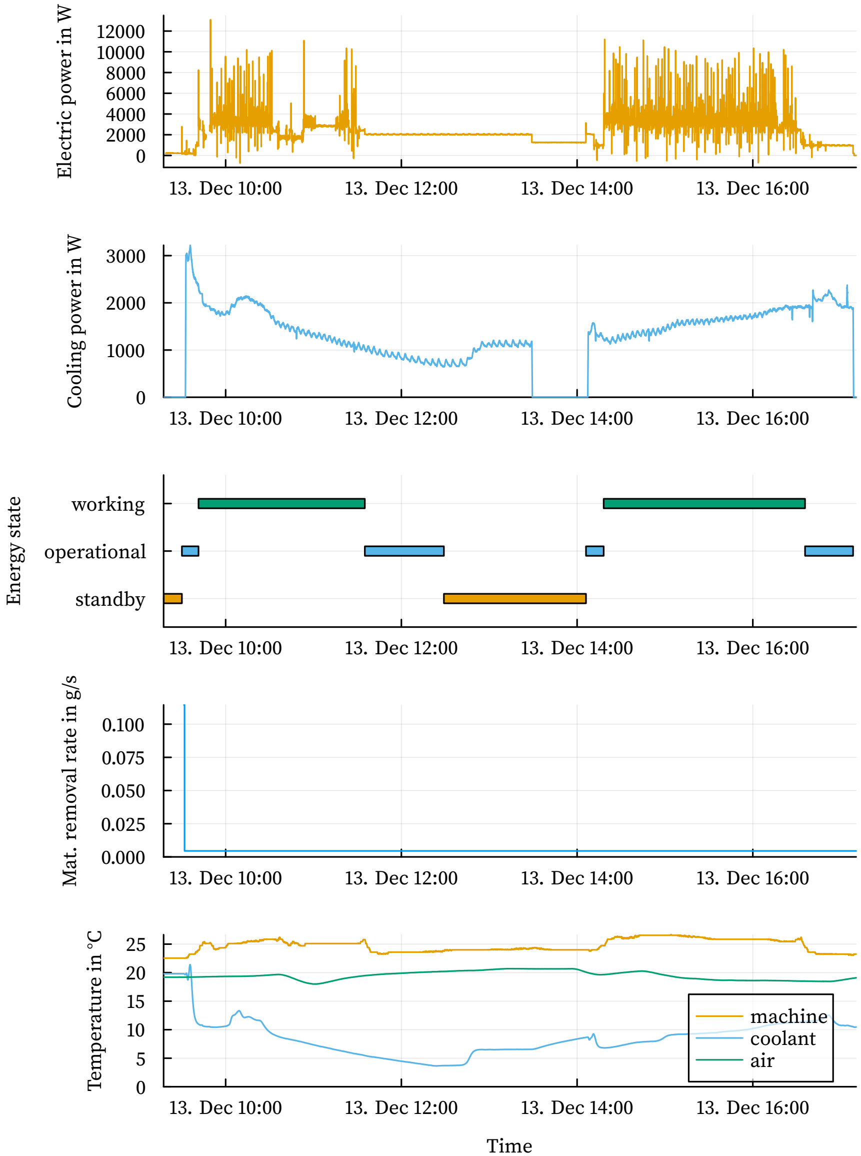


Figure 6.4: Gathered data for the EMAG VLC100 GT machine tool during the experiments on 13 December 2022.

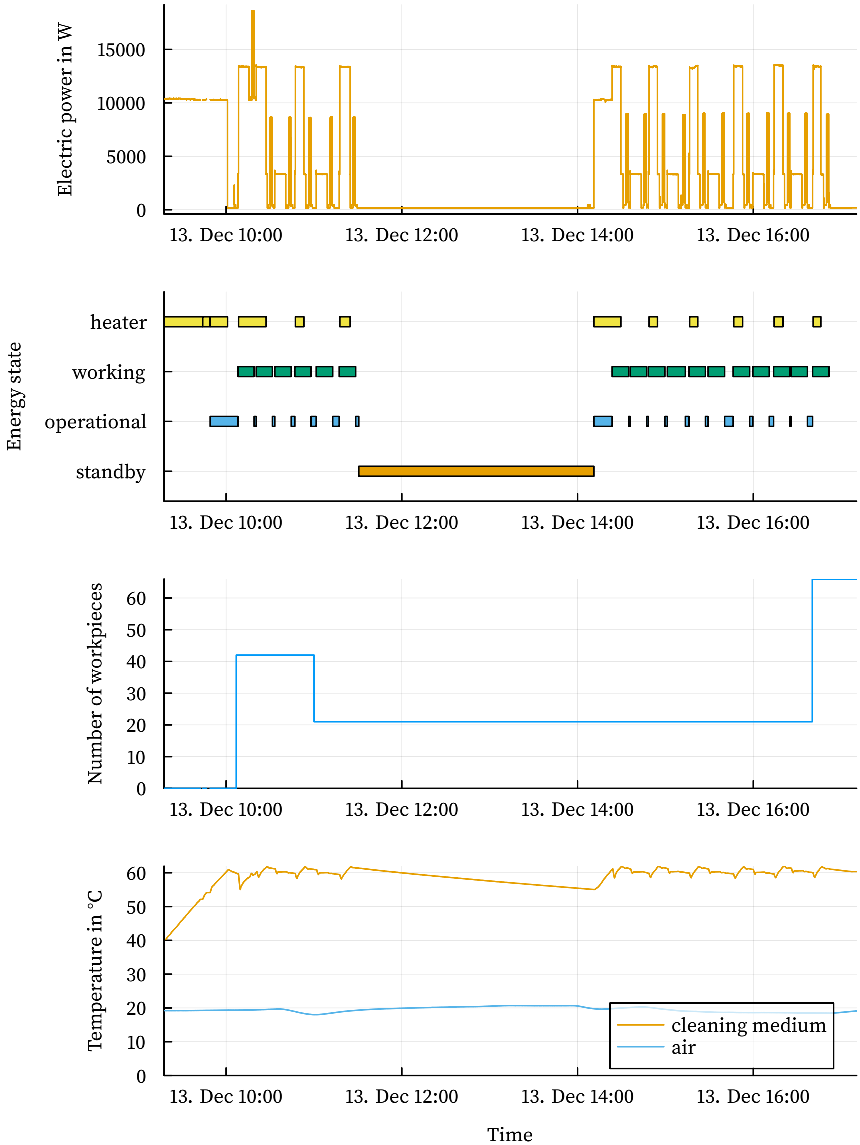


Figure 6.5: Gathered data for the MAFAC KEA cleaning machine during the experiments on 13 December 2022.

Table 6.4: Configuration parameters for the production machines in the ETA Research Factory.

Parameter	EMAG VLC100 Y	EMAG VLC100 GT	MAFAC JAVA	MAFAC KEA
Identifier	971	972	981	985
Capacity	1 piece	1 piece	42 pieces	42 pieces
Unique Job	true	true	false	false
Upper temp. limit	-	-	62 °C	62 °C
Lower temp. limit	-	-	58 °C	58 °C

relatively simple. The existing machine tool model presented in Section 5.2.5 captures the machine tools' energy consumption well. The cleaning machine model presented in Section 5.2.5 reflects the behaviour of the MAFAC JAVA, and the cleaning machine model adapted for electrically heated cleaning machines presented in Section 6.2.1 represents the MAFAC KEA.

The parameter estimation is performed as described in Section 6.2.2. The parameter estimation uses the CPLEX solver (IBM Corporation, 2022) to estimate the regression parameters of the production machine energy models. The parameter estimation uses data from 13 and 14 December 2022. The testing and deployment phase (refer to Section 6.2.3) shows the parameter estimation results in more detail.

Configure Production Machines. As the data-gathering plans in Section 6.2.1 state, some data about the production machines must be collected manually. Table 6.4 displays the complete configuration for all involved production machines. These parameters are in addition to the parameters estimated by the model parameter estimation process.

Collect Production Process Data. The collection of production process data for all machines also occurred during the experiment days in December 2022. Five workpieces were identified and measured throughout their production process to ensure the reliability of the collected data. The relevant process data identified during the discovery and planning phase includes the weights before and after the processing steps on the machine tools as well as the processing time on the machine tools.

Table 6.5 lists the weights of each workpiece after the completion of the respec-

Table 6.5: Weights of workpieces after each operation as measured during production on 14 December 2022.

Workpiece	Raw	OP 10	OP 11	OP 40	OP 41
1	671.24 g	378.11 g	266.74 g	257.63 g	256.98 g
2	678.60 g	381.79 g	266.91 g	257.79 g	256.85 g
3	677.43 g	377.09 g	266.94 g	257.78 g	257.01 g
4	684.70 g	386.15 g	266.59 g	257.57 g	256.72 g
5	671.85 g	383.34 g	266.38 g	257.55 g	256.84 g
Average	676.76 g	381.30 g	266.71 g	257.66 g	256.88 g

Table 6.6: Processing times of workpieces as measured during production on 14 December 2022.

Operation:	OP 10	OP 11	OP 40	OP 41
Processing time:	326 s	113 s	79 s	174 s

tive operation. Notably, the weights after operation OP 41 are very similar, while the weights of the raw parts show much more significant deviations. These differences are most likely due to the increasingly tight tolerances. Table 6.6 details the processing times for each operation. Since the processing times were measured using a stopwatch and the deviations between different measurements were less than one second, the table only shows one value rounded to whole seconds.

Configure Products and Processes. The product configuration uses the previously collected data. The process-dependent parameter z_{src}^{proc} for the cleaning machines is the number of workpieces, and for the machine tools it is the material removal rate in g/s. The material removal rate can be calculated using the data from Tables 6.5 and 6.6.

Configure Internal and External Communication. As previously discussed, this case study does not consider external communication. Additionally, the case study does not directly validate internal communication apart from what is necessary to gather data for model parameter estimation. Grosch, Fuhrländer-Völker, et al. (2022) and Fuhrländer-Völker et al. (2023) perform preliminary studies of communication and some aspects of rescheduling. Section 6.2.3 discusses some additional preliminary experiments; however, fully configuring and controlling the entire ETA Research Factory production system exceeds the scope of this work.

Export Configuration. After completing the configuration, there are two JSON files representing the production system of the ETA Research Factory. After inserting manually collected information, the energy model parameter estimation automatically generates the first file containing the machine configuration. The second file contains the product and process configuration and describes the operations to produce the hydraulic control plate as described in Section 6.1.1. This second file was created manually. A third JSON configuration file contains the configuration for internal and external communication using the connectors.

Results. The development and configuration phase required implementing the objective criteria and executing the software implementation plans created in the previous phase. The software implementation plans included the creation of an adapted cleaning machine model. The most important part of this phase was performing the experiments. The experiments revealed some limitations of the research production system, which is less stable than an actual production system; however, it also validated many of the processes proposed by the implementation procedure.

The configuration files created during this phase include the information necessary to operate the production scheduling system. This includes information representing the production system structure, the estimated parameters for the production machine energy models and data for internal and external communication.

6.2.3. Testing and Deployment Phase

After preparing the energy-aware production scheduling system during the development and configuration phase, the testing and deployment phase ensures that the system is suitable for scheduling the actual production system. Section 4.2.3 introduces the steps of the testing and deployment phase – refer to Figure 4.4 on page 77 for an overview of the steps. This phase performs various tests, such as of the parametrized model's quality and communications. Additionally, the testing and deployment phase includes user training and tests with real orders and users before deploying the system. Since this thesis performs an initial evaluation of the proposed concepts according to the Design Research Methodology, as discussed in Section 1.2, the following provides preliminary test results – future research should perform additional evaluations of these concepts.

Table 6.7: Collected machine data used for model parameter estimation. BOD: beginning of day, EOD: end of day.

Machine	13 December	14 December
Emag VLC100 Y	BOD to EOD	BOD to EOD
Emag VLC100 GT	13:20 to EOD	10:00 to 11:30 and 13:10 to EOD
Mafac JAVA	10:10 to EOD	10:00 to EOD
Mafac KEA	10:00 to EOD	9:50 to EOD

Check Parametrized Model's Quality. The energy model parameter estimation uses the production machine data collected during the development and configuration phase to find the best parameters for the production machine energy models. This thesis uses data collected on 13 and 14 December 2022 to estimate the model parameters and then uses data collected on 21 December to verify their quality. Table 6.7 contains the exact times used for parameter estimation. These fixed time frames exclude some invalid data, which would otherwise significantly decrease the estimated parameter's quality. Data validity problems occurred most often at the beginning of the day, while data from the end of each day was usually valid.

In addition to the periods with invalid data, another problem for the model parameter estimation is the measuring uncertainty in the temperature measurements. The uncertainty introduces noise in the signal and creates problems with the temperature gradient calculation. Since the gradient is calculated using adjacent temperature measurements, even slightly inaccurate measurements can cause significant inaccuracy in the temperature gradient. To reduce noise in the temperature measurements, the model parameter estimation uses moving averages over 20 s intervals. The moving average filter is well suited for this purpose because it is good at reducing noise in the time domain while preserving good step response (Smith, 1999, p. 277)

Figures 6.6 and 6.7 illustrate the testing data collected on 21 December 2022 and the corresponding predictions from the parametrized production machine energy models. Data for the machines not shown here is available in Appendix C. Figure 6.6 presents the test data for the Emag VLC100 GT. Since the machine depends on raw parts from the Emag VLC100 Y, there were not enough parts available for production in the morning; thus, the machine only operated in the afternoon. Before noon the machine remained in the operational or standby state. As the figure illustrates, the thermal power in the operational state is similar to that in the working state, while the electric power is significantly less.

Figure 6.6 illustrates that the energy models have some inaccuracies. For ex-

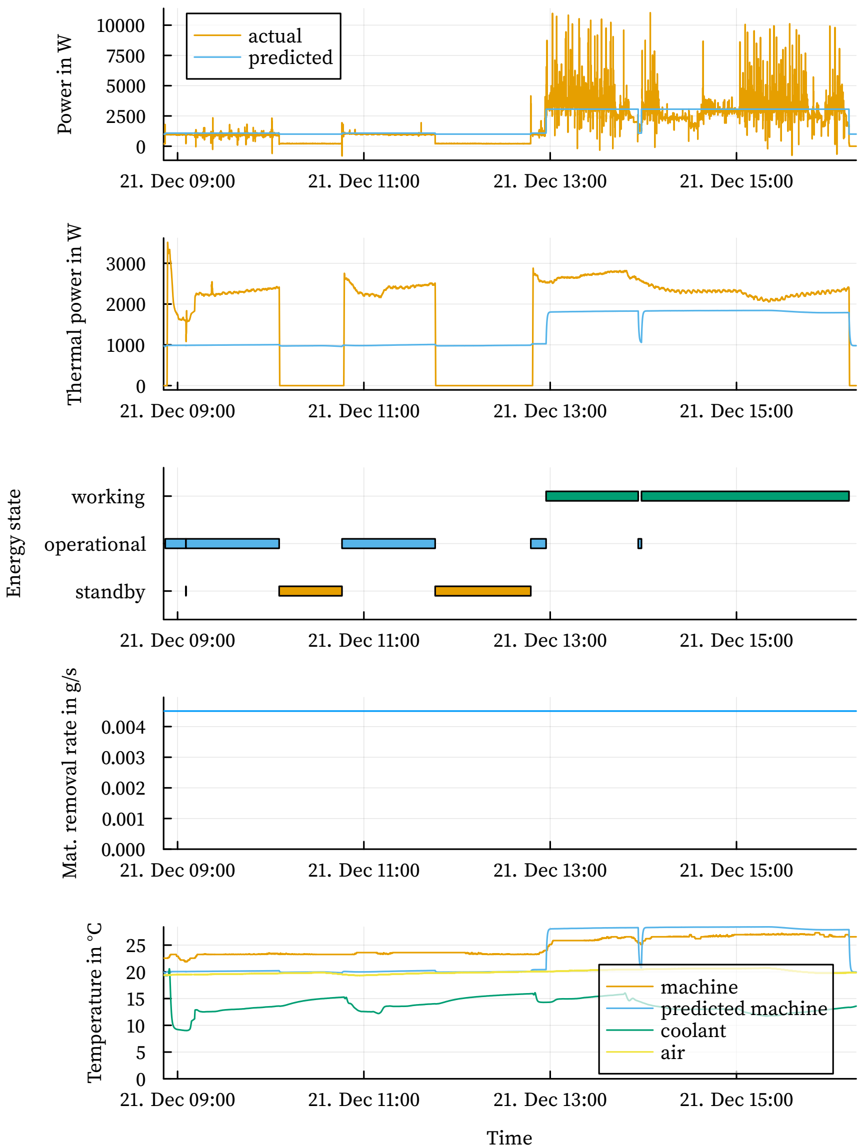


Figure 6.6: Testing data for the EMAG VLC100 GT machine tool during the experiments on 21 December 2022.

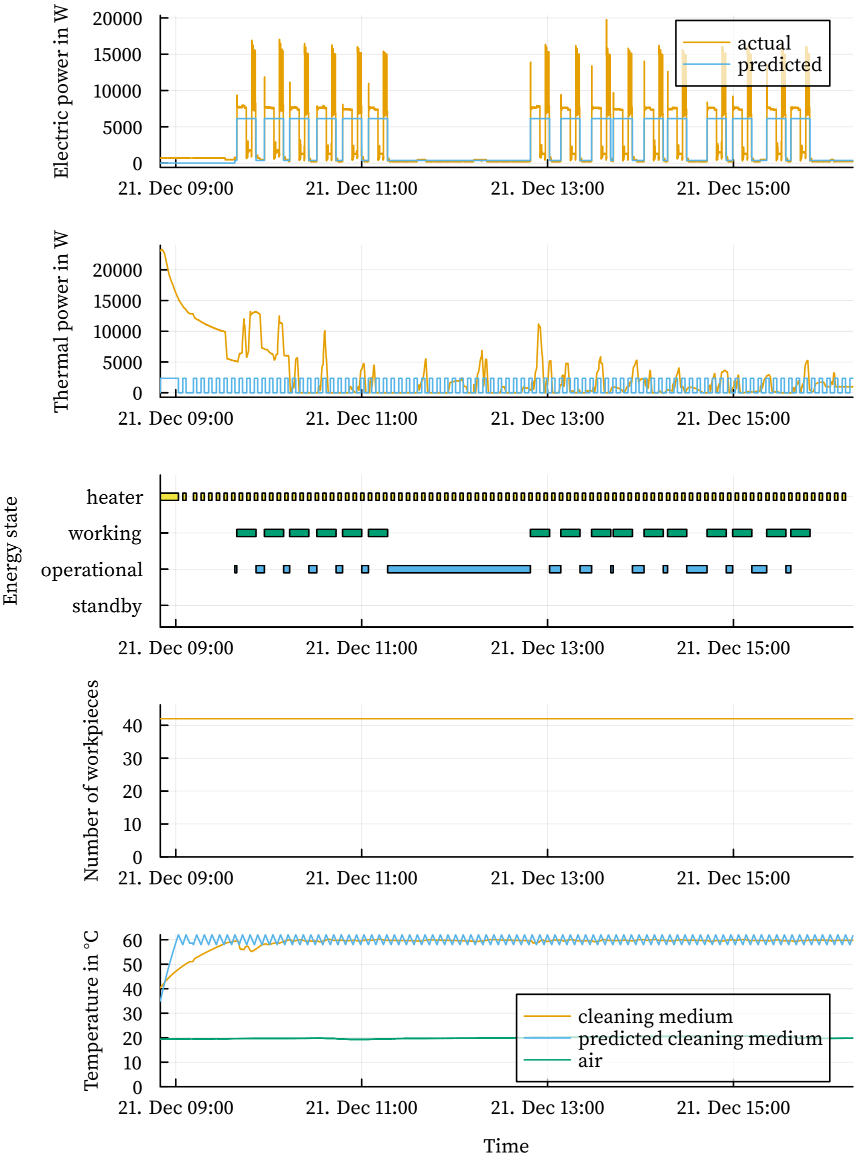


Figure 6.7: Testing data for the MAFAC JAVA cleaning machine during the experiments on 21 December 2022.

Table 6.8: Estimated parameters for each production machine. - indicates parameters not applicable to the respective model.

Parameter	EMAG VLC100 Y	EMAG VLC100 GT	MAFAC JAVA	MAFAC KEA
β^{el}_{st}	876.2	999.2	0.0	176.3
β^{el}_{op}	0.0	82.7	368.1	776.2
β^{el}_{wk}	2713.8	0.0	5765.4	3795.6
β^{el}_{proc}	0.0	1977.6	-	-
β^{c}_{M}	9475.8	7393.7	41 691.6	43 293.3
β^{th}_{cool}	135.5	102.4	-	-
β^{th}_{wp}	-	-	0.0	0.4
β^{th}_{spray}	-	-	1.0	7.8
β^{th}_{heater}	-	-	2349.9	3704.9
β^{th}_{env}	233.0	157.2	24.8	4.5

ample, the machine energy model does not accurately represent the machine's electric and thermal standby power consumption. While this might be due to inaccuracies in the mathematical models presented in Section 5.1.5, it could also be attributed to insufficient training data. Table 6.8 presents the estimated regression parameters for each production machine model. Notably, some parameters, for example, the β^{el}_{op} parameter of both machine tools, are 0 or very close to 0. This could indicate linear dependencies between parameters resulting from insufficient training data for the respective states. Such dependencies are even more apparent for the machine tools' β^{el}_{wk} and β^{el}_{proc} parameters, one of which is 0 for both machines because the variation of material removal rates for different parts on the machine tools is low.

Figure 6.7 illustrates data from the MAFAC JAVA cleaning machine. Between approximately 9 and 10 am, the machine was pre-heating its cleaning medium tank and started production soon after this process was complete. The figure shows indications that the parameter estimation underestimates the cleaning medium tank's heat capacity because the tank heater activates much more often than it does in reality. Looking at the model parameters in Table 6.8, it is clear that the energy model parameter estimation needs additional data to correctly estimate the effects of different numbers of workpieces processed concurrently. The models did not find a relation between the number of workpieces and the thermal power because limitations in the data available from this machine necessitated constantly cleaning the same number of workpieces. Overall, for both cleaning machines the heating power predictions are less accurate than the predictions of electric power. For the MAFAC KEA, the lower heating power prediction accuracy directly translates to lower electric power accuracy because it is heated electrically and

Table 6.9: Error measures for predicted values for 21 December 2022. - indicates measures not applicable to the respective model.

Parameter	Emag VLC100 Y	Emag VLC100 GT	Mafac JAVA	Mafac KEA
RMSE				
P^{el}	1163.7 W	555.6 W	2294.8 W	4716.9 W
$P^{\mathrm{th,cool}}$	396.8 W	991.2 W	-	-
$P^{\mathrm{th,heater}}$	-	-	5163.9 W	-
T^{M}	3.4 °C	2.8 °C	4.7 °C	2.8 °C
MAE				
P^{el}	887.2 W	438.1 W	1462.4 W	3492.1 W
$P^{\mathrm{th,cool}}$	291.1 W	916.7 W	-	-
$P^{\mathrm{th,heater}}$	-	-	3052.1 W	
T^{M}	2.8 °C	2.6 °C	2.3 °C	1.8 °C
Total Energy Percentage Error				
P^{el}	6.7 %	15.3 %	−1.8 %	−12.6 %
$P^{\mathrm{th,cool}}$	−14.1 %	−24.0 %	-	-
$P^{\mathrm{th,heater}}$	-	-	−64.5 %	-

heating power is part of its total electric power.

Table 6.9 summarizes the error measures Mean Absolute Error (MAE) and Root-Mean-Squared Error (RMSE) for each machine. The table also shows the total energy percentage error, which looks at the predicted total energy consumption over the entire day compared to the measured energy consumption. Since the production machine energy models only predict the average power for each machine energy state, the evaluation uses a moving average over 30 seconds for the measured electric power consumption. The smoothing decreases the impact of the high positive and negative deviations of measured values from predicted values due to the short measuring interval of 1 s compared to the much longer intervals between energy state transitions ranging in the realm of minutes.

