Taylor & Francis
Taylor & Francis Group



International Journal of Production Research

ISSN: 0020-7543 (Print) 1366-588X (Online) Journal homepage: www.tandfonline.com/journals/tprs20

Plug and schedule: scalable and modular architecture for integrated and dynamic scheduling based on graph neural networks and reinforcement learning

Yannik Hermann , Vincent Schroettle , Benedikt Kaupp , Martin Benfer & Gisela Lanza

To cite this article: Yannik Hermann , Vincent Schroettle , Benedikt Kaupp , Martin Benfer & Gisela Lanza (08 Oct 2026): Plug and schedule: scalable and modular architecture for integrated and dynamic scheduling based on graph neural networks and reinforcement learning, International Journal of Production Research, DOI: 10.1080/00207543.2026.2741380

To link to this article: https://doi.org/10.1080/00207543.2026.2741380

© 2026 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group.

Published online: 08 Oct 2026.

Submit your article to this journal

Article views: 326

View related articles

View Crossmark data

Full Terms & Conditions of access and use can be found at
https://www.tandfonline.com/action/journalInformation?journalCode=tprs20

INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH
https://doi.org/10.1080/00207543.2026.2741380

Taylor & Francis
by informa

OPEN ACCESS Check for updates

Plug and schedule: scalable and modular architecture for integrated and dynamic scheduling based on graph neural networks and reinforcement learning

Yannik Hermann, Vincent Schroettle, Benedikt Kaupp, Martin Benfer and Gisela Lanza

wbk – Institute of Production Science, Karlsruhe Institute of Technology, Karlsruhe, Germany

ABSTRACT

Manufacturing companies increasingly require scheduling approaches that integrate practical constraints such as workforce availability, maintenance, setup dependencies, energy objectives, and uncertainty, yet existing research offers limited support for transferring them into heterogeneous industrial data landscapes. This paper proposes a scalable and modular architecture for integrated and dynamic scheduling that combines an Asset Administration Shell-based middleware for standardised data integration, a heterogeneous graph representation of scheduling instances, a graph neural network-based State Modeler, and a Proximal Policy Optimisation-based RL-Scheduler coupled with event-based simulation. The architecture is validated on the production system of a German machinery manufacturer with 59 manufacturing resources and 39 employees and on public benchmark instances. On the industrial instances, the learned policy reduces the makespan against the mean of seven dispatching baselines by 2.93% to 6.76% across six evaluation cells covering small-part, large-part, and combined production under deterministic and stochastic machine failures, with statistical significance in every scenario. The energy-aware configuration preserves this makespan performance while significantly reducing the relative energy cost in most scenarios. The findings indicate that the proposed architecture provides a practically transferable pathway for configurable RL-based scheduling in heterogeneous production environments.

ARTICLE HISTORY
Received 1 April 2026
Accepted 12 September 2026

KEYWORDS
Scheduling; Asset Administration Shell; graph neural network; reinforcement learning; software-defined manufacturing

SUBJECT CLASSIFICATIONS
Advanced planning and scheduling systems; automatic control; dynamic scheduling

Introduction

In Germany, challenges such as elevated energy costs and strained supply chains have contributed to a prolonged phase of weak economic growth in recent years. In this light competitive strength becomes of utmost importance. Within manufacturing companies scheduling of production orders is one central driver of competitive performance. It formally refers to the allocation of individual operations of a production order onto the necessary machines being able to perform predefined manufacturing tasks in order to complete given orders while improving production, cost or other targets and, thus securing competitive competence (Rossit, Tohmé, and Frutos 2019). Typically, it primarily aims at optimising the monetary and economic side of business activities, for example through improving throughput, minimising delivery times or reducing production cost (Serrano-Ruiz, Mula, and Poler 2021). This perspective is also profound in literature with a multitude of approaches (for example: Kazemi, Mazdeh, and Rostami 2017; Monkman, Morrice, and Bard 2008; Oğuz, Sibel Salman, and Bilgintürk Yalçın 2010). Besides this purely economic focus, the realm of sustainability has become another important perspective in scheduling. This aspect is not only suitable to increase profits through 'green products' and concurrent higher willingness to pay by consumers (Tully and Winer 2014) but more so necessary to tackle disruptions and increasing scarcity in raw materials and energy (Akbar and Irohara 2018). The latter challenges are not only caused by continuously tightening legislation, in the European Union for example in the form of the Industrial Emissions Directive containing both stricter emission limits as well as profound sanction actions, but also large-scale exogenous factors like the Russian war on Ukraine which led to sharply increased and persistently volatile energy costs within the European Union.

In a practical setting at the partner machinery company of this work these challenges can also be identified and result in average throughput times of more than ten days, although the actual processing time per order amounts to only a few hours. As a consequence, assembly

CONTACT Yannik Hermann yannik.hermann@kit.edu wbk – Institute of Production Science, Karlsruhe Institute of Technology, Karlsruhe 76133, Germany

© 2026 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group.
This is an Open Access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. The terms on which this article has been published allow the posting of the Accepted Manuscript in a repository by the author(s) or with their consent.

lines are often not supplied in time from the company's own manufacturing site, which leads to delivery delays at the customer side.

Additionally, the respective manufacturing system is highly labour-intensive, with an operator-to-machine ratio of approximately one to one. However, operators are mainly occupied with setup tasks, which occur frequently due to the small lot sizes. In addition, they determine the order sequence in a decentralised way by selecting parts from the buffer of their workstation. This individual decision-making further increases variability in flow and workload distribution. Finally, uncoordinated maintenance activities also cause additional disruptions since they lead to unplanned machine downtimes and further delays.

Besides unclear order scheduling and recurrent delivery delays due to lacking scheduling the high level of energy consumption resulting from the intensive use of machine tools and peripheral equipment is also a critical challenge of the focus company. The resulting energy costs represent a significant burden with inefficient scheduling amplifying this issue because machines often are operated in non-optimal sequences, leading to avoidable peaks in energy demand.

Optimising not only the underlying scheduling, but also integrating the aforementioned other subproblems like energy-efficiency or operator planning not only offers a large potential to improve achievable solution quality, but is even more necessary in order to enable real-world applicability of generated solutions (Avalos-Rosales et al. 2018; Geurtsen, Adan, and Akçay 2024). In order to achieve such a multi-faceted optimisation the scheduling approach must take in available data and respect the actual production system to be optimised (Cowling and Johansson 2002). This leads to two new challenges: Firstly, it requires the integration of scheduling approaches into existing production systems with means to respect a variety of diverse system setups, data formats and interfaces. Additionally, new technologies must be employed not only to tackle the increased complexity from integrating the subdecisions but also in order to enable real-time adaption and the consideration of uncertainties of the scheduling problem through the use of sophisticated sensor technology or more efficient solving through Artificial Intelligence (AI)-driven solution algorithms (Monostori et al. 2016).

However, an in-depth literature review performed by the authors in section 2 shows that there is currently no approach identifiable in literature that combines all aspects (sustainability, workforce scheduling, maintenance scheduling, set-up times, real-time adaptability, consideration of uncertainties) into a holistic scheduling framework. As such the purpose of this paper is not only to further describe the importance of each individual element, but to combine them into a systematic landscape of functional units allowing not only the scheduling problem to be solved, but also to provide a way to uniformly integrate diverse system architectures in real companies by proposing a comprehensive and modular system architecture (Pinedo 2022). This fills a long-standing gap in literature (Parente et al. 2020).

Literature review

Flexible job shop scheduling problem

The issue of scheduling can be formally described as the sequence of operations o_j belonging to a job j within a set of jobs J on a set of machines $m \in M$ with M being the total set of all machines within the system (Rossit, Tohmé, and Frutos 2019). Each operation contains a distinct processing time and each job a distinct sequence of its operations (Pinedo 2022). This so-called *Job Shop Scheduling Problem* (*JSSP*) can be used to model a variety of production system layouts including more limited systems like single or parallel machine environments as well as so called *flowshop environments*. Additionally, it can be further modified into the *Flexible Job Shop Scheduling Problem* (*FJSSP*) by allowing for specific machine sets that can perform one specific job operation instead of only one distinct machine. This added flexibility allows production modelling largely without abstractions and is, thus, necessary when trying to produce an approach suitable for real-life scheduling tasks. An optimal solution for this problem is then a sequence of operations on each machine, that collectively minimise a set of target functions $Z_1, \ldots, Z_n$. These usually include but are not limited to classical aspects like the makespan *Cmax*, tardiness of jobs when including deadlines as well as cost aspects like the production cost (Dauzère-Pérès et al. 2024).

Resembling physical production systems the FJSSP also includes a set of common restrictions shared among authors. These include the limitation of parallel processing with each machine only being able to process a single operation at once and each operation only being able to be processed by one machine at a time (Blazewicz et al. 2019). Additionally, all machines and jobs are usually assumed to be available at the start of scheduling. A third restriction concerns preemption, with interruption of operations not being allowed when processing. The problem description can also be further expanded to include other aspects and restrictions depending on the specific system to be modelled (Dauzère-Pérès et al. 2024).

Solution approaches for the *FJSSP* are diverse, but are commonly separated into three distinct categories (Xie et al. 2019). The most fundamental way to identify

a solution is to model the problem as an optimisation problem and to utilise some solving algorithm to identify an optimal solution to the specific problem instance. However, these approaches struggle with problem sizes to be seen in real-world circumstances (Fattahi, Saidi Mehrabad, and Jolai 2007). As the FJSSP is a NP-hard problem, the computational complexity increases significantly with problem size preventing achieving an optimal solution of more complex instances in reasonable time (Ho et al. 2024). Therefore, other solution approaches are more commonly used. A second way of solving scheduling problems are heuristics. These aim at utilising comparably easy-to-calculate rules in order to devise both an assignment of operations to machines as well as a sequence of operations on a single machine (Ziaee 2014). Common rules are for example ranking operations and machines based on shortest or longest processing times (SPT / LPT) and then creating an assignment by choosing the highest ranked operation or machine respectively. Novel approaches extend on this rule-based assignment by identifying more complex, yet still computationally inexpensive rules for example through the use of genetic programming or improve the selection of suitable heuristic rules utilising AI (F. Zhang et al. 2024; Luo 2020).

A third stream of solution approaches focusses on so-called metaheuristics in order to improve solution quality compared to heuristics while simultaneously achieving lower computation times compared to exact solutions. In this domain, approaches utilising genetic algorithms are some of the most popular ones. These use nature-inspired methods to successively conduct exploration (to randomly search the problem domain) and exploitation (to improve upon identified solutions through various means) (Mirjalili and Gandomi 2023).

As mentioned before AI also becomes increasingly common in the domain of scheduling and has started to form a fourth category. Apart from adapting existing solution approaches with means of AI, for example through an intelligent selection of individuals within genetic algorithms, there are also approaches utilising for example deep learning in order to train a scheduling agent to choose the most suitable actions to minimise set goals like the makespan (Chen et al. 2020).

In order to test and compare the achievable solution quality of devised algorithms benchmark datasets can be used (Dauzère-Pérès et al. 2024). While there are some well-established datasets (Brandimarte 1993; Hurink, Jurisch, and Thole 1994), there also is a multitude of new proposals each targeting specific variants of classical scheduling problems (Kasapidis et al. 2021).

Scheduling architectures

The FJSSP is highly challenging due to its NP-hardness and the dynamic, uncertain nature of real manufacturing environments (e.g. machine breakdowns or the arrival of new orders) (Toptal and Sabuncuoglu 2010). The growing need for flexibility, adaptability, and scalability has driven the development of sophisticated scheduling architectures and mechanisms. Classical static scheduling approaches, which rely on fully known information in advance, often fail to respond effectively to workload fluctuations or unforeseen disruptions. Consequently, research increasingly focuses on architectures and methods designed for compatibility and extensibility in dynamic scenarios (Jalali Khalil Abadi and Mansouri 2024).

Several architectural forms exist. Hierarchical and decentralised architectures aim to combine advantages of centralised and decentralised control (Toptal and Sabuncuoglu 2010). In hierarchical systems, tasks typically pass through multiple schedulers arranged in layers, where higher levels address more abstract problems. Decentralised systems allow local decision-makers or agents to create schedules autonomously. A common variant is quasi-decentralised control, in which local schedulers make autonomous decisions while a superior manager agent resolves conflicts or sets global objectives. Such structures are widely used in job shops and flexible manufacturing systems (Jules and Saadat 2017).

Fuzzy logic is particularly suitable for situations with imprecisely defined or uncertain parameters, which frequently occur in dynamic environments. It supports modelling vague information and decision-making with relatively low computational complexity. By explicitly incorporating uncertainty, fuzzy approaches can improve robustness. Moreover, hierarchical fuzzy inference systems can reduce the complexity of rule design. Fuzzy systems have been successfully applied to scheduling-related tasks in distributed computing environments, for example to compute fitness functions, assign tasks to fog/cloud nodes, or cluster resources (Jalali Khalil Abadi and Mansouri 2024).

The review of the literature presented indicates a lack of holistic and integrated models for complex sociotechnical production environments. Although researchers have investigated individual aspects such as complex job routes, sequence dependent setup times, heterogeneous worker skills, multi resource requirements including machines, labour, tools, energy, buffers, maintenance, and inventories, as well as workforce factors, comprehensive approaches that represent and solve these requirements simultaneously and in a synergistic manner within a single model remain scarce (Grumbach et al. 2023).

Integrated flexible job shop scheduling

Besides the issue of formulating and solving the scheduling problem enhancing its practicability also forms an increasingly important aspect of scheduling literature. In order to structure the available input factors, target functions, and other related scheduling elements (henceforth referenced as scheduling components) a structured literature analysis was performed to ensure a holistic perspective including the central components of modern production paradigms. To reveal as much of the relevant information as possible, the three widely used databases 'Web of Science', 'Scopus' and 'EBSCOhost' are used to source literature. Identified results were first screened based on title and abstract, after which the remaining ones were analysed in more detail based on the full text. Based on predefined inclusion criteria like the integration of sustainability and a focus on (short-term) scheduling the identified works were successively filtered resulting in 241 papers being selected from the initial 12,115 results.

One important development to be identified, is the integration of multi-objective decision making. Instead of only optimising for a single target, the objective function often features multiple, potentially conflicting targets. In line with the increasing focus on sustainability these enhanced target functions also often include sustainability aspects besides purely economic objectives. Another development is an integration of other subproblems into the selection of optimal scheduling decisions. These range from smaller additions like the modelling of setup times up to the inclusion of highly important practical decisions like the concurrent scheduling of maintenance and employees. Both aspects, however, drastically increase the complexity of the scheduling problem and are, thus, mostly only included separately and not within one comprehensive framework. The following sections review the current state of the literature on the key scheduling components considered in this paper, namely workforce scheduling, maintenance, and energy.

Workforce scheduling

The realisation of flexible manufacturing systems, characteristic of Industry 4.0, requires sophisticated scheduling models that accurately account for both technical and human resources. The incorporation of human capital transforms the classical Flexible Job-shop Scheduling Problem (FJSSP) into sociotechnical variants, typically referred to as the Flexible Job Shop Scheduling Problem with Worker Flexibility (FJSSP-W) or the Dual Resource Constrained Flexible Job-shop Scheduling Problem (DRC-FJSSP). In these models, human resources are treated as critical, flexible constraints whose properties must be explicitly considered during the planning process (Hutter, Steinberger, and Hellwig 2025).

The integration of the workforce into the scheduling mechanism is defined by several key parameters. Fundamentally, for an operation to be processed, each machine requires a worker to be present. Crucially, the assignment of a worker to an operation not only fulfils this requirement but also additionally affects the processing times (Hutter, Steinberger, and Hellwig 2025). Research addressing complex industry requirements models specific factors such as flexible human workers and their associated worker capabilities. Furthermore, modern production environments often involve human-machine-collaboration and (partially) automated tasks which must be mapped in the DRC-FJSSP formulation (Grumbach et al. 2023).

For specialised operational demands, such as quality control or fault analysis, highly experienced specialists are required. These specialists, alongside tools, are modelled as flexible, required resources that can perform one or more operations depending on their unique properties within the Multi-resource Partial-ordering Flexible Job-shop Scheduling (MPF-JSS) problem. A recognised challenge in environments relying on specialists is that decisions made at the local level can be suboptimal because each expert possesses only a local view on the total set of operations to be performed (Francescutto, Schekotihin, and El-Kholany 2021).

To address the assignment and sequencing decisions under these constraints reliably and efficiently, advanced solution frameworks are utilised. These frameworks aim to solve the DRC-FJSSP by improving sequencing and assignment decisions, often through the integration of reinforcement learning (RL) with hybrid metaheuristics, such as Memetic Algorithms (Grumbach et al. 2023).

Maintenance scheduling

The integrated consideration of production scheduling and maintenance is increasingly recognised as essential in flexible job shop environments, yet maintenance is still frequently modelled in a simplified manner or treated separately from production planning decisions (Geurtsen, Adan, and Akçay 2024). Earlier work primarily represented maintenance through machine-availability restrictions, such as time-indexed availability calendars, blocking intervals, maintenance operations, or stochastic availability models (Sanlaville and Schmidt 1998).

More recent studies increasingly emphasise that production and maintenance should be optimised jointly rather than sequentially. In particular, simulation-based and stochastic FJSSP formulations show that the integrated determination of job sequences and maintenance

timing can improve schedule robustness and overall system performance, especially under uncertain machine availability and failure-prone operating conditions (Ali, Saif, and Ghasemi 2022). Practice-oriented contributions likewise report efficiency gains when maintenance activities are aligned with production-related idle periods or other opportunistic decision windows (Thörnblad et al. 2015). This development is further reinforced by recent joint-optimisation models that integrate machine assignment, sequencing, and maintenance decisions in a unified optimisation framework and consistently outperform sequential planning approaches (Y. Wang et al. 2024).

At the same time, first RL approaches indicate that learning-based methods are particularly promising for maintenance-aware scheduling in dynamic multi-machine systems. Another work demonstrates that heterogeneous GNN-based RL can learn graph-based resource and maintenance policies that outperform heuristic baselines (Z. Wang and Gombolay 2022). These developments suggest that effective maintenance-aware scheduling requires not only integrated decision models, but also state representations and control mechanisms that can capture dynamic system interactions, variable resource availability, and the longer-term consequences of local scheduling decisions.

Energy-aware scheduling

Energy-aware scheduling has attracted increasing research interest in recent years, with the literature showing strong overall growth and a particularly pronounced rise in mathematically oriented approaches (Gao et al. 2020). AI-based methods have also gained importance more recently, indicating that intelligent energy-aware production control is becoming an emerging research direction (Loffredo et al. 2024). For example a Q-learning-based cooperative meta-heuristic (CMAQ), combining a constructive heuristic, a bi-population framework, and a knowledge-based energy-saving strategy, to minimise makespan and total energy consumption in the distributed no-wait flow-shop scheduling problem with sequence-dependent setup times is proposed (F. Zhao, Jiang, and Wang 2023). Another work proposes a hybrid memetic algorithm (HMA) with Q-learning-based local search operators and an energy-saving operator to minimise makespan and total energy consumption costs in the distributed two-stage flexible job shop scheduling problem under time-of-use electricity pricing (Geng, Liu, and Wu 2024).