The error measures show that future work should perform additional analyses of the production machine energy models and check whether there are ways to improve the thermal model quality. The total energy percentage errors reiterate that the electric power predictions are more accurate over the entire day than the thermal power predictions. All models appear to slightly overestimate electric energy consumption while they underestimate thermal energy consumption. This assumption could also be valid for the electric cleaning machine model, which sums the electric and thermal energy before calculating the total (see equation (6.3)).

Table 6.10: Stored items after each operation and orders for finished workpieces.

Experiment	OP 10	OP 11	OP 20	OP 40	OP 41	OP 50	Orders
1	0	0	0	0	0	0	5
2	0	0	0	0	0	0	10
3	0	0	0	0	0	0	15
4	0	0	2	0	2	0	10
5	5	0	5	5	0	0	15

A comparison between the RMSE and MAE for electric power emphasizes that many large short-term deviations lead to a significantly higher RMSE than MAE. Since the models do not aspire to represent the very short-term behaviour of the production machines accurately, this effect does not negatively impact the model's quality. Comparing the RMSE and MAE for thermal power and for the machine temperature shows less pronounced differences, likely due to thermal power's less volatile nature. Overall, the MAE for the production machine energy models ranges between approximately 10 % – 20 % of the machine's average power consumption. While this result is not perfect, the models are sufficient for the preliminary analysis in this thesis. Future work could try to improve the parameter estimation accuracy, for instance, by collecting additional data from the production system.

Test Data Acquisition. Data acquisition consists of two parts – reading data about the current production system state from the production machines and other Industrial Internet of Things devices in the factory and gathering data from users about the production scheduling problem. The step *test controlling production machines* details some experiments testing reading and writing data from production machines.

On the other hand, users supply data in a configuration file. The file contains information about customer orders and stored workpieces, as previously examined in Section 5.2.8. Data acquisition from users generally works properly. The final quality checks use the test configurations given in Table 6.10. The first three tests do not have any stored items but use increasing numbers of orders. Tests four and five additionally have some stored items. There are intentionally no stored items for OP 50 because this would be equivalent to reducing the number of orders.

Test Controlling Production Machines. As indicated in Section 6.2.2, this case study does not perform a complete validation of controlling the machines because this validation would exceed the scope of the initial descriptive study-II, which is the goal of this thesis. In addition to the experiments published in Grosch, Fuhrländer-Völker, et al. (2022) and Fuhrländer-Völker et al. (2023), which validated controlling the MAFAC KEA with the connectors and structure provided by the *eta_utility* framework, additional control experiments with the machines were performed.

The control experiments included all machines in the production system of the ETA Research Factory. During these experiments, the machines operated according to rules set by a simple controller. The experiments showed that controlling all machines is generally possible. The configuration file structure provided by the framework proved helpful in this case because the machine tools have three different Industrial Internet of Things devices, which the energy-aware production scheduling system must communicate with. Besides the machines themselves these devices include a separate power meter for each machine, and both machines connect to the unified DC supply. The configuration allows combining the separate Industrial Internet of Things devices for each machine into a single system such that they appear as one in the energy-aware production scheduling system.

Train Users. Validating user training exceeds the scope of this case study; however, according to literature (see: Schuh et al., 2015), this is an essential step for implementation in actual production systems. As mentioned in Section 4.2.3, the implementation team should resort to established training methods to ensure adequate training outcomes.

Test with Real Orders and Users. While testing with real orders and users is not directly possible due to the production system's origin in research, the following tests of the energy-aware production scheduling system use the configurations proposed in Table 6.10. Prior to performing the tests, preliminary evaluations are needed to determine appropriate parameters for the optimization algorithm. This thesis performs the preliminary evaluations with the second case from Table 6.10. This experiment provides a good combination of complexity and speed for many runs with varying parameters. The preliminary evaluations to determine appropriate algorithm parameters are executed in two incremental phases. The termination condition $g_{\max}$ equals 500 generations for all runs which, as Figure 6.8 shows, is sufficient to ensure that all experiments are mostly converged.

Table 6.11 shows the parameters analyzed during each phase of the algorithm

Table 6.11: Analyzed parameter values for the algorithm parameter variation.

Population $w_{\max}$	Crossover rate κ	Mutation rate ω
Phase 1		
50	0.1	0.025
100	0.15	0.05
200	0.2	0.075
	0.25	0.1
	0.3	0.125
		0.15
Phase 2		
100	0.2	0.02
200	0.275	0.035
	0.35	0.05

parameter variation. Every possible combination of these parameters was investigated – 90 combinations in phase one and an additional 18 combinations in phase two. The first phase uses a wider variety and number of parameters to roughly establish the appropriate ranges, while the second phase performs a more detailed analysis. The first phase shows that the mutation rate becomes more relevant during later generations when the solutions are overall more refined. In contrast, the crossover rate is more relevant during earlier generations at the beginning of the optimization.

The first phase of parameter variation shows that the refinement of solutions during later generations becomes significantly worse when the mutation rate is too high. The algorithm cannot find satisfactory solutions within the allotted termination condition of 500 generations when the mutation rate is above 0.05. On the other hand, the mutation rate of 0.025 shows an increasing number of collisions with solutions which the algorithm has seen before, leading to decreased improvement rates. Crossover rates are much less sensitive, but higher crossover rates generally lead to faster convergence toward adequate solutions. Similar to the mutation rates, if the crossover rates become too high, the final solution after 500 generations will be slightly worse. During the first phase crossover rates above 0.2 appear to perform well. Finally, the population size 50 is too small to find suitable solutions quickly while larger population sizes show diminishing returns, especially considering the increased overall duration of the optimization.

The second phase validates the findings of the first phase; larger populations improve convergence slightly at the cost of significantly increased optimization duration. Therefore, the following tests use a population of 100. Higher crossover

Table 6.12: Algorithm configuration after parameter variation.

Parameter	Value
Population $w_{\max}$	100
Crossover rate κ	0.4
Mutation rate ω	0.025
Learning rate	linear decreasing 1 to 0.5
Generations $g_{\max}$	500 (experiment 3: 1000)

and mutation rates also increase convergence speed at the beginning while decreasing the solution quality toward the end of the optimization. The best values according to the tests are a crossover rate of 0.35 and a mutation rate of 0.02. Since lower crossover and mutation values during later generations improve the solution quality, the following tests use a configuration with a linearly decreasing learning rate between 1 and 0.5 and starting points of 0.4 for the crossover rate and 0.025 for the mutation rate. The crossover and mutation rates are multiplied by the learning rate for each new generation. With the proposed configuration, the crossover rate during the last generation before the algorithm terminates will be 0.2, and the final mutation rate will be 0.0125. Table 6.12 summarizes the algorithm configuration after the parameter variation. The configuration with the learning rate converged approximately 100 generations earlier than the best configuration without the learning rate decreasing the crossover and mutation rates. The preliminary evaluations also verify the findings from Grosch et al. (2021): The hybrid initialization outperforms the normal random initialization (refer to Section 5.2.1).

In addition to the algorithm parameters in Table 6.12, the energy-aware production scheduling system needs starting values for the schedule optimization and production machine energy models. Table 6.13 shows these starting values. The scenario dates for energy prices and temperature measurements are two days for which data was available. Future work could implement predictions for energy prices and temperature measurements in addition to the energy consumption of production machines. The energy prices are 15 min intraday prices from EPEX Spot, and the temperature measurements were performed in the ETA Research Factory during experiments on 21 December 2022. The possible pause times are the durations of pauses that the algorithm can choose from using the time chromosome (refer to Section 5.2.2). The scheduler will wait 180 s before switching a production machine to the standby state (refer to Section 5.1.5). The production machine energy models also need starting temperatures – the starting temperatures are close to the environment's temperature for the two machine tools. The

Table 6.13: Parameters and starting values for the production schedule optimization.

Parameter	Value
Scenario Dates	
Date of energy prices	29 November 2021, 8:00 to 16:00
Date of temperature measurements	21 December 2022, 8:00 to 16:00
Pause Scheduling	
Possible pauses in s	0, 180, 300, 600, 900, 1800, 3600
Wait before standby	180 s
Starting Temperatures	
EMAG VLC100 Y T_1^{M}	20 °C
EMAG VLC100 GT T_1^{M}	20 °C
MAFAC JAVA T_1^{M}	60 °C
MAFAC KEA T_1^{M}	60 °C
Thermal Power Conversion	
Cooling efficiency	0.8
Heating efficiency	0.95
Random Seed for Reproducible Results	
Random number seed	542346723

cleaning machines' starting temperatures assume they have been pre-heated to their operating temperature of approximately 60 °C. The thermal power conversions are necessary because this thesis considers a simplified energy conversion system, as introduced in Section 6.1. These factors are chosen arbitrarily for this initial study of the energy-aware production scheduling system's performance. Future work could include more detailed models of the factory's building and energy supply systems.

The actual scheduling experiments can now be performed with the algorithm parameters from Table 6.12 and the other parameters and starting values from Table 6.13. Figure 6.8 illustrates the convergence for the experiments and both objective criteria. Most experiments converge well before the optimization's termination at 500 generations $g_{\max}$. Only the third experiment benefited from additional optimization time, so this run was repeated with a termination condition of 1000 generations $g_{\max}$. The figure confirms that the chosen algorithm parameters work well for all experiments. Note that the energy cost can only be calculated once the makespan *MKSP* is smaller than the available energy price data. Due to this limitation, the energy cost is not available for the earlier generations of some of the scheduling experiments.

Regarding the optimization duration, experiment 1 was the fastest, with 2 min 16 s, and experiment 3 was the slowest, with 11 min and 5 s. All experiments were

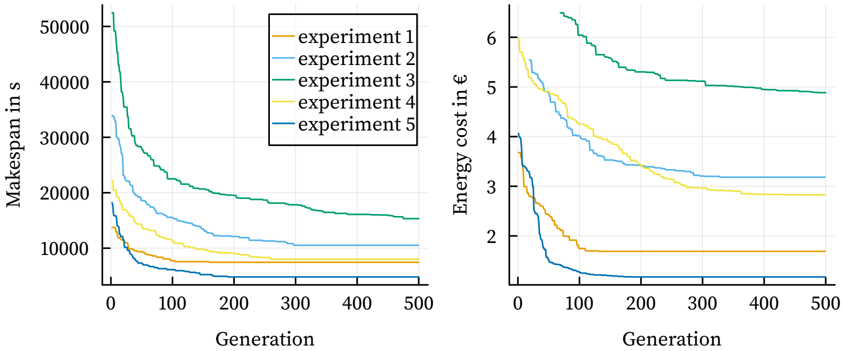


Figure 6.8: Convergence for both objective criteria and all experiments (only first 500 generations shown).

Table 6.14: Best solutions from each optimization experiment and comparison to the benchmark.

Experiment	Only SPT rule		Best *MKSP* solution		Best *ERC* solution	
	MKSP	*ERC*	*MKSP*	*ERC*	*MKSP*	*ERC*
1	7453 s	1.90 €	7573 s	1.76 €	7772 s	1.69 €
2	11 344 s	4.29 €	10 640 s	3.18 €	10 640 s	3.18 €
3	14 376 s	5.71 €	14 035 s	5.29 €	15 327 s	4.28 €
4	8411 s	3.63 €	8010 s	2.83 €	8010 s	2.83 €
5	5464 s	1.74 €	5391 s	1.17 €	5391 s	1.17 €

performed on a laptop with an 11th Generation Intel Core i7 1185G7 processor. The processor has four cores and eight threads with a base clock speed of 3 GHz. The implementation in Julia led to significant speed improvements over the Python implementations used in Grosch et al. (2019) and Grosch et al. (2021), where similar optimization runs would routinely take many hours on much more capable computers.

Table 6.14 presents the makespan and energy-related cost values for the best solutions for each objective criterion and every experiment. The table also notes the results achieved using just the SPT dispatching rule without further optimization. Looking at the generated schedules, which Appendix D illustrates, it is clear that the results generated by the SPT rule are an excellent benchmark for the makespan optimization. Since the energy price only significantly begins to rise after about 7000 s, some experiments only have one best solution for both dimensions because the solutions do not differ in their energy cost. For experiment 1 there are two solutions but the difference between the two best solutions is minimal.

Overall, the energy-aware production scheduling optimization can almost always find better solutions than the SPT rule. When looking at makespan as the only objective criterion, the best solutions found by the optimization are, on average, 3 % better than those found by the SPT rule. The energy-related cost is also improved by 13 % for the same solutions compared to the SPT rule. When choosing the solutions with the best energy-related cost instead, the makespan generally increases, but the energy-related cost also decreases significantly – for example, experiment 3 benefits most from the energy cost optimization due to its comparatively long makespan. Compared to the solution generated by the SPT rule, the best *ERC* solution for experiment 3 improves energy-related cost by 25 % while increasing makespan by 7 %. When comparing the same solution to the best *MKSP* solution generated by the optimization algorithm, there is an improvement of 19 % in *ERC* and an increase of 9 % in *MKSP*.

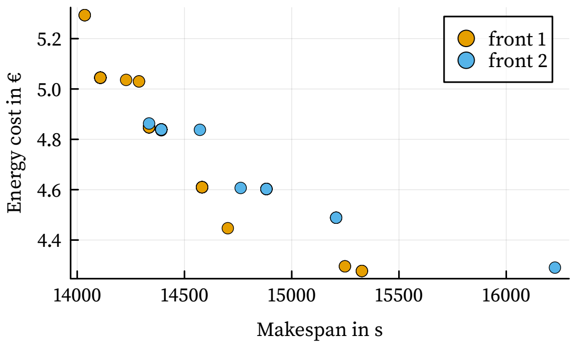


Figure 6.9: Solution space with two pareto fronts for experiment 3.

Figure 6.9 depicts the solution space for experiment 3. While Table 6.14 only shows the best solutions for either makespan or energy cost, the figure also highlights that multiple intermediate solutions are available. For example, if a user wanted a schedule below a makespan of 15 000 s, a significant improvement of the energy-related cost would still be possible by selecting the solution at approximately 14 700 s and 4.42 €.

Figures 6.10 and 6.11 show the best solutions from experiment 3 for makespan and energy-related cost, respectively. The topmost plot indicates when machines need to undergo setup to switch between operations, when there are scheduled pauses and when the production of a specific product takes place. The plots reflect

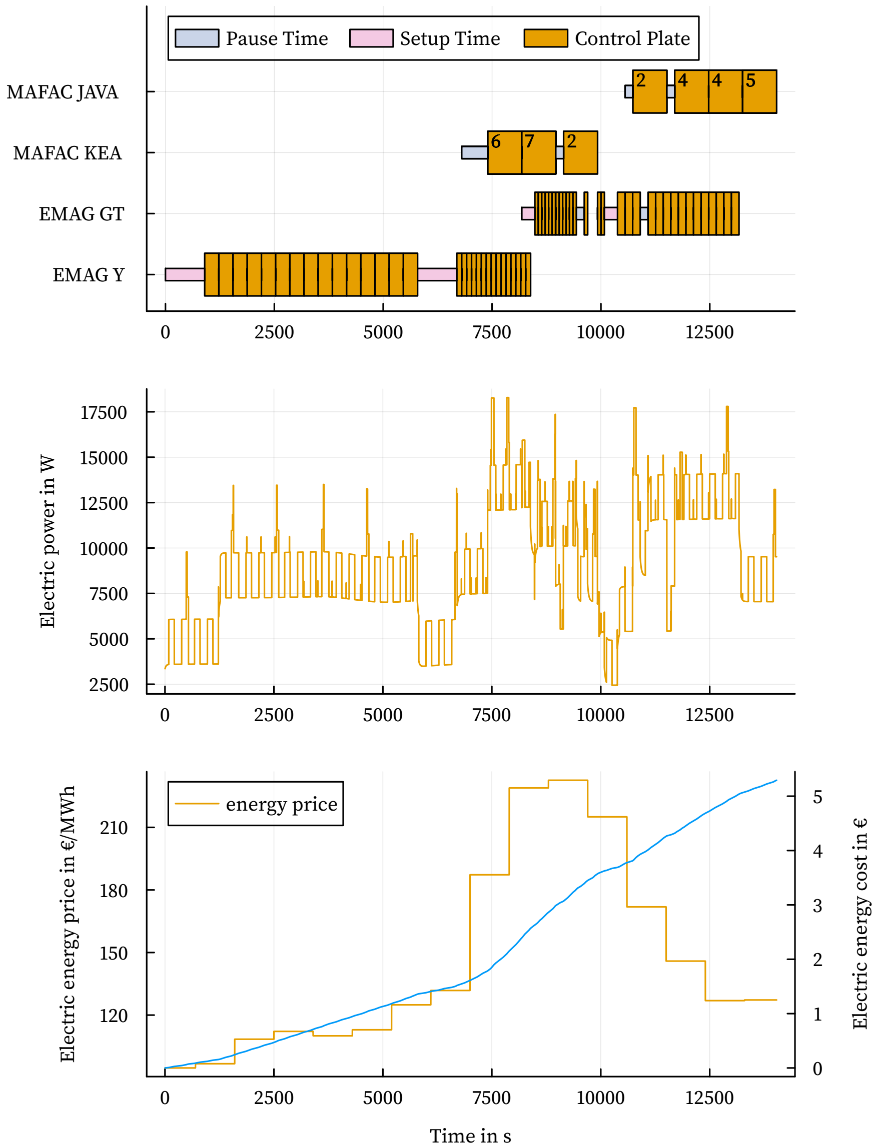


Figure 6.10: Solution with the best makespan from experiment 3.

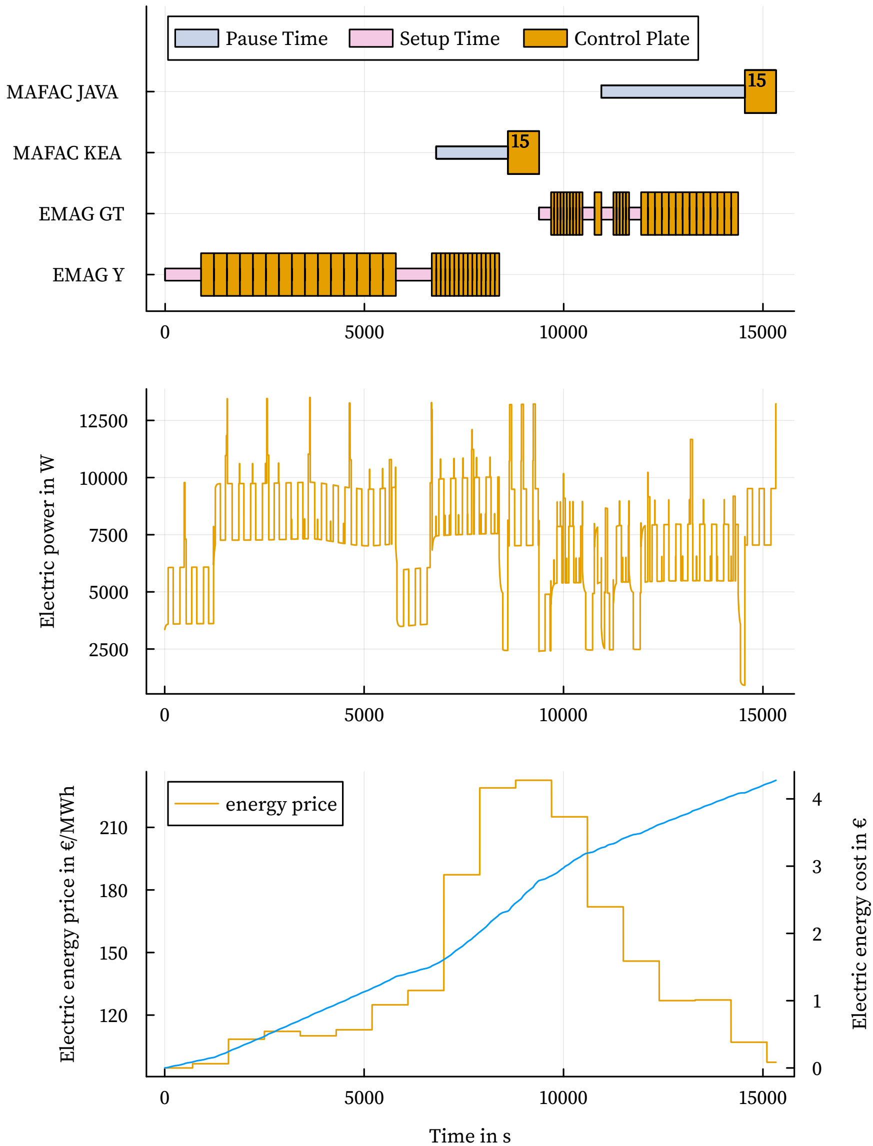


Figure 6.11: Solution with the best energy-related cost from experiment 3.

that the ETA Research Factory only produces a single product. The topmost plot also indicates the number of parts being processed at once for machines with a capacity greater than one. The figures' middle plot illustrates the factory's total electric power, and the bottom plot shows the energy price and cumulated energy cost. Appendix D contains additional plots for the other experiments and the schedules generated by the SPT rule.

Figures 6.10 and 6.11 also illustrate some differences between the solutions with the best makespan and with the best energy-related cost. It can be beneficial to be very eager when processing parts on machines with higher capacity to achieve the best makespan. For example, in Figure 6.10, the algorithm chooses to produce only six parts when it first uses the MAFAC KEA. This choice allows the EMAG VLC100 GT to begin processing the first parts slightly earlier than in Figure 6.11. The same is also true for the MAFAC JAVA, although in this case it does not serve a real purpose because no further operations follow it. In the best energy-related cost solution in Figure 6.11, the algorithm waits until the cleaning machines can process all parts at once. A similar comparison with the SPT rule (refer to Appendix D) is possible – it begins processing as soon as the first part becomes available and continuously utilizes all machines whenever feasible. However, it does not take advantage of the additional machine capacity.

The two figures and the additional figures in Appendix D highlight even more aspects. For example, the optimization algorithm has problems avoiding unnecessary setup times and sometimes schedules unnecessary pauses. This aspect becomes apparent when looking at the EMAG VLC100 GT in Figure 6.11, where the algorithm schedules operation OP 41 once while processing OP 40, leading to three unnecessary setup times. One reason for this might be the limitation to fixed pause durations. In the case mentioned above, a pause of 1800 s would likely have been too short to gather all workpieces for the MAFAC JAVA, while a pause of 3600 s is too long because it would leave the entire factory idle at the end of the day. More fine-grained control over the pause times could improve upon this; however, it might also lead to problems with convergence by increasing the number of possible solutions.

Check Final Quality Gate and Deploy System. The tests show that the production scheduling system performs well and achieves the goals set during the discovery and planning phase in Section 6.2.1. The goals were to exceed the performance of SPT dispatching rules, perform scheduling for an entire day within a maximum of 15 min, and to represent the ETA Research Factory production system with a single product and four machines while considering the electricity cost of production. Section 6.4 provides a more detailed discussion of how well the proposed concepts meet the requirements set in Section 4.1.

Continuous Support and Improvements. While this step is vital for actual industrial implementations, the research environment cannot sufficiently evaluate it. See Section 7.1 for possible improvements to the entire implementation procedure and energy-aware production scheduling system architecture that future research should analyze.

Results. The testing and deployment phase aims to check whether the proposed energy-aware production scheduling system suits the use case and whether the results conform to the requirements set during the discovery and planning phase. The requirements were that the system should be able to schedule for a day within a maximum of 15 min, which the system achieved by scheduling a day within approximately 11 min. The system should also be able to represent the ETA Research Factory production system while being able to extend to multiple products. The testing and deployment phase showed that the system could represent the ETA Research Factory reasonably well. The energy-aware production scheduling system is also extensible to multiple products. Finally, the system should consider the total energy-related cost of the schedules it creates. This consideration is included in the energy-aware production scheduling system, and it can improve energy-related cost while improving the makespan simultaneously, as shown in Table 6.14.

6.3. Transfer to another Production System

Validating whether the energy-aware production scheduling system can transfer to different production systems is fundamental while answering the research questions introduced in Section 1.1. The case study in the previous section validates the suitability of the energy-aware production scheduling system for a single use case. Additionally, a master's thesis supervised by the author of this work analyzed transferring the energy-aware production scheduling system to another production system [1]. The production system in question belongs to an industry partner in the KI4ETA project (Projektträger Jülich, 2021).

[1] Hirsch, Jonas. (2023). *Application-oriented analysis of the adaptivity of systems for energy-aware production scheduling* (Unpublished master's thesis). Technische Universität Darmstadt. Fachbereich Maschinenbau. Unfinished at the time of writing.

The master's student executed the implementation procedure to implement the energy-aware production scheduling system for the industry partner. Since this transfer is also a prototype, the student could not evaluate all steps of the implementation procedure; nevertheless, preliminary results showed a promising tendency that the implementation procedure and energy-aware production scheduling can be easily transferred to various production systems. The master's thesis was still incomplete when this thesis was finalized; thus, the final results were unavailable. The following is an account of preliminary observations gathered in meetings with the master's student and the industry partner.

During the implementation procedure it quickly became apparent that clarifying the project scope, goals, and requirements is critical once the implementation team is assembled. A clear project goal helps to manage the expectations of the implementation's stakeholders. The production system for this case study consists of four parallel production machines; any machine can produce all products. All machines operate solely with electric energy, but two production machines are more efficient than the others. Additionally, the production machines have multiple relevant energy states while processing workpieces.

The checks on whether the energy-aware production scheduling system could represent the production system showed that additional implementation work was required to correctly model the production system as well as the production machines. The production machines are similar to the existing `MachineTool` class but have additional energy states. The production system structure additionally required adaptations to the production scheduling model because the model previously did not support multiple parallel machines with equivalent capabilities.

Based on this information, the student could create software implementation plans and quickly progress with the implementation. Additionally, the student communicated with the industry partner to create the configuration concepts and data-gathering plans.

Executing the software implementation plans and creating a testing environment with simulated data went smoothly; however, an essential outcome of the development and implementation phase is that data-gathering in an industrial environment can be very tedious and time-consuming due to organizational barriers – the data-gathering was incomplete when this thesis was finalized.

Overall, the preliminary case study transferring the proposed implementation procedure and energy-aware production scheduling system architecture showed that the proposed concepts are well suited for an industrial use case. The discovery and planning phase emerged as an essential part of the implementation

Table 6.15: Fulfilment of functional requirements. Checkmarks in parentheses indicate partially fulfilled requirements.

Requirement	Fulfilment
Simplicity	✓
Robustness	(✓)
Controllability	✓
Adaptivity	✓
Completeness	✓
Interactivity	(✓)
Connectivity	(✓)

procedure to unify the understanding of goals and requirements as well as the production system itself. The plans produced during the discovery and planning phase supported the development and configuration phase well.

6.4. Evaluation and Discussion

The experiments and case studies in the preceding sections reveal that the proposed implementation procedure and energy-aware production scheduling system architecture generally fulfil their purpose of performing energy-aware production scheduling. This section checks all requirements and success criteria set in Section 4.1 to analyze how deeply the proposed concepts fulfill them.

The functional requirements form the basis for the proposed solution because they are generic requirements for decision-support systems, as Little (1970) proposed. They also enable deriving the more detailed technical requirements discussed later in this section. Table 6.15 summarizes the fulfilment of the functional requirements. The following paragraphs elaborate why each requirement is fully or partially fulfilled.

Simplicity. According to Little (1970), the system should only incorporate the required factors and exclude everything else. The proposed system includes the necessary elements for performing production scheduling. The system also provides flexibility regarding adding further energy models or changing the production scheduling model. The proposed concepts are kept as simple as possible, although the inherent complexity of the energy-aware job shop scheduling problem with rescheduling restrains simplicity. Overall, the proposed energy-aware production scheduling system fulfils this requirement well.

Robustness. The energy model parameter estimation should enable system users to configure the production machine energy models easily by avoiding configuration errors. The models should also be more accurate than the manual configuration of multiple energy states where users might obtusely derive values from the machine's rated power. Despite these advantages, the energy model parameter estimation also has the problem that the parameter estimation quality strongly depends on the available data. In case good data is unavailable, the parameter estimation might fail to find reasonable estimates for the parameters, in which case the implementation team needs the expertise to improve the energy model parameter estimation results. On the other hand, the robustness regarding production scheduling is quite high due to the multi-objective optimization, which empowers users to select the most satisfactory solution. Ultimately, the implementation procedure's robustness could be improved by enhancing robustness of the energy model parameter estimation.

Controllability. Little (1970) demands that changes made by the user should lead to the intended consequences. When executing production scheduling, the user usually changes the number of customer orders and the number of workpieces in storage. Both of these directly impact the objective criteria; the exact impact depends on the precise objective criteria chosen during the implementation procedure. Changes occurring due to rescheduling behave the same because these usually manifest as changes in the number of workpieces in storage. The proposed system and implementation procedure fulfil this requirement completely.

Adaptivity. To achieve adaptivity, Little (1970) requires that the models' parameters and structure should be updatable. The proposed energy-aware production scheduling system architecture realizes both aspects by providing configuration files read before every scheduling run. Therefore, the proposed energy-aware production scheduling system always uses the most current available data. The implementation of storage is another crucial part of adaptivity because it facilitates adapting to inaccurate timing of production jobs through rescheduling. With the implementation procedure and energy model parameter estimation, the proposed energy-aware production scheduling system architecture can adapt to various uses and production systems. The system is extensible such that new functionality can be added when required. The transfer to another production system in Section 6.3 delineates adding additional aspects to the production scheduling model and the production machine energy models, which is easily possible. Thus, the proposed concepts statisfy this requirement.

Completeness. This requirement asks for a compromise with the simplicity requirement. As detailed in the paragraph about the simplicity requirement, the proposed system includes the necessary components to successfully model and optimize the energy-aware job shop scheduling problem. The system also provides some extensibility to other scheduling problems through the implementation procedure and energy model parameter estimation. Regarding the production machine energy models, the *evaluate energy supply* step is essential to the implementation procedure. As the case study showed, the evaluation provides the implementation team with a better understanding of the requirements for the energy models. In that step the implementation team can also decide whether simplifications of the energy supply system are necessary to achieve the previously determined goals. Overall, the proposed energy-aware production scheduling system is flexible enough to fulfil various completeness requirements.

Interactivity. Since this is a research prototype, Section 4.1 intentionally left the requirement for interactivity loose; however, the configuration files, which cover the production system's structure as well as the production machine energy models and the product, order, and storage configurations allow for easy manipulation of the production scheduling aspects of the system. While this interactivity is helpful for users of the energy-aware production scheduling system after the implementation is complete, the implementation team does need substantial programming knowledge. There will often be cases when the implementation team identifies particularities of a production system that must be implemented in the software or when additional energy models are needed. The proposed system's energy model parameter estimation aspects also require manual scripting because the available data often needs some pre-processing to rename values, remove outliers, or calculate derived variables. Overall, the user-facing aspects of the energy-aware production scheduling system could be implemented in a graphical user interface suitable for managers without programming knowledge. However, the implementation procedure will likely continue to require some programming knowledge.

Connectivity. Providing connectivity to the production machines and external services is another aspect of implementing an energy-aware production scheduling system in industrial settings. While evaluating connectivity exceeded the

Table 6.16: Fulfilment of technical requirements. Checkmarks in parentheses indicate partially fulfilled requirements.

Requirement	Fulfilment
Collection of Requirements	√
Production System Configuration	√
Standardized Models	√
Automatic Model Parameter Estimation	√
Customer Order Input	√
Individualized Objective Criteria	√
Multi-Objective Optimization	√
Rescheduling	(√)
Production System Data Acquisition	(√)
Controlling Production Machines	(√)
Connectivity with External Services	(√)

scope of the initial case study in Section 6.2, the evaluations we published in Grosch, Fuhrländer-Völker, et al. (2022) and Fuhrländer-Völker et al. (2023) and additional experiments conducted in the ETA Research Factory showed that the proposed system in conjunction with the *eta_utility* framework could acquire data and control production machines. The connectors provided by the *eta_utility* framework also enabled data-gathering for the implementation procedure and energy model parameter estimation. However, since this requirement was not fully tested, it is considered partially fulfilled.

In summary, the proposed implementation procedure with energy model parameter estimation and the energy-aware production scheduling system architecture fulfil the functional requirements well. At the same time, areas for future enhancements include the robustness of the energy modelling process and interactivity during the implementation procedure and the usage phase. Further research into various levels of detail for the energy models and the connectivity aspects would also be valuable.

Section 4.1 derives technical requirements from the functional requirements to more precisely identify the most pertinent conditions that the technical implementation has to consider. Table 6.16 marks the degree of fulfilment for each technical requirement. Requirements with a checkmark in parentheses are not fulfilled entirely, or the case study did not wholly evaluate them.

The table shows that this thesis makes substantial strides toward fulfilling the technical requirements for energy-aware production scheduling. The formalized implementation procedure and its collection of requirements from stakeholders ensured the integration of their perspectives and needs into the system and unified their understanding. The energy-aware production scheduling system architecture's interfaces for configuring production systems improved adaptability to various environments, and the standardized interface for production machine models demonstrated the potential for scalability and applicability to different production systems.

In combination with the automatic model parameter estimation, the standardized models showed promise in reducing the effort required for model creation; however, the transfer to another production system also highlighted that energy models can be very specialized such that they do need to be implemented individually. The energy-aware production scheduling system can also read customer orders from a configuration file enabling users to generate production schedules quickly. The energy-aware production scheduling system also provides the ability to select individualized objective criteria during the implementation procedure contributing to user flexibility and ensuring that the specific needs of all stakeholders are addressed.

While the energy-aware production scheduling system's rescheduling capability, production system data acquisition, machine control, and connectivity with external services were not thoroughly tested, this thesis recognizes their importance and emphasizes the potential for future research in these directions. Once fully implemented, these aspects are crucial to allow the scheduling system to dynamically adapt to unexpected events, acquire real-time data for informed decision-making, automate production machine control, and incorporate external factors such as energy prices into the optimization process.

The success criteria defined in Section 4.1 should provide a quantitative evaluation of the functional and technical requirements as well as the general performance of the energy-aware production scheduling system. The first two success criteria focus on fulfilling the functional and technical requirements. The success criterion *accurate representation of the production system* ensures that the energy models represent the actual production machine's behaviour, and the two success criteria *improved energy-awareness of production scheduling* and *sufficient performance in production-related objectives* concentrate on the production schedule optimization. The last criterion gauges the transferability to a variety of production systems.

Functional and Technical Requirements Fulfilled. In combination, the proposed energy-aware production scheduling system architecture and implementation procedure meet most of the functional and technical requirements. Of the seven functional requirements, the case studies only could not evaluate the *connectivity* requirement thoroughly due to the vast scope of evaluating this requirement; however, we published some initial results showing the general capabilities of the *eta_utility* framework in the previously mentioned articles. Regarding the technical requirements, the proposed concepts also achieved a satisfying fulfillment rate, fully meeting seven of eleven requirements. Although some requirements were only partially met, it is worth noting that none of them were outright unmet. Like the incomplete functional requirement, the four technical requirements not exhaustively validated in this thesis are also related to connectivity.

Accurate Representation of the Production System. The production machine energy models achieved reasonable results reflecting the production machine's energy consumption; however, additional data collection from the production system could help to improve the prediction accuracy. In terms of the total energy consumption over the entire test data set, gathered during production on 21 December 2022, the models had absolute percentage errors for the prediction of electric energy between approximately 2 % – 15 %. However, the absolute percentage errors for thermal energy prediction over the entire day were significantly higher, between approximately 14 % – 24 % for the machine tools and at 64.5 % for the cleaning machine with a heat exchanger. The MAE and RMSE paint a similar picture – the RMSE additionally indicates that momentary peaks of the measured power have a much more significant impact on electric power predictions than thermal power predictions. The higher volatility of electric power compared to thermal power might be responsible for this impact.

Accurate representations of the production system's energy consumption are crucial to energy-aware production scheduling. The accuracy achieved by the proposed models is sufficient to validate the entire energy-aware production scheduling system architecture proposed in this thesis. However, future research should look into improving the accuracy of energy models for production scheduling in combination with energy model parameter estimation.

Improved Energy-Awareness and Sufficient Performance in Production-Related Objectives. The energy-aware production scheduling system also successfully reduced the energy consumption of production schedules without compromising production-related objectives. This performance signifies the scheduling system's effectiveness in improving energy-awareness while maintaining satisfactory production-related performance. The energy-aware production scheduling system contributes to more sustainable and cost-effective production by optimizing energy-related objective criteria in the production scheduling process.

Transferable Scheduling System Architecture. The case study further demonstrates the transferability of the scheduling system across various machines of the same type using the energy model parameter estimation component. The energy model's ability to capture the energy consumption of the two machine tools and two cleaning machines, respectively, using a single model for each type of machine, symbolizes this flexibility. The energy model parameter estimation enables this and allows the system to adapt to different production systems, albeit with some development work required if additional types of machines exist. The case study and the transfer to a second production system also showcase the implementation procedure's capability to accommodate and facilitate adaptations. Thus, the implementation procedure ensures the system's transferability and applicability to diverse manufacturing environments.

In conclusion, the energy-aware production scheduling system and implementation procedure exhibit a solid performance in fulfilling a substantial portion of the requirements established in Section 4.1. While there is room for improvement in some technical areas, especially regarding the evaluation of connectivity and rescheduling, the proposed concepts show promising results in accurately representing the production system, enhancing energy-awareness, and maintaining production-related performance. The energy-aware production scheduling system architecture and implementation procedure also improve the adaptivity to diverse production systems and consider stakeholder requirements during implementation to implement systems that support the user's goals. These achievements lay a foundation for future research into implementing energy-aware production scheduling in real industrial settings.

7. Summary

The motivation for this thesis, as discussed in Section 1, is recognizing climate change as a pressing global challenge, as emphasized by world leaders and reports from organizations like the Intergovernmental Panel on Climate Change (Intergovernmental Panel on Climate Change, 2018). Urgent action is required to reduce greenhouse gas emissions, mitigate the impacts of climate change, and limit global warming to 1.5 °C. As a large energy consumer, the industrial sector plays a crucial role in reducing emissions and advancing sustainable practices. Improving the integration of renewable energy into the power grid through demand-side integration is one avenue for achieving these goals (Bundesministerium für Umwelt, 2019, p. 90; Walther et al., 2022). Additionally, trends such as Industry 4.0 (Lasi et al., 2014) and smart manufacturing (Kang et al., 2016) offer opportunities for enhancing energy efficiency and sustainability within the industrial sector (Bunse et al., 2011; Mohamed et al., 2019).

Against this backdrop, this thesis focuses on developing an implementation procedure and an energy-aware production scheduling system architecture to optimize production schedules while concurrently considering production-related and energy-related objective criteria. By improving the integration of energy considerations into production scheduling, the proposed concepts aim to improve energy efficiency, utilize opportunities for demand-side integration and minimize the environmental impact of production. Under the assumption that actual industrial implementations of energy-aware production scheduling are still lacking, the research goal for this thesis is:

Research Goal

This thesis aims to simplify the implementation of energy-aware production scheduling systems in real production systems by manufacturing companies using generalized modelling and tooling.

To achieve this goal, this thesis addresses three key research questions, to determine why there is a lack of actual implementations and to identify an implementation procedure as well as an energy-aware production scheduling system architecture with energy model parameter estimation to support the implementation procedure.

1. “Can a lack of implementations be attributed to an absence of procedures and architectures for implementing energy-aware production scheduling

systems in job shops?”
2. “Can a standardized and partially automated implementation procedure for the adoption of energy-aware production scheduling systems be proposed such that an energy-aware production scheduling system can be more easily applied to real industrial use cases?”
3. “How should the architecture of an energy-aware production scheduling system be designed to support the implementation procedure, and which additional tooling is needed to reduce the implementation efforts?”

This research explicitly focuses on implementing energy-aware production scheduling for job shops in the discrete manufacturing industry. At the beginning of the research project, the literature review (refer to Section 3) aimed to answer the first research question. It revealed two significant research gaps in the field of energy-aware production scheduling. Firstly, there is a need for accurate representations of production systems that can tolerate deviations during the execution of production schedules because much of the existing research focuses on optimization models and algorithms instead of actual implementations. Secondly, since it is challenging to implement energy-aware production scheduling in actual industrial environments, evaluating the factors that may influence such implementations is necessary. To overcome this challenge, realistic research production systems like the ETA Research Factory can provide valuable insights. Studying realistic research production systems can help to assess the feasibility, performance, and potential obstacles to implementing energy-aware production scheduling systems in actual production environments. Overall, the literature review showed that some of the implementation shortfall in the industry can likely be attributed to the absence of standardized procedures and system architectures and a resulting knowledge gap in industrial companies.

Addressing the identified research gaps by answering the second and third research questions is crucial to disseminating energy-aware production scheduling systems. Accurate representations of production systems with rescheduling capabilities enable the system to adapt to dynamic conditions and deviations, ensuring the feasibility and efficiency of the production schedule. Additionally, evaluations in realistic research production systems provide valuable insights into the challenges and opportunities associated with implementing such systems in industrial settings. Through these contributions, this thesis seeks to provide practical solutions and insights that facilitate the adoption of energy-aware production scheduling.

This thesis proposes an implementation procedure and an energy-aware production scheduling system architecture. The implementation procedure proposed in Section 4.2 standardizes a structured approach to configuring and deploying the energy-aware production scheduling system, ensuring alignment with stakeholder requirements. The implementation procedure begins with a discovery and planning phase to aggregate goals and requirements and create plans for additional software implementations, data-gathering, and configuration of the energy-aware production scheduling system's components. The development and configuration phase executes these plans by gathering data from and about the production system, creating software implementations and generating configuration files. Finally, the testing and deployment phase checks if the completed implementation fulfils the goals established during the development and configuration phase. The testing and deployment phase also confirms that the system performs as expected, provides user training and validates whether the system works for real users and orders.

The energy-aware production scheduling system architecture discussed in Section 5 implements a cyber-physical production system with a virtual representation of the actual production system, as illustrated in Figure 5.5. The system architecture is based on the *eta_utility* framework introduced by Grosch, Ranzau, et al. (2022). It incorporates the NSGA-II optimization algorithm, enabling multi-objective optimization and individualized objective criteria. The algorithm uses a graph-based encoding for the production scheduling solutions, which helps quickly evaluate solutions and avoids infeasible solutions. The second major component of the energy-aware production scheduling system architecture is the production system environment. The production system environment is at the scheduling system's core because it is the virtual representation of the actual production system. The production system environment is an adaptive component which can be adjusted to the specific requirements of a particular production system using software implementation (e.g., for objective criteria) and configuration files (e.g., for the production system structure). As a final component, the energy model parameter estimation supports the implementation procedure by simplifying the configuration process through automatic parameter estimation for the production machine energy models. The proposed energy models, in combination with the energy model parameter estimation, accurately represent the energy consumption characteristics of production processes. The architecture is designed to be adaptive and scalable, ensuring its applicability to various industrial contexts and production environments.

This thesis evaluates the proposed implementation procedure, energy-aware production scheduling system architecture and energy model parameter estimation in Section 6. The ETA Research Factory serves as a use case for the energy-aware production scheduling system deployment. The proposed concepts' evaluation focuses on assessing the fulfillment of the requirements and success criteria outlined in Section 4.1. During the case study in the ETA Research Factory, the energy-aware production scheduling system was successfully tested in a real-world production system. The case study indicates that the functional requirements are generally fulfilled, with the system demonstrating simplicity, robustness, controllability, adaptivity, and completeness. Areas for improvement include the robustness of energy modelling and enhancing interactivity for non-technical users. A detailed evaluation of connectivity exceeds the scope of this work, although preliminary studies (see: Grosch, Fuhrländer-Völker, et al., 2022) showed promising results.

The production machine energy models also captured reasonably accurate representations of the production machines, enabling the energy-aware production scheduling system to reduce energy consumption while maintaining production-related objective criteria. The system is compared to the SPT dispatching rules as a traditional production scheduling approach that does not consider energy constraints to evaluate the energy efficiency optimization, and the results demonstrated by the energy-aware production scheduling system show energy cost savings of 13 % on average, while slightly improving the production-related objectives as well. When accepting slightly worse performance in the production-related objective, the energy-aware production scheduling system achieves energy cost savings of 18 % on average while decreasing the production-related performance by 5 % over all experiments. A preliminary evaluation of transferring the proposed concepts to another production system additionally shows that the implementation procedure and energy-aware production scheduling system architecture facilitate implementations in various production systems with diverse structures.

In summary, this thesis contributes to the field of energy-aware production scheduling by developing an implementation procedure and a system architecture that fulfill most of the identified requirements and demonstrate satisfactory performance in all success criteria. The proposed implementation procedure is a structured approach to implementing energy-aware production scheduling in a variety of production systems. Combined with the energy-aware production scheduling system architecture, it forms a framework for disseminating energy-aware production scheduling in industrial applications. The energy-aware production scheduling system architecture also provides a basis for further research. This thesis concludes that the proposed concepts significantly contribute to the field, promoting environmentally conscious and economically viable production practices.

7.1. Outlook

The implementation procedure and energy-aware production scheduling system architecture proposed in this thesis form a starting point for filling an essential research gap regarding the dissemination of energy-aware production scheduling in real industrial applications; however, there are also many remaining areas for future research to address limitations and leverage potentials for further improvement.

There are edge cases where the solution encoding and decoding strategies do not perform as well as expected. For example, when multiple operations of a job have to be performed by the same machine, the solution encoding leads to many infeasible solutions. Thus, one area of focus should be addressing edge cases where the optimization algorithm may break down. Further investigation and refinement of the optimization algorithm and solution encoding could improve its robustness and ensure reliable performance across various scenarios.

Enhancing the robustness of the energy model parameter estimation process is another important aspect for future research. This could involve refining the energy models and modelling techniques to handle uncertainties, variations, and complex energy consumption patterns more effectively. By improving the production machine energy models' accuracy and reliability, the system could provide even more precise optimization outcomes. Regarding energy models, exploring various levels of detail and considering the interconnections between production systems and technical building services would also be valuable. Considering these interconnections also involves incorporating more granular models of technical building services and considering aspects like thermal energy conversion.

Regarding the production scheduling model, integrating product order due dates could be an appealing area for future research. By incorporating due date constraints, the system can ensure the timely completion of customer orders, particularly when scheduling over longer timeframes. Furthermore, additional information could be incorporated into the scheduling model to provide a more comprehensive description of the production process. Such information includes considering factors beyond processing and setup times, such as complex dependencies between production steps and the inclusion of additional input resources.

Moreover, adopting capability-based resource assignment mechanisms in reconfigurable production systems could enhance scheduling efficiency and energy utilization (D'Addona & Teti, 2019). This approach would enable assigning jobs to different resources with similar capabilities, potentially leading to higher energy efficiency by rescheduling to underutilized, more efficient resources.

Integrating energy price predictions or energy trading mechanisms could contribute to evaluating more realistic scenarios. By incorporating market dynamics and economic factors, the system can optimize production schedules considering energy consumption, cost-efficiency, and financial implications. These factors are vital to bringing energy-aware production scheduling closer to actual implementations considering demand response in addition to energy efficiency during production scheduling. In combination with pricing schemes, direct evaluation of carbon emissions could provide further insights into the environmental impact of production scheduling decisions facilitating greener decision-making.

Improving interactivity during the implementation procedure and the usage phase of the system is also an area for future enhancements. Research in this direction could involve analyzing stakeholder preferences and developing user-friendly interfaces, providing more intuitive visualization and decision-support tools. Research should also be done to analyze how users of a production scheduling system select the production schedule they implement and whether automation, visualizations or decision-support tools could be proposed to improve users' understanding of the trade-offs between energy consumption, production cost and timely completion of production jobs.

Since this thesis only performed an initial evaluation of the case study, some of the implementation procedure steps involving user training and deploying the system while actually controlling production machines require further evaluation. Future research should focus on assessing these steps in real-world industrial settings and identifying any challenges or improvements needed for a successful implementation.

In conclusion, future research should address the identified gaps and further advance the energy-aware production scheduling system. By addressing challenges related to optimization algorithms, energy modelling, interactivity, and real-world implementation and by incorporating additional factors and dependencies into the models, the system can continue to evolve and contribute to sustainable and efficient industrial production.