Integration approaches in real-world data landscapes

A further gap in literature is a clear analysis and integration of the system landscape and real production environment, in which the resulting scheduling programme shall operate (Oluyisola et al. 2022). This, however, is necessary for usable solution concepts with the resulting gap impeding or even preventing the transfer of resulting solution into practical situations (Framinan and Ruiz 2010). Romero-Silva, Santos, and Hurtado-Hernández (2024), thus, argue: 'one of the biggest challenges for production scheduling has been to transfer its body of theoretical knowledge to actual practical settings'. Not only do existing works lack methods and approaches on how to integrate the existing enterprise systems, but also do not take into account the possibilities new developments like Industry 4.0 can provide. One such development is concepts like the AAS (Bakopoulos et al. 2024).

Literature gap

The literature analysis shows that there is still no modular approach that integrates the key scheduling components

- Set-up sequences
- Maintenance scheduling
- Workforce scheduling
- Sustainability aspects such as energy consumption
- Uncertainties such as machine failures and target times

into a scheduling solution that also considers the underlying data landscape of companies and is scalable for factories of different sizes and process principles. The aim of this paper is, therefore, to first develop a scalable architecture that is able to handle the complexity of the underlying scheduling problem. The above-mentioned factors are then integrated into the architecture. In order to make it universally applicable the architecture is connected to a middleware enhancing easier deployment to practical settings.

Scalable and modular architecture for intelligent scheduling

Goal of the architecture

As seen from the previous paragraph scheduling still draws a lot of attention of scholars with a multitude of approaches. However, there is still fundamental gap between real-world requirements and theoretical examination, mainly due to a lack of necessary elements for an

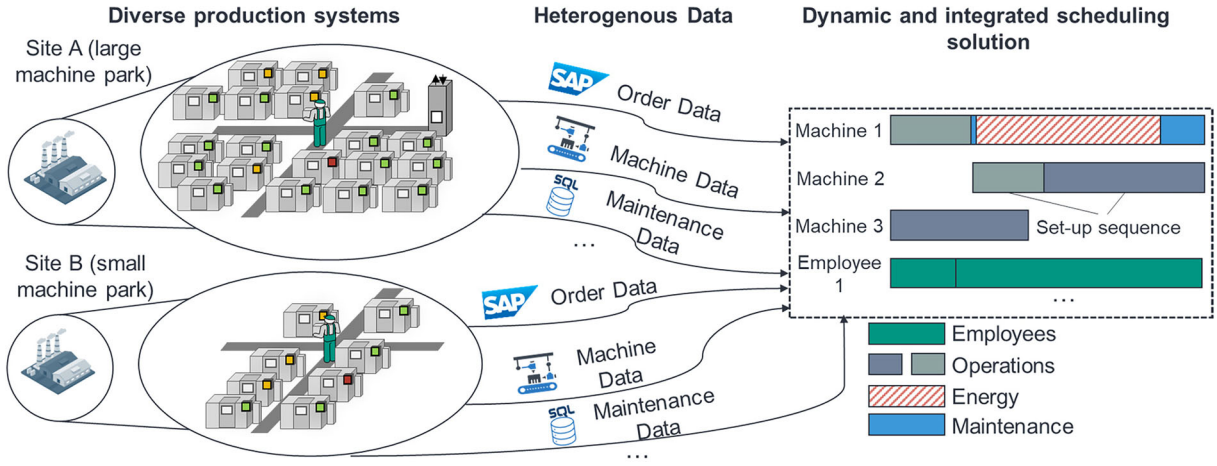


Figure 1. Fundamental motivational aspects for the presented methodology.

application within a company. This degrades the usability of many approaches as they may achieve high quality in benchmarks, but are not usable in real-world settings due to their perspective being limited to solution quality without regard to how data can be fed into the algorithms and how different parts of the programmes can and must interact with surrounding systems within companies like ERP, MES or smart machinery.

Therefore, the following methodology, visualised in Figure 1, first aims at developing a scalable and modular architecture that not only focuses on solving the integrated and dynamically the FJSSP, but sees the integration into the data landscape as an integral part of its conception. It shall be able to dynamically calculate an optimal selection of operations, machines, maintenance and workers even under uncertain environmental conditions and volatile energy prices. Additionally, the architecture shall be easily implementable for companies through an extensive use of standardisation.

Currently, these concurrent decisions, like selecting the best employee for a set-up task or scheduling maintenance inbetween orders, are taken sequentially due to the resulting complexity (Haït and Artigues 2009). However, integrating these decisions not only provides a better decision support for companies, but also offers the possibility of improving the overall solution quality (Geurtsen, Adan, and Akçay 2024).

In order to achieve the set goals, the methodology involves multiple steps. In a first step an initial architecture is developed. This initial architecture aims to provide a comprehensive solution approach for the FJSSP while simultaneously enabling its integration into existing software landscapes and production system configurations.

Following this, further parts of the architecture are implemented and validated. This step aims at extending the initial concept through integrating important aspects of scheduling leading to a more holistic view and therefore demonstrates the modular and scalable approach of the developed architecture. These important aspects feature two different subgroups. Enhanced input factors for decision making, modelling setup sequences and stochastic downtimes as well as machine availability enables a closer reflection of real production. A second group is built by the integration of employee and maintenance scheduling as well as energy costs in order to improve solution quality compared to a sequential optimisation through an interconnection and information exchange between subproblems (Jiang et al. 2022). The resulting integrative approach is able to identify a collectively optimised solution and offer companies a comprehensive support to integrate various decision problems in order to transfer theoretical power to practical use (Alemao et al. 2022). This is necessary for modern scheduling solutions (Dauzère-Pérès et al. 2024; Parente et al. 2020).

General overview of the architecture

The baseline architecture is divided into several modules as shown in Figure 2. This enables easier adaption not only to different companies and production system requirements, but also easier adaption of individual modules in case functional changes may be necessary in the future (Behrendt et al. 2023). Additionally, the necessary abstraction and standardisation associated with such modular architecture avoids the use of proprietary interfaces and connectors potentially alleviating the necessity to adapt the software and, thus, reducing the cost associated with using such software (Harjunkoski and Bauer 2014). As such the framework can be positioned close to so-called 'operation standard software' with standard components, clear documentations and lower cost than

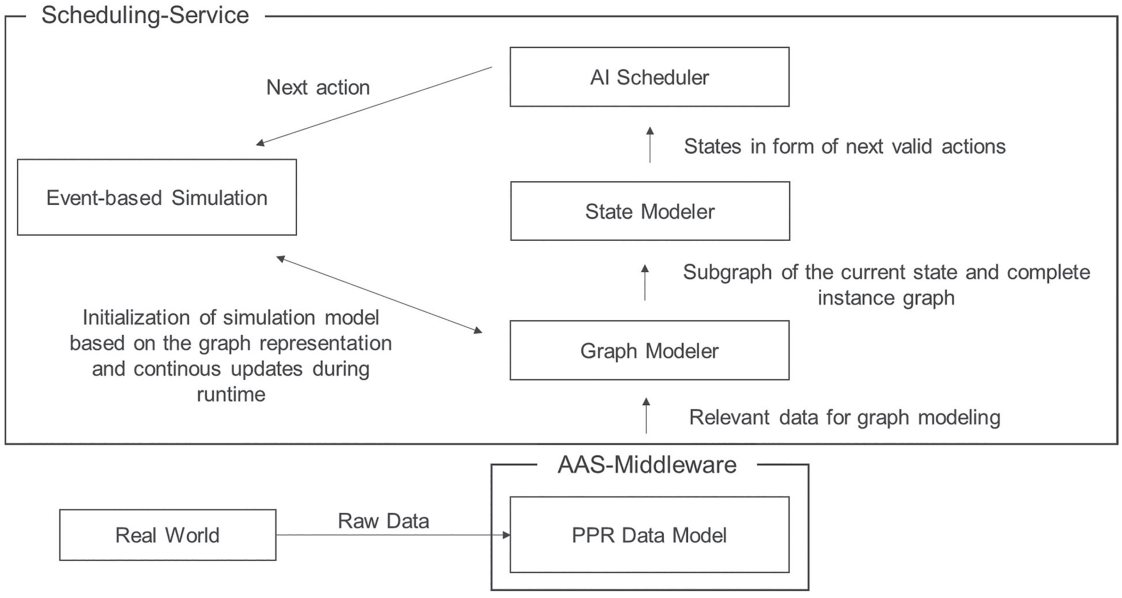


Figure 2. Modules of SMAS and its interconnections.

company-specific developments (Damsgaard and Karlsbjerg 2010; Venters et al. 2018).

A description of a preliminary version of the framework, herein referred to as SMAS (Scalable and Modular Architecture for Intelligent Scheduling), can also be found in (Hermann et al. 2025). The basic structure of the architecture consists of the *AAS-Middleware* for standardised modelling and representation of data; the *Graph Modeler (GM)*, which models the scheduling problem as a heterogeneous graph; the *State Modeler (SM)*, which translates the graph into embeddings using a graph neural network; the *RL-Scheduler*, which uses a RL agent to determine a probability distribution for the next possible actions; and the *event-based* (, discrete material flow) *simulation*, which maps the uncertainties and executes the actions. In the proposed architecture, the scheduling functionality itself is implemented as a service on top of the AAS middleware. This service-oriented design allows the scheduling modules to operate on a standardised data basis and is intended to ensure interoperability between heterogeneous systems, interfaces, and production environments.

The main purpose of the architecture is to determine a production schedule (in the form of operation sequences for each individual machine) through successively determining the most appropriate action at each decision point. Decision points are points in time on which an operation can potentially be assigned onto a machine, either determined through simulation or requests from the physical production system. The specific modules are further described in the following sections.

AAS-middleware

The physical production system and its digital counterpart, a standardised data model realised through (AAS), constitute the foundation of the architecture. The physical system serves as the starting point of the process and provides real-time data on job status, machine conditions, and other control-relevant information. As discussed above, insufficient integration of existing systems and data can significantly hinder, or even prevent, the implementation of novel concepts. For many companies, this lack of integration capability remains one of the main reasons why AI-based solutions have so far only been adopted to a limited extent (Oluyisola et al. 2022).

To address this issue, the approach builds on the concept of the AAS, which has been established in the context of the German Industry 4.0 strategy as a means of enabling efficient integration of heterogeneous machines and system components. As a vendor-independent framework, AAS provides a modular abstraction of machines, equipment, and other production assets and defines standardised requirements for accessing their services and data. Owing to this standardised structure, AAS has proven applicable beyond purely production-related use cases and offers a suitable basis for interoperable system design (J. Zhang et al. 2025).

In the present approach, the use of AAS enables the abstraction of the real production system in a form that is sufficiently standardised for a modular architecture while still allowing the formulation of application-oriented requirements for real factory environments. The AAS acts as a middleware in this scenario allowing heterogeneous physical components to offer unified digital

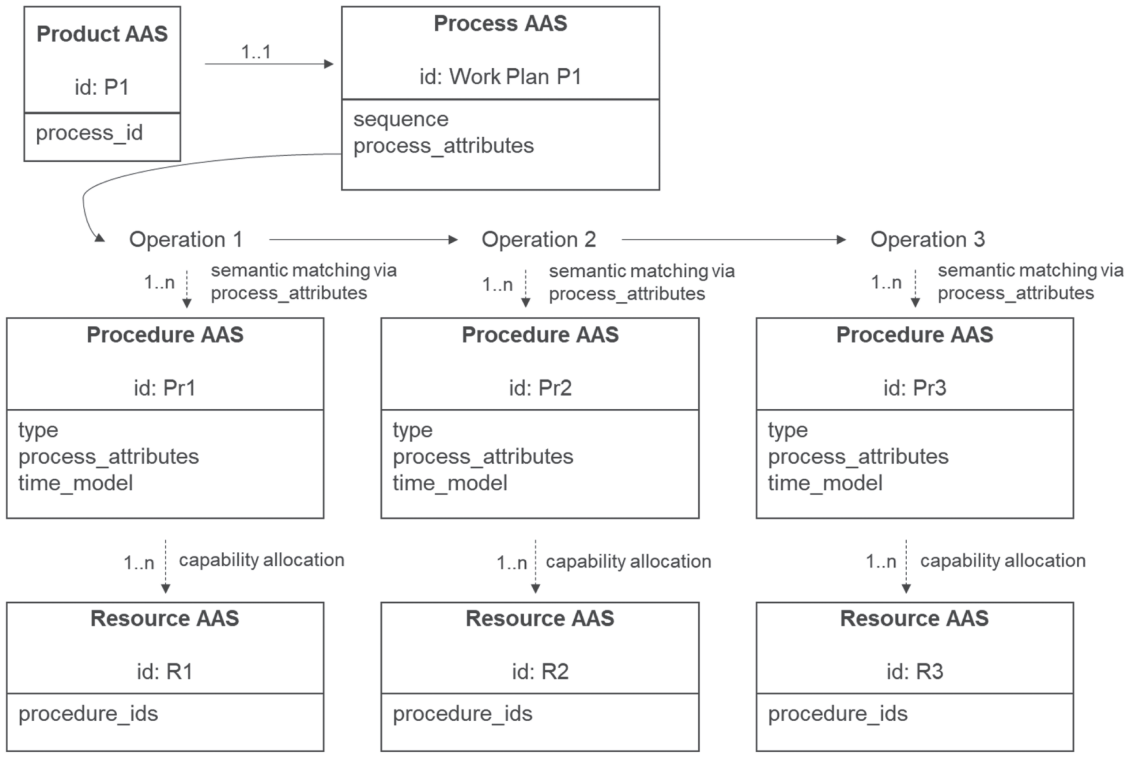


Figure 3. Used entities from SDM reference data model including its AAS components for creating scheduling problem instances.

interfaces. This separation supports system-independent analysis and development and thereby increases the flexibility of the overall approach while reducing restrictions with respect to the specific design of the underlying production environment.

Within the AAS the physical production resources and system parts are modelled using a combination of 'Product-Process-Resource'- or PPR-model as well as an implementation of the AAS from the package https://pypi.org/project/sdm-reference-model/. The PPR-model is a way of modelling production systems based on the distinction between products, processes and resources in order to manage modularity in modern systems. A product 'denotes a final or intermediate product type' with processes being used to represent changes to a product during production (Pfrommer, Schleipen, and Beyerer 2013). Resources then 'execute processes and handle products'.

This model is aligned with the ISA-95 and ISA-88 standards, which ensures compatibility with established hierarchical structures and production system modelling principles. ISA-95 defines the hierarchical structure of production systems, including levels such as enterprise, site, area, work centre, and workstation. ISA-88, in contrast, focuses on batch control and provides a foundation for process modelling (Vegetti and Henning 2014). Based on the standard frameworks mentioned above, the data points relevant for scheduling can now be integrated into a respective AAS, as shown in Figure 3.

The FJSSP instances, in form of diverse Product AAS (P), are then represented by a Process AAS (P), executable operations by Procedure AASs, and eligible resources by Resource AAS (R). Process-step identifiers in the Process AAS define the precedence structure, while their semantic requirements are specified through *process_attributes*. Suitable procedures are determined by semantic matching, and feasible resources are assigned explicitly through *capabilities.procedure_ids*. Hence, the resulting mapping is process step → procedure(s) → resource(s), which provides a compact and flexible AAS-based representation of FJSSP structure and resource alternatives.

Graph Modeler

The data provided by the AAS middleware is subsequently transferred to the GM, which introduces an additional abstraction layer between the physical system and its virtual representation. In close interaction with the simulation module, the GM transforms the scheduling problem into a heterogeneous graph that stores decision-relevant information in a compact and structured form. This representation provides a simple, extensible, and scalable basis for modelling the problem and consists of distinct node types for machines and operations, each with potentially different but systematically structured

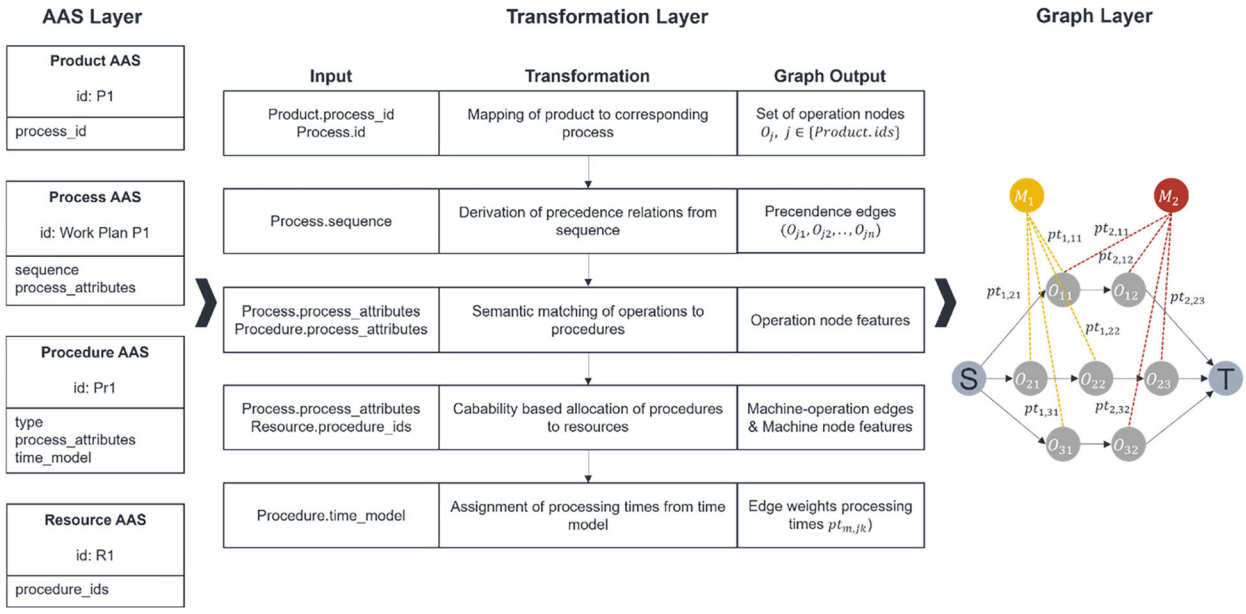


Figure 4. Transformation process from the data of AAS middleware layer to graph representations of scheduling problems that is performed in the GM module.

feature sets (C. Zhang et al. 2019; Rui et al. 2024; W. Song et al. 2023).

As illustrated in Figure 4, the transformation from the AAS-based data model to the scheduling graph is performed in several steps. First, product information is mapped to the corresponding process description, which enables the instantiation of job-specific operation nodes. Second, the sequence information contained in the process model is used to derive the technological precedence relations between operations. Third, process-specific attributes are semantically matched with procedure descriptions in order to assign relevant operation features. Fourth, the procedures are allocated to eligible resources based on capability information, resulting in feasible machine–operation relations and corresponding machine node attributes. Finally, processing times are assigned from the time model and transferred to the graph as edge weights. In this way, the semantically structured AAS representation is converted into a scheduling-specific graph structure that can be consistently used by the simulation, the SM, and the RL-Scheduler.

To satisfy the Markov property, the graph contains all information relevant for decision-making at a given scheduling instant. This is achieved through state augmentation, which extends static machine and operation attributes with dynamic information updated after each decision. These dynamic features include, for example, the current status of machines and operations as well as the remaining processing time of unfinished operations. Additional node features capture further system characteristics, such as reliability-related information. To ensure comparability across heterogeneous features, all numerical values are normalised using min–max scaling to the interval [0,1].

State Modeler

The next component of the architecture is the SM. A fundamental challenge in applying the machine learning and RL methods lies in their limited ability to process inputs of variable size (Liu, Piplani, and Toro 2022; L. Zhang et al. 2023). Although the graph representation already provides a unified description of the underlying scheduling problem, its size may still vary depending on the production system and the number of jobs considered. For subsequent processing, the graph must therefore be transformed into a representation of fixed dimension. Existing approaches to this problem often rely on selected summary indicators or on restricting the problem to a predefined maximum size (Bezoui et al. 2024; Felder et al. 2023; Kuhnle et al. 2021). However, such simplifications may reduce the expressive power of the representation and thereby impair decision quality (Y. Zhao, Luo, and Zhang 2024).

For this reason, the present work adopts a separate SM component. Its purpose is to encode the information contained in heterogeneous scheduling graphs into compact and informative vector embeddings while preserving as much problem-specific structure as possible. The SM is implemented as an adapted heterogeneous graph neural network based on (C. Zhang et al. 2019). The selected model architecture is particularly suitable in

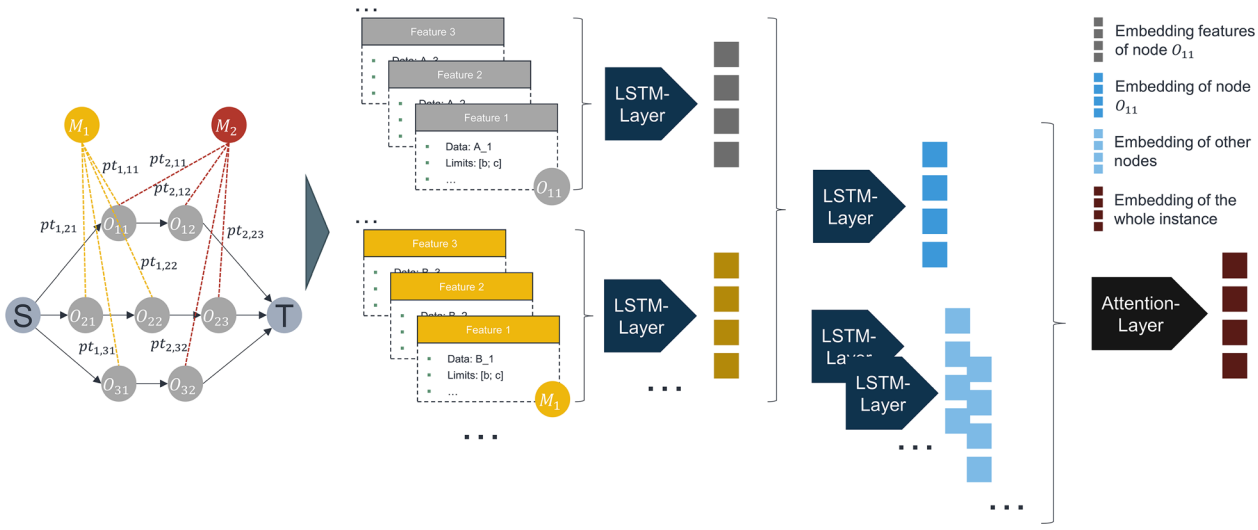


Figure 5. Transformation of the heterogeneous graph in embeddings through a HetGNN.

the present context because it explicitly supports heterogeneous node sets, here corresponding to machines and operations, and further allows self-supervised learning. As a result, the model can identify and exploit relevant structural relationships in the data without relying on manually specified target labels that may bias the learning process. To account more explicitly for the importance of processing times in scheduling, the model is extended such that these are incorporated into the weighting of node information during representation learning.

To obtain a common latent representation for heterogeneous node information, the different input types are first processed separately due to their potentially varying feature sets. Categorical inputs, such as node identities or discrete structural indicators, are mapped by embedding layers, while numerical attributes are projected into the same latent space by Long Short-Term Memory (LSTM) layers. After this pre-aggregation step, all generated vectors have a common dimensionality and can be processed jointly by the aggregation layers visualised in Figure 5.

In the baseline formulation from (C. Zhang et al. 2019), the representation of a node v is obtained by aggregating the representations of the elements $i \in C_v$ through bidirectional LSTM layers. Formally, the node embedding is given by

$$f(v) = \frac{1}{|C_v|} \sum_{i \in C_v} (\overrightarrow{LSTM}(f(x_i, \theta_i)) || \overleftarrow{LSTM}(f(x_i, \theta_i))),$$

where C_v denotes the set of node features or intermediate node representations associated with node v. $f(x_i, \theta_i)$ denotes the transformed input representation of element i and $||$ denotes vector concatenation. The bidirectional processing allows the model to capture dependencies among the aggregated elements in both directions, while the averaging operation produces a fixed-length embedding independent of the local neighbourhood size.

However, the aggregation mechanism is modified in the present work in order to better reflect the specific characteristics of the scheduling graph. In contrast to the graph settings considered in prior work, the problem instances addressed here are substantially smaller and structurally more specialised. Moreover, the neighbourhood relations in the scheduling graph are not only determined by topology, but also by edge-specific information that carries operational meaning. For this reason, the use of randomly sampled walk-based neighbours is replaced by a complete aggregation over all adjacent neighbours of a node. The aggregation is therefore reformulated as

$$f(v) = \sum_{i \in N_v} w_i (\overrightarrow{LSTM}(f(x_i, \theta_i)) || \overleftarrow{LSTM}(f(x_i, \theta_i))),$$

with

$$w_i = \frac{1}{g_{v,i}} * \frac{1}{\sum_{j \in N_v} \frac{1}{g_{v,j}}},$$

$$N_v = \{u \in V | AZ_{vu} = 1\}$$

Here, N_v denotes the set of nodes adjacent to node v, identified by the adjacency matrix AZ, and $g_{v,i}$ represents the edge weight between nodes v and i. Through this weighting scheme, neighbours connected via smaller edge weights exert a stronger influence on the resulting node embedding due to potentially lower impact of this node combination on final target functions to be minimised. This allows the SM to account not only for structural adjacency, but also for problem-specific differences in the strength or relevance of individual node

relations. As a result, the learned representation better reflects the operational characteristics of the scheduling problem than a purely topology-based aggregation.

The learning objective encourages similar representations for nodes that are structurally related, while representations of unrelated nodes are pushed apart. This contrastive learning principle enables the network to capture relational information from the graph without requiring explicit supervisory labels for each node. During training, the model parameters are optimised by backpropagation such that the latent representations increasingly reflect the structural patterns that are relevant for the later control task.

Overall, the proposed SM combines heterogeneous node attributes, graph topology, and edge-specific relational information in a unified neural representation model. By extending the original aggregation scheme towards a complete and weighted neighbourhood aggregation, the module generates richer and more problem-appropriate node embeddings. These embeddings form the basis for the subsequent decision-making process and thus represent an essential link between the graph-based state description and the learning-based control of the production system.

Event-based simulation

The simulation model enables a digital representation of the production system and thereby allows production runs to be simulated independently of the real system, in this case for training and inference runs of the developed RL agent.

A key design decision in this context concerns the definition of the control points at which scheduling decisions are generated. In the present work, a discrete event-based simulation (DES) was chosen, as the focus lies on discrete manufacturing environments. Such systems are typically characterised by discrete state transitions in practice, for instance from processing to idle states, which makes DES a natural and efficient modelling choice.

To make the setting used for RL explicit, the interaction between the discrete-event simulation and the RL-Scheduler is formalised as a Markov decision process (MDP) defined over the event timeline of the simulation. The MDP is characterised by its decision points, state representation, action space, transition dynamics, and reward signal, which are specified as follows.

Decision points. A scheduling decision is requested at the initial simulation time and, thereafter, only when an event occurs that may require an adaptation of the schedule. In the baseline setting these events comprise:

- the completion of one or more operations,
- machine failures
- the completion of repair processes

Between two consecutive decision points the simulation advances to the next triggering event, so that no computation is spent on time steps at which no decision is required.

State. At each decision point the environment exposes a fixed-dimensional latent state vector produced by the SM. It combines a static embedding of the overall scheduling instance, embeddings of the currently feasible actions, and the current simulation time stamp, as described in the SM chapter.

Action space. The action space consists of all feasible machine–operation assignments that satisfy the technological and operational constraints active at the current decision point, complemented by a no-operation alternative for cases in which no immediate (re-)assignment is beneficial.

Transition dynamics and uncertainty. Given the selected action, the environment applies the corresponding assignment and advances the simulation to the next event. Processing times are treated as deterministic, whereas disruptions are modelled stochastically: times to failure follow a Weibull distribution (MTBF) and repair durations a clipped Gaussian distribution (MTTR). The resulting next state is therefore governed jointly by the deterministic progression of assigned operations and by the stochastic breakdown and recovery behaviour of the resources.

Reward. After each transition the agent receives a scalar reward that combines a dense intermediate signal with a terminal schedule-level evaluation, as specified in the reward-function description below.

RL-Scheduler

The RL-Scheduler constitutes the central decision-making component of the proposed approach, as it determines which feasible scheduling action should be executed at each decision point. Its main purpose is to map the current production state to a suitable machine–operation assignment while accounting for the sequential and dynamic nature of the scheduling problem. In contrast to static scheduling procedures, the RL-Scheduler is designed to operate under continuously changing system conditions and to generate decisions online during the course of production. To this end, it is embedded in a RL framework that enables the agent to improve its policy through repeated interaction with the simulation environment.

From a functional perspective, the RL-Scheduler receives the current system state as input together with

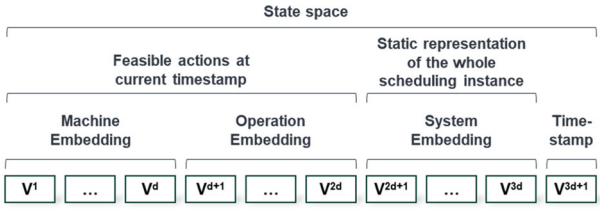


Figure 6. State-space representation of the proposed scheduling approach, combining embeddings of feasible actions at the current time step with a static embedding of the overall scheduling instance and the current time stamp.

the set of admissible actions, visualised in Figure 6. The state representation is provided by the upstream modelling components and contains the information required for decision-making at the respective scheduling instant and is always the same in its dimensionality due to the uniform vector magnitude. The action space consists of all feasible machine–operation assignments that satisfy the technological and operational constraints of the system. In addition, the action set may include a no-action alternative if no immediate reassignment is beneficial or required. It should be noted that the number of feasible actions may vary across decision points. To ensure applicability across different scheduling problems, the present approach therefore adopts a commonly used strategy for modelling variable action spaces (Kool, van Hoof, and Welling 2018; P. Song et al. 2025). Specifically, feasible actions are processed sequentially as inputs to the network, which computes an individual score for each action. This formulation ensures that action selection is restricted to valid decisions and that the agent always operates within the feasible solution space of the scheduling problem.

To process this information, the scheduler is implemented as a neural policy model following the PPO paradigm featuring an actor–critic architecture. This design is chosen because it combines direct policy learning with a value-based estimate of long-term decision quality, which improves optimisation stability in sequential control problems. More specifically, the actor network maps the input state to a numerical preference for each admissible action, whereas the critic network estimates the expected return associated with the current state. The actor is therefore responsible for action selection, while the critic provides an additional learning signal that supports stable and sample-efficient policy improvement.

Formally, let s_t denote the system state at decision point t, and let $\mathrm{A}(s_t)$ denote the set of feasible actions in this state. The actor network produces for each action $a \in \mathrm{A}(s_t)$ a scalar output z_t^a, which can be interpreted as an unnormalised action score or logit. These logits are transformed into a probability distribution over the feasible actions by means of the softmax function:

$$\pi_\theta(a|s_t) = \frac{\exp(z_t^a)}{\sum_{a' \in \mathrm{A}(s_t)} \exp(z_t^{a'})}, a \in \mathrm{A}(s_t),$$

where $\pi_\theta(a|s_t)$ denotes the probability of selecting action a in state s_t under policy parameters θ. This transformation serves two important purposes. First, it converts the raw network outputs into normalised probabilities, which allows the model to compare admissible alternatives on a common scale. Second, it enables stochastic action selection during training, which supports exploration of the action space and helps the agent avoid premature convergence to suboptimal scheduling policies. In application mode, the learned policy can then be used either stochastically or by selecting the action with the highest probability.

The critic network complements this mechanism by estimating the state value function $V_\phi(s_t)$, parameterised by ϕ, which approximates the expected cumulative return starting from system state s_t. This estimate is used during training to evaluate how favourable a state is under the current policy and to compute the advantage of selected actions. In the present scheduling setting, this estimate reflects how favourable the current configuration of machine loads, waiting operations, and remaining processing structure is under the learned policy. The critic thus provides a baseline against which the quality of a selected dispatching decision can be assessed. It is trained by minimising the discrepancy between predicted values and empirical return estimates by means of the loss

$$L_{CRIT}(\phi) = \hat{\mathbb{E}}_t[(V_\phi(s_t) - \hat{R}_t)^2]$$

where $\hat{R}_t$ denotes the return estimate associated with state s_t. This value-based baseline is used to compute the advantage

$$\hat{A}_t = \hat{R}_t - V_\phi(s_t),$$

which indicates whether a selected scheduling action performed better or worse than expected given the current state. In operational terms, the critic therefore helps distinguish between generally favourable states and genuinely beneficial scheduling decisions within those states.

The scheduler is trained using Proximal Policy Optimisation (PPO), which is adopted because of its robustness, good empirical performance, and stable update behaviour in high-dimensional sequential decision problems. PPO updates the policy by maximising a clipped surrogate objective that restricts excessively large policy changes between successive iterations. This is particularly beneficial in the present setting, as the scheduling

environment is highly dynamic and unstable updates may otherwise lead to deteriorating performance. The corresponding objective can be written as

$$L^{CLIP}(\theta) = \hat{\mathbb{E}}_t[\min(r_t(\theta)\hat{A}_t, clip(r_t(\theta), 1-\varepsilon, 1+\varepsilon)\hat{A}_t)],$$

with

$$r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)}$$

where $\hat{A}_t$ denotes the estimated advantage at time step t, θ_{old} are the policy parameters before the update, and ε is a clipping parameter that limits the update range. In the scheduling case, this restriction is beneficial because it prevents abrupt changes in dispatching behaviour between successive training iterations and thereby reduces the risk of unstable or erratic learning dynamics.

In addition to the clipped policy objective and the critic loss, training also includes an entropy-based regularisation term. The rationale behind this component is that the scheduler should not collapse too early to a narrow set of apparently favourable scheduling rules, especially in environments with heterogeneous jobs and flexible machine eligibility where the local preference for one machine–operation assignment may later prove inferior once congestion, blocking, or precedence-induced side effects emerge. The entropy bonus therefore supports broader exploration of alternative scheduling patterns before the policy becomes more concentrated by pushing the individual probabilities of actions to be more uniform. The entropy $\mathcal{H}$ of the action distribution is given by

$$\mathcal{H}(\pi_\theta(\cdot|s_t)) = -\sum_{a \in \mathrm{A}(s_t)} \pi_\theta(a|s_t) \log \pi_\theta(a|s_t),$$

and encourages the policy to maintain sufficient diversity in the selection of feasible actions during training.

Accordingly, the overall optimisation objective combines actor improvement, critic learning, and entropy regularisation, for example in the form

$$L_{PPO}(\theta, \phi) = L^{CLIP}(\theta) - \alpha L_{CRIT}(\phi) + \beta \hat{\mathbb{E}}_t[\mathcal{H}(\pi_\theta(\cdot|s_t))],$$

where α and β control the relative influence of value learning and entropy regularisation.

Training is carried out through repeated rollouts in the simulation environment. During each episode, the scheduler successively observes production states, selects feasible dispatching actions, triggers the resulting system transitions, and receives reward signals reflecting the quality of the emerging schedule. The collected trajectories are then used to estimate returns and advantages and to update the actor and critic over multiple optimisation epochs. This repeated interaction enables the scheduler to gradually learn decision patterns that are beneficial not only at the level of individual assignments but also with respect to the overall system objective.

Reward design represents a central element of the proposed RL approach, as it defines how the quality of scheduling decisions is communicated to the agent during training. In contrast to supervised learning, where desired outputs are directly available, the agent in the present setting must infer suitable dispatching behaviour from scalar feedback signals that evaluate the consequences of its actions. The purpose of the reward function is therefore not to replace the actual scheduling objective, but to provide a dense learning signal that guides the agent towards policies that improve system performance over time. In dynamic production scheduling, this design task is particularly important because the impact of an individual dispatching decision often only becomes visible through its interaction with subsequent decisions and later system states. As already noted, the agent seeks to maximise the cumulative reward collected over the course of an episode. This objective can generally be expressed as

$$R_{episode} = \sum_{t=1}^{T} \gamma^{t-1} R_{a_t} + \gamma^{T} R_{final},$$

where R_{a_t} denotes the reward assigned to action a_t at decision point t, R_{final} denotes an additional terminal reward at the end of the episode, T is the number of decision points, and γ is the discount factor. This formulation reflects the sequential structure of the scheduling problem by combining immediate decision-level feedback with a final assessment of the resulting schedule. The discount factor determines the relative importance of near-term and future rewards and thereby influences how strongly the agent values later consequences of current scheduling choices.

Overall, the RL-Scheduler provides a flexible and adaptive decision mechanism for dynamic production scheduling. By combining a learned state representation with a PPO-based actor–critic architecture, the module is able to evaluate feasible alternatives, generate probabilistic action preferences via softmax, and improve its policy through repeated interaction with the environment. It thus forms the core control component of the proposed approach and establishes the connection between the graph-based state description and the final scheduling decisions executed in the system.

Enhancing SMAS

As outlined above, SMAS is intended to serve as a generic, modular, and scalable architecture for scheduling applications. To demonstrate this flexibility, the following sections integrate setup dependencies, workforce scheduling, maintenance scheduling, and additional system-specific aspects into the proposed architecture, which results in the problem modelling shown in Figure 7. The integration of these factors results in the creation of an integrated schedule.