References

Abele, E., Beck, M., Flum, D., Schraml, P., Panten, N., Junge, F., Bauerdick, C., Helfert, M., Sielaff, T., Daume, C., Weber, M., Strobel, N., Redelberger, A., Pototzky, L., Rothenbücher, S., Landgraf, G., Rummel, W., Heimbach, K., Haase, P., ... Krönauer, A. (2019). *Gemeinsamer Schlussbericht zum Projekt ETA-Fabrik: Energieeffiziente Fabrik für interdisziplinäre Technologie- und Anwendungsforschung*. Technische Universität Darmstadt. https://doi.org/10.2314/KXP:1667471384

Abele, E., Schneider, J., Beck, M., & Maier, A. (2018). *ETA - die Modell-Fabrik: Energieeffizienz weiter gedacht*. Technische Universität Darmstadt. https://www.ptw.tu-darmstadt.de/media/fachgebietptw/dokumente_3/wissenssammlung_ptw/buecher_und_broschueren_1/ETA_Broschuere_en.pdf

Aizpurua, J. I., Stewart, B. G., McArthur, S. D. J., Jajware, N., Kearns, M., Garro, U., Muxika, E., & Mendicute, M. (2021). A Diagnostics Framework for Underground Power Cables Lifetime Estimation Under Uncertainty. *IEEE Transactions on Power Delivery, 36*(4), 2014–2024. https://doi.org/10.1109/TPWRD.2020.3017951

Alotaibi, A., Lohse, N., & Vu, T. M. (2016). Dynamic Agent-based Bi-objective Robustness for Tardiness and Energy in a Dynamic Flexible Job Shop. *Procedia CIRP, 57*, 728–733. https://doi.org/10.1016/j.procir.2016.11.126

Alvarez-Meaza, I., Zarrabeitia-Bilbao, E., Rio-Belver, R.-M., & Garechana-Anacabe, G. (2021). Green scheduling to achieve green manufacturing: Pursuing a research agenda by mapping science. *Technology in Society, 67*, 101758. https://doi.org/10.1016/j.techsoc.2021.101758

Arbeitsgemeinschaft Energiebilanzen e.V. (2020). *Anwendungsbilanzen zur Energiebilanz Deutschland: Endenergieverbrauch nach Energieträgern und Anwendungszwecken*.

Ayyoubzadeh, B., Ebrahimnejad, S., Bashiri, M., Baradaran, V., & Hosseini, S. M. H. (2021). Modelling and an improved NSGA-II algorithm for sustainable manufacturing systems with energy conservation under environmental uncertainties: a case study. *International Journal of Sustainable Engineering, 14*(3), 255–279. https://doi.org/10.1080/19397038.2021.1923083

Back, M.-G., Lee, D.-K., Shin, J.-G., & Woo, J.-H. (2016). A study for production simulation model generation system based on data model at a shipyard. *International Journal of Naval Architecture and Ocean Engineering, 8*(5), 496–510. https://doi.org/10.1016/j.ijnaoe.2016.05.005

Bai, F., Shen, T., Yu, Z., Zeng, K., & Gong, B. (2022). Trustworthy Blockchain-Empowered Collaborative Edge Computing-as-a-Service Scheduling and Data Sharing in the IIoE. *IEEE Internet of Things Journal, 9*(16), 14752–14766. https://doi.org/10.1109/JIOT.2021.3058125

Bänsch, K., Busse, J., Meisel, F., Rieck, J., Scholz, S., Volling, T., & Wichmann, M. G. (2021). Energy-aware decision support models in production environments: A systematic literature review. *Computers and Industrial Engineering, 159*, 107456. https://doi.org/10.1016/J.CIE.2021.107456

Bayerisches Landesamt für Umwelt. (2006). *Energieeinsparung in Lackierbetrieben - Langfassung: Klima schützen - Kosten senken*. Bayerisches Landesamt für Umwelt. https://www.bestellen.bayern.de/application/eshop_app000003?SID=1390523755&

Beeson, K. (2022). *ERP implementation plan*. Retrieved March 24, 2023, from https://www.erpfocus.com/erp-implementation-plan.html

Benndorf, R., Bernicke, M., Bertram, A., Butz, W., Dettling, F., Drotleff, J., Elsner, C., Fee, E., Gabler, C., Galander, C., Hargita, Y., Herbener, R., Hermann, T., Jäger, F., Kanthak, J., Kessler, H., Koch, Y., Kuntze, D., Lambrecht, M., ... Zietlow, B. (2014). *Treibhausgasneutrales Deutschland im Jahr 2050*. Umweltbundesamt. https://www.umweltbundesamt.de/publikationen/treibhausgasneutrales-deutschland-im-jahr-2050-0

Berti, N., Finco, S., Battaïa, O., & Delorme, X. (2021). Ageing workforce effects in Dual-Resource Constrained job-shop scheduling. *International Journal of Production Economics, 237*, 108151. https://doi.org/10.1016/j.ijpe.2021.108151

Bertsch, J., Schweter, H., Sitzmann, A., Buhl, H. U., Fridgen, G., Körner, M.-F., Michaelis, A., Rägo, V., Sachs, T., Schöpf, M., & Schott, P. (2019). *Ausgangsbedingungen für die Vermarktung von Nachfrageflexibilität* (2nd edition). Energiewirtschaftliches Institut an der Universität zu Köln. https://www.ewi.uni-koeln.de/de/publikationen/ausgangsbedingungen-fuer-die-vermarktung-von-nachfrageflexibilitaet-status-quo-analyse-und-metastudie-2/

Biel, K., & Glock, C. H. (2016). Systematic literature review of decision support models for energy-efficient production planning. *Computers and Industrial Engineering, 101*, 243–259. https://doi.org/10.1016/j.cie.2016.08.021

Blesl, M., & Kessler, A. (2021). *Energy Efficiency in Industry*. Berlin, Heidelberg, Germany, Springer. https://doi.org/10.1007/978-3-662-63923-8

Blessing, L. T., & Chakrabarti, A. (2009). *DRM, a Design Research Methodology*. Springer London. https://doi.org/10.1007/978-1-84882-587-1

Bokah, A., & Maheri, A. (2021). Multiobjective Optimisation of Job Shop Scheduling of Renewable Powered Machinery (R. A. Angelova & R. Velichkova, Eds.). In R. A. Angelova & R. Velichkova (Eds.), *Proceedings of the 2021 6th International Symposium on Environment Friendly Energies and Applications (EFEA)*, IEEE. https://doi.org/10.1109/EFEA49713.2021.9406238

Boyes, H., Hallaq, B., Cunningham, J., & Watson, T. (2018). The industrial internet of things (IIoT): An analysis framework. *Computers in Industry, 101*, 1–12. https://doi.org/10.1016/J.COMPIND.2018.04.015

Brauner, P., Dalibor, M., Jarke, M., Kunze, I., Koren, I., Lakemeyer, G., Liebenberg, M., Michael, J., Pennekamp, J., Quix, C., Rumpe, B., van der Aalst, W., Wehrle, K., Wortmann, A., & Ziefle, M. (2022). A Computer Science Perspective on Digital

Transformation in Production. *ACM Transactions on Internet of Things, 3*(2), 15:1–15:32. https://doi.org/10.1145/3502265

Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., & Zaremba, W. (2016). *OpenAI Gym*. arXiv. https://doi.org/10.48550/arXiv.1606.01540

Bundesministerium für Umwelt. (2016). *Klimaschutzplan 2050 - Klimaschutzpolitische Grundsätze und Ziele der Bundesregierung*. Bundesministerium für Umwelt, Naturschutz, Bau und Reaktorsicherheit. https://www.umweltbundesamt.de/publikationen/klimaschutzplan-2050-der-bundesregierung

Bundesministerium für Umwelt. (2019). *Klimaschutzprogramm 2030 der Bundesregierung zur Umsetzung des Klimaschutzplans 2050*. Bundesministerium für Umwelt, Naturschutz, Bau und Reaktorsicherheit. https://www.bundesregierung.de/breg-de/schwerpunkte/klimaschutz/massnahmenprogramm-klima-1679498

Bundesnetzagentur. (2022). *Bericht zum Redispatch nach Artikel 13 Verordnung (EU) 2019/943. Entwicklungsstand, Maßnahmen und nächste Schritte*. Bundesnetzagentur für Elektrizität, Gas, Telekommunikation, Post und Eisenbahnen. https://www.bundesnetzagentur.de/DE/Fachthemen/ElektrizitaetundGas/Versorgungssicherheit/Netzengpassmanagement/start.html

Bunse, K., Vodicka, M., Schönsleben, P., Brülhart, M., & Ernst, F. O. (2011). Integrating energy efficiency performance in production management – gap analysis between industrial needs and scientific literature. *Journal of Cleaner Production, 19*(6-7), 667–679. https://doi.org/10.1016/j.jclepro.2010.11.011

Cao, W., Jiang, P., Liu, B., & Jiang, K. (2018). Real-time order scheduling and execution monitoring in public warehouses based on radio frequency identification. *International Journal of Advanced Manufacturing Technology, 95*(5-8), 2473–2494. https://doi.org/10.1007/s00170-017-1381-z

Cassandras, C. G., & Lafortune, S. (2008). *Introduction to Discrete Event Systems* (2nd edition). Springer New York. https://doi.org/10.1007/978-0-387-68612-7

Cheng, Q., Liu, C., Chu, H., Liu, Z., Zhang, W., & Pan, J. (2020). A New Multi-Objective Hybrid Flow Shop Scheduling Method to Fully Utilize the Residual Forging Heat. *IEEE Access, 8*, 151180–151194. https://doi.org/10.1109/ACCESS.2020.3017239

Coca, G., Castrillon, O. D., Ruiz, S., Mateo-Sanz, J. M., & Jimenez, L. (2019). Sustainable evaluation of environmental and occupational risks scheduling flexible job shop manufacturing systems. *Journal of Cleaner Production, 209*, 146–168. https://doi.org/10.1016/j.jclepro.2018.10.193

Cramton, P. (2017). Electricity market design. *Oxford Review of Economic Policy, 33*(4), 589–612. https://doi.org/10.1093/oxrep/grx041

D'Addona, D. M., & Teti, R. (2019). Planning. In S. Chatti, L. Laperrière, G. Reinhart, & T. Tolio (Eds.), *CIRP Encyclopedia of Production Engineering*. Springer Berlin, Heidelberg. https://doi.org/10.1007/978-3-662-53120-4_6566

Dai, M., Ji, Z., & Wang, Y. (2018). Energy-aware integrated optimization of process planning and scheduling considering transportation. *Modern Physics Letters B*, *32*(34n36), 1840114. https://doi.org/10.1142/S0217984918401140

Dai, M., Zhang, Z., Giret, A., & Salido, M. A. (2019). An Enhanced Estimation of Distribution Algorithm for Energy-Efficient Job-Shop Scheduling Problems with Transportation Constraints. *Sustainability*, *11*(11), 3085. https://doi.org/10.3390/su11113085

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, *6*(2), 182–197. https://doi.org/10.1109/4235.996017

Denkena, B., Abele, E., Brecher, C., Dittrich, M.-A., Kara, S., & Mori, M. (2020). Energy efficient machine tools. *CIRP Annals - Manufacturing Technology*, *69*(2), 646–667. https://doi.org/10.1016/j.cirp.2020.05.008

Denkena, B., Schinkel, F., Pirnay, J., & Wilmsmeier, S. (2021). Quantum algorithms for process parallel flexible job shop scheduling. *CIRP Annals - Manufacturing Technology*, *33*, 100–114. https://doi.org/10.1016/j.cirpj.2021.03.006

Deutsches Institut für Normung e. V. (2018). *Energiemanagementsysteme - Messung und Verifizierung der energiebezogenen Leistung von Organisationen: Allgemeine Grundsätze und Leitlinien* (DIN ISO 50015). Beuth Verlag. https://www.beuth.de/de/norm/din-iso-50015/284893520

Deutsches Institut für Normung e.V. (2016). *Referenzarchitekturmodell Industrie 4.0 (RAMI4.0)* (DIN SPEC 91345). Beuth Verlag. https://www.beuth.de/de/technische-regel/din-spec-91345/250940128

Dickmann, P. (2015). *Schlanker Materialfluss: mit Lean Production, Kanban und Innovationen*. Springer Berlin, Heidelberg. https://doi.org/10.1007/978-3-662-44869-4

Dos Santos, S. A. B., Soares, J. M., Barroso, G. C., & de Athayde Prata, B. (2023). Demand response application in industrial scenarios: A systematic mapping of practical implementation. *Expert Systems with Applications*, *215*, 119393. https://doi.org/10.1016/j.eswa.2022.119393

Duflou, J. R., Sutherland, J. W., Dornfeld, D., Herrmann, C., Jeswiet, J., Kara, S., Hauschild, M., & Kellens, K. (2012). Towards energy and resource efficient manufacturing: A processes and systems approach. *CIRP Annals - Manufacturing Technology*, *61*(2), 587–609. https://doi.org/10.1016/j.cirp.2012.05.002

Elahi, B. (2021). Manufacturing Plant Layout Improvement: Case study of a High-Temperature Heat Treatment Tooling Manufacturer in Northeast Indiana. *Procedia Manufacturing*, *53*, 24–31. https://doi.org/10.1016/j.promfg.2021.06.006

Elliott, R. T., Fernández-Blanco, R., Kozdras, K., Kaplan, J., Lockyear, B., Zyskowski, J., & Kirschen, D. S. (2019). Sharing Energy Storage Between Transmission and Distribution. *IEEE Transactions on Power Systems*, *34*(1), 152–162. https://doi.org/10.1109/TPWRS.2018.2866420

Fairbanks, J., Besan, M., Simon, S., Hoffiman, J., Eubank, N., & Karpinski, S. (2021). *JuliaGraphs/Graphs.jl: An optimized graphs package for the Julia programming language*. GitHub. https://github.com/JuliaGraphs/Graphs.jl/

Feng, Y., Hong, Z., Li, Z., Zheng, H., & Tan, J. (2020). Integrated intelligent green scheduling of sustainable flexible workshop with edge computing considering uncertain machine state. *Journal of Cleaner Production, 246*, 119070. https://doi.org/10.1016/j.jclepro.2019.119070

Fuhrländer-Völker, D., Grosch, B., & Weigold, M. (2023). Modelling and Control of Aqueous Parts Cleaning Machines for Demand Response. In D. Herberger, M. Hübner, & V. Stich (Eds.), *Proceedings of the Conference on Production Systems and Logistics* (pp. 790–800). publish-Ing. https://doi.org/10.15488/13498

Gahm, C., Denz, F., Dirr, M., & Tuma, A. (2016). Energy-efficient scheduling in manufacturing companies: A review and research framework. *Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems, 248*(3), 744–757. https://doi.org/10.1016/j.ejor.2015.07.017

Gao, K., Huang, Y., Sadollah, A., & Wang, L. (2020). A review of energy-efficient scheduling in intelligent production systems. *Complex & Intelligent Systems, 6*(2), 237–249. https://doi.org/10.1007/s40747-019-00122-6

Garetti, M., & Taisch, M. (2012). Sustainable manufacturing: Trends and research challenges. *Production Planning and Control, 23*(2-3), 83–104. https://doi.org/10.1080/09537287.2011.591619

Garwood, T. L., Hughes, B. R., Oates, M. R., O'Connor, D., & Hughes, R. (2018). A review of energy simulation tools for the manufacturing sector. *Renewable and Sustainable Energy Reviews, 81*, 895–911. https://doi.org/10.1016/j.rser.2017.08.063

Gayko, J., Leboucher, Y., Wei, S., Wang, C., Hankel, M., Schewe, F., & Hoffmeister, M. (2018). *Alignment Report for Reference Architectural Model for Industrie 4.0 / Intelligent Manufacturing System Architecture*. Bundesministerium für Wirtschaft und Energie. https://www.plattform-i40.de/IP/Redaktion/DE/Downloads/Publikation/hm-2018-manufactoring.html?nn=1105090

Giret, A., Trentesaux, D., & Prabhu, V. (2015). Sustainability in manufacturing operations scheduling: A state of the art review. *Journal of Manufacturing Systems, 37*, 126–140. https://doi.org/10.1016/j.jmsy.2015.08.002

Grabner, T. (2017). *Operations Management*. Springer Gabler Wiesbaden. https://doi.org/10.1007/978-3-658-14484-5

Groombridge, D. (2021). *Top Strategic Technology Trends for 2022*. Gartner. https://www.gartner.com/en/documents/4006913

Groover, M. P. (2008). *Automation, production systems, and computer-integrated manufacturing: Pearson International Edition*. Prentice Hall.

Grosch, B., Fuhrländer-Völker, D., Stock, J., & Weigold, M. (2022). Cyber-physical production system for energy-flexible control of production machines. *Procedia CIRP, 107*, 221–226. https://doi.org/10.1016/j.procir.2022.04.037

Grosch, B., Kohne, T., & Weigold, M. (2021). Multi-objective hybrid genetic algorithm for energy adaptive production scheduling in job shops. *Procedia CIRP*, *98*, 294–299. https://doi.org/10.1016/j.procir.2021.01.106

Grosch, B., Ranzau, H., Dietrich, B., Kohne, T., Fuhrländer-Völker, D., Sossenheimer, J., Lindner, M., & Weigold, M. (2022). A framework for researching energy optimization of factory operations. *Energy Informatics*, *5*, 15–16. https://doi.org/10.1186/s42162-022-00207-6

Grosch, B., Weitzel, T., Panten, N., & Abele, E. (2019). A metaheuristic for energy adaptive production scheduling with multiple energy carriers and its implementation in a real production system. *Procedia CIRP*, *80*, 203–208. https://doi.org/10.1016/j.procir.2019.01.043

Guzman, E., Andres, B., & Poler, R. (2021). Models and algorithms for production planning, scheduling and sequencing problems: A holistic framework and a systematic review. *Journal of Industrial Information Integration*, *2*(27), 100287. https://doi.org/10.1016/j.jii.2021.100287

Haapala, K. R., Zhao, F., Camelio, J., Sutherland, J. W., Skerlos, S. J., Dornfeld, D. A., Jawahir, I. S., Zhang, H. C., & Clarens, A. F. (2011). A Review of Engineering Research in Sustainable Manufacturing. *ASME 2011 International Manufacturing Science and Engineering Conference, Volume 2*, *135*(4), 599–619. https://doi.org/10.1115/MSEC2011-50300

Hax, A. C., & Meal, H. C. (1973). *Hierarchical Integration of Production Planning and Scheduling*. Sloan School of Management, Massachusetts Institute of Technology. http://hdl.handle.net/1721.1/1868

He, Y., & Smith, M. (2021). FMS Scheduling Integration for Mass Customization (A. Dolgui, A. Bernard, D. Lemoine, G. von Cieminski, & D. Romero, Eds.). In A. Dolgui, A. Bernard, D. Lemoine, G. von Cieminski, & D. Romero (Eds.), *Advances in Production Management Systems. Artificial Intelligence for Sustainable and Resilient Production Systems*, Springer, Cham. https://doi.org/10.1007/978-3-030-85902-2_54

Hesselbach, J. (2012). *Energie- und klimaeffiziente Produktion*. Vieweg+Teubner Verlag. https://doi.org/10.1007/978-3-8348-9956-9

Hochrein, S., & Glock, C. H. (2012). Systematic literature reviews in purchasing and supply management research: A tertiary study. *International Journal of Integrated Supply Management*, *7*(4), 215–245. https://doi.org/10.1504/IJISM.2012.052773

Hornberg, C., Niekisch, M., Calliess, C., Kemfert, C., Lucht, W., Messari-Becker, L., & Rotter, V. S. (2020). *Für eine entschlossene Umweltpolitik in Deutschland und Europa: Umweltgutachten 2020*. Sachverständigenrat für Umweltfragen. https://www.umweltrat.de/SharedDocs/Downloads/DE/01_Umweltgutachten/2016_2020/2020_Umweltgutachten_Entschlossene_Umweltpolitik.html

Huang, J., Yu, B., Xing, C.-C., Cerny, T., & Ning, Z. (2021). Online Energy Scheduling Policies in Energy Harvesting Enabled D2D Communications. *IEEE Transactions on Industrial Informatics, 17*(8), 5678–5687. https://doi.org/10.1109/TII.2020.3005440

IBM Corporation. (2022). *ILOG CPLEX Optimization Studio: User's Manual for CPLEX* (Version 22.1.0). https://www.ibm.com/docs/en/icos/22.1.0?topic=optimizers-users-manual-cplex

Ibrahim, M., Harb, H., Nasser, A., Mansour, A., & Osswald, C. (2022). Aggregation-Scheduling Based Mechanism for Energy-Efficient Multivariate Sensor Networks. *IEEE SENSORS JOURNAL, 22*(16), 16662–16672. https://doi.org/10.1109/JSEN.2022.3189431

Intergovernmental Panel on Climate Change. (2018). *Global Warming of 1.5 °C. An IPCC Special Report on the impacts of global warming of 1.5 °C above pre-industrial levels and related global greenhouse gas emission pathways, in the context of strengethening the global response to the threat of climate change, sustainable development, and efforts to eradicate poverty*. Masson-Delmotte, Valérie; Zhai, Panmao; Pörtner, Hans-Otto; Roberts, Debra; Skea, Jim; Shukla, Priyadarshi R.; Pirani, Anna; Moufouma-Okia, Wilfran; Péan, Clotilde; Pidcock, Roz; Connors, Sarah; Matthews, J. RobinB.; Chen, Yang; Zhou, Xiao; Gomis, Melissa I.; Lonnoy, Elisabeth; Maycock, Tom; Tignor, Melinda; Waterfield, Tim. https://doi.org/10.1017/9781009157940

International Energy Agency. (2020). *Key World Energy Statistics 2020*. https://www.iea.org/reports/key-world-energy-statistics-2020

International Monetary Fund. (2021). *World Economic Outlook: Managing Divergent Recoveries*. https://www.imf.org/en/Publications/WEO/Issues/2021/03/23/world-economic-outlook-april-2021

International Organization for Standardization. (2017). *Machine tools - Environmental evaluation of machine tools: Part 1: Design methodology for energy efficient machine tools* (ISO 14955-1). https://www.iso.org/standard/70035.html

International Organization for Standardization. (2021). *Internet of things (IoT) - Vocabulary* (ISO IEC 20924). https://www.iso.org/standard/82771.html

Jiang, X., Tian, Z., Liu, W., Suo, Y., Chen, K., Xu, X., & Li, Z. (2022). Energy-efficient scheduling of flexible job shops with complex processes: A case study for the aerospace industry complex components in China. *Journal of Industrial Information Integration, 27*, 100293. https://doi.org/10.1016/j.jii.2021.100293

Jodlbauer, H. (2008). *Produktionsoptimierung: Wertschaffende sowie kundenorientierte Planung und Steuerung*. Springer Vienna. https://doi.org/10.1007/978-3-211-78141-8

Joshi, R., Didier, P., Holmberg, C., Jimenez, J., & Carey, T. (2022). *The Industrial Internet of Things Connectivity Framework*. Industry IoT Consortium. https://www.iiconsortium.org/IICF/

Kang, H. S., Lee, J. Y., Choi, S., Kim, H., Park, J. H., Son, J. Y., Kim, B. H., & Noh, S. D. (2016). Smart Manufacturing: Past Research, Present Findings, and Future Directions. *International Journal of Precision Engineering and Manufacturing, 3*(1), 111–128. https://doi.org/10.1007/s40684-016-0015-5

Kobiela, G., Samadi, S., Kurwan, J., Tönjes, A., Fischedick, M., Koska, T., Lechtenböhmer, S., März, S., & Schüwer, D. (2020). *CO2-neutral bis 2035: Eckpunkte eines deutschen Beitrags zur Einhaltung der 1,5-°C-Grenze*. Wuppertal Institut. https://epub.wupperinst.org/frontdoor/index/index/docId/7606

Kowalski, M., Steinboeck, A., & Kugi, A. (2022). Scheduling Multiple Groups of Jobs for a Multi-Line Steel Hot Rolling Mill. *IFAC-PapersOnLine, 55*(21), 168–173. https://doi.org/10.1016/j.ifacol.2022.09.262

Kropik, M. (2009). *Produktionsleitsysteme in der Automobilfertigung*. Springer Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-88991-5

Kuhrke, B. (2011). *Methode zur Energie- und Medienbedarfsbewertung spanender Werkzeugmaschinen* (Doctoral dissertation). Technische Universität Darmstadt. Fachbereich Maschinenbau. https://tubiblio.ulb.tu-darmstadt.de/55900/

Lampropoulos, I., Kling, W. L., Ribeiro, P. F., & van den Berg, J. (2013). History of demand side management and classification of demand response control schemes, In *IEEE Power and Energy Society General Meeting*, IEEE. https://doi.org/10.1109/PESMG.2013.6672715

Lasi, H., Kemper, H.-G., Fettke, P., Feld, T., & Hoffmann, M. (2014). Industry 4.0. *Business & Information Systems Engineering*, (4), 239–242. https://doi.org/10.1007/s12599-014-0334-4

Lauth, J., & Kowalczyk, J. (2022). *Thermodynamik: Eine Einführung*. Springer Spektrum Berlin, Heidelberg. https://doi.org/10.1007/978-3-662-64707-3

Lewrick, M., Link, P., Leifer, L., & Langensand, N. (2018). *Das Design Thinking Playbook: Mit traditionellen, aktuellen und zukünftigen Erfolgsfaktoren* (2nd edition). Verlag Franz Vahlen; Versus Verlag.