Set up sequencing

To account for setup dependencies within SMAS, the existing information flow is extended by machine and operation related setup-groups and integrated into the established processing sequence of the architecture. At the level of the AAS-based data representation, this includes the current setup state of each machine, the setup requirement associated with an operation, and the corresponding setup time. The reference model already includes the procedure type SETUP, which allows setup activities to be modelled as dedicated procedures. In addition, setup-related distinctions can be encoded semantically through process_attributes, using the existing structure of attribute_carrier, general_attribute, predicate_type and attribute_value. Resource-specific executability can then be linked via procedures_ids in the capabilities model.

These attributes are transferred to the GM together with the remaining technological and operational information and represented as additional machine- and operation-related node features within the heterogeneous scheduling graph. As a result, setup-related information becomes part of the same graph-based representation that already captures precedence relations, machine eligibility, processing times, and dynamic status information.

The SM processes this augmented graph together with the remaining node attributes and edge relations to derive the state representation at each decision point. Since the current setup state of each machine is included as dynamic information, the resulting state contains the information required to determine whether a machine–operation assignment involves an additional setup process. More specifically, setup time is added whenever the current setup state of the selected machine differs from the setup requirement of the assigned operation. This additional time component is then considered together with the remaining technological and operational constraints during the determination of feasible actions.

Within the event-based simulation, setup is represented explicitly as a process phase preceding the actual machining step. During this phase, the corresponding machine is blocked for the required setup duration before processing can start. After completion of the setup process, the setup state of the machine is updated in the graph and thereby made available for subsequent scheduling decisions in the same way as the other dynamic system attributes.

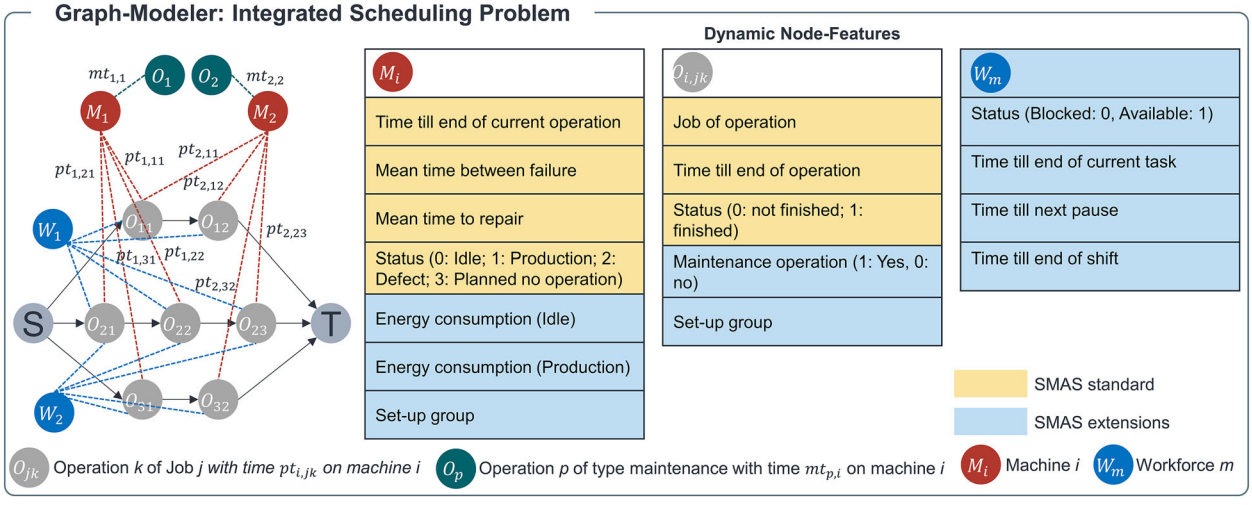


Figure 7. Graph-based modelling of the integrated scheduling problem, comprising the heterogeneous problem graph (left) and the dynamic node features of operation, maintenance, machine, and workforce nodes (right), distinguishing SMAS standard features from the proposed extensions.

Maintenance scheduling

Maintenance can be modelled using existing procedure-related elements of the SDM reference model. In particular, the enumeration of procedure types already includes MAINTENANCE, allowing maintenance activities to be described as dedicated procedures. Their duration can be specified through the existing TimeModel, while current resource states can be represented via ResourceStatus using status and status_start. To support maintenance-aware scheduling, the reference model is extended at the Resource level by a dedicated reliability-related submodel. The existing model already provides general resource information, executable procedures, and a current resource status through ResourceInformation, Capabilities.procedures_ids, and ResourceConfiguration.status, while maintenance activities themselves are already represented via ProcedureTypeEnum.MAINTENANCE and a corresponding TimeModel. The extension therefore adds explicit reliability indicators, in particular mean_time_between_failure and mean_time_to_repair, and may further include distribution-based parameters for failure and repair processes. In this way, the existing procedure-oriented maintenance representation is complemented by a condition- and reliability-oriented description that can be used directly for maintenance planning, simulation, and scheduling.

To enhance the FJSSP within a realistic simulation environment, maintenance operations are included to build a stable learning environment. As maintenance operations are essentially treated as jobs, they are modelled in a consistent way to maintain system integrity.

An additional binary feature was created within the operation node to separate order-related tasks from actual maintenance. This feature is an essential part of the feature engineering for the HetGNN to encode the specific priority and rigidity of these tasks. In this scenario, regular characteristics must be adapted, as maintenance cannot be assigned to alternative machines unlike normal operations. Consequently, the maintenance flag is checked during initialisation, and if active, the logic for finding alternative machines is skipped to enforce a fixed resource allocation. Additionally, these tasks always require the presence of one worker. The setup time from normal operations acts as a placeholder for the maintenance duration, as this specification is independent of quantity, while the process time is set to a dummy value of one second to ensure simulation stability. Adjacent to this, upper limits ensure that the simulation is not negatively influenced by outlier data.

Energy awareness

Energy-related information can be represented using existing elements of the SDM reference model. At the resource level, the energy_interfaces submodel already provides attributes such as nominal_power_consumption, peak_power_consumption, and standby_power_consumption, which can be used to describe machine-related energy demand.

To account for energy-related aspects within the baseline SMAS architecture, the GM is extended such that the scheduling graph captures not only technological and resource-related information, but also energy-relevant system characteristics. While the basic heterogeneous graph still consists of machine and operation nodes, the associated feature sets are enriched by energy-related attributes. In particular, machine nodes include the features of processing-related and idle-state energy demand, whereas edge attributes continue to represent machine-specific processing times and thereby preserve the relation between temporal and energy-relevant execution characteristics.

The state description is extended by additional information that reflects the current system-wide energy context. The state-vector structure is extended by an additional energy-prediction vector. This vector provides external information relevant to energy-aware scheduling, such as predicted price developments or expected energy availability over future periods in form of an energy cost vector with respect to the time.

The simulation environment is likewise extended to reflect the energy-aware decision context. As in the baseline architecture, the production system is modelled as a discrete-event simulation, but the event logic is augmented by energy-related state changes. In particular, a new decision point is added:

- changes in the external energy situation, due to time-varying electricity prices

The reward function of the RL-Scheduler is extended by an additional energy-cost component in order to represent the scheduling problem as a bi-criteria optimisation task. In this way, the agent is encouraged to learn decision policies that not only improve conventional production objectives, but also respond to time-dependent energy conditions in form of the energy costs of predicted schedules.

Workforce scheduling

For workforce-aware scheduling, the reference model is extended at the Resource level by a dedicated temporal availability submodel for human resources. While the

current model already contains general resource descriptors such as name, production_level, resource_type, and status, and links executable activities indirectly through procedures_ids, it does not yet represent workforce-specific temporal information explicitly. The extension therefore introduces attributes such as shift_start, shift_end, break_start_times and break_end_times. This preserves the existing resource-centric structure of the reference model while making it directly applicable to workforce-integrated scheduling.

A key extension of the GM is the inclusion of workforce nodes in the instance graph alongside machine and operation nodes. In addition, edges between operation and workforce nodes are introduced to encode qualification relationships and setup-related information. This graph extension provides the structural basis for ensuring that the RL agent only considers feasible combinations of operations, machines, and qualified workers. To consistently integrate workforce-related information into the graph representation, the adjacency and neighbourhood structures are further adapted, including the computation of Katz centrality and the derivation of neighbourhood lists. As a result, workforce-specific relations are incorporated not only formally but also functionally into the graph-based scheduling model and can be treated equivalently to machine-related information in subsequent modelling and decision-making steps.

In the extended target system, the SM processes workforce-specific features in addition to machine and operation information and integrates them into the neural aggregation mechanism. To this end, dedicated embedding layers for workforce nodes are implemented within the HetGNN. At the same time, the neighbourhood logic is extended such that workforce-related information directly contributes to the embeddings of operations and thereby becomes part of the decision basis. Particular attention is paid to the consistency of data fusion. Workforce, machine, and operation information are processed jointly in a unified representation without introducing redundancy or losing relevant information. The result is a comprehensive state description that captures all relevant resource types and enables the agent to make scheduling decisions that explicitly account for workforce dependencies.

In the extended system, the simulation model is augmented by additional functions for modelling human resources. These include, in particular, the management of workforce states such as breaks, shift endings, and worker availability, all of which are dynamically updated during the simulation. A central function of the simulation is the synchronisation of machine and workforce availability. An operation can only be scheduled if both resource types are available at the same time and the assigned worker possesses the required qualification. The event logic is further extended by the mechanisms of shift changes and staggered break structures, which improve the realism of the production dynamics. In addition, the reward design is adapted to avoid undesirable states such as workforce-related deadlocks, without explicitly incorporating broader social criteria such as worker satisfaction or fatigue into the optimisation objective. Overall, the simulation environment captures the close interdependence between machine and workforce capacities and thus enables a more realistic assessment of complex scheduling decisions.

The original binary assignment mechanism of the RL-Scheduler between operation and machine is thus extended to a ternary decision structure comprising operation, machine, and workforce. To process this enlarged action space, the underlying actor–critic networks are functionally adapted, in particular by integrating additional workforce embeddings into the decision process. The objective is to learn policies that systematically account for workforce-related constraints while simultaneously avoiding infeasible allocations and deadlock situations.

Integration of enhanced SMAS at machinery manufacturer

A real-world industrial case is considered to demonstrate the applicability and potential of the proposed dynamic scheduling approach. The case company is a German medium-sized machine tool manufacturer that produces woodworking machinery. The overall production is divided into manufacturing and assembly, with the focus of this study on the manufacturing domain.

The manufacturing system can be described as a flexible job shop environment with heterogeneous resources. The shop floor consists of 59 manufacturing machines, like automatic sawing machines, welding stations, blasting machines, CNC machines for large-scale components such as machine beds, 3-axis milling machines, 5-axis milling machines, turning machines, sheet-metal processing machines, and a painting line. In addition to these machine-based operations, manual finishing steps such as deburring are required and must be integrated into the scheduling process. On average, about 400 operations are executed per shift. Around 70 percent of production orders involve small batch sizes between one and five units, with strongly varying processing times, setup times, and operation sequences. The workforce of the manufacturing system is made up of 39 employees, each with different qualifications.

Integration of relevant data in middleware

To illustrate the data integration process, one representative order is used throughout the example. In the order data from SAP, order 8002145130 contains two operations. The first operation lists three alternative CNC machines for large components, namely Uniport 7000, Uniport 6000 and Uniport 6000 Pendel with an equal set-up and processing time. The second operation has one executable alternative machine for painting large components with a respective set-up and processing time. In the SDM reference structure, this order is first represented as a Product AAS with the identifier product_8002145130. Its routing is represented by the corresponding Process AAS process_8002145130, whose ordered process sequence consists of the two logical operations op_8002145130_10 and op_8002145130_15. The product-specific fields job, material number, and the ordered operation numbers from the operation field therefore define the basic Product–Process structure of the instance.

The crucial transformation from the Process AAS to the Procedure AAS is performed at the level of each operation in the process sequence. For every operation node contained in process_8002145130, the entry of the order data alternative machines is parsed first. In the present example, operation op_8002145130_10 generates three executable Procedure AAS objects: procedure_8002145130_10_uniport7000, procedure_8002145130_10_uniport6000, and procedure_8002145130_10_uniport6000_pendel. Each of these procedures inherits the same logical operation identity from the Process AAS, but differs in the assigned resource and therefore represents one concrete execution alternative. The machine-specific values are transferred into the procedure-level time_model, with their setup time and processing time as explicit procedure parameters. Analogously, operation op_8002145130_15 generates the single procedure procedure_8002145130_15_paint_job_large. In this way, the Process AAS defines the logical job structure, whereas the Procedure AAS instantiates the executable alternatives of each process step.

The machine resources are then created from the workplace and MES information. For the present case, the Resource AAS objects resource_machine_uniport7000, resource_machine_uniport6000, resource_machine_uniport6000_pendel, and resource_paint_job_large are instantiated. Their procedural capabilities are represented through procedures_ids, for example resource_machine_uniport7000.procedures_ids = [procedure_8002145130_10_uniport7000] and resource_machine_paint_job_large.procedures_ids = [procedure_8002145130_15_paint_job_large]. The MES records are used to populate execution history and resource state information. In addition, maintenance-related parameters are attached to the machine resources via the reliability extension. For example, the MES data reports regarding the MTBF/MTTR for an Uniport machine the values Mean Time Between Failure = 51.553663 h and Mean Time to Repair = 5.026473 h, which are assigned to the corresponding resource as mean_time_between_failure and mean_time_to_repair.

The workforce resources are created directly from a qualification matrix excel sheet. Each row in that sheet becomes one human-resource Resource AAS. Accordingly, a respective resource resource_worker_WF12 is created and receives the executable procedure references procedure_8002145130_10_uniport7000, procedure_8002145130_10_uniport6000, and procedure_8002145130_10_uniport6000_pendel in its procedures_ids.

Finally, the graph instance is derived directly from these AAS objects. The ordered operations from the Process AAS become the operation nodes op_8002145130_10 and op_8002145130_15, connected by one precedence edge. The machine resources become machine nodes, and the workforce resources become workforce nodes. Machine–operation edges are generated from the procedure assignments of the machine resources, and workforce–operation edges are generated from the worker-specific procedures_ids. For the present case, this yields three feasible machine–operation edges for operation 10, one machine–operation edge for operation 15, three workforce–operation edges from resource_worker_WF12-type resources to operation 10, and one workforce–operation edge from resource_worker_WF31-type resources to operation 15. In this way, the raw order file defines the logical routing and executable alternatives, the MES data provide execution- and state-related machine information, the reliability workbook adds maintenance-related parameters, and the qualification matrix generates the workforce resources required for workforce-aware scheduling.

The energy model reuses the machine-level attributes already provided by the EnergyInterface Submodel Element Collection of the SDM reference model, namely nominal, peak, and standby power consumption. For scheduling, each machine is characterised by a productive power demand that applies while an operation is being processed and by a reduced idle or standby demand that applies during non-productive states such as maintenance. External conditions are represented by a time-resolved electricity-price vector that specifies the price for each period over the considered horizon and is exposed to the agent through the energy-prediction vector of the state. The energy cost of an individual action is obtained by weighting the power demand of the selected

machine over the duration of the action with the electricity price prevailing in each covered period, i.e. by summing the price over the action window and multiplying it by the machine's power demand. The total energy consumption of a completed schedule is the sum of these action-level contributions over the entire rollout, so that both the timing of an action and the choice of executing machine jointly determine the realised energy consumption.

The mapping is set up once and then only needs to be expanded if new factors need to be taken into account for scheduling.

Training and evaluation State Modeler

For the industrial case considered in this study, the SM had to encode production states of a highly heterogeneous manufacturing environment rather than abstract benchmark graphs. In practical terms, the model had to distinguish between scheduling situations in which several interchangeable resources were available for one operation and situations in which downstream processing was effectively tied to a single bottleneck resource. This becomes particularly relevant in the present use case because many machining operations can be routed flexibly across several eligible machines, whereas downstream steps such as painting are much more restrictive. The SM therefore had to learn embeddings that preserved exactly those structural asymmetries that drive scheduling difficulty in the partner company's production system.

From a feature-design perspective, the graph representation was configured to reflect the operational logic of this environment as closely as possible. According to Figure 7, the graph contained seven machine features, five operation features, and four workforce features. The machine-side representation therefore captured not only generic status and timing information, but also dynamic characteristics relevant for integrated scheduling, whereas the operation-side representation encoded the technological and temporal properties required for sequencing decisions. On the workforce side, four features were used to describe worker availability, time until completion of the current task, time until the next pause, and time until the end of shift. This is important in the case setting because the workforce is not merely an additional annotation, but a real feasibility constraint. An operation can only be executed when a qualified worker is simultaneously available for the selected resource.

The selected graph-construction settings further underline this case specificity. The message-passing horizon was restricted to one hop, which is consistent with the operational requirement of capturing the most relevant local interactions at a scheduling decision point rather than diffusing information over unnecessarily large graph distances. At the same time, graph centrality and neighbourhood construction were guided by Katz-based relations with an alpha value of 0.005 and beta equal to 1, while the inverse graph variant was activated and self-loops were retained. For neighbourhood reduction, the graph builder kept the top two neighbours for machine, operation, and workforce relations, respectively. This compact neighbourhood design is meaningful in the present use case because decision quality depends particularly strongly on the most relevant local alternatives, for example the most plausible substitute machines for a machining operation or the most relevant qualified workers for a setup-intensive task. The graph input was capped at 1,024 nodes, which provided sufficient capacity for the industrial instances considered while maintaining computational stability during training.

The neural architecture of the SM was intentionally compact but tailored to the complexity of the partner company's data structure. In the final reported setup, the model used two graph-neural layers, a hidden feature dimension of 40, and an output embedding dimension of 40. Dropout was set to 0.2 and layer normalisation was enabled. The encoder considered six neighbouring nodes during state construction, and the mini-batch size was fixed at 80. Training was performed for 2,000 optimisation iterations with a learning rate of 0.001, while 300 samples per graph were generated both for training and evaluation. The evaluation interval was set to 25 iterations. Operationally, this architecture means that every production state, regardless of the exact number of active jobs, feasible machine alternatives, or workforce-resource links, was compressed into a fixed 40-dimensional latent representation that could subsequently be processed by the RL-Scheduler. This compactness was not chosen for methodological elegance alone, but because the industrial scheduling environment requires a representation that is expressive enough to preserve routing flexibility and resource dependencies while still being stable enough for repeated downstream RL training.

A further point that is relevant for interpreting the results concerns the dataset split. The final run reported for the present paper is the dedicated optimal-split experiment documented in the results workbook, which used 14 training instances in form of orders of a whole day and 4 held-out evaluation instances also in form of orders of a whole day, corresponding to an approximately 80/20 split over 18 total instances. This split is particularly meaningful in the present context because the industrial instances differ substantially with respect to routing structure, machine flexibility, and the resulting congestion patterns. A useful SM therefore had to generalise

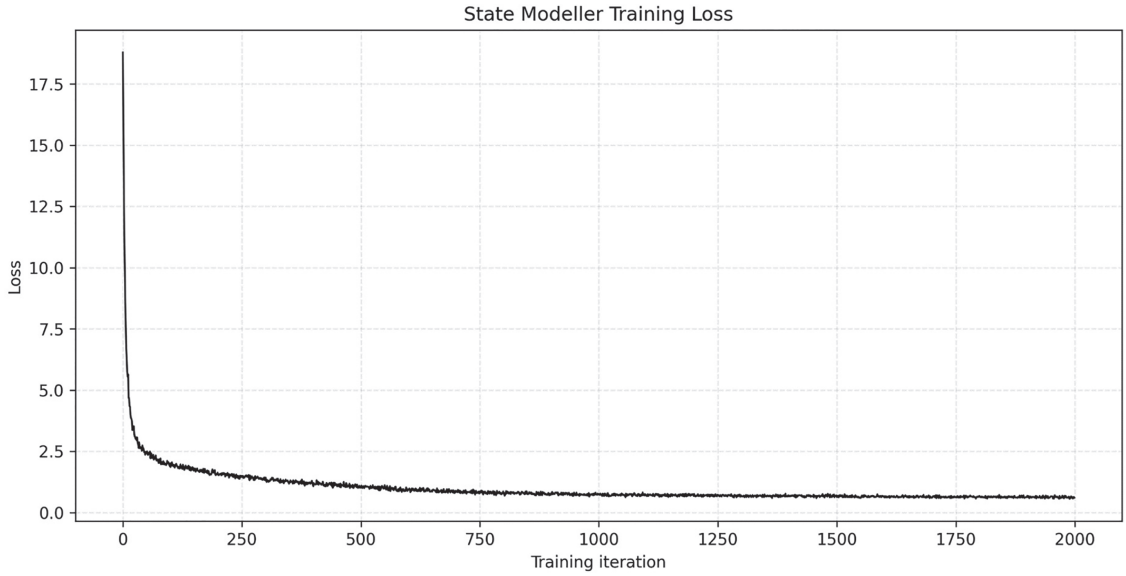


Figure 8. Training loss of the SM over 2000 optimisation iterations. The curve shows a steep decrease during the early training phase followed by a gradual convergence towards a stable low-loss region, indicating that the heterogeneous graph representation was learned efficiently and refined progressively over the course of training.

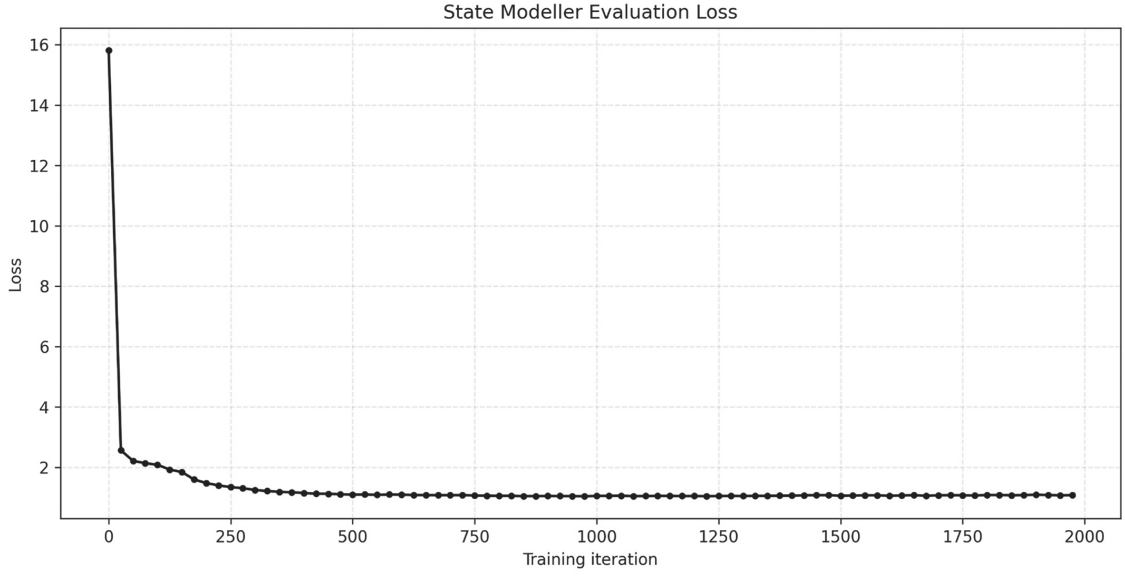


Figure 9. Evaluation loss of the SM on the held-out instance set. The curve decreases strongly in the initial phase and then stabilises around a low level, suggesting that the learned graph embeddings generalise to previously unseen industrial scheduling instances and do not merely reflect memorisation of the training data.

across structurally different production situations rather than merely interpolate within one narrow family of graphs.

The observed training and evaluation results indicate that this objective was achieved to a meaningful extent. The training loss decreased from 18.786 at the beginning of optimisation to 0.605 at the final iteration, with the best training loss of 0.557 reached at iteration 1,989. On the held-out evaluation set, the loss decreased from 15.813 initially to 1.080 at the end of training, with the best evaluation value of 1.045 observed at iteration 975. This development is visualised in Figures 8 and 9. As shown in Figure 8, the training loss dropped very sharply during the initial phase of optimisation and then continued to decline more gradually until reaching a low and stable level towards the end of the 2,000 iterations. This pattern suggests that the model first learned the dominant structural regularities of the industrial graphs quickly and subsequently refined the latent representation more incrementally. A comparable development can

be observed for the evaluation loss in Figure 9. After the strong initial decrease, the curve flattened at a substantially lower level and remained stable over the remaining training horizon, indicating that the learned graph representation generalised beyond the training set rather than being limited to memorising recurring training constellations.

From a substantive perspective, the central result is therefore not merely that the loss decreased, but that the SM was able to transform a highly case-specific integrated scheduling state into a stable and transferable latent representation. In the present use case, this means that the model could jointly encode flexible routing alternatives, dynamic machine states, workforce-related feasibility, and maintenance-sensitive information in a single compact vector without requiring a fixed problem size. This is particularly relevant for the subsequent RL-Scheduler, because the quality of these embeddings is decisive for whether the RL agent can later distinguish between, for example, a genuinely favourable local machine-operation-worker constellation and a superficially attractive assignment that creates downstream congestion. The observed convergence behaviour and held-out performance indicate that the trained SM provided a sufficiently robust abstraction of the partner company's production system to fulfil exactly this role.

Training and evaluation RL-Scheduler

Components of the RL agent

For the industrial use case, the RL-Scheduler was implemented as a PPO-based actor-critic agent that operated on the 40-dimensional latent state representation produced by the SM and evaluated up to 80 feasible actions at each event-driven decision point. Both the actor and the critic were configured with three layers in total, corresponding to two hidden layers and one output layer, with 128 neurons per hidden layer, ReLU activation functions, and no batch normalisation. The output dimension was one, such that the actor produced one scalar score per feasible action and the critic one scalar state-value estimate per scheduling state. The training configuration comprised learning rates of 0.0001 for both actor and critic, 2,000 PPO updates, five optimisation epochs per update, 50 rollout steps, a mini-batch size of 128, a clipping parameter of 0.3, a discount factor of 0.99, and a GAE factor of 0.99. Entropy regularisation was set to 0.005, gradient clipping to 2.0, and both learning-rate decay and entropy decay were applied over training with step sizes of 800 and 400 and decay factors of 0.4 and 0.1, respectively. Training and evaluation were each conducted with one agent, and the simulation was run until episode completion so that the full schedule quality could be reflected in the reward signal.

Table 1. Training configuration and simulation-environment settings of the RL-Scheduler in the industrial use case.

Parameter	Value
Algorithm	PPO (actor–critic)
State dimension	40
Max. feasible actions per decision point	80
Hidden layers (actor & critic)	2 × 128, ReLU
Batch normalisation	None
Output dimension	1
Learning rate (actor & critic)	1×10^{-4}
PPO updates	2000
Optimisation epochs per update	5
Rollout steps	50
Mini-batch size	128
Clipping parameter ϵ	0.3
Discount factor γ	0.99
GAE factor λ	0.99
Entropy coefficient	0.005
Gradient clipping	2.0
Learning-rate decay (step size / factor)	800 / 0.4
Entropy decay (step size / factor)	400 / 0.1
Processing times	Deterministic
Time to failure	Weibull (MTBF)
Repair time	Clipped Gaussian (MTTR), σ factor 0.6
Agents (training / evaluation)	1 / 1
Episode termination	Full schedule completion

The agent was trained in a simulation environment that combined deterministic processing times with stochastic disruption behaviour through the MTBF/MTTR logic. In this setting, times to failure were modelled by a Weibull distribution and repair times by a clipped Gaussian distribution, with the global repair-time dispersion governed by an MTTR sigma factor of 0.6. This design is particularly relevant for the industrial application because it exposed the policy not only to sequencing and routing decisions, but also to stochastic breakdown and recovery dynamics that affect downstream congestion and resource availability. The resulting training environment therefore represented a more realistic scheduling context than a purely deterministic shop-floor model. The complete training configuration is summarised in Table 1.

Reward function

The reward formulation consists of two clearly separated parts: a dense intermediate reward that is returned during the rollout at each decision point, and a terminal reward that is added once the complete schedule has been generated. This separation is intended to combine local decision guidance with a final episode-level performance evaluation. Accordingly, for all non-terminal decision points the agent receives only the intermediate contribution, whereas at the final transition the terminal

contribution is added to the last intermediate reward. In compact form, this interaction can be written as

$$r_t = \begin{cases} r_t^{\text{int}}, & t < T \\ r_T^{\text{int}} + r_T^{\text{term}}, & t = T \end{cases}$$

where r_t^{int} denotes the intermediate reward at decision point t, r_T^{term} denotes the terminal reward at the end of the rollout, and T is the final decision point of the episode. In this way, the formulation makes it explicit that dense and terminal feedback play different roles: the former guides local action selection during the rollout, while the latter evaluates the quality of the completed schedule as a whole.

Both the intermediate reward and the terminal reward follow the same global trade-off between a time-oriented objective and an energy-oriented objective. This trade-off is controlled by a weighting factor $\omega \in [0,1]$, which determines the relative influence of time and energy throughout the entire reward structure. The intermediate reward is therefore defined as

$$r_t^{\text{int}} = (1-\omega) r_t^{\text{int},time} + \omega r_t^{\text{int},energy}$$

while the terminal reward is given by

$$r_T^{\text{term}} = w_f[(1-\omega) r_T^{\text{mk}} + \omega r_T^{\text{E}}]$$

where $r_t^{\text{int},time}$ and $r_t^{\text{int},energy}$ denote the time-oriented and energy-oriented intermediate components, r_T^{mk} and r_T^{E} denote the makespan-based and energy-based terminal components, and w_f is a terminal reward weight. This structure ensures that the same trade-off principle is used consistently at both levels: local scheduling decisions are evaluated with respect to time and energy, and the final schedule is assessed with respect to the same two objectives at the episode level.

The time-oriented intermediate reward is designed to provide dense feedback on whether a selected action improves the projected future completion effort of the system. Instead of relying directly on the final makespan, the formulation uses a normalised workload-based potential Φ_t, which is derived from an aggregated proxy of the remaining completion effort. This proxy combines the current simulation time with the still outstanding workload represented in the graph-based state and the remaining workload of currently active machines in the event-based simulation. The actual progress signal is then obtained from the potential difference between two consecutive decision points,

$$\Delta\Phi_t = \Phi_t - \Phi_{t-1}$$

A positive potential difference therefore indicates that the selected decision has reduced the projected future workload of the system. To extend this progress signal by further operational considerations, the time-oriented intermediate component is formulated as

$$r_t^{\text{int},time} = \lambda_t(w_p \Delta\Phi_t - w_i I_t + b_t)$$

Here, I_t denotes the ratio of idling machines, and b_t is an optional bonus for setup-efficient assignments. The coefficients w_p and w_i determine the relative influence of these components, while λ_t controls the overall scale of the dense time-oriented signal. In functional terms, this means that the intermediate reward captures progress towards lower completion times, penalises low machine utilisation, and rewards setup-efficient assignments.

The energy-oriented intermediate reward evaluates the energetic cost of the selected action at the current point in time. Its purpose is to make the scheduler sensitive to both machine-dependent power demand and time-dependent electricity prices. In an aggregated form, this component can be written as

$$r_t^{\text{int},energy} = -\frac{P_{m,t} \sum_{\tau=t}^{t+d_t-1} c_\tau}{P_{\max} c_{\max} d_t}$$

where $P_{m,t}$ denotes the action-dependent power demand of the selected machine, d_t is the duration of the selected action, c_τ is the electricity price at time τ, P_{max} is a reference maximum power demand, and c_{max} is the corresponding maximum electricity price within the considered horizon. For productive operations, the productive power demand of the selected machine is used, whereas for maintenance-related actions the standby demand can be applied. This leads to a stronger negative reward contribution when an action is executed on an energetically unfavourable machine or during periods with high electricity prices. Consequently, the energy-oriented intermediate component complements the time-oriented shaping signal by introducing a local cost perspective that directly depends on when and where an action is executed.

Whereas the intermediate reward evaluates local decisions during schedule construction, the terminal reward evaluates the final outcome of the completed schedule. It therefore assesses the realised makespan and total energy consumption relative to corresponding reference values. The makespan-based terminal component is defined as

$$r_T^{\text{mk}} = \frac{2}{\pi} arctan\left(g_C \frac{B-C}{B}\right)$$

where C denotes the realised final makespan, B is a baseline reference for the considered instance, and g_C is a gain parameter controlling the sensitivity of the transformation. In the same way, the energy-based terminal

component is defined as

$$r_T^{\mathrm{E}} = \frac{2}{\pi} arctan\left(g_E \frac{E_{max} - E}{E_{max}}\right)$$

where E denotes the realised total energy consumption, E_{max} is an instance-specific energy reference, and g_E controls the sensitivity of the energy transformation. Both terminal terms compare the realised rollout outcome with normalised reference values and then map the result to a bounded range. This prevents extreme schedule outcomes from producing disproportionately large terminal signals. By combining these two terminal terms through the global weighting factor ω, the final reward remains fully consistent with the trade-off already used at the intermediate level.

A final feature of the formulation is that the scale of the dense time-oriented signal is not fixed throughout training. Instead, the factor λ_t is adjusted across rollouts in order to maintain a stable relationship between cumulative intermediate feedback and terminal reward contribution. The purpose of this adaptation is not to change the optimisation objective itself, but to prevent either the dense local shaping signal or the final schedule evaluation from dominating the learning process over extended training phases. Overall, the resulting formulation provides a clearly structured objective in which intermediate and terminal rewards are functionally separated, while time and energy are treated consistently through a common weighting principle at both levels.

Results

Two final reward parameterisations were evaluated. The first configuration, $AIS(\omega_0)$, represented a pure makespan-oriented policy with $\omega = 0$. In this run, the terminal reward weight w_f was set to 2.0, the makespan terminal gain g_C to 30.0, the initial scaling of the time-oriented intermediate reward λ_0 to 0.6, the progress weight w_p to 0.4, the idle weight w_i to 0.02, and the setup-efficiency bonus b_t to 0.02. Since $\omega = 0$, the energy component did not contribute to the effective reward signal of this run.

The second configuration, $AIS(\omega_{0,1})$, assigned 10% of the reward weight to energy and 90% to throughput time. Here, the terminal reward weight w_f again amounted to 2.0, while the makespan terminal gain g_C was reduced to 25.0 and the energy terminal gain g_E was 12.5. The initial scaling of the time-oriented intermediate reward λ_0 was 1.0, the progress weight w_p was set to 1.0, the idle weight w_i to 0.05, and the setup-efficiency bonus b_t again to 0.02. In both runs, the baseline references used in the terminal terms were dynamic and instance-specific. Thus, the two evaluated policies differed not in architecture, but in the operational priority encoded in the reward design.

The evaluation was extended along three dimensions in order to separate the contribution of the architecture from case-specific tuning. First, the industrial evaluation was widened from a single order list to the daily order lists of two production areas of the application partner, comprising 18 day instances of small-part production, 18 day instances of large-part production and 18 combined day instances in which both areas are scheduled jointly on the shared resource pool. Second, the identical architecture was evaluated on 25 public benchmark instances drawn from four established flexible job shop families, namely Brandimarte Mk01 to Mk10 and the Hurink edata, rdata and vdata sets, fixed before the evaluation on the basis of instance size and routing flexibility. Since these instances do not contain the integrated problem features addressed in this work, they were reproducibly augmented with synthetic setup groups, worker qualifications, maintenance operations and machine energy demands as well as real day-ahead electricity prices, while the routing structure and processing times of the originals were left unchanged. Reliability information, by contrast, was not synthesised, as artificial MTBF and MTTR parameters would bear no correspondence to the calibrated failure behaviour on which the policy is trained. The stochastic regime is therefore evaluated exclusively on the industrial instances, whose failure models were fitted to the MES failure logs of the application partner. Third, every industrial instance family was evaluated both in a deterministic regime without machine failures and in a stochastic regime in which failures and repairs are sampled from the fitted distributions, which together with the deterministic benchmark evaluation yields the seven evaluation cells reported below.

The learned policy is compared against seven dispatching baselines that are re-evaluated within the identical simulation environment, namely the shortest processing time rule (SPT), the longest processing time rule (LPT), random action selection (RANDOM), first in first out (FIFO), the two work-remaining rules most operations remaining (MOR) and most work remaining (MWR), and the machine affinity rule (MAF), which serves as an additional assignment-oriented reference. MAF prioritises jobs with the largest number of remaining operations that can be processed on the machine under consideration, thereby favouring assignments that preserve routing flexibility on the respective machine. Unlike the established priority rules, MAF constitutes a problem-specific heuristic introduced in this study to reflect the machine-related structure of the industrial setting.

Each instance was evaluated in five independent runs with distinct random seeds. Inference was performed with the stochastic policy rather than with a

greedy arg-max decision, so that the reported dispersion reflects both the sampling of the policy and the stochastic dynamics of the environment. All comparisons are reported as instance-wise relative deviations of the RL-Scheduler from a baseline, computed on the seed means of the respective instance, with negative values indicating an advantage of the RL-Scheduler. Because the instance-wise deviations are heavy-tailed and contain individual outliers, statistical significance is assessed by a two-sided Wilcoxon signed-rank test over the instances of an evaluation cell rather than by a t-test, and is complemented by the number of instances on which the RL-Scheduler is superior. The seed-level dispersion of the RL-Scheduler itself is low in the deterministic regimes, with a mean coefficient of variation of the makespan of 1.2% to 2.0% on the industrial instances and 4.5% to 4.9% on the benchmark instances, and rises to 5.8% to 8.8% in the stochastic regimes. This increase reflects the variance introduced by the sampled failure events rather than instability of the learned policy.