Liao, W., & Wang, T. (2019). A Novel Collaborative Optimization Model for Job Shop Production–Delivery Considering Time Window and Carbon Emission. *Sustainability, 11*(10), 2781. https://doi.org/10.3390/su11102781

Lin, S.-W., Murphy, B., Clauer, E., Loewen, U., Neubert, R., Bachmann, G., Pai, M., & Hankel, M. (2017). *Architecture Alignment and Interoperability: An Industrial Internet Consortium and Plattform Industrie 4.0 Joint Whitepaper*. Industry IoT Consortium; Plattform Industrie 4.0. https://www.plattform-i40.de/IP/Redaktion/DE/Downloads/Publikation/whitepaper-iic-pi40.html

Lin, S.-W., Simmon, E., Young, D., Miller, B., Durand, J., Bleakley, G., Chigani, A., Martin, R., Murphy, B., & Crawford, M. (2022). *The Industrial Internet Reference Architecture: An Industry IoT Consortium Foundational Document*. Industry IoT Consortium. https://www.iiconsortium.org/iira/

Little, J. D. C. (1970). Models and Managers: The Concept of a Decision Calculus. *Management Science, 16*(8), B466–B485. https://doi.org/10.1287/mnsc.1040.0267

Liu, Z., Guo, S., & Wang, L. (2019). Integrated green scheduling optimization of flexible job shop and crane transportation considering comprehensive energy consumption. *Journal of Cleaner Production, 211,* 765–786. https://doi.org/10.1016/j.jclepro.2018.11.231

Liu, Z., Guo, S., Wang, L., Li, Y., & Li, X. (2019). A Benefit Conflict Resolution Approach for Production Planning Energy Optimization of Group Manufacturing. *IEEE Access, 7,* 45012–45031. https://doi.org/10.1109/ACCESS.2019.2908934

Liu, Z., Luo, Q., Wang, L., Tang, H., & Li, Y. (2022). The Low-Carbon Scheduling Optimization of Integrated Multispeed Flexible Manufacturing and Multi-AGV Transportation. *Processes, 10*(10), 1944. https://doi.org/10.3390/pr10101944

Lotfi, M., Osorio, G., Javadi, M., Ashraf, A., Zahran, M., Samih, G., & Catalao, J. P. S. (2021). A Dijkstra-Inspired Graph Algorithm for Fully Autonomous Tasking in Industrial Applications. *IEEE Transactions on Industry Applications, 57*(5), 5448–5460. https://doi.org/10.1109/TIA.2021.3091418

Lubin, M., Dowson, O., Garcia, J. D., Huchette, J., Legat, B., & Vielma, J. P. (2023). JuMP 1.0: Recent improvements to a modeling language for mathematical optimization. *Mathematical Programming Computation,* (Preprint). https://doi.org/10.48550/arXiv.2206.03866

Lv, Y., Li, C., Tang, Y., & Kou, Y. (2022). Toward Energy-Efficient Rescheduling Decision Mechanisms for Flexible Job Shop With Dynamic Events and Alternative Process Plans. *IEEE Transactions on Automation Science and Engineering, 19*(4), 3259–3275. https://doi.org/10.1109/TASE.2021.3115821

Mirahmadi, N., & Taghipour, S. (2019). Energy-efficient optimization of Flexible Job Shop Scheduling and Preventive Maintenance, In *Annual Symposium on Reliability and Maintainability (RAMS),* IEEE. https://doi.org/10.1109/RAMS.2019.8768908

Mohamed, N., Al-Jaroodi, J., & Lazarova-Molnar, S. (2019). Leveraging the Capabilities of Industry 4.0 for Improving Energy Efficiency in Smart Factories. *IEEE Access, 7,* 18008–18020. https://doi.org/10.1109/ACCESS.2019.2897045

Monostori, L. (2019). Cyber-Physical Systems. In S. Chatti, L. Laperrière, G. Reinhart, & T. Tolio (Eds.), *CIRP Encyclopedia of Production Engineering*. Springer Berlin, Heidelberg. https://doi.org/10.1007/978-3-662-53120-4_16790

Musil, J. (2018). *SAP Activate guides every step on your journey to SAP S/4HANA Cloud*. Retrieved March 24, 2023, from https://blogs.sap.com/2018/05/03/sap-activate-guides-every-step-on-your-journey-to-sap-s4hana-cloud/

Napoleone, A., Macchi, M., & Pozzetti, A. (2020). A review on the characteristics of cyber-physical systems for the future smart factories. *Journal of Manufacturing Systems, 54,* 305–335. https://doi.org/10.1016/J.JMSY.2020.01.007

Narciso, D. A., & Martins, F. G. (2020). Application of machine learning tools for energy efficiency in industry: A review. *Energy Reports, 6,* 1181–1199. https://doi.org/10.1016/j.egyr.2020.04.035

Orozco-Santos, F., Sempere-Paya, V., Silvestre-Blanes, J., & Albero-Albero, T. (2021). Multicast Scheduling in SDN WISE to Support Mobile Nodes in Industrial Wireless Sensor Networks. *IEEE Access, 9,* 141651–141666. https://doi.org/10.1109/ACCESS.2021.3120917

Paterakis, N. G., Erdinç, O., & Catalão, J. P. (2017). An overview of Demand Response: Key-elements and international experience. *Renewable and Sustainable Energy Reviews, 69,* 871–891. https://doi.org/10.1016/J.RSER.2016.11.167

Pereira, M. S., & Lima, F. (2018). A Machine Learning Approach Applied to Energy Prediction in Job Shop Environments, In *IECON 2018 - 44th Annual Conference of the IEEE Industrial Electronics Society,* IEEE. https://doi.org/10.1109/IECON.2018.8592763

Pfeiffer, S. (2017). The Vision of “Industrie 4.0” in the Making - a Case of Future Told, Tamed, and Traded. *Nanoethics,* (11), 107–121. https://doi.org/10.1007/s11569-016-0280-3

Phylipsen, G., Blok, K., & Worrell, E. (1997). International comparisons of energy efficiency-Methodologies for the manufacturing industry. *Energy Policy, 25*(7-9), 715–725. https://doi.org/10.1016/S0301-4215(97)00063-3

Pinedo, M. L. (2016). *Scheduling: Theory, algorithms, and systems*. Springer Cham. https://doi.org/10.1007/978-3-319-26580-3

Projektträger Jülich. (2021). *EnArgus Vorhaben ‘03EN2053A’: Joint Projects: KI4ETA - Artificial Intelligence for energy technologies and application in production; sub-projects: Project management and development of applications for data acquisition, data-based modelling and operation optimisation*. Retrieved June 21, 2023, from https://www.enargus.de/detail/?id=1903685

Qu, M., Zuo, Y., Xiang, F., & Tao, F. (2022). An improved electromagnetism-like mechanism algorithm for energy-aware many-objective flexible job shop scheduling. *The International Journal of Advanced Manufacturing Technology, 119*(7-8), 4265–4275. https://doi.org/10.1007/s00170-022-08665-8

Raffin, A., Hill, A., Gleave, A., Kanervisto, A., Ernestus, M., & Dormann, N. (2021). Stable-Baselines3: Reliable Reinforcement Learning Implementations. *Journal of Machine Learning Research, 22*(268), 1–8. http://jmlr.org/papers/v22/20-1364.html

Rager, M. (2008). *Energieorientierte Produktionsplanung: Analyse, Konzeption und Umsetzung* (Doctoral dissertation). Universität Augsburg. https://doi.org/10.1007/978-3-8350-5569-8

Ramos, C., Barreto, R., Mota, B., Gomes, L., Faria, P., & Vale, Z. (2020). Scheduling of a textile production line integrating PV generation using a genetic algorithm. *Energy Reports, 6,* 148–154. https://doi.org/10.1016/j.egyr.2020.11.093

Ramya, G., Chandrasekaran, M., & Shankar, E. (2019). Case Study Analysis of Job Shop Scheduling and its Integration with Material Requirement Planning. *Materials Today: Proceedings, 16*(2, SI), 1034–1042. https://doi.org/10.1016/j.matpr.2019.05.192

Rencher, A. C., & Christensen, W. F. (2012). *Methods of Multivariate Analysis*. John Wiley & Sons. https://doi.org/10.1002/9781118391686

Rubaiee, S., Cinar, S., & Yildirim, M. B. (2019). An Energy-Aware Multiobjective Optimization Framework to Minimize Total Tardiness and Energy Cost on a Single-Machine Nonpreemptive Scheduling. *IEEE Transactions on Engineering Management*, *66*(4), 699–714. https://doi.org/10.1109/TEM.2018.2846627

Schrems, S. (2014). *Methode zur modellbasierten Integration des maschinenbezogenen Energiebedarfs in die Produktionsplanung* (Doctoral dissertation). Technische Universität Darmstadt. Fachbereich Maschinenbau. https://tubiblio.ulb.tu-darmstadt.de/67653/

Schuh, G., Gottschalk, S., & Höhne, T. (2007). High Resolution Production Management. *CIRP Annals*, *56*(1), 439–442. https://doi.org/10.1016/J.CIRP.2007.05.105

Schuh, G. (2019). Enterprise Resource Planning. In S. Chatti, L. Laperrière, G. Reinhart, & T. Tolio (Eds.), *CIRP Encyclopedia of Production Engineering*. Springer Berlin, Heidelberg. https://doi.org/10.1007/978-3-662-53120-4_6673

Schuh, G., Stich, V., Runge, S., Brosze, T., Brandenburg, U., Cuber, S., Schenk, M., Quick, J., Schmidt, C., Helmig, J., Schürmeyer, M., Hering, N., Meier, C., Bauhoff, F., Schoth, A., & Kompa, S. (2015). *Produktionsplanung und -steuerung 1: Grundlagen der PPS* (4th edition, Vol. 1). Springer Gabler Berlin, Heidelberg. https://doi.org/10.1007/978-3-662-43542-7

Schwarz, L. (2022). *6 Key Phases of an ERP Implementation Plan*. Retrieved March 24, 2023, from https://www.netsuite.com/portal/resource/articles/erp/erp-implementation-phases.shtml

Selmair, M., Claus, T., Trost, M., Bley, A., & Herrmann, F. (2016). Job Shop Scheduling With Flexible Energy Prices (T. Claus, F. Herrmann, M. Manitz, & O. Rose, Eds.). In T. Claus, F. Herrmann, M. Manitz, & O. Rose (Eds.), *30th European Conference on Modelling and Simulation (ECMS 2016)*, Curran Associates. https://doi.org/10.7148/2016-0488

Shao, C., Ding, Y., Siano, P., & Song, Y. (2021). Optimal Scheduling of the Integrated Electricity and Natural Gas Systems Considering the Integrated Demand Response of Energy Hubs. *IEEE Systems Journal*, *15*(3), 4545–4553. https://doi.org/10.1109/JSYST.2020.3020063

Sihn, W., Sobottka, T., Heinzl, B., & Kamhuber, F. (2018). Interdisciplinary multi-criteria optimization using hybrid simulation to pursue energy efficiency through production planning. *CIRP Annals*, *67*(1), 447–450. https://doi.org/10.1016/j.cirp.2018.04.059

Silver, E. A., Pyke, D. F., & Peterson, R. (1998). *Inventory management and production planning and scheduling*. John Wiley & Sons. https://archive.org/details/inventorymanagem00silv

Smith, S. W. (1999). *The scientist and engineer's guide to digital signal processing* (2. ed.). California Technical Publishing. https://www.analog.com/en/education/education-library/scientist_engineers_guide.html

Stricker, N., Kuhnle, A., Sturm, R., & Friess, S. (2018). Reinforcement learning for adaptive order dispatching in the semiconductor industry. *CIRP Annals*, *67*(1), 511–514. https://doi.org/10.1016/j.cirp.2018.04.041

Sun, Z., Dababneh, F., & Li, L. (2020). Joint Energy, Maintenance, and Throughput Modeling for Sustainable Manufacturing Systems. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, *50*(6), 2101–2112. https://doi.org/10.1109/TSMC.2018.2799740

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An Introduction* (2nd edition). The MIT Press.

The White House. (2021). *National Climate Task Force*. Retrieved March 2, 2023, from https://www.whitehouse.gov/climate/

Thiede, S., Schönemann, M., Kurle, D., & Herrmann, C. (2016). Multi-level simulation in manufacturing companies: The water-energy nexus case. *Journal of Cleaner Production*, *139*, 1118–1127. https://doi.org/10.1016/j.jclepro.2016.08.144

Tian, Z., Jiang, X., Liu, W., & Li, Z. (2023). Dynamic energy-efficient scheduling of multi-variety and small batch flexible job-shop: A case study for the aerospace industry. *Computers and Industrial Engineering*, *178*, 109111. https://doi.org/10.1016/j.cie.2023.109111

Ujazdowski, T., & Piotrowski, R. (2022). Task Scheduling–Review of Algorithms and Analysis of Potential Use in a Biological Wastewater Treatment Plant. *IEEE Access*, *10*, 45230–45240. https://doi.org/10.1109/ACCESS.2022.3170105

United Nations. (2015). Transforming our world: the 2030 Agenda for Sustainable Development. *General Assembley 70 session*, *16301*(October), 1–35. https://doi.org/10.1007/s13398-014-0173-7.2

U.S. Department of Energy. (2006). *Benefits of Demand Response in Electricity Markets and Recommentations for Achieving Them: A Report to the United States Congress Prusuant to Section 1252 of the Energy Policy Act of 2005*. https://www.energy.gov/oe/articles/benefits-demand-response-electricity-markets-and-recommendations-achieving-them-report

Vahid-Ghavidel, M., Javadi, M. S., Santos, S. F., Gough, M., Mohammadi-Ivatloo, B., Shafie-Khah, M., & Catalao, J. P. S. (2021). Novel Hybrid Stochastic-Robust Optimal Trading Strategy for a Demand Response Aggregator in the Wholesale Electricity Market. *IEEE Transactions on Industry Applications*, *57*(5), 5488–5498. https://doi.org/10.1109/TIA.2021.3098500

VDI/VDE-Gesellschaft Mess- und Automatisierungstechnik. (2013). *Cyber-Physical Systems: Chancen und Nutzen aus Sicht der Automation*. Verein Deutscher Ingenieure e.V. https://www.vdi.de/ueber-uns/presse/publikationen/details/cyber-physical-systems-chancen-und-nutzen-aus-sicht-der-automation

Verein Deutscher Ingenieure e.V. (2020). *Energieflexible Fabrik: Grundlagen* (VDI 5207). Beuth Verlag. https://www.beuth.de/de/technische-regel/vdi-5207-blatt-1/323854546

Verein Deutscher Maschinen- und Anlagenbauer. (2019). *Messvorschrift zur Bestimmung des Energie- und Medienbedarfs von Werkzeugmaschinen in der Serienfertigung* (VDMA 34179). Beuth Verlag. https://www.beuth.de/de/technische-regel/vdma-34179/302624354

Walther, J., Dietrich, B., Grosch, B., Lindner, M., Fuhrländer-Völker, D., Strobel, N., & Weigold, M. (2022). A Methodology for the Classification and Characterisation of Industrial Demand-Side Integration Measures. *Energies*, *15*(3), 923–949. https://doi.org/10.3390/EN15030923

Wang, H., Jiang, Z., Wang, Y., Zhang, H., & Wang, Y. (2018). A two-stage optimization method for energy-saving flexible job-shop scheduling based on energy dynamic characterization. *Journal of Cleaner Production*, *188*, 575–588. https://doi.org/10.1016/j.jclepro.2018.03.254

Wang, J., Liu, Y., Ren, S., Wang, C., & Wang, W. (2021). Evolutionary game based real-time scheduling for energy-efficient distributed and flexible job shop. *Journal of Cleaner Production*, *293*, 126093. https://doi.org/10.1016/j.jclepro.2021.126093

Wang, J., Yang, J., Zhang, Y., Ren, S., & Liu, Y. (2020). Infinitely repeated game based real-time scheduling for low-carbon flexible job shop considering multi-time periods. *Journal of Cleaner Production*, *247*, 119093. https://doi.org/10.1016/j.jclepro.2019.119093

Wang, J., Zhang, Y., Liu, Y., & Wu, N. (2019). Multiagent and Bargaining-Game-Based Real-Time Scheduling for Internet of Things-Enabled Flexible Job Shop. *IEEE Internet of Things Journal*, *6*(2), 2518–2531. https://doi.org/10.1109/JIOT.2018.2871346

Wang, Y., Li, K., Gan, S., & Cameron, C. (2019). Analysis of energy saving potentials in intelligent manufacturing: A case study of bakery plants. *Energy*, *172*, 477–486. https://doi.org/10.1016/j.energy.2019.01.044

Wang, Z., Xu, Z., Liu, B., Zhang, Y., & Yang, Q. (2022). A Hybrid Cleaning Scheduling Framework for Operations and Maintenance of Photovoltaic Systems. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, *52*(9), 5925–5936. https://doi.org/10.1109/TSMC.2021.3131031

Waschneck, B., Reichstaller, A., Belzner, L., Altenmüller, T., Bauernhansl, T., Knapp, A., & Kyek, A. (2018). Optimization of global production scheduling with deep reinforcement learning. *Procedia CIRP*, *72*, 1264–1269. https://doi.org/10.1016/j.procir.2018.03.212

Wiendahl, H.-P., & Wiendahl, H.-H. (2019). *Betriebsorganisation für Ingenieure*. Carl Hanser Verlag. https://doi.org/10.3139/9783446460614

World Meteorological Organization. (2023). *State of the Global Climate 2022*. https://library.wmo.int/index.php?lvl=notice_display&id=22265

Xu, J., & Wang, L. (2017). A Feedback Control Method for Addressing the Production Scheduling Problem by Considering Energy Consumption and Makespan. *Sustainability, 9*(7), 1185. https://doi.org/10.3390/su9071185

Yang, X., Zeng, Z., Wang, R., Sun, X., & Song, H. (2016). Bi-Objective Flexible Job-Shop Scheduling Problem Considering Energy Consumption under Stochastic Processing Times. *PLOS ONE, 11*(12), e0167427. https://doi.org/10.1371/journal.pone.0167427

Yin, L., Li, X., Gao, L., Lu, C., & Zhang, Z. (2017). A novel mathematical model and multi-objective method for the low-carbon flexible job shop scheduling problem. *Sustainable Computing: Informatics and Systems, 13,* 15–30. https://doi.org/10.1016/j.suscom.2016.11.002

Zarte, M., Pechmann, A., & Nunes, I. L. (2021). A Fuzzy Inference Model for Social-Sustainability Production Planning (I. L. Nunes, Ed.). In I. L. Nunes (Ed.), *Advances in Human Factors and System Interactions,* Springer, Cham. https://doi.org/10.1007/978-3-030-79816-1_17

Zeng, Q., Wang, M., Shen, L., & Song, H. (2019). Sequential Scheduling Method for FJSP with Multi-Objective under Mixed Work Calendars. *Processes, 7*(12), 888. https://doi.org/10.3390/pr7120888

Zeng, Z., Hong, M., Man, Y., Li, J., Zhang, Y., & Liu, H. (2018). Multi-object optimization of flexible flow shop scheduling with batch process — Consideration total electricity consumption and material wastage. *Journal of Cleaner Production, 183,* 925–939. https://doi.org/10.1016/j.jclepro.2018.02.224

Zhang, W., Zheng, Y., & Ahmad, R. (2023). An energy-efficient multi-objective scheduling for flexible job-shop-type remanufacturing system. *Journal of Manufacturing Systems, 66,* 211–232. https://doi.org/10.1016/j.jmsy.2022.12.008

Zhang, X., Zhang, H., & Yao, J. (2020). Multi-Objective Optimization of Integrated Process Planning and Scheduling Considering Energy Savings. *Energies, 13*(23), 6181. https://doi.org/10.3390/en13236181

Zhang, Y. W. (2022). Energy-Aware Nonpreemptive Scheduling of Mixed-Criticality Real-Time Task Systems. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 41*(9), 2891–2900. https://doi.org/10.1109/TCAD.2021.3120326

Zhang, Y., Wang, J., & Liu, Y. (2017). Game theory based real-time multi-objective flexible job shop scheduling considering environmental impact. *Journal of Cleaner Production, 167,* 665–679. https://doi.org/10.1016/j.jclepro.2017.08.068

Zhu, S., Zhang, H., Jiang, Z., & Hon, B. (2020). A carbon efficiency upgrading method for mechanical machining based on scheduling optimization strategy. *Frontiers of Mechanical Engineering, 15*(2), 338–350. https://doi.org/10.1007/s11465-019-0572-8

Own Publications

The following publications were published while working on this thesis.

2023 Fuhrländer-Völker, D., Grosch, B., & Weigold, M. (2023). Modelling and Control of Aqueous Parts Cleaning Machines for Demand Response. In D. Herberger, M. Hübner, & V. Stich (Eds.), *Proceedings of the Conference on Production Systems and Logistics* (pp. 790–800). publish-Ing. https://doi.org/10.15488/13498

Lindner, M., Grosch, B., Elserafi, G., Dietrich, B., & Weigold, M. (2023). Holistic Approach for an Energy-Flexible Operation of a Machine Tool with Cooling Supply. *Energies, 16*(9), 3943. https://doi.org/10.3390/en16093943

2022 Grosch, B., Fuhrländer-Völker, D., Stock, J., & Weigold, M. (2022). Cyber-physical production system for energy-flexible control of production machines. *Procedia CIRP, 107*, 221–226. https://doi.org/10.1016/j.procir.2022.04.037

Grosch, B., Petruschke, L., Burkhardt, M., Kohne, T., Wächter, A., & Weigold, M. (2022). Energieeffizienz in der Produktion umsetzen/Implementing energy efficiency measures in industrial production – Motivating factors and implementation barriers from the ETA-Transfer project. *wt Werkstattstechnik online, 112*(01-02), 97–102. https://doi.org/10.37544/1436-4980-2022-01-02-101

Grosch, B., Ranzau, H., Dietrich, B., Kohne, T., Fuhrländer-Völker, D., Sossenheimer, J., Lindner, M., & Weigold, M. (2022). A framework for researching energy optimization of factory operations. *Energy Informatics, 5*, 15–16. https://doi.org/10.1186/s42162-022-00207-6

Walther, J., Dietrich, B., Grosch, B., Lindner, M., Fuhrländer-Völker, D., Strobel, N., & Weigold, M. (2022). A Methodology for the Classification and Characterisation of Industrial Demand-Side Integration Measures. *Energies, 15*(3), 923–949. https://doi.org/10.3390/EN15030923

2021 Grosch, B., Kohne, T., & Weigold, M. (2021). Multi-objective hybrid genetic algorithm for energy adaptive production scheduling in job shops. *Procedia CIRP, 98*, 294–299. https://doi.org/10.1016/j.procir.2021.01.106

Petruschke, L., Burkhardt, M., Grosch, B., Kohne, T., Weigold, M., Zens, M., & Frings, L. (2021). Energieeffizienz von lufttechnischen Anlagen. In *wt Werkstattstechnik online* (pp. 31–36), VDI Fachmedien. https://e-paper.vdi-fachmedien.de/werkstattstechnik/2021#01-02-2021

2020 Grosch, B., Seyfried, S., Strobel, N., & Weigold, M. (2020). Chancen durch den Einsatz Künstlicher Intelligenz im industriellen Energiemanagement. In *Mittelstand-Digital Magazin: Wissenschaft trifft Praxis* (pp. 15–20), Mittelstand-Digital c/o WIK-Consult GmbH. https://www.mittelstand-digital.de/MD/Redaktion/DE/Publikationen/Wissenschaft-trifft-Praxis/magazin-wissenschaft-trifft-praxis-ausgabe-14.pdf

Weyand, A., Seyfried, S., Gosch, B., Fuhrländer-Völker, D., Panten, N., & Abele, E. (2020). Method for transforming energy and production systems into mobile demonstrators. *Procedia Manufacturing, 45,* 134–139. https://doi.org/10.1016/j.promfg.2020.04.084

2019 Grosch, B., Weitzel, T., Panten, N., & Abele, E. (2019). A metaheuristic for energy adaptive production scheduling with multiple energy carriers and its implementation in a real production system. *Procedia CIRP, 80,* 203–208. https://doi.org/10.1016/j.procir.2019.01.043

Credits

Figure 2.7 on page 30 reproduced with permission of the Verein Deutscher Ingeniuere e.V.
Lower portion of Figure 5.7 on page 96 from (© IEEE, 2002).
Figure 5.8 on page 97 from (© IEEE, 2002).

A. Data-gathering plans

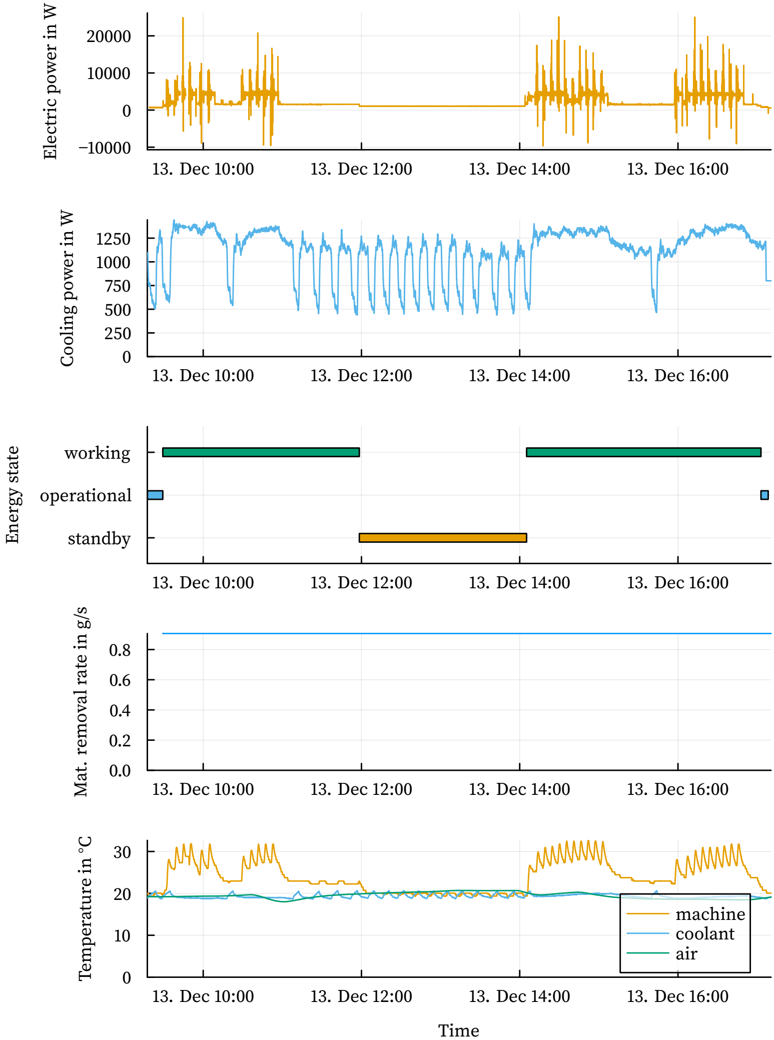
Data-gathering plan for EMAG VLC100 GT machine tool.

Variable Name	Data Source	Type of Sensor	Unit
Electric power	calculated		W
Machine (Alternating Current)	Janitza UMG 604	current transformer	W
Drives (DC)	QI-POWER-485-300	current transformer	W
Cooling power	calculated		W
Feed temperature	Grundfos Magna 3	internal (unknown)	°C
Return temperature	Grundfos Magna 3	internal (unknown)	°C
Volume flow	Grundfos Magna 3	internal (unknown)	L/min
Energy states			
standby	machine PLC	software	binary
operational	machine PLC	software	binary
working	machine PLC	software	binary
Material removal rate	calculated		g/s
Weight before	manual	KERN 824	g
Weight after	manual	KERN 824	g
Processing time	manual	stop watch	s
Product	machine PLC	software	-
Machine Temperature	machine PLC	internal sensor	°C

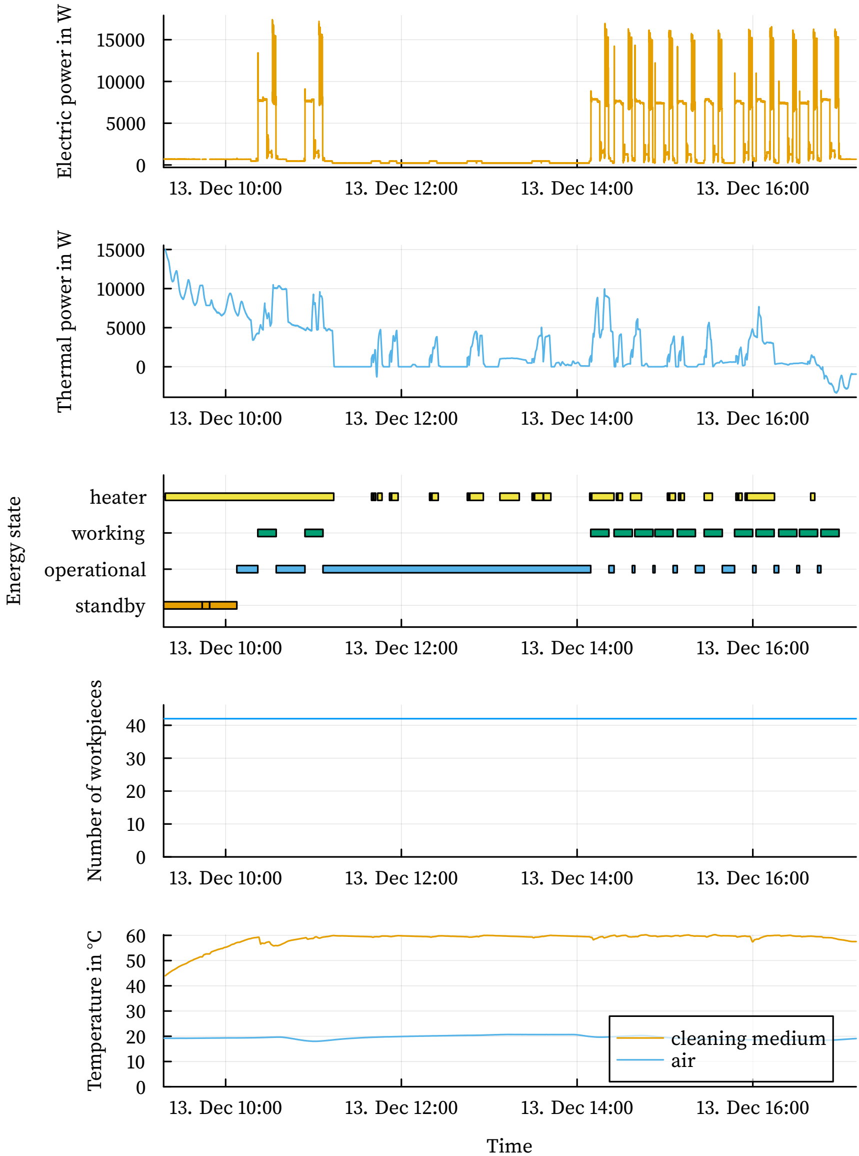
Data-gathering plan for MAFAC KEA cleaning machine.

Variable Name	Data Source	Type of Sensor	Unit
Electric power machine	Janitza UMG 96 RM	current transformer	W
Medium tank temp.	machine PLC	PT 100	°C
Energy states			
standby	machine PLC	software	binary
operational	machine PLC	software	binary
working	machine PLC	software	binary
heater operation	calculated	electric power > 9500 W	binary
Batch size	machine PLC	counted / software	pieces
Capacity	manual	documentation	pieces
Lower temperature limit	manual	documentation	°C
Upper temperature limit	manual	documentation	°C

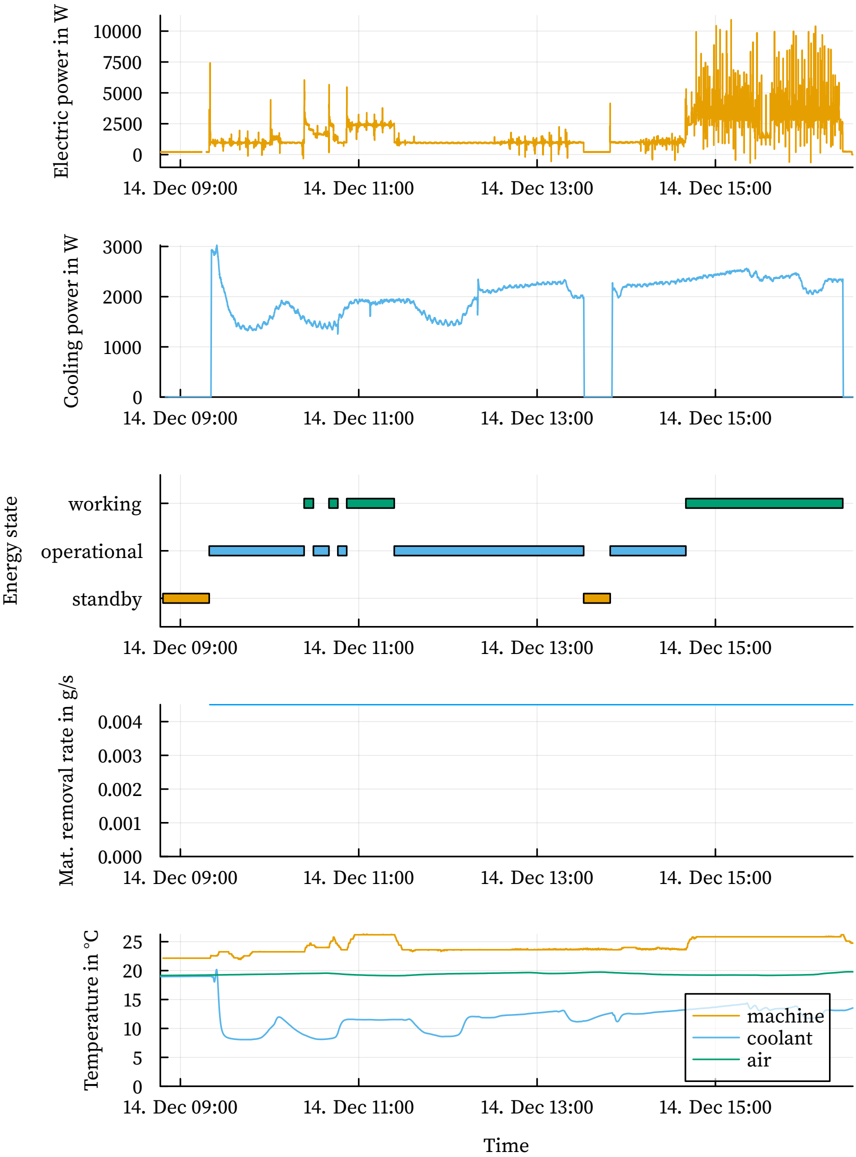
B. Experiment Data Plots



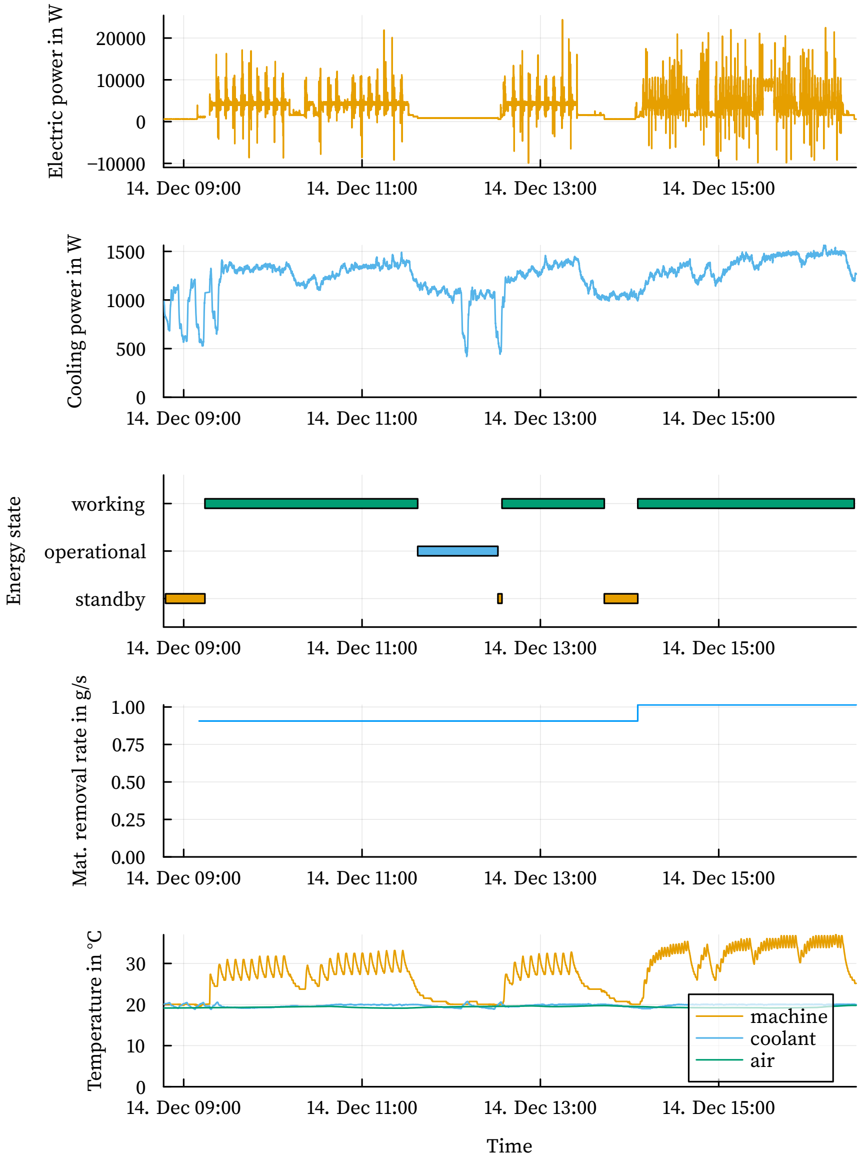
Gathered data for the EMAG VLC100 Y machine tool during the experiments on 13 December 2022.



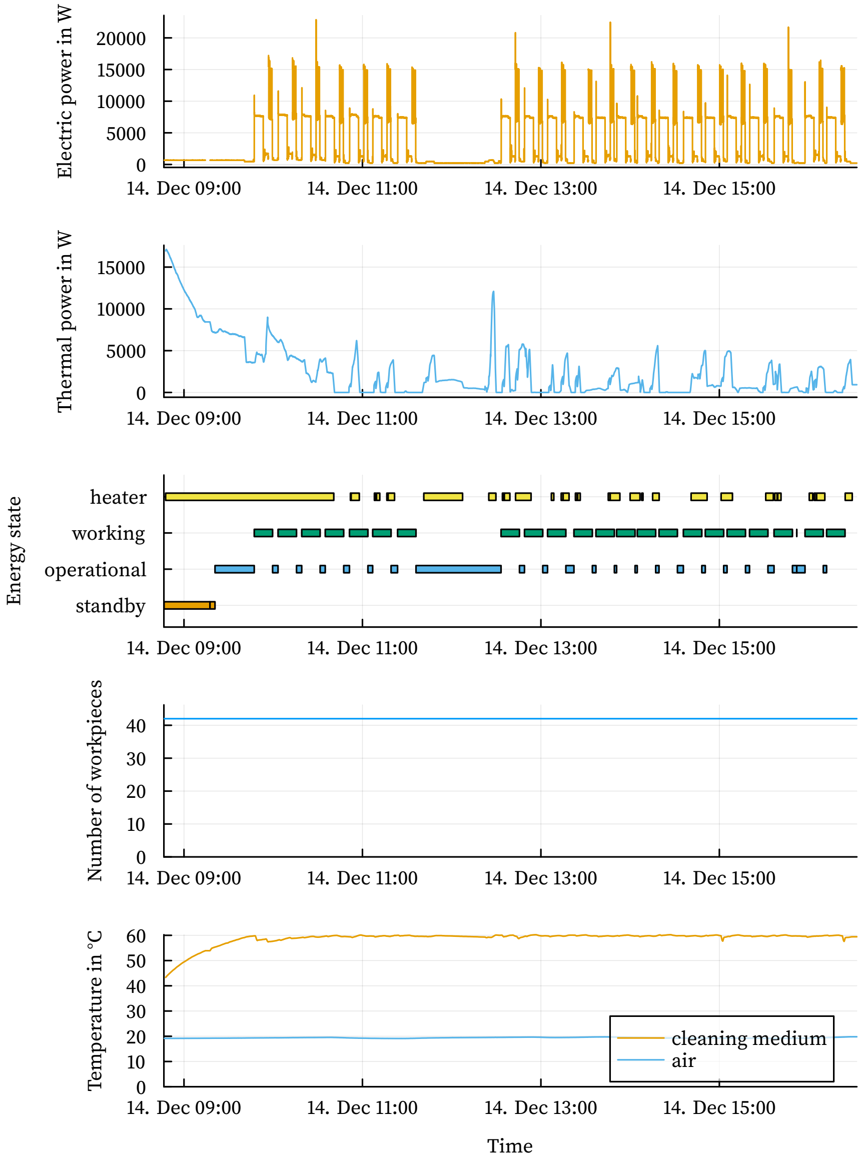
Gathered data for the MAFAC JAVA cleaning machine during experiments on 13 December 2022.



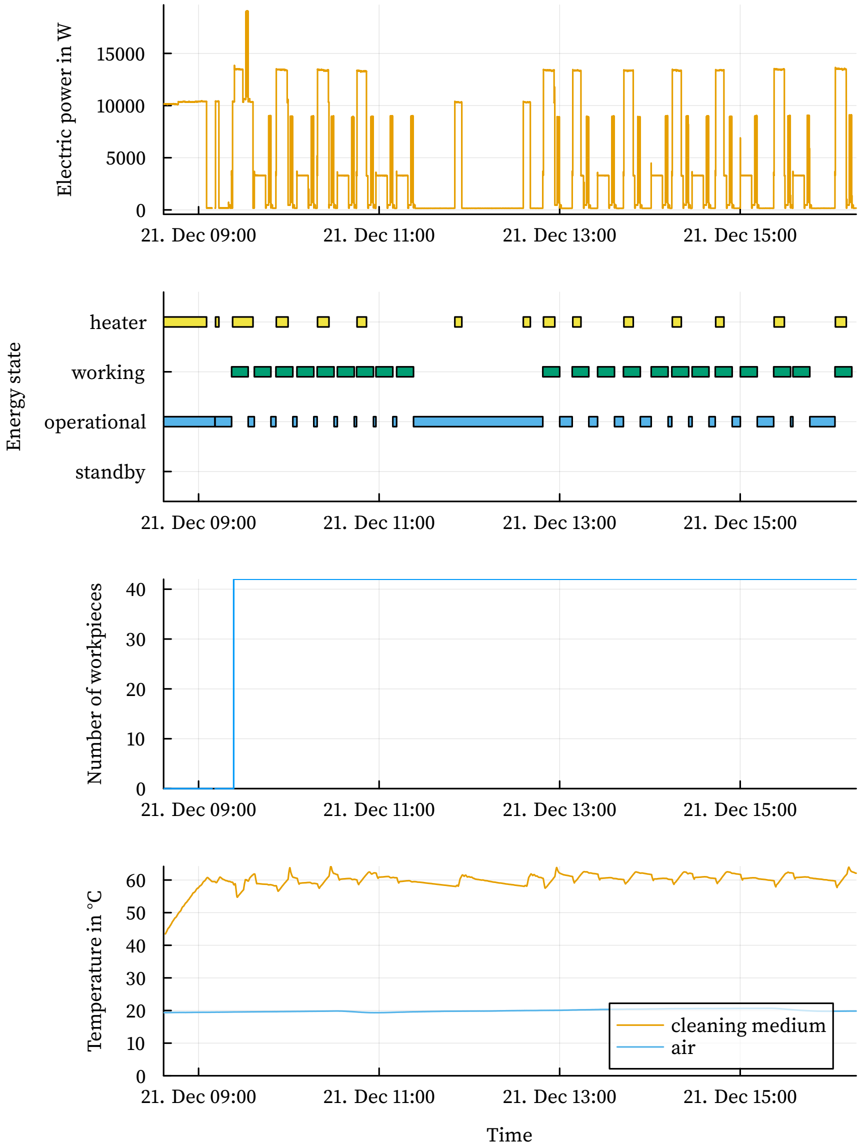
Gathered data for the EMAG VLC100 GT machine tool during the experiments on 14 December 2022.



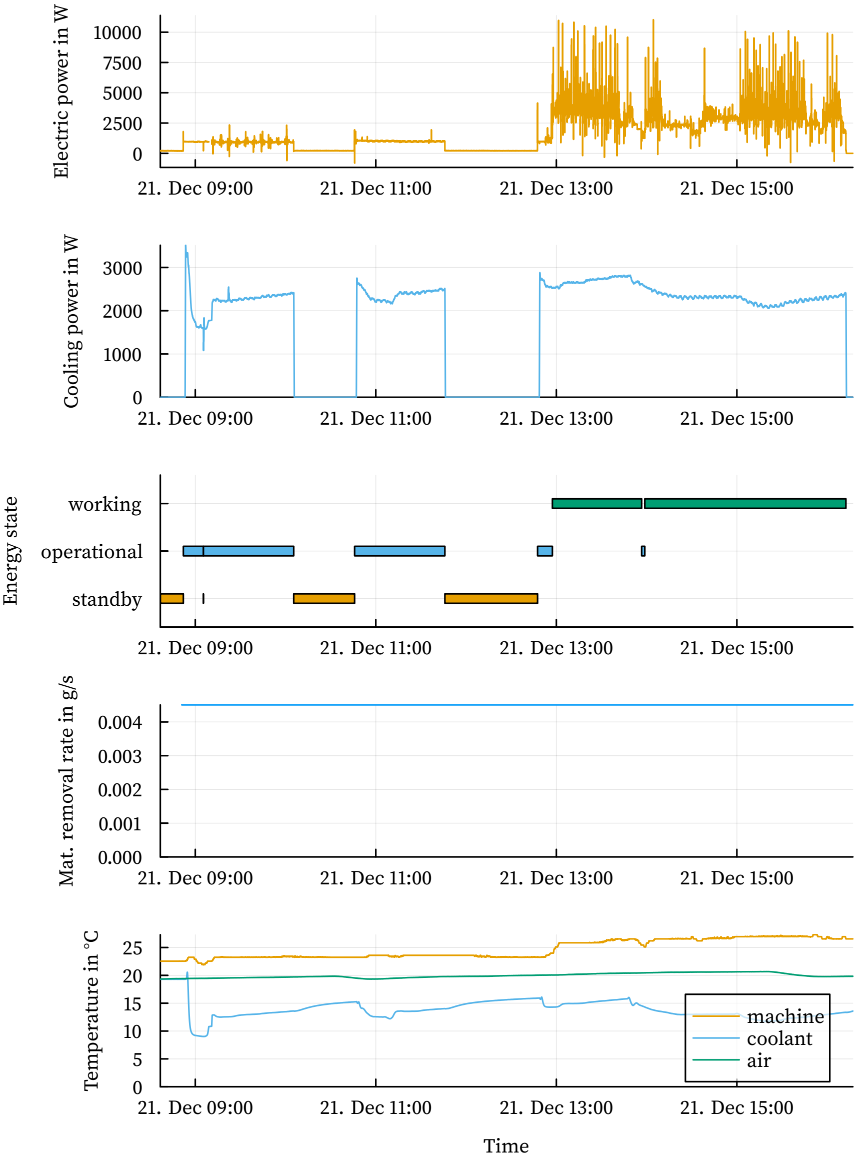
Gathered data for the Emag VLC100 Y machine tool during the experiments on 14 December 2022.