Table 2 reports the aggregated results against the mean of the seven dispatching baselines. On the industrial instances the makespan-oriented configuration reduces the makespan by 5.62% on the small-part line and by 4.68% on the large-part line in the deterministic regime, outperforming the baseline mean on 18 of 18 and on 15 of 18 instances respectively. Under stochastic machine failures the advantage decreases only moderately, to 4.27% and 4.37% on 14 of 18 instances in both cases. On the combined day instances, in which small-part and large-part production are scheduled jointly, the advantage is preserved but attenuated relative to the separate lines: the makespan-oriented configuration reduces the makespan against the baseline mean by 3.92% on 18 of 18 instances in the deterministic regime and by 2.93% on 14 of 18 instances under stochastic failures, both statistically significant. This attenuation is plausible, as the joint scheduling of both lines pools the workload across the shared resource set, so that the relative benefit of exploiting the integrated feature set of any single line is diluted. For the energy-aware configuration the relative energy cost is likewise reduced on the small-part line, by 3.51% in the deterministic and by 2.58% in the stochastic regime, whereas on the large-part line the energy effect is smaller and does not reach significance. The realised energy cost of the makespan-oriented configuration is not reported, since energy does not enter its objective and the corresponding values constitute a by-product rather than an optimisation result. On the combined instances the energy-aware configuration reduces the relative energy cost by 2.45% in the deterministic regime and by 2.62% under stochastic failures, both statistically significant. Averaged over the seven evaluation cells, the learned policy reduces the makespan against the baseline mean by 4.35% in the makespan-oriented and by 4.45% in the energy-aware configuration, and the energy-aware configuration reduces the relative energy cost by 2.41%; each of these advantages is consistent across all seven cells and significant in a two-sided Wilcoxon signed-rank test over the cell means ($p = 0.016$).

On the public benchmark instances the picture is more differentiated, and it is best read from the baseline-wise results in Table 3. In the deterministic regime the RL-Scheduler outperforms the baseline mean by 4.67% on 22 of 25 instances, but this aggregate conceals a systematic difference between weak and strong dispatching rules. The advantage over LPT, FIFO and random selection is substantial, amounting to 12.49%, 8.99% and 6.81% respectively, whereas the two work-remaining rules MOR

Table 2. Aggregated evaluation results of the RL-Scheduler against the mean of the seven dispatching baselines.

Evaluation cell	ω	n	Δ makespan (%)	Better on	p	Δ energy cost (%)	Better on	p
Public benchmark, deterministic	0	25	-4.67 ± 4.76	22/25	<0.001	–	–	–
Public benchmark, deterministic	0.1	25	-5.48 ± 3.55	24/25	<0.001	-2.69 ± 3.65	18/25	0.001
Industrial small parts, deterministic	0	18	-5.62 ± 3.61	18/18	<0.001	–	–	–
Industrial small parts, deterministic	0.1	18	-6.76 ± 6.25	17/18	<0.001	-3.51 ± 7.29	15/18	0.018
Industrial small parts, stochastic	0	18	-4.27 ± 6.00	14/18	0.010	–	–	–
Industrial small parts, stochastic	0.1	18	-3.85 ± 6.67	14/18	0.030	-2.58 ± 6.19	14/18	0.024
Industrial large parts, deterministic	0	18	-4.68 ± 4.75	15/18	<0.001	–	–	–
Industrial large parts, deterministic	0.1	18	-4.03 ± 4.98	15/18	0.003	-1.79 ± 3.41	11/18	0.074
Industrial large parts, stochastic	0	18	-4.37 ± 7.41	14/18	0.018	–	–	–
Industrial large parts, stochastic	0.1	18	-4.60 ± 8.26	14/18	0.024	-1.24 ± 6.30	12/18	0.417
Industrial combined, deterministic	0	18	-3.92 ± 2.71	18/18	<0.001	–	–	–
Industrial combined, deterministic	0.1	18	-3.23 ± 3.25	17/18	<0.001	-2.45 ± 3.82	17/18	0.001
Industrial combined, stochastic	0	18	-2.93 ± 5.11	14/18	0.024	–	–	–
Industrial combined, stochastic	0.1	18	-3.21 ± 3.58	15/18	0.002	-2.62 ± 3.34	16/18	0.003
Mean over evaluation cells	0	7	-4.35 ± 0.82	7/7	0.016	–	–	–
Mean over evaluation cells	0.1	7	-4.45 ± 1.29	7/7	0.016	-2.41 ± 0.72	7/7	0.016

Notes: Negative values indicate an advantage of the RL-Scheduler. p-values from a two-sided Wilcoxon signed-rank test over the instances of the respective evaluation cell. Relative energy costs are reported only for the energy-aware configuration ($\omega = 0.1$), since the makespan-oriented configuration does not include energy in its objective. The final rows report the unweighted mean over the seven evaluation cells, given as mean ± standard deviation of the cell means; the corresponding p-values result from a two-sided Wilcoxon signed-rank test over the seven cell means.

Table 3. Instance-wise mean relative deviation of the RL-Scheduler from each individual dispatching baseline in percent.

Evaluation cell	ω	Metric	SPT	LPT	RANDOM	FIFO	MOR	MWR	MAF
Public benchmark, deterministic	0	Makespan	−4.20	−12.49	−6.81	−8.99	+2.08	+2.25	−4.55
Public benchmark, deterministic	0.1	Makespan	−4.97	−13.27	−7.59	−9.77	+1.25	+1.37	−5.39
Public benchmark, deterministic	0.1	Energy cost	+1.14	−7.87	−2.98	−4.52	−1.28	−0.49	−2.81
Industrial small parts, deterministic	0	Makespan	−8.13	−4.42	−9.75	−6.24	−3.06	−2.62	−5.11
Industrial small parts, deterministic	0.1	Makespan	−9.21	−5.70	−10.86	−7.43	−3.99	−3.84	−6.27
Industrial small parts, deterministic	0.1	Energy cost	−4.40	−3.23	−7.28	−3.70	−1.15	−1.81	−2.98
Industrial small parts, stochastic	0	Makespan	−7.03	−4.33	−8.58	−5.62	−1.95	+0.01	−2.36
Industrial small parts, stochastic	0.1	Makespan	−6.69	−4.04	−8.18	−5.30	−1.55	+0.63	−1.85
Industrial small parts, stochastic	0.1	Energy cost	−5.07	−2.96	−5.94	−3.06	+0.28	+0.48	−1.82
Industrial large parts, deterministic	0	Makespan	−9.42	−3.64	−3.48	−5.96	−3.37	−1.13	−5.77
Industrial large parts, deterministic	0.1	Makespan	−8.90	−2.99	−2.83	−5.23	−2.77	−0.39	−5.07
Industrial large parts, deterministic	0.1	Energy cost	−5.37	−0.77	−1.91	−2.78	−0.57	+0.64	−1.77
Industrial large parts, stochastic	0	Makespan	−7.27	−3.73	−6.88	−5.20	−3.48	+0.84	−4.90
Industrial large parts, stochastic	0.1	Makespan	−7.50	−3.87	−7.01	−5.52	−3.60	+0.58	−5.27
Industrial large parts, stochastic	0.1	Energy cost	−4.84	−0.35	−2.06	−1.77	+0.12	+2.01	−1.77
Industrial combined, deterministic	0	Makespan	−5.20	−3.96	−6.94	−4.08	−1.11	−1.59	−4.58
Industrial combined, deterministic	0.1	Makespan	−4.53	−3.25	−6.28	−3.37	−0.44	−0.88	−3.90
Industrial combined, deterministic	0.1	Energy cost	−3.19	−2.55	−5.55	−1.88	−0.34	−1.55	−2.13
Industrial combined, stochastic	0	Makespan	−4.78	−2.86	−5.07	−3.48	−1.21	+0.27	−3.39
Industrial combined, stochastic	0.1	Makespan	−5.00	−3.09	−5.36	−3.75	−1.64	−0.03	−3.64
Industrial combined, stochastic	0.1	Energy cost	−4.18	−2.69	−4.50	−2.30	−1.30	−0.99	−2.38
Mean over evaluation cells	**0**	**Makespan**	**−6.57**	**−5.06**	**−6.79**	**−5.65**	**−1.73**	**−0.28**	**−4.38**
Mean over evaluation cells	**0.1**	**Makespan**	**−6.69**	**−5.17**	**−6.87**	**−5.77**	**−1.82**	**−0.36**	**−4.48**
Mean over evaluation cells	**0.1**	**Energy cost**	**−3.70**	**−2.92**	**−4.32**	**−2.86**	**−0.61**	**−0.25**	**−2.24**

Notes: Negative values indicate an advantage of the RL-Scheduler. Relative energy costs are reported only for the energy-aware configuration ($\omega = 0.1$). The final rows report the unweighted mean over the seven evaluation cells.

and MWR outperform the learned policy by 2.08% and 2.25%.

This contrast between the benchmark and the industrial instances is informative rather than incidental. The public instances were augmented with the same integrated feature set, that is setup groups, workforce qualifications and maintenance operations, yet these attributes were generated synthetically and are therefore largely uncorrelated with the routing and processing structure of the original instances. In such a setting a well-chosen priority rule such as most-work-remaining remains strong, since the synthetic attributes offer little exploitable regularity. The industrial instances, by contrast, exhibit systematically interrelated setup, qualification and load structures that emerge from a real production environment, which is precisely the structure the learned policy is able to exploit through the graph and state representation. It is on these instances that the learned policy is superior to, or at least slightly ahead of, all seven baselines, outperforming MOR by between 0.44% and 3.99%. Relative to the strongest rule MWR, the learned policy remains slightly ahead on average across the industrial instances, by 0.70% in the makespan-oriented and by 0.65% in the energy-aware configuration, and this advantage is significant for the makespan-oriented configuration under a one-sided Wilcoxon signed-rank test of the directional hypothesis that the learned policy does not fall behind MWR ($p = 0.028$), whereas the energy-aware configuration narrowly misses the significance level ($p = 0.057$). The cell-wise deviations under stochastic conditions and on the combined day instances nevertheless stay below the robustness threshold of approximately two percentage points discussed below.

The comparison of the two reward parameterisations does not reproduce the simple trade-off that was originally expected. The energy-aware configuration does not systematically sacrifice makespan. In the deterministic regimes it is at least as strong as the makespan-oriented configuration, improving the makespan advantage from 4.67% to 5.48% on the benchmark instances and from 5.62% to 6.76% on the small-part line. Its effect on energy cost is therefore assessed against the dispatching baselines rather than against the makespan-oriented configuration. The energy-aware configuration reduces the relative energy cost significantly on the public benchmark instances, on the small-part line in both regimes, and on the combined instances in both regimes, whereas on the large-part line the reduction does not reach significance in either regime. The energy term therefore acts as a mild regulariser whose benefit is real but modest and not uniform across settings. A systematic energy improvement at constant makespan is not supported by the present data. This result should, however, be read together with the comparison against the strongest dispatching rule. Whereas a priority rule such as MWR optimises a single implicit criterion and offers no lever for adjusting the operational objective, the two evaluated configurations share the identical architecture, training procedure, and hyperparameters and differ only in the reward weighting. The learned scheduler thus attains at least the makespan level of MWR while allowing the operational objective to be shifted, for example towards energy cost under volatile

electricity prices or, in future extensions, towards due-date adherence or workforce levelling, through a reconfiguration of the reward rather than through a replacement of the scheduling logic. The practical argument for the learned scheduler over a fixed priority rule therefore does not rest on the sub-percentage makespan margin alone, but on providing a configurable and integrated decision capability at no cost in makespan performance.

Module reusability and transfer across instances

The proposed architecture is designed as a set of loosely coupled modules so that adaptation to a new plant or production instance is confined to well-defined components rather than requiring a redesign of the overall system. Table 4 summarises, for each module, whether it can be reused across plants and instances without change and which parts must be re-calibrated or retrained when the architecture is transferred to a new environment.

Together, Tables 4 and 5 substantiate the plug-and-schedule character of the proposed architecture. Both the core modules and the functional extensions can be adopted as-is, without redeveloping the scheduling logic for a new deployment. Table 4 describes the reusable base framework, that is, the modules that constitute the architecture independently of the concrete use case, whereas Table 5 lists the functional extensions that are layered onto this framework and that may vary depending on the underlying production system. Which extensions are required, and to what extent they are parameterised, is therefore a matter of configuration rather than a redesign of the architecture. In the context of this work, the selection and scope of the extensions were driven by the requirements of the industrial application partner from the machinery manufacturing sector. A different production system may call for a different subset or additional extensions, while the base framework shown in Table 4 remains unchanged.

Managerial insights

From a managerial perspective, the proposed architecture is relevant because it lowers several of the barriers that typically prevent advanced scheduling methods from being transferred into industrial practice. In contrast to many highly specialised scheduling solutions, the present approach is not limited to the integrated flexible job shop alone, but can in principle also be transferred to structurally simpler settings such as flow shops or even single-machine environments, provided that the corresponding production structure is mapped consistently into the graph-based representation. At the same time, the paper shows that this abstraction remains sufficiently expressive to account for the operational characteristics that matter in practice, such as machine alternatives, workforce restrictions, maintenance requirements, setup dependencies, uncertainty and energy efficiency. This implies that investments into the architecture are not tied to one narrowly defined scheduling problem, but can form the basis for a broader and more scalable digital decision-support capability.

A second implication concerns implementation effort. The main initial effort lies in the one-time mapping of company data into the proposed structure and in the calibration of the most relevant hyperparameters of the SM and RL-Scheduler. Once this mapping has been established, however, the architecture can be extended incrementally when additional decision factors become relevant. In the present case, this concerns for example the

Table 4. Transfer matrix for reusability of the architecture modules across plants and scheduling instances.

Module	Reusable across plants?	Re-calibration of data / retraining needed
AAS-Middleware (data integration)	Yes (standard AAS interfaces)	Map plant data sources to reference submodels
GM	Yes (generic graph logic based on AAS SDM reference model)	Feature definitions of node types for new attributes
SM (embeddings)	Architecture yes; embeddings no	Retrain embeddings on target-plant data; Definition of embedding size
Event-based Simulation	Yes (generic DES engine based on graph from GM)	Parameters: times, MTBF/MTTR, energy (will be automatically transferred from graph)
RL-Scheduler	Framework yes; weights no	Retrain / fine-tune policy and reward weights

Table 5. Integration of the functional extensions into the SMAS architecture.

Extension	Required data (AAS model)	Architecture components adapted	Change type
Set-up sequencing	Setup state, requirement, time	Setup features in graph and state; setup phase in simulation	Retrained; no structural change
Maintenance scheduling	Maintenance orders marked by a flag in the feature vector	Flag triggers the corresponding resource binding in the simulation; no dedicated adaptation	None required; inactive when the flag is 0
Energy awareness	Power values + price vector	Energy features + prediction vector; price events; reward term	Retrained; adds bi-criteria objective
Workforce scheduling	Availability: shifts, breaks, qualifications	Workforce nodes/edges; HetGNN embeddings; worker sync in simulation	Structural: ternary action space, adapted networks

integration of maintenance, workforce constraints, and energy-related information. This modularity is important from a managerial point of view because it makes implementation more manageable. Companies do not need to redesign the whole logic whenever they want to include another planning factor, but can build upon an existing representation and expand it step by step. Closely related to this is the fact that no completely separate optimisation-specific simulation model has to be developed in the traditional sense. However, the event-based execution logic must still be adapted whenever company-specific production characteristics are to be represented adequately.

A further managerial insight is that the approach reduces dependence on manually designed scheduling heuristics. In many companies, practical scheduling still relies either on human scheduling experience or on rule sets that have to be specified, maintained, and adapted explicitly. Such heuristics can work reasonably well in stable environments, but they become increasingly difficult to design once multiple interacting objectives and constraints have to be considered simultaneously. The proposed approach shifts this burden from manual rule engineering towards data-driven policy learning. In other words, instead of investing substantial effort into constructing elaborate rule systems for every special case, companies can train the RL agent to learn suitable scheduling behaviour directly from repeated interaction with the modelled production environment. This is particularly valuable in complex environments where workforce availability, maintenance events, setup dependencies, and stochastic disruptions interact in ways that are difficult to translate into fixed rules. The findings of the paper therefore suggest that RL can serve as a practical alternative to extensive heuristic design, especially when scheduling complexity exceeds the range that can still be handled efficiently by manually specified scheduling logic.

The training concept itself also has practical implications. Once the underlying representation has been set up, the system can be trained directly and subsequently adapted through pretraining or retraining when system conditions change, for example after changes in routing structures, workforce availability, or maintenance behaviour.

Finally, the proposed architecture may be particularly useful for making RL-based scheduling more accessible to smaller and medium-sized manufacturers. Such firms often lack the resources for highly customised optimisation software or large-scale in-house AI development. The modular structure proposed in this paper addresses this issue by combining standardised data integration, reusable scheduling components, and configurable learning logic in a way that is intended to accommodate the specific characteristics of different production systems without requiring a completely bespoke solution in each case.

Conclusion and outlook

This paper proposed Plug and Schedule, a scalable and modular architecture for integrated and dynamic scheduling that combines an AAS-based middleware for standardised data integration, a graph-based problem representation, a graph neural network-based SM, an event-based simulation, and a Proximal Policy Optimisation-based RL-Scheduler. The architecture integrates setup sequencing, maintenance scheduling, workforce constraints, and energy awareness into a single decision framework and was validated both on the production system of an industrial application partner and on established public benchmark instances.

The evaluation supports a specific rather than a general claim. On the industrial instances the learned policy reduces the makespan against the mean of seven dispatching baselines by between 2.93% and 6.76% across six evaluation cells covering small-part, large-part, and combined production under deterministic and stochastic machine failures, with statistical significance in every cell, and remains slightly ahead of the strongest individual rule MWR on average. Beyond these numerical margins, the practical value of the approach lies in its configurability. The operational objective can be shifted through the reward weighting without any change to the architecture itself, as demonstrated by the energy-aware configuration, which preserves the makespan performance while reducing the relative energy cost in most evaluation cells.

Several limitations remain. The industrial validation is based on a single production system, since the underlying data are proprietary. The advantage of the learned policy under machine failures is established only for failure models calibrated from the MES logs of the application partner, so that its robustness in systems without calibrated reliability data remains untested. Furthermore, the effect of the energy-aware configuration on energy cost is modest and not uniform across the evaluation cells, and differences below approximately two percentage points should not be interpreted as robust on individual instances.

Future work will therefore focus on the scalability and transfer of the architecture. This includes its application towards non-discrete production environments, and its transfer to further industrial domains, with concrete follow-up activities planned in the semiconductor and heat-pump industries in further research projects. In

addition, the reward design will be extended towards further operational objectives such as due-date adherence and workforce levelling.

Acknowledgements

The authors would like to thank 'INSPIRE' under the Project Number 543072266 for the useful insights into the development of the paper.

Author contributions

CRediT: **Yannik Hermann:** Conceptualization, Data curation, Formal analysis, Methodology, Validation, Visualization, Writing – original draft, Writing – review & editing; **Vincent Schroettle:** Conceptualization, Methodology, Visualization; **Benedikt Kaupp:** Methodology, Validation, Visualization; **Martin Benfer:** Supervision, Writing – original draft, Writing – review & editing; **Gisela Lanza:** Conceptualization, Supervision

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

We gratefully acknowledge the support for this research through the 'CLiCE-DiPP' project funded by the Federal Ministry for Economic Affairs and Climate Action (BMWK) through the 'Greentech' program [grant number 01MN23 023G], the Project Inspire (Project number: 543072266) funded by German Research Foundation (DFG).