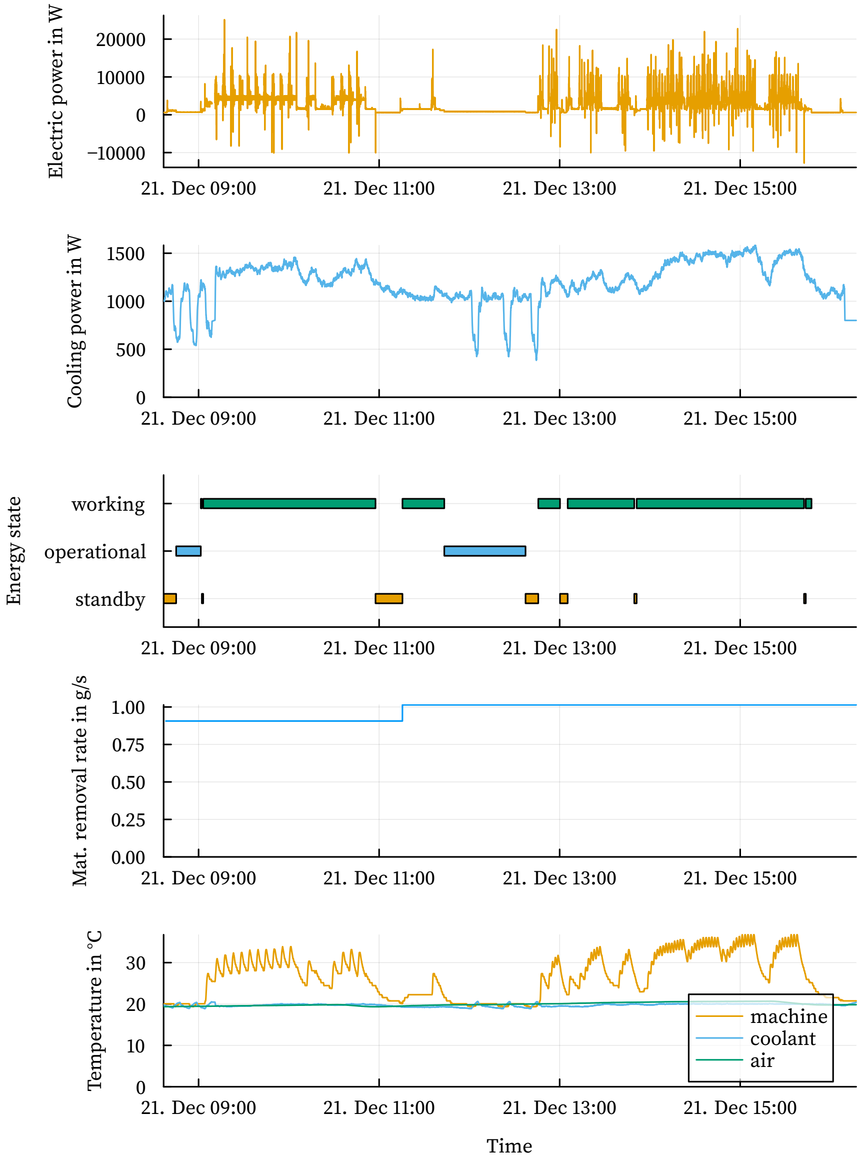
Gathered data for the MAFAC JAVA cleaning machine during experiments on 14 December 2022.



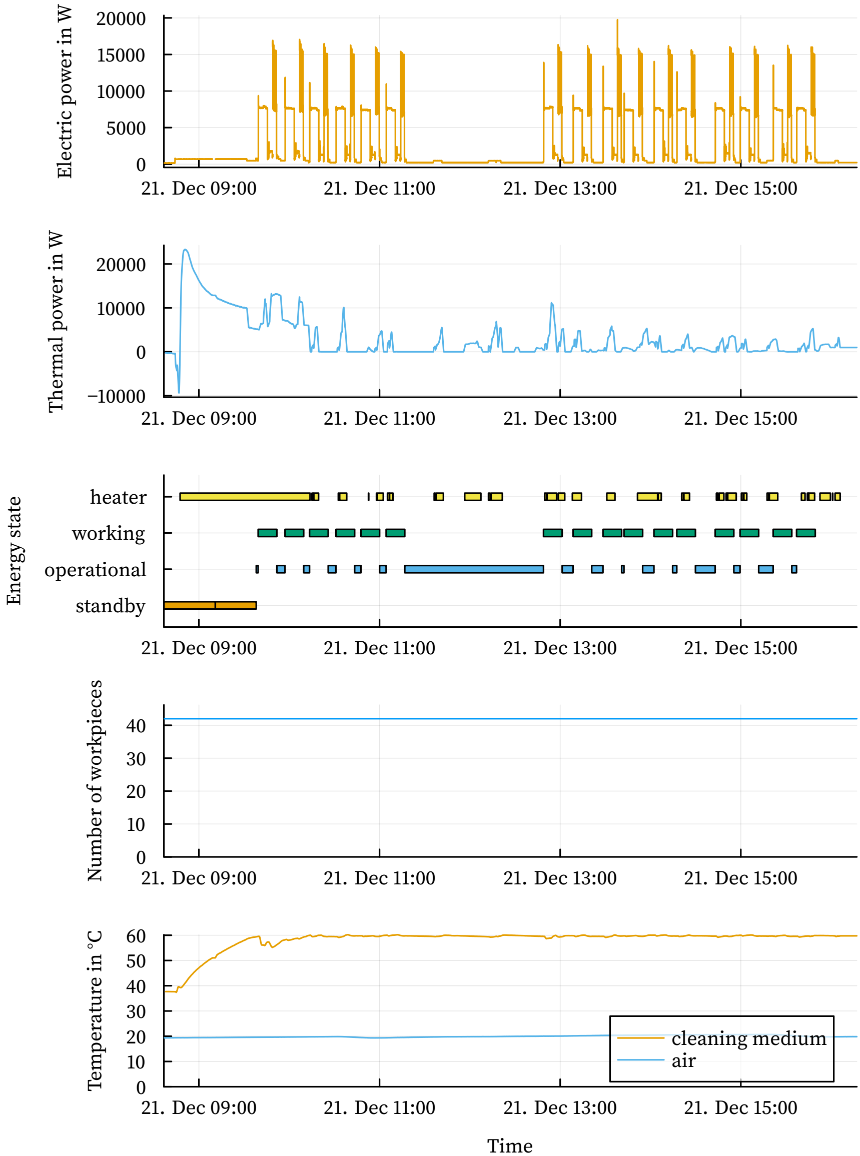
Gathered data for the MAFAC KEA cleaning machine during experiments on 14 December 2022.



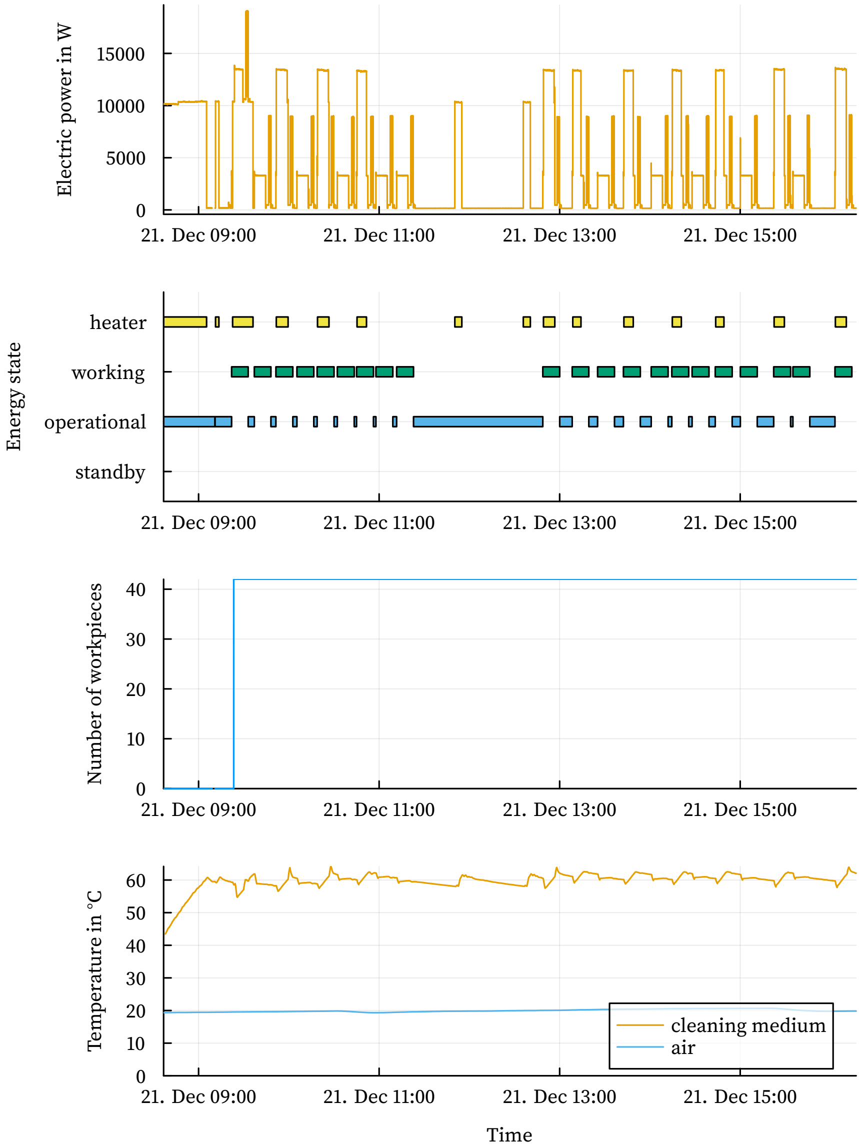
Gathered data for the EMAG VLC100 GT machine tool during the experiments on 21 December 2022.



Gathered data for the EMAG VLC100 Y machine tool during the experiments on 21 December 2022.

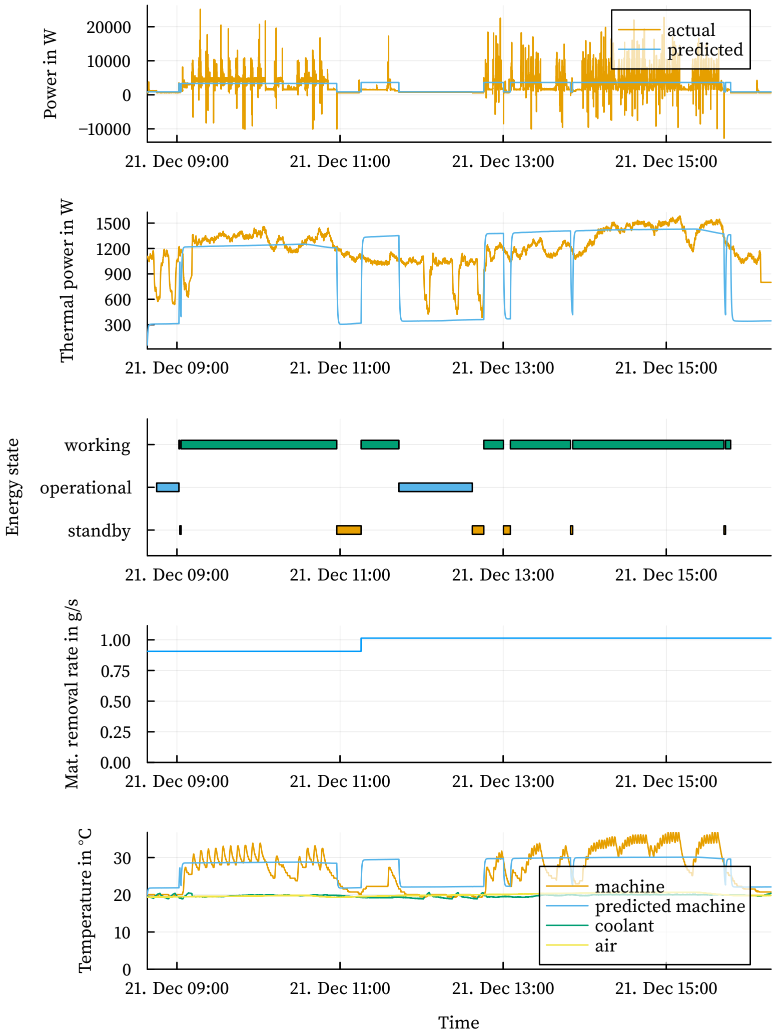


Gathered data for the MAFAC JAVA cleaning machine during experiments on 21 December 2022.

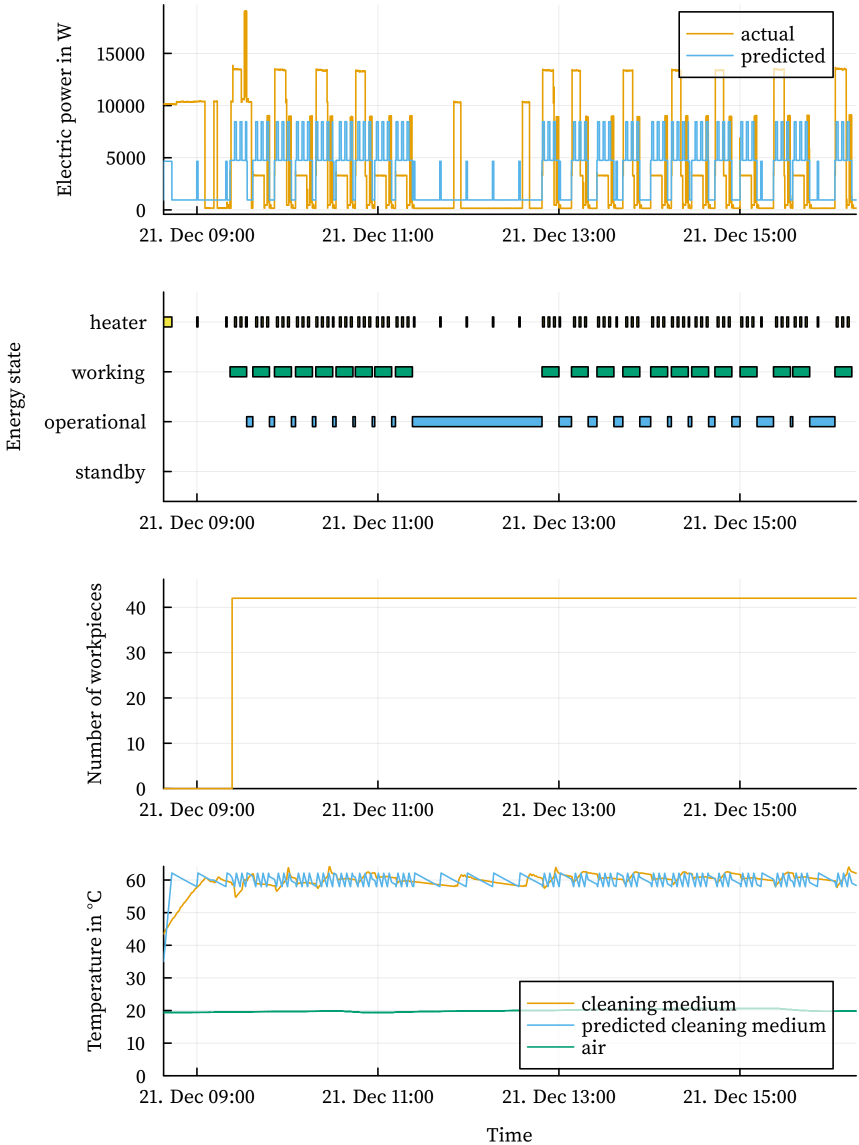


Gathered data for the MAFAC KEA cleaning machine during experiments on 21 December 2022.

C. Test Data Plots

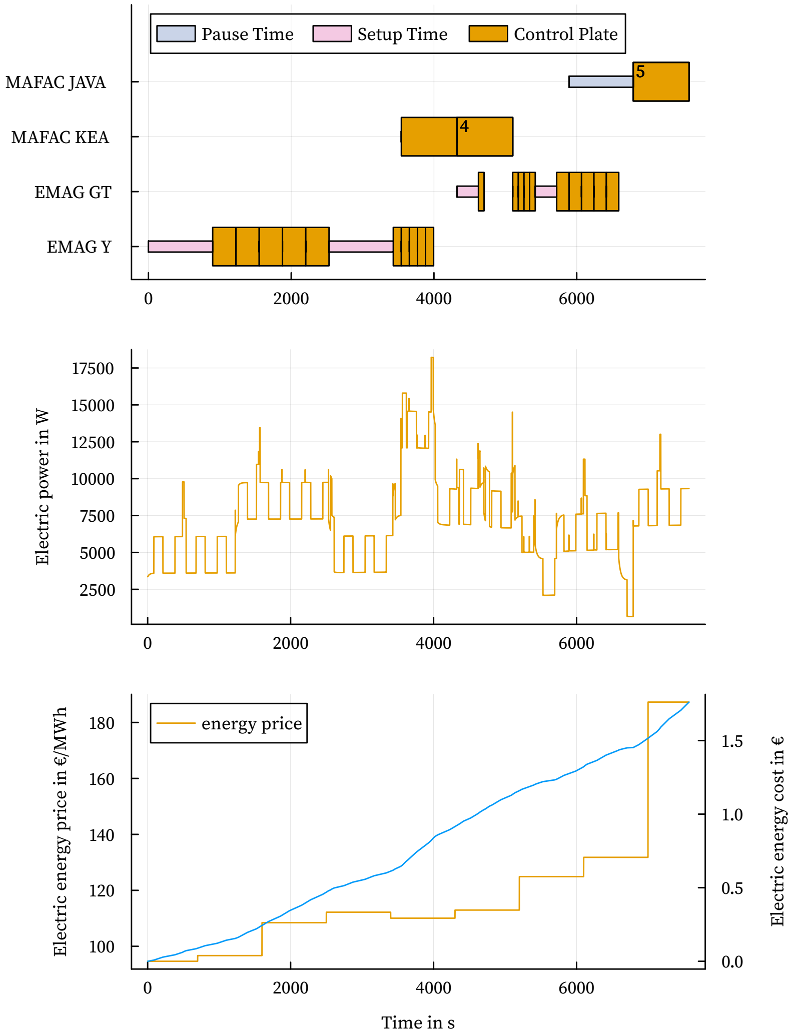


Testing data for the EMAG VLC100 Y machine tool during the experiments on 21 December 2022.

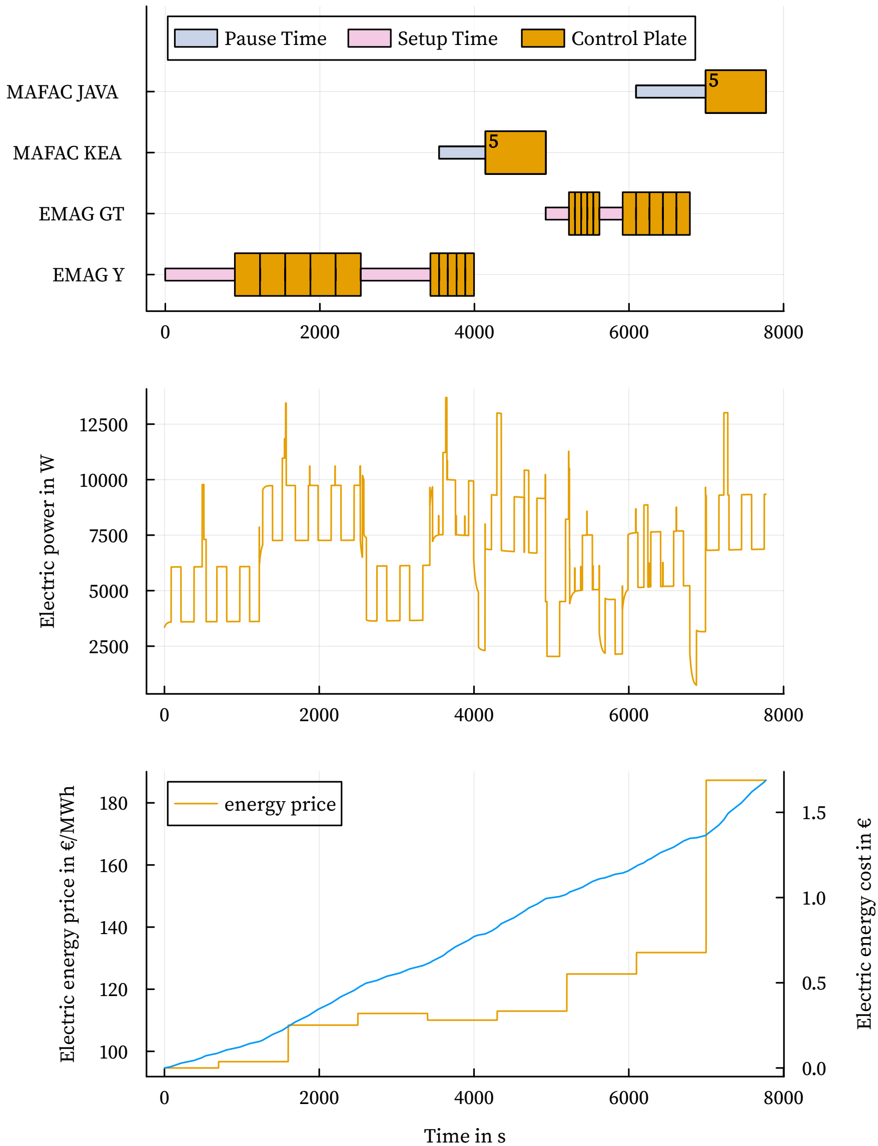


Testing data for the MAFAC KEA cleaning machine during the experiments on 21 December 2022.