Notes on contributors



Yannik Hermann, M.Sc. studied industrial engineering at KIT in Karlsruhe and is a research assistant at wbk Institute of Production Science since 2023. He conducts his Ph.D. in the field of Production Systems Planning and Control.



Vincent Schröttle, M.Sc. studied industrial engineering at KIT in Karlsruhe and is currently pursuing his Ph.D. at the Technical University of Munich in the field of circular economy.



Benedikt Kaupp, M.Sc. studied industrial engineering at KIT in Karlsruhe. He conducted his Master Thesis at the wbk Institute of Production Science in 2025.



Martin Benfer, Ph.D. is the former Deputy Head of Production Systems at wbk Institute of Production Science. He studied mechanical engineering at RWTH Aachen University before his dissertation at wbk Institute of Production Science at Karlsruhe Institute of Technology (KIT). His research focused on model-based decision-making in global production networks.



Prof. Dr-Ing. Gisela Lanza is member of the management board at wbk Institute of Production Science of the Karlsruhe Institute of Technology (KIT). She heads the Production Systems division, dealing with topics of Global Production Strategies, Production System Planning, and Quality Assurance.

Data availability statement

The data that support the findings of this study are not publicly available due to confidentiality restrictions and the proprietary nature of the industrial case data. The data are available from the corresponding author upon reasonable request and subject to approval by the industrial project partner.

References

Akbar, M., and T. Irohara. 2018. "Scheduling for Sustainable Manufacturing: A Review." *Journal of Cleaner Production* 205:866–883. https://doi.org/10.1016/j.jclepro.2018.09.100.

Alemao, D., A. D. Rocha, S. Nikghadam-Hojjati, and J. Barata. 2022. "How to Design Scheduling Solutions for Smart Manufacturing Environments Using RAMI 4.0?" *IEEE Access* 10:71284–71298. https://doi.org/10.1109/ACCESS.2022.3187974.

Ali, M. H., A. Saif, and A. Ghasemi. 2022. "Robust Job Shop Scheduling with Condition-Based Maintenance and Random Breakdowns." *IFAC-PapersOnLine* 55 (10): 1225–1230. https://doi.org/10.1016/j.ifacol.2022.09.557.

Avalos-Rosales, O., F. Angel-Bello, A. Álvarez, and Y. Cardona-Valdés. 2018. "Including Preventive Maintenance Activities in an Unrelated Parallel Machine Environment with Dependent Setup Times." *Computers & Industrial Engineering* 123:364–377. https://doi.org/10.1016/j.cie.2018.07.006.

Bakopoulos, E., K. Sipsas, N. Nikolakis, and K. Alexopoulos. 2024. "A Digital Twin and Data Spaces Framework towards Resilient Manufacturing Value Chains." *IFAC-PapersOnLine* 58 (19): 163–168. https://doi.org/10.1016/j.ifacol.2024.09.129.

Behrendt, S., F. Stamer, M. C. May, and G. Lanza. 2023. "Towards a Service-Oriented Architecture for Production Planning and Control: A Comprehensive Review and Novel Approach." In *CPSL – Conference on Production Systems and Logistics*, November 14–17, Stellenbosch, South Africa. https://doi.org/10.15488/15271.

Bezoui, M., A. Kermali, A. Bounceur, S. M. Qaisar, and A. T. Almaktoom. 2024. "Deep Reinforcement Learning for Multiobjective Scheduling in Industry 5.0 Reconfigurable Manufacturing Systems." In *Machine Learning for Networking*, Vol

14525, edited by É. Renault, S. Boumerdassi, and P. Mühlethaler, 90–107. Cham: Springer Nature. https://doi.org/10.1007/978-3-031-59933-0_7.

Blazewicz, J., K. H. Ecker, E. Pesch, G. Schmidt, M. Sterna, and J. Weglarz. 2019. *Handbook on Scheduling: From Theory to Practice*. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-99849-7.

Brandimarte, P. 1993. "Routing and Scheduling in a Flexible Job Shop by Tabu Search." *Annals of Operations Research* 41 (3): 157–183. https://doi.org/10.1007/BF02023073.

Chen, R., B. Yang, S. Li, and S. Wang. 2020. "A Self-learning Genetic Algorithm Based on Reinforcement Learning for Flexible Job-Shop Scheduling Problem." *Computers & Industrial Engineering* 149:106778. https://doi.org/10.1016/j.cie.2020.106778.

Cowling, P., and M. Johansson. 2002. "Using Real Time Information for Effective Dynamic Scheduling." *European Journal of Operational Research* 139 (2): 230–244. https://doi.org/10.1016/S0377-2217(01)00355-1.

Damsgaard, J., and J. Karlsbjerg. 2010. "Seven Principles for Selecting Software Packages." *Communications of the ACM* 53 (8): 63–71. https://doi.org/10.1145/1787234.1787252.

Dauzère-Pérès, S., J. Ding, L. Shen, and K. Tamssaouet. 2024. "The Flexible Job Shop Scheduling Problem: A Review." *European Journal of Operational Research* 314 (2): 409–432. https://doi.org/10.1016/j.ejor.2023.05.017.

Fattahi, P., M. Saidi Mehrabad, and F. Jolai. 2007. "Mathematical Modeling and Heuristic Approaches to Flexible Job Shop Scheduling Problems." *Journal of Intelligent Manufacturing* 18 (3): 331–342. https://doi.org/10.1007/s10845-007-0026-8.

Felder, M., D. Steiner, P. Busch, M. Trat, C. Sun, J. Bender, and J. Ovtcharova. 2023. "Energy-Flexible Job-Shop Scheduling Using Deep Reinforcement Learning." In *Conference on Production Systems and Logistics CPSL 2023*, February 28–March 2, Querétaro, Mexico https://doi.org/10.15488/13454.

Framinan, J. M., and R. Ruiz. 2010. "Architecture of Manufacturing Scheduling Systems: Literature Review and an Integrated Proposal." *European Journal of Operational Research* 205 (2): 237–246. https://doi.org/10.1016/j.ejor.2009.09.026.

Francescutto, G., K. Schekotihin, and M. M. S. El-Kholany. 2021. "Solving a Multi-resource Partial-Ordering Flexible Variant of the Job-Shop Scheduling Problem with Hybrid ASP." In *Logics in Artificial Intelligence*, Vol. 12678, edited by W. Faber, G. Friedrich, M. Gebser, and M. Morak, 313–328. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-75775-5_21.

Gao, K., Y. Huang, A. Sadollah, and L. Wang. 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.

Geng, K., L. Liu, and S. Wu. 2024. "A Reinforcement Learning Based Memetic Algorithm for Energy-Efficient Distributed Two-Stage Flexible Job Shop Scheduling Problem." *Scientific Reports* 14 (1): 30816. https://doi.org/10.1038/s41598-024-81064-z.

Geurtsen, M., J. Adan, and A. Akçay. 2024. "Integrated Maintenance and Production Scheduling for Unrelated Parallel Machines with Setup Times." *Flexible Services and Manufacturing Journal* 36 (3): 1046–1079. https://doi.org/10.1007/s10696-023-09511-z.

Grumbach, F., N. E. A. Badr, P. Reusch, and S. Trojahn. 2023. "A Memetic Algorithm with Reinforcement Learning for Sociotechnical Production Scheduling." *IEEE Access* 11:68760–68775. https://doi.org/10.1109/ACCESS.2023.3292548.

Haït, A., and C. Artigues. 2009. "Scheduling Parallel Production Lines with Energy Costs." *IFAC Proceedings Volumes* 42 (4): 1274–1279. https://doi.org/10.3182/20090603-3-RU-2001.0411.

Harjunkoski, I., and R. Bauer. 2014. "Sharing Data for Production Scheduling Using the ISA-95 Standard." *Frontiers in Energy Research* 2: 14–15. https://doi.org/10.3389/fenrg.2014.00044.

Hermann, Y., V. Schröttle, M. Benfer, and G. Lanza. 2025. "A Scalable and Modular Architecture for Intelligent Scheduling." *IFAC-PapersOnLine* 59 (10): 1337–1342. https://doi.org/10.1016/j.ifacol.2025.09.225.

Ho, K.-H., J.-Y. Cheng, J.-H. Wu, F. Chiang, Y.-C. Chen, Y.-Y. Wu, and I.-C. Wu. 2024. "Residual Scheduling: A New Reinforcement Learning Approach to Solving Job Shop Scheduling Problem." *IEEE Access* 12:14703–14718. https://doi.org/10.1109/ACCESS.2024.3357969.

Hurink, J., B. Jurisch, and M. Thole. 1994. "Tabu Search for the Job-Shop Scheduling Problem with Multi-purpose Machines." *OR Spektrum* 15 (4): 205–215. https://doi.org/10.1007/BF01719451.

Hutter, D., T. Steinberger, and M. Hellwig. 2025. "A Parameter Adaptive Genetic Algorithm with Restarts for Flexible Job Shop Scheduling Problems with Worker Flexibility." In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, Malaga, Spain, 523–526. https://doi.org/10.1145/3712255.3726573.

Jalali Khalil Abadi, Z., and N. Mansouri. 2024. "A Comprehensive Survey on Scheduling Algorithms Using Fuzzy Systems in Distributed Environments." *Artificial Intelligence Review* 57 (1): 4. https://doi.org/10.1007/s10462-023-10632-y.

Jiang, T., H. Zhu, L. Liu, and Q. Gong. 2022. "Energy-conscious Flexible Job Shop Scheduling Problem considering Transportation Time and Deterioration Effect Simultaneously." *Sustainable Computing: Informatics and Systems* 35:100680. https://doi.org/10.1016/j.suscom.2022.100680.

Jules, G., and M. Saadat. 2017. "Agent Cooperation Mechanism for Decentralized Manufacturing Scheduling." *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 47 (12): 3351–3362. https://doi.org/10.1109/TSMC.2016.2578879.

Kasapidis, G. A., D. C. Paraskevopoulos, P. P. Repoussis, and C. D. Tarantilis. 2021. "Flexible Job Shop Scheduling Problems with Arbitrary Precedence Graphs." *Production and Operations Management* 30 (11): 4044–4068. https://doi.org/10.1111/poms.13501.

Kazemi, H., M. M. Mazdeh, and M. Rostami. 2017. "The Two Stage Assembly Flow-Shop Scheduling Problem with Batching and Delivery." *Engineering Applications of Artificial Intelligence* 63:98–107. https://doi.org/10.1016/j.engappai.2017.05.004.

Kool, W., H. van Hoof, and M. Welling. 2018. "Attention, Learn to Solve Routing Problems! (Version 3)." arXiv. https://doi.org/10.48550/ARXIV.1803.08475.

Kuhnle, A., J.-P. Kaiser, F. Theiß, N. Stricker, and G. Lanza. 2021. "Designing an Adaptive Production Control System Using Reinforcement Learning." *Journal of Intelligent*

Manufacturing 32 (3): 855–876. https://doi.org/10.1007/s10845-020-01612-y.

Liu, R., R. Piplani, and C. Toro. 2022. "Deep Reinforcement Learning for Dynamic Scheduling of a Flexible Job Shop." *International Journal of Production Research* 60 (13): 4049–4069. https://doi.org/10.1080/00207543.2022.2058432.

Loffredo, A., M. C. May, A. Matta, and G. Lanza. 2024. "Reinforcement Learning for Sustainability Enhancement of Production Lines." *Journal of Intelligent Manufacturing* 35 (8): 3775–3791. https://doi.org/10.1007/s10845-023-02258-2.

Luo, S. 2020. "Dynamic Scheduling for Flexible Job Shop with New Job Insertions by Deep Reinforcement Learning." *Applied Soft Computing* 91:106208. https://doi.org/10.1016/j.asoc.2020.106208.

Mirjalili, S., and A. H. Gandomi. 2023. *Comprehensive Metaheuristics: Algorithms and Applications*. London: Academic Press.

Monkman, S. K., D. J. Morrice, and J. F. Bard. 2008. "A Production Scheduling Heuristic for an Electronics Manufacturer with Sequence-Dependent Setup Costs." *European Journal of Operational Research* 187 (3): 1100–1114. https://doi.org/10.1016/j.ejor.2006.06.063.

Monostori, L., B. Kádár, T. Bauernhansl, S. Kondoh, S. Kumara, G. Reinhart, O. Sauer, G. Schuh, W. Sihn, and K. Ueda. 2016. "Cyber-physical Systems in Manufacturing." *CIRP Annals* 65 (2): 621–641. https://doi.org/10.1016/j.cirp.2016.06.005.

Oğuz, C., F. Sibel Salman, and Z. Bilgintürk Yalçın. 2010. "Order Acceptance and Scheduling Decisions in Make-to-Order Systems." *International Journal of Production Economics* 125 (1): 200–211. https://doi.org/10.1016/j.ijpe.2010.02.002.

Oluyisola, O. E., S. Bhalla, F. Sgarbossa, and J. O. Strandhagen. 2022. "Designing and Developing Smart Production Planning and Control Systems in the Industry 4.0 Era: A Methodology and Case Study." *Journal of Intelligent Manufacturing* 33 (1): 311–332. https://doi.org/10.1007/s10845-021-01808-w.

Parente, M., G. Figueira, P. Amorim, and A. Marques. 2020. "Production Scheduling in the Context of Industry 4.0: Review and Trends." *International Journal of Production Research* 58 (17): 5401–5431. https://doi.org/10.1080/00207543.2020.1718794.

Pfrommer, J., M. Schleipen, and J. Beyerer. 2013. "PPRS: Production Skills and Their Relation to Product, Process, and Resource." In *2013 IEEE 18th Conference on Emerging Technologies & Factory Automation (ETFA)*, Cagliari, Italy, 1–4. https://doi.org/10.1109/ETFA.2013.6648114.

Pinedo, M. 2022. *Scheduling: Theory, Algorithms, and Systems*. 6th ed. Cham: Springer.

Romero-Silva, R., J. Santos, and M. Hurtado-Hernández. 2024. "A Conceptual Framework of the Applicability of Production Scheduling from a Contingency Theory Approach: Addressing the Theory-Practice Gap." *Production Planning & Control* 35 (3): 262–282. https://doi.org/10.1080/09537287.2022.2076627.

Rossit, D. A., F. Tohmé, and M. Frutos. 2019. "A Data-Driven Scheduling Approach to Smart Manufacturing." *Journal of Industrial Information Integration* 15:69–79. https://doi.org/10.1016/j.jii.2019.04.003.

Rui, Z., X. Zhang, M. Liu, L. Ling, X. Wang, C. Liu, and M. Sun. 2024. "Graph Reinforcement Learning for Flexible Job Shop Scheduling under Industrial Demand Response: A Production and Energy Nexus Perspective." *Computers & Industrial Engineering* 193:110325. https://doi.org/10.1016/j.cie.2024.110325.

Sanlaville, E., and G. Schmidt. 1998. "Machine Scheduling with Availability Constraints." *Acta Informatica* 35 (9): 795–811. https://doi.org/10.1007/s002360050143.

Serrano-Ruiz, J. C., J. Mula, and R. Poler. 2021. "Smart Manufacturing Scheduling: A Literature Review." *Journal of Manufacturing Systems* 61:265–287. https://doi.org/10.1016/j.jmsy.2021.09.011.

Song, P., H. Chen, K. Cui, J. Wang, and D. Shi. 2025. "Meta-learning for Dynamic Multi-robot Task Scheduling." *Computers & Operations Research* 182:107109. https://doi.org/10.1016/j.cor.2025.107109.

Song, W., X. Chen, Q. Li, and Z. Cao. 2023. "Flexible Job-Shop Scheduling via Graph Neural Network and Deep Reinforcement Learning." *IEEE Transactions on Industrial Informatics* 19 (2): 1600–1610. https://doi.org/10.1109/TII.2022.3189725.

Thörnblad, K., A. B. Strömberg, M. Patriksson, and T. Almgren. 2015. "Scheduling Optimisation of a Real Flexible job Shop including Fixture Availability and Preventive Maintenance." *European Journal of Industrial Engineering* 9 (1): 126. https://doi.org/10.1504/EJIE.2015.067451.

Toptal, A., and I. Sabuncuoglu. 2010. "Distributed Scheduling: A Review of Concepts and Applications." *International Journal of Production Research* 48 (18): 5235–5262. https://doi.org/10.1080/00207540903121065.

Tully, S. M., and R. S. Winer. 2014. "The Role of the Beneficiary in Willingness to Pay for Socially Responsible Products: A Meta-analysis." *Journal of Retailing* 90 (2): 255–274. https://doi.org/10.1016/j.jretai.2014.03.004.

Vegetti, M., and G. Henning. 2014. *ISA-88 Formalization. A Step Towards Its Integration with the ISA-95 Standard*.

Venters, C. C., R. Capilla, S. Betz, B. Penzenstadler, T. Crick, S. Crouch, E. Y. Nakagawa, C. Becker, and C. Carrillo. 2018. "Software Sustainability: Research and Practice from a Software Architecture Viewpoint." *Journal of Systems and Software* 138:174–188. https://doi.org/10.1016/j.jss.2017.12.026.

Wang, Z., and M. Gombolay. 2022. "Stochastic Resource Optimization over Heterogeneous Graph Neural Networks for Failure-Predictive Maintenance Scheduling." *Proceedings of the International Conference on Automated Planning and Scheduling* 32:527–536. https://doi.org/10.1609/icaps.v32i1.19839.

Wang, Y., T. Xia, Y. Xu, G. Si, D. Wang, E. Pan, and L. Xi. 2024. "Quality-centered Production and Maintenance Scheduling for Multi-machine Manufacturing Systems under Variable Operating Conditions." *Reliability Engineering & System Safety* 250:110264. https://doi.org/10.1016/j.ress.2024.110264.

Xie, J., L. Gao, K. Peng, X. Li, and H. Li. 2019. "Review on Flexible Job Shop Scheduling." *IET Collaborative Intelligent Manufacturing* 1 (3): 67–77. https://doi.org/10.1049/iet-cim.2018.0009.

Zhang, J., C. Ellwein, M. Heithoff, J. Michael, and A. Wortmann. 2025. "Digital Twin and the Asset Administration Shell: An Analysis of the Three Types of AASs and Their Feasibility for Digital Twin Engineering." *Software and Systems Modeling* 24 (3): 771–793. https://doi.org/10.1007/s10270-024-01255-0.

Zhang, L., Y. Feng, Q. Xiao, Y. Xu, D. Li, D. Yang, and Z. Yang. 2023. "Deep Reinforcement Learning for Dynamic Flexible

Job Shop Scheduling Problem considering Variable Processing Times." *Journal of Manufacturing Systems* 71:257–273. https://doi.org/10.1016/j.jmsy.2023.09.009.