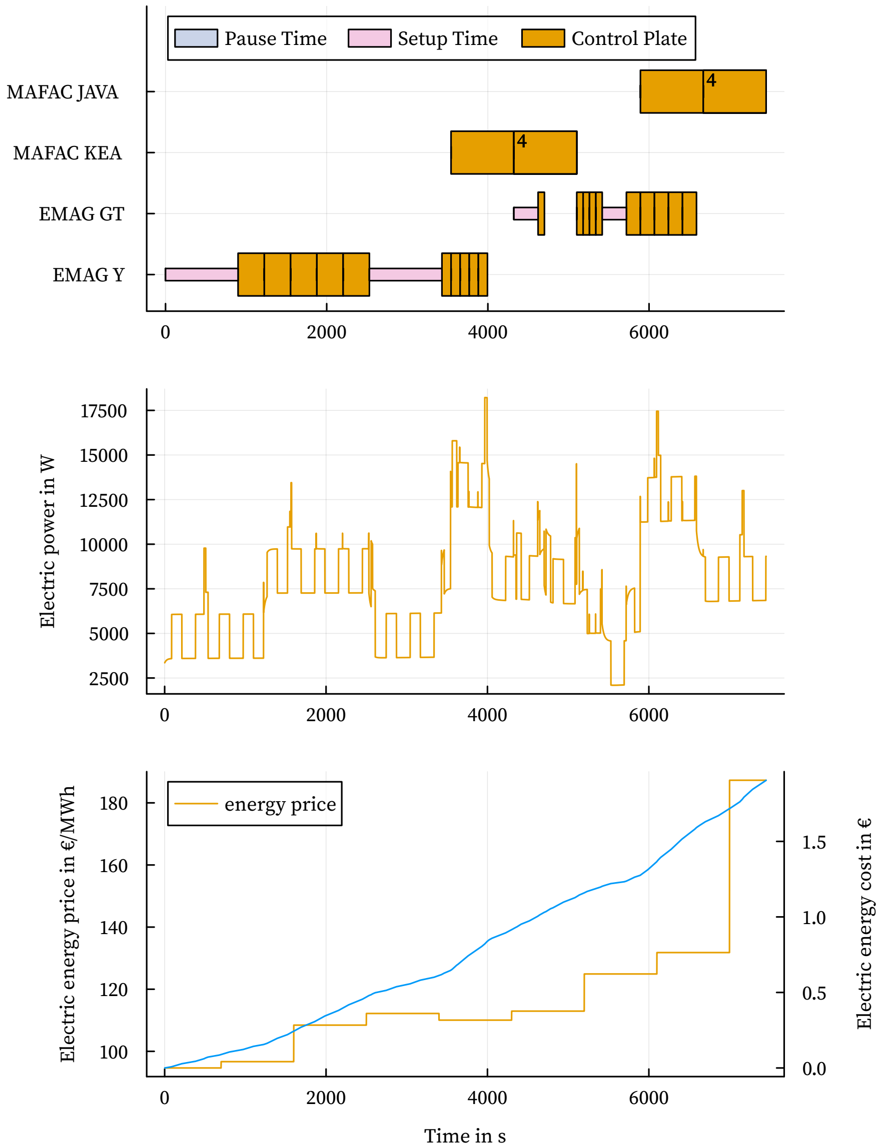
D. Scheduling Experiment Plots



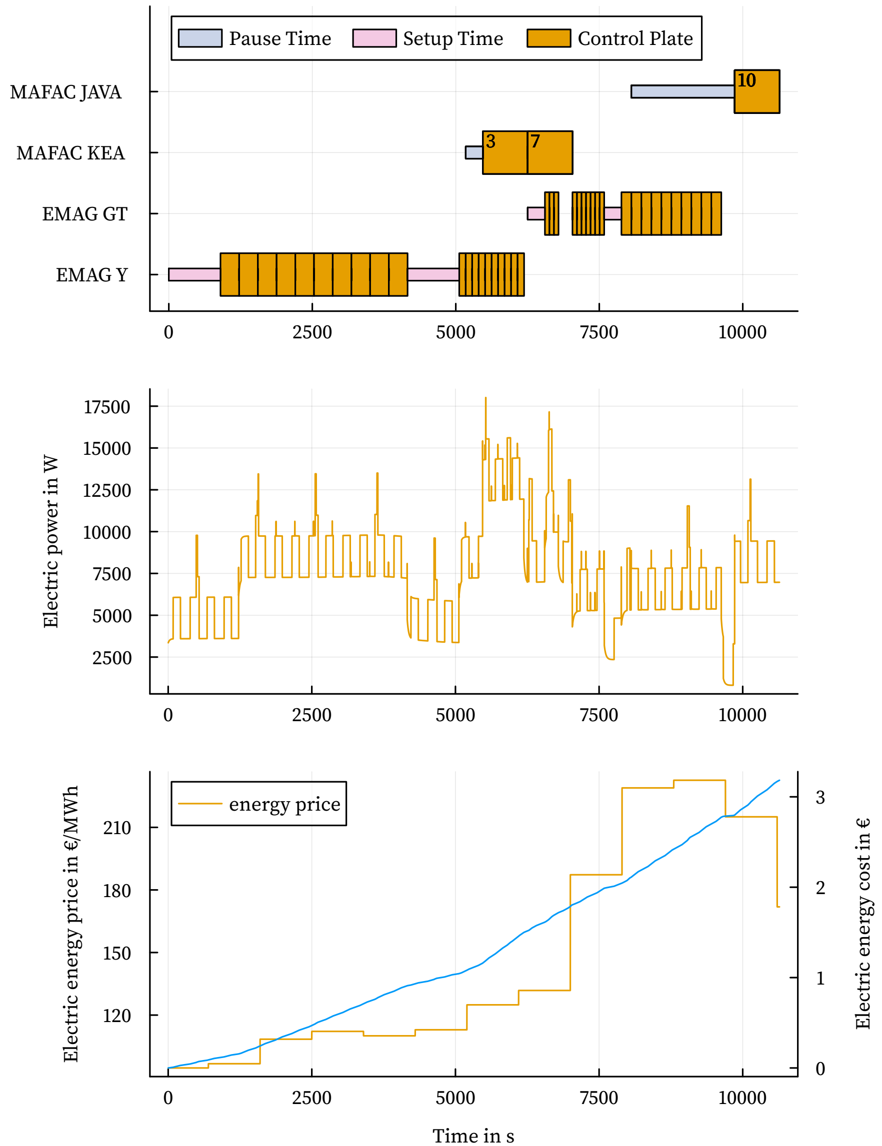
Solution with the best makespan from experiment 1.



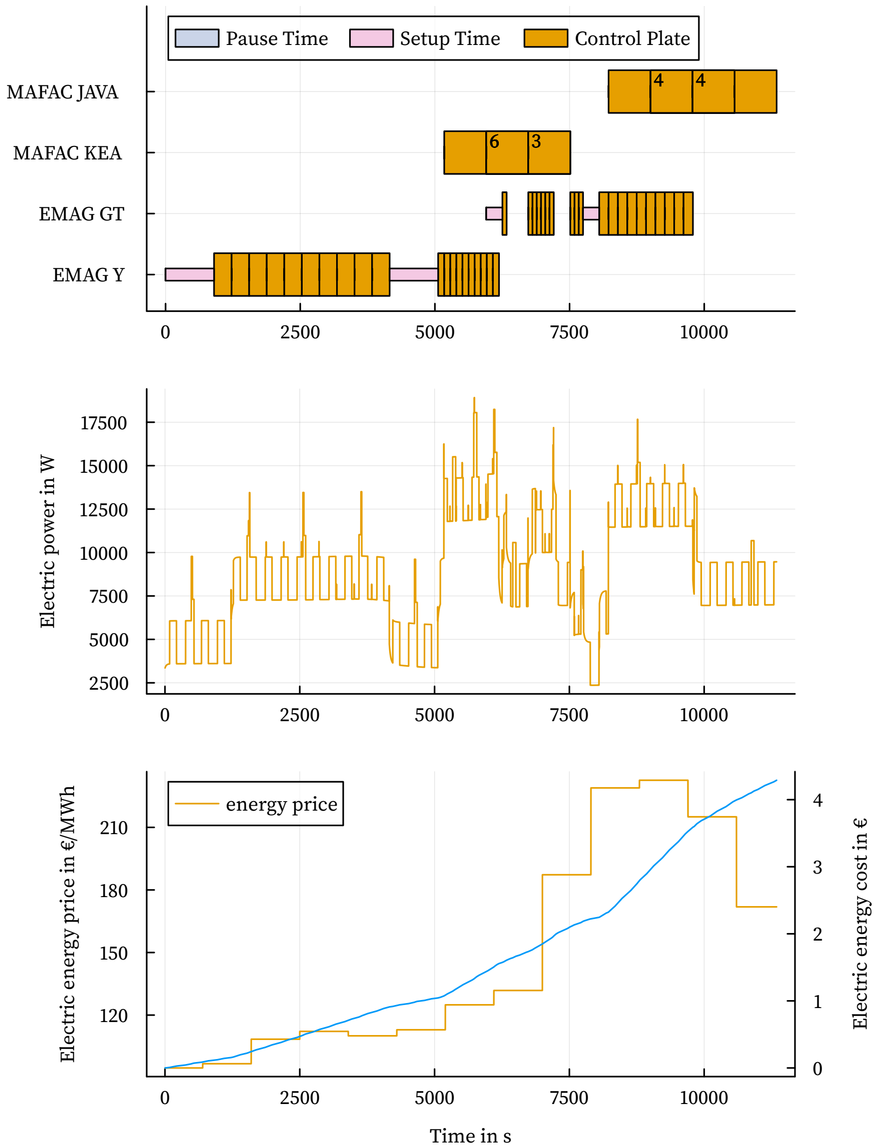
Solution with the best energy-related cost from experiment 1.



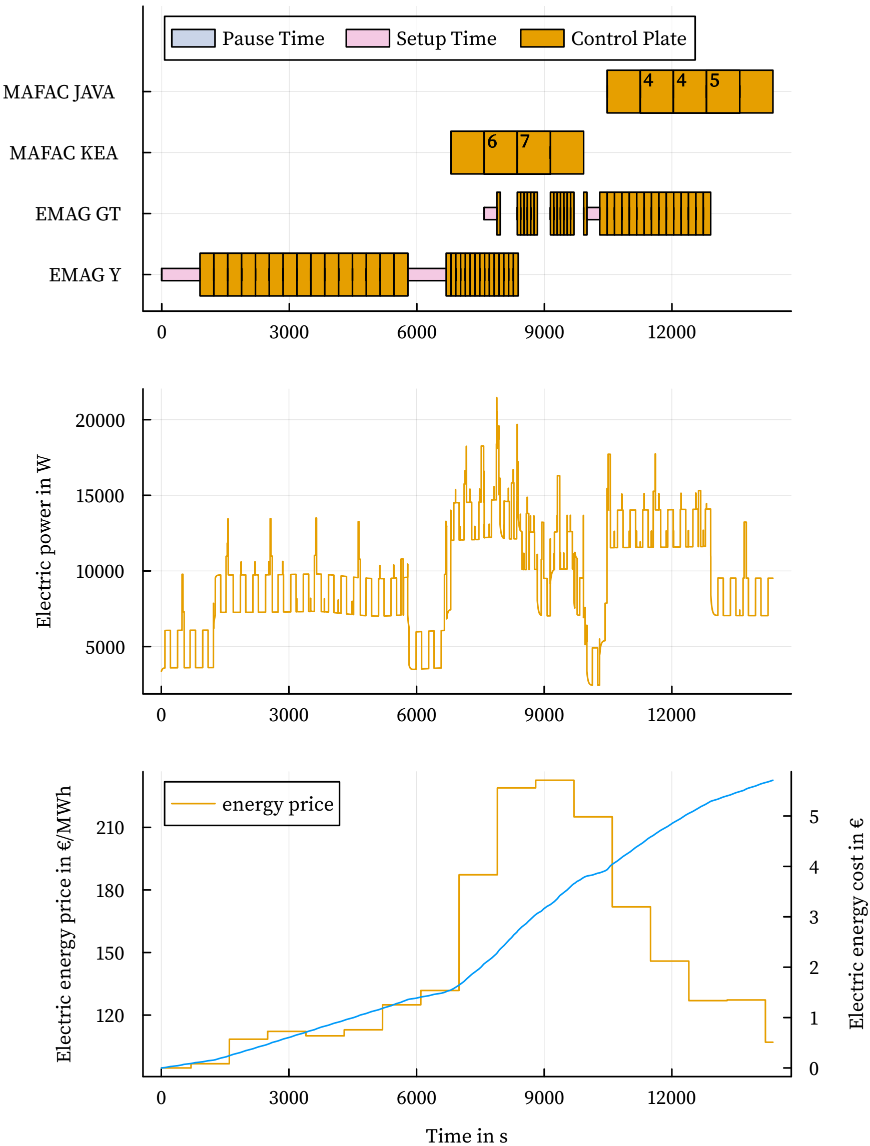
Schedule for experiment 1, generated by the SPT rule.



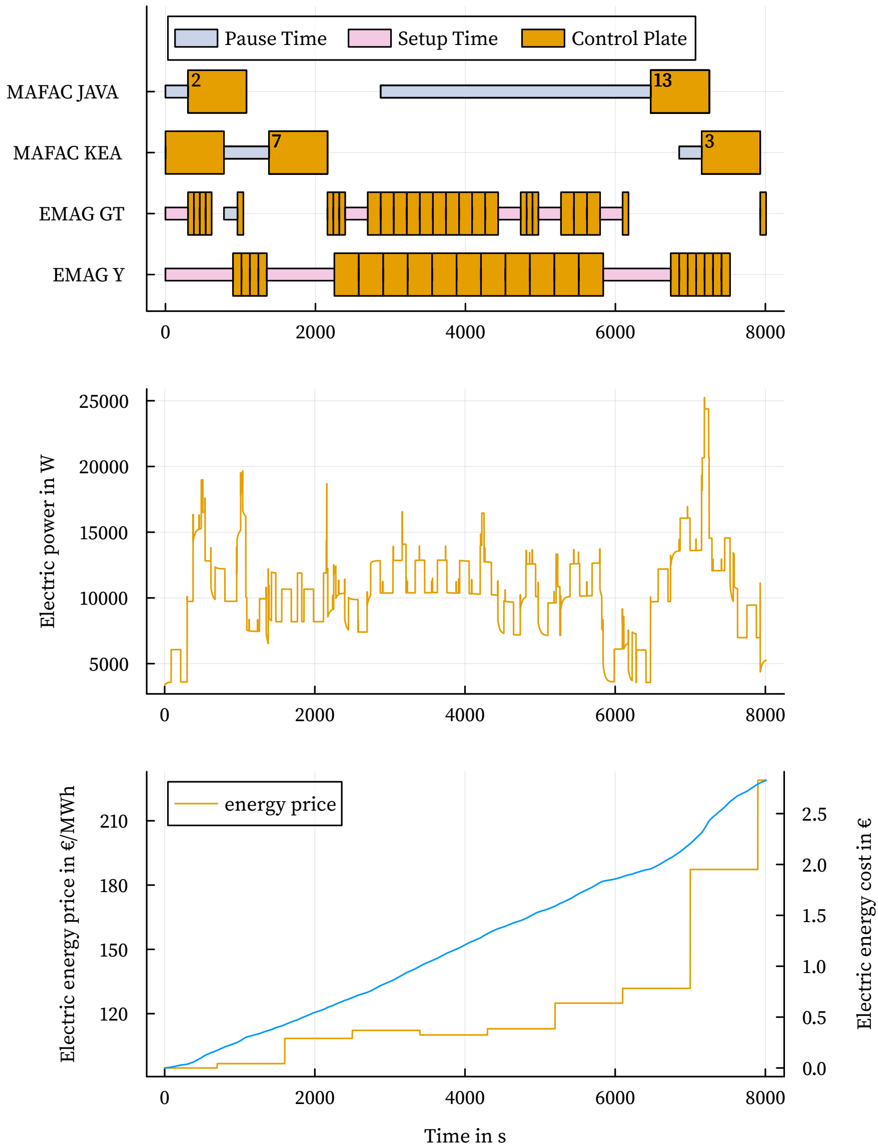
Best solution from experiment 2.



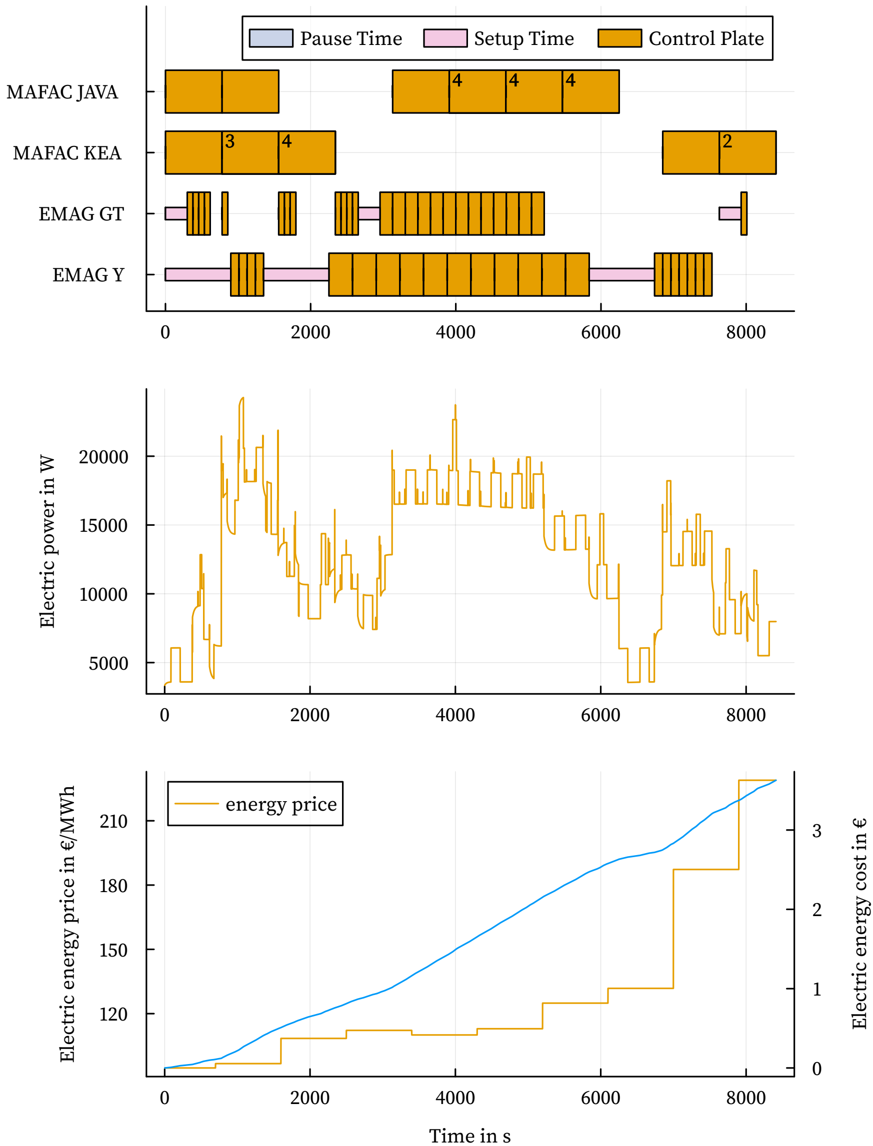
Schedule for experiment 2, generated by the SPT rule.



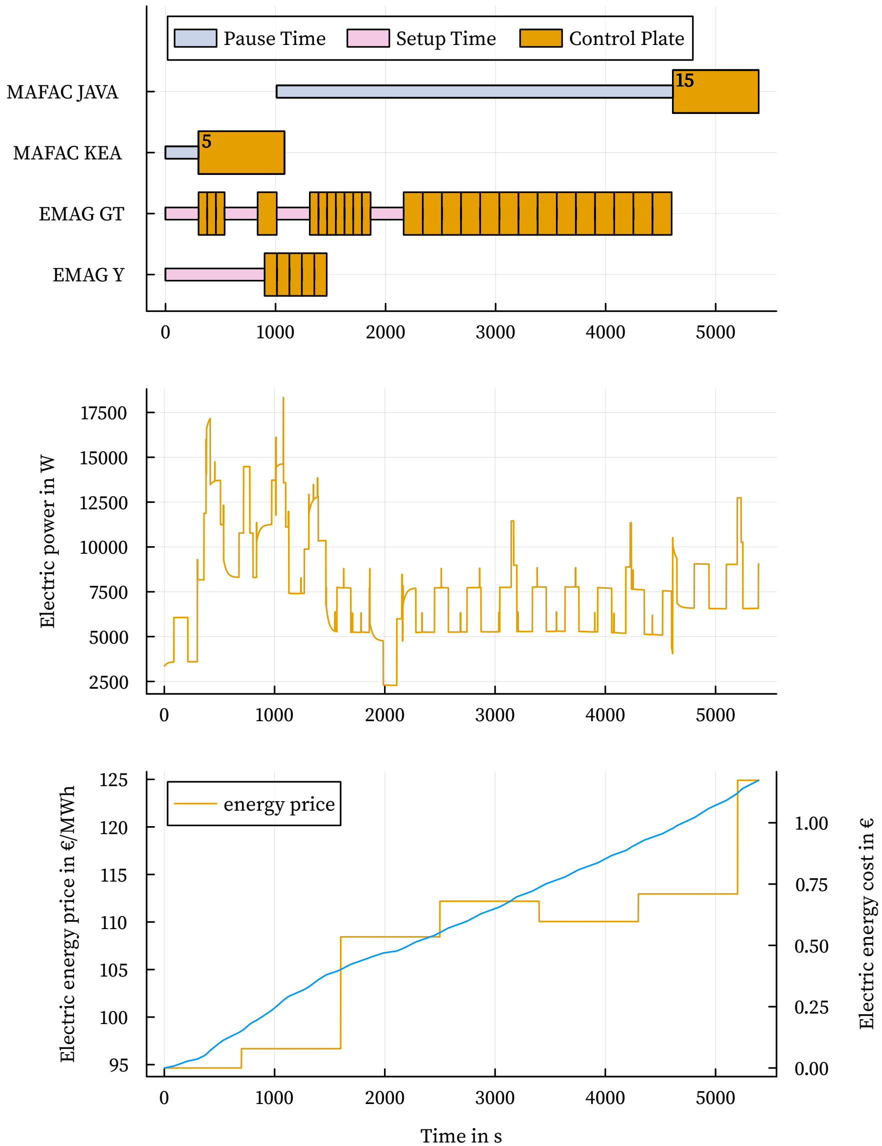
Schedule for experiment 3, generated by the SPT rule.



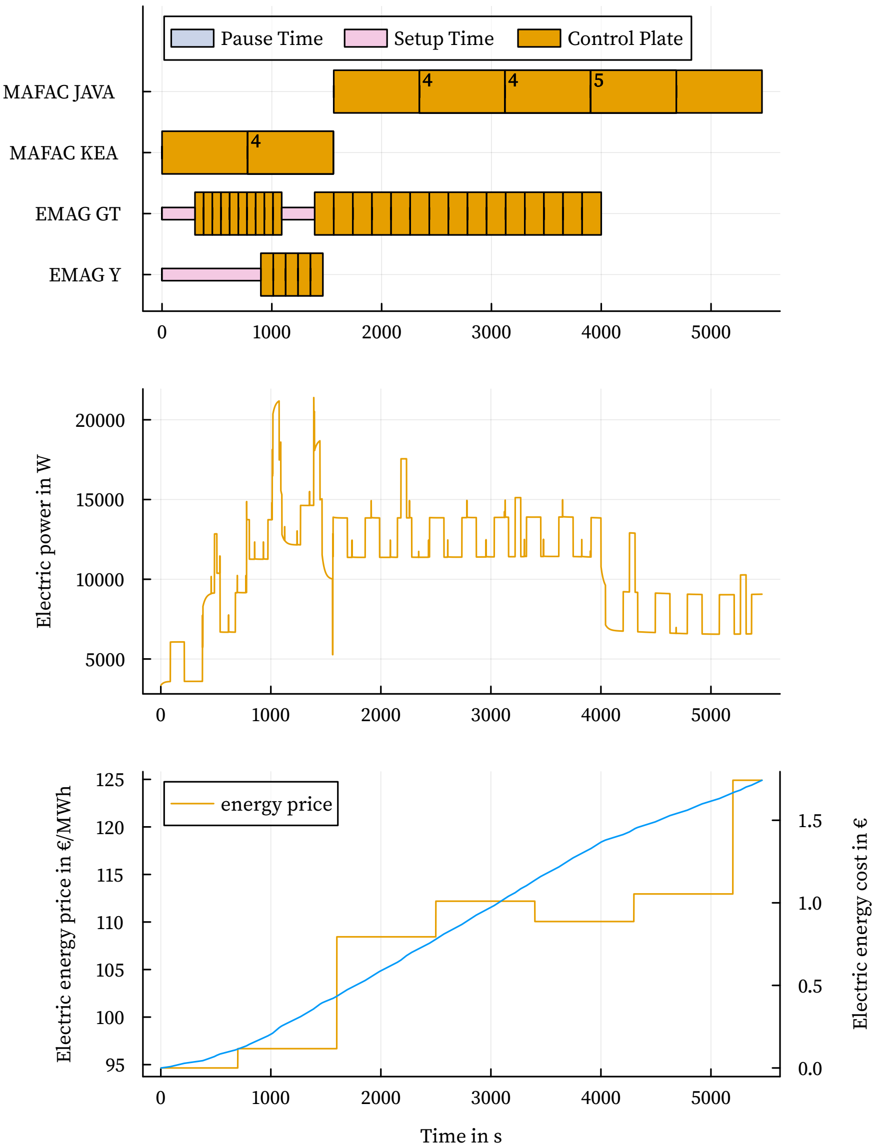
Best solution from experiment 4.



Schedule for experiment 4, generated by the SPT rule.



Best solution from experiment 5.



Schedule for experiment 5, generated by the SPT rule.