Zhang, F., Y. Mei, S. Nguyen, and M. Zhang. 2024. "Survey on Genetic Programming and Machine Learning Techniques for Heuristic Design in Job Shop Scheduling." *IEEE Transactions on Evolutionary Computation* 28 (1): 147–167. https://doi.org/10.1109/TEVC.2023.3255246.

Zhang, C., D. Song, C. Huang, A. Swami, and N. V. Chawla. 2019. "Heterogeneous Graph Neural Network." In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, Anchorage, AK, 793–803. https://doi.org/10.1145/3292500.3330961.

Zhao, F., T. Jiang, and L. Wang. 2023. "A Reinforcement Learning Driven Cooperative Meta-heuristic Algorithm for Energy-Efficient Distributed No-Wait Flow-Shop Scheduling with Sequence-Dependent Setup Time." *IEEE Transactions on Industrial Informatics* 19 (7): 8427–8440. https://doi.org/10.1109/TII.2022.3218645.

Zhao, Y., X. Luo, and Y. Zhang. 2024. "The Application of Heterogeneous Graph Neural Network and Deep Reinforcement Learning in Hybrid Flow Shop Scheduling Problem." *Computers & Industrial Engineering* 187:109802. https://doi.org/10.1016/j.cie.2023.109802.

Ziaee, M. 2014. "A Heuristic Algorithm for Solving Flexible Job Shop Scheduling Problem." *The International Journal of Advanced Manufacturing Technology* 71 (1–4): 519–528. https://doi.org/10.1007/s00170-013-5510-z.

Appendices

Appendix 1. Employed search terms

A.1 1. Employed search terms for 'Web of Science'

TI ((‘Production’ OR ‘manufacturing’ OR ‘fabrication’ OR ‘assembly’) AND (‘control’ OR ‘management‘ OR ‘operation’ OR ‘administration’ OR ‘regulation’ OR ‘planning’ OR ‘scheduling’) AND (‘sustainable’ OR ‘green’ OR ‘ecological’ OR ‘environment-friendly’ OR ‘environment friendly’ OR ‘eco-friendly’ OR ‘eco friendly‘ OR ‘energy-efficient’)) OR AB ((‘sustainable’ OR ‘green’ OR ‘ecological’ OR ‘environment-friendly’ OR ‘environment friendly’ OR ‘eco-friendly’ OR ‘eco friendly‘ OR ‘energy-efficient’) AND (‘manufacturing’ OR ‘fabrication’ OR ‘assembly’) AND (‘control’ OR ‘management‘ OR ‘operation’ OR ‘administration’ OR ‘regulation’ OR ‘planning’ OR ‘scheduling’)) NOT TI (‘food’ OR ‘plant’ OR ‘crop’ OR ‘farm∗’ OR ‘construction’ OR ‘Agricultur∗’ OR ‘Nano∗’ OR ‘supply chain management’ OR ‘SCM’ OR ‘human resource’ OR ‘HRM’) NOT AB (‘food’ OR ‘plant’ OR ‘crop’ OR ‘farm∗’ OR ‘construction’ OR ‘agricultur∗’ OR ‘Nano∗’)

A.2 2. Employed search terms for EBSCOhost

(TITLE ((‘Production’ OR ‘manufacturing’ OR ‘fabrication’ OR ‘assembly’) AND (‘control’ OR ‘management’ OR ‘operation’ OR ‘administration’ OR ‘regulation’ OR ‘planning’ OR ‘scheduling’) AND (‘sustainable’ OR ‘green’ OR ‘ecological’ OR ‘environment-friendly’ OR ‘environment friendly’ OR ‘eco-friendly’ OR ‘eco friendly’ OR ‘energy-efficient’)) OR ABS ((‘sustainable’ OR ‘green’ OR ‘ecological’ OR ‘environment-friendly’ OR ‘environment friendly’ OR ‘eco-friendly’ OR ‘eco friendly’ OR ‘energy-efficient’) AND (‘manufacturing’ OR ‘fabrication’ OR ‘assembly’) AND (‘control’ OR ‘management’ OR ‘operation’ OR ‘administration’ OR ‘regulation’ OR ‘planning’ OR ‘scheduling’)) AND NOT TITLE (‘food’ OR ‘plant’ OR ‘crop’ OR ‘farm∗’ OR ‘construction’ OR ‘Agricultur∗’ OR ‘Nano∗’ OR ‘supply chain management’ OR ‘SCM’ OR ‘human resource’ OR ‘HRM’) AND NOT ABS (‘food’ OR ‘plant’ OR ‘crop’ OR ‘farm∗’ OR ‘construction’ OR ‘agricultur∗’ OR ‘Nano∗’)) AND PUBYEAR > 2015 AND PUBYEAR < 2025 AND (LIMIT-TO (DOCTYPE, ‘ar’) OR LIMIT-TO (DOCTYPE, ‘cp’) OR LIMIT-TO (DOCTYPE, ‘cr’) OR LIMIT-TO (DOCTYPE, ‘sh’) OR LIMIT-TO (DOCTYPE, ‘no’) OR LIMIT-TO (DOCTYPE, ‘tb’) OR LIMIT-TO (DOCTYPE, ‘ed’) OR LIMIT-TO (DOCTYPE, ‘dp’)) AND (LIMIT-TO (LANGUAGE, ‘English’) OR LIMIT-TO (LANGUAGE, ‘German’)) AND (LIMIT-TO (EXACTKEYWORD, ‘Manufacture’) OR LIMIT-TO (EXACTKEYWORD, ‘Sustainability’) OR LIMIT-TO (EXACTKEYWORD, ‘Energy Efficiency’) OR LIMIT-TO (EXACTKEYWORD, ‘Manufacturing’) OR LIMIT-TO (EXACTKEYWORD, ‘Sustainable Manufacturing’) OR LIMIT-TO (EXACTKEYWORD, ‘Decision Making’) OR LIMIT-TO (EXACTKEYWORD, ‘Green Manufacturing’) OR LIMIT-TO (EXACTKEYWORD, ‘Industry 4.0’) OR LIMIT-TO (EXACTKEYWORD, ‘Scheduling’) OR LIMIT-TO (EXACTKEYWORD, ‘Production Control’))

A.3 3. Employed search terms for Scopus

((TI = (AND (‘Production’ OR ‘manufacturing’ OR ‘fabrication’ OR ‘assembly’) AND (‘control’ OR ‘management‘ OR ‘operation’ OR ‘administration’ OR ‘regulation’ OR ‘planning’ OR ‘scheduling’) AND (‘sustainable’ OR ‘green’ OR ‘ecological’ OR ‘environment-friendly’ OR ‘environment friendly’ OR ‘eco-friendly’ OR ‘eco friendly‘ OR ‘energy-efficient’))) OR AB = ((‘sustainable’ OR ‘green’ OR ‘ecological’ OR ‘environment-friendly’ OR ‘environment friendly’ OR ‘eco-friendly’ OR ‘eco friendly‘ OR ‘energy-efficient’) AND (‘manufacturing’ OR ‘fabrication’ OR ‘assembly’) AND (‘control’ OR ‘management‘ OR ‘operation’ OR ‘administration’ OR ‘regulation’ OR ‘planning’ OR ‘scheduling’))) NOT TI = (‘food’ OR ‘plant’ OR ‘crop’ OR ‘farm∗’ OR ‘construction’ OR ‘agricultur∗’ OR ‘Nano∗’ OR ‘supply chain management’ OR ‘SCM’ OR ‘human resource’ OR ‘HRM’) NOT AB = (‘food’ OR ‘plant’ OR ‘crop’ OR ‘farm∗’ OR ‘construction’ OR ‘agricultur∗’ OR ‘Nano∗’)

Appendix 2. Analysed final literature sources

Aikenov, T., Hidayat, R. and Wicaksono, H. 2024. "Power Consumption and Process Cost Prediction of Customized Products Using Explainable AI: A Case in the Steel Industry." Paper presented at Lecture Notes in Mechanical Engineering. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85172726580&doi=10.1007%2f978-3-031-38165-2_135&partnerID=40&md5=7f18bc86646d59c61a89b1dc0ee40324.

Attia, A.M., Alatwi, A.O., Al Hanbali, A. and Alsawafy, O.G. 2024. "Joint maintenance planning and production scheduling optimization model for green environment." *Journal of Quality in Maintenance Engineering* 30 (1): 153–174.

Bok, Y., Lee, N.K., Jo, S., Lee, S., Kweon, S.J. and Na, H.S. 2024. "The production scheduling problem employing non-identical parallel machines with due dates considering carbon emissions and multiple types of energy sources." *EXPERT SYSTEMS WITH APPLICATIONS* 238.

Chawla, S. and Singari, R.M. 2024. "Energy-Efficient Fuzzy Scheduling System for Crankcase Covers Manufacturing." *Journal of The Institution of Engineers (India): Series C*.

Chien, C.F., Kuo, *P*.C., Sun, *P*.C. and Kuo, H.A. 2024. "Green production planning for circular supply chain and resource management: An empirical study for high-tech textile dyeing." *RESOURCES CONSERVATION AND RECYCLING* 204.

Du, S.-L., Zhou, W.-J., Fei, M.-R., Nee, A. and Ong, S.K. 2024. "Bi-objective scheduling for energy-efficient distributed assembly blocking flow shop." *CIRP Annals*.

Ertem, M. 2024. "Renewable Energy-Aware Machine Scheduling Under Intermittent Energy Supply." *IEEE ACCESS* 12: 23613–23625.

Feng, X. and Peng, H. 2024. "Robust identical parallel machine scheduling with two-stage time-of-use tariff and not-all-machine option." *INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH* 62 (1–2): 380–403.

Fu, Y., Zhou, M., Guo, X., Qi, L., Gao, K. and Albeshri, A. 2024. "Multiobjective Scheduling of Energy-Efficient Stochastic Hybrid Open Shop With Brain Storm Optimization and Simulation Evaluation." *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 1–13.

Hudaifah, H., Andriansyah, A., Al-Shareef, K., Darghouth, M.N. and Saleh, H. 2024. "Genetic algorithm with normal boundary intersection for multi-objective early/tardy scheduling problem with carbon-emission consideration: a Pareto-optimum solution." *NEURAL COMPUTING & APPLICATIONS* 36 (5): 2493–2506.

Kong, M., Wu, *P*., Zhang, Y., Wang, W., Deveci, M. and Kadry, S. 2024. "Energy-efficient scheduling model and method for assembly blocking permutation flow-shop in industrial robotics field." *ARTIFICIAL INTELLIGENCE REVIEW* 57 (3).

Liu, H., Zhuang, *P*.T., Zhang, J. and Hu, K. 2024. "Optimizing Low-Carbon Job Shop Scheduling in Green Manufacturing with the Improved NSGAII Algorithm." Paper presented at Communications in Computer and Information Science. https://www.scopus.com/inward/record.uri?eid = 2-s2.0-85189524302&doi = 10.1007%2f978-3-031-56373-7_7&partnerID = 40&md5 = d0f2b57dd0be18469f7d6351b83c7158.

Loffredo, A., Frigerio, N., Lanzarone, E. and Matta, A. 2024. "Energy-efficient control in multi-stage production lines with parallel machine workstations and production constraints." *IISE TRANSACTIONS* 56 (1): 69–83.

Luo, C., Gong, W., Ming, F. and Lu, C. 2024. "A Q-learning memetic algorithm for energy-efficient heterogeneous distributed assembly permutation flowshop scheduling considering priorities." *SWARM AND EVOLUTIONARY COMPUTATION* 85.

Ostovari, A., Benyoucef, L., Benderbal, H.H. and Delorme, X. 2024. "Multi-Objective Workforce and Process Planning For Socio-Economic Sustainable RMS: Lp-metric vs Epsilon Constraint." Paper presented at Procedia Computer Science. https://www.scopus.com/inward/record.uri?eid = 2-s2.0-85189828464&doi = 10.1016%2fj.procs.2024.01.045&partnerID = 40&md5 = 9ae3bad863a951cc5bca352408f52148.

Pan, Z., Wang, L., Wang, J. and Zhang, Q. 2024. "A Bi-Learning Evolutionary Algorithm for Transportation-Constrained and Distributed Energy-Efficient Flexible Scheduling." *IEEE Transactions on Evolutionary Computation*, *p*. 1.

Tian, G., Wang, W., Zhang, H., Zhou, X., Zhang, C. and Li, Z. 2024. "Multi-objective optimization of energy-efficient remanufacturing system scheduling problem with lot-streaming production mode." *EXPERT SYSTEMS WITH APPLICATIONS* 237.

Tian, Z. and Zheng, L. 2024. "Single machine parallel-batch scheduling under time-of-use electricity prices: New formulations and optimisation approaches." *EUROPEAN JOURNAL OF OPERATIONAL RESEARCH* 312 (2): 512–524.

Wang, J., Tang, H. and Lei, D. 2024. "A feedback-based artificial bee colony algorithm for energy-efficient flexible flow shop scheduling problem with batch processing machines." *APPLIED SOFT COMPUTING* 153.

Wang, X.H., Dai, Y.*P*. and Jia, Z.Y. 2024. "Energy-efficient on/off control in serial production lines with Bernoulli machines." *FLEXIBLE SERVICES AND MANUFACTURING JOURNAL* 36 (1): 103–128.

Yağmur, E. and Kesen, S.E. 2024. "Integrated production scheduling and vehicle routing problem with energy efficient strategies: Mathematical formulation and metaheuristic algorithms." *EXPERT SYSTEMS WITH APPLICATIONS* 237.

Yu, F., Lu, C., Zhou, J.J., Yin, L. and Wang, K.*P*. 2024. "A knowledge-guided bi-population evolutionary algorithm for energy-efficient scheduling of distributed flexible job shop problem." *ENGINEERING APPLICATIONS OF ARTIFICIAL INTELLIGENCE* 128.

Zhang, F., Li, R. and Gong, W. 2024. "Deep reinforcement learning-based memetic algorithm for energy-aware flexible job shop scheduling with multi-AGV." *Computers and Industrial Engineering* 189.

Zhang, G., Yan, S., Song, X., Zhang, D. and Guo, S. 2024. "Evolutionary algorithm incorporating reinforcement learning for energy-conscious flexible job-shop scheduling problem with transportation and setup times." *ENGINEERING APPLICATIONS OF ARTIFICIAL INTELLIGENCE* 133.

Zheng, X. and Chen, Z. 2024. "An improved deep Q-learning algorithm for a trade-off between energy consumption and productivity in batch scheduling." *COMPUTERS & INDUSTRIAL ENGINEERING* 188.

An, X.X., Si, G.J., Xia, T.B., Wang, D., Pan, E.S. and Xi, L.F. 2023. "An energy-efficient collaborative strategy of maintenance planning and production scheduling for serial-parallel systems under time-of-use tariffs." *APPLIED ENERGY* 336.

Bao, H., Pan, Q., Rong, M., Yang, A. and Wang, X. 2023. "A Multi-action Reinforcement Learning Algorithm for Energy-efficiency Blocking Flow-shop Scheduling Problem." Paper presented at Proceedings of the 2023 26th International Conference on Computer Supported Cooperative Work in Design, CSCWD 2023. https://www.scopus.com/inward/record.uri?eid = 2-s2.0-85164739318&doi = 10.1109%2fCSCWD57460.2023.10152003&partnerID = 40&md5 = 913f7a5f1926f090214ac9314150735a.

Carlucci, D., Renna, *P*. and Materi, S. 2023. "A Job-Shop Scheduling Decision-Making Model for Sustainable Production Planning With Power Constraint." *IEEE TRANSACTIONS ON ENGINEERING MANAGEMENT* 70 (5): 1923–1932.

Chen, M. and Zio, E. 2023. "Energy-Efficient Scheduling of Flexible Manufacturing Systems Based on Petri Net." Paper presented at 2023 7th International Conference on System Reliability and Safety, ICSRS 2023. https://www.scopus.com/inward/record.uri?eid = 2-s2.0-85183457954&doi = 10.1109%2fICSR

S59833.2023.10381099&partnerID = 40&md5 = 147d37d1c8b8939b4ca7e544851eda84.

Cherian Jos, B., Varghese, A., Viswanath, K., Padmakumar, N. and Kurian, M. 2023. "A sustainable production scheduling model for parallel manufacturing units with multiple speed levels." *Materials Today: Proceedings*.

Destouet, C., Tlahig, H., Bettayeb, B. and Mazari, B. 2023. "NSGA-II for Solving a Multi-objective, Sustainable and Flexible Job Shop Scheduling Problem." Paper presented at IFIP Advances in Information and Communication Technology. https://www.scopus.com/inward/record.uri?eid = 2-s2.0-85174445432&doi = 10.1007%2f978-3-031-43670-3_38&partnerID = 40&md5 = cfab42c5bfa1142434d202bcce7d4fcf.

Füchtenhans, M. and Glock, C.H. 2023. "The impact of incentive-based programmes on job-shop scheduling with variable machine speeds." *INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH*.

Gaggero, M., Paolucci, M. and Ronco, R. 2023. "Exact and heuristic solution approaches for energy-efficient identical parallel machine scheduling with time-of-use costs." *EUROPEAN JOURNAL OF OPERATIONAL RESEARCH* 311 (3): 845–866.

Hager, T. and Hichem, H. 2023. "A PSO Approach for Job Shop Scheduling Problem with Energy Constraint." Paper presented at Lecture Notes in Mechanical Engineering. https://www.scopus.com/inward/record.uri?eid = 2-s2.0-85137003102&doi = 10.1007%2f978-3-031-14615-2_85&partnerID = 40&md5 = 5891b07ec60cd6c00ec662a30a9024ac.

Kaur, G. and Kander, R. 2023. "Supply Chain Simulation of Manufacturing Shirts Using System Dynamics for Sustainability." *SUSTAINABILITY* 15 (21).

Krstevski, S., Fatahi Valilai, O. and Wicaksono, H. 2023. "Integrating Real-Time Dynamic Electricity Price Forecast into Job Shop Production Scheduling Model with Multiple Machine Environments." Paper presented at ACM International Conference Proceeding Series. https://www.scopus.com/inward/record.uri?eid = 2-s2.0-85162878927&doi = 10.1145%2f3587889.3587905&partnerID = 40&md5 = 11a5faaa5b4c294c5b2e3ac8a3bfe338.

Li, C., Zhang, B., Han, Y., Wang, Y., Li, J. and Gao, K. 2023. "Energy-Efficient Hybrid Flowshop Scheduling with Consistent Sublots Using an Improved Cooperative Coevolutionary Algorithm." *MATHEMATICS* 11 (1).

Li, J., Li, H.K., He, *P*.B., Xu, L.*P*., He, K. and Liu, S.H. 2023. "Flexible Job Shop Scheduling Optimization for Green Manufacturing Based on Improved Multi-Objective Wolf Pack Algorithm." *APPLIED SCIENCES-BASEL* 13 (14).

Li, R., Gong, W., Lu, C. and Wang, L. 2023. "A Learning-Based Memetic Algorithm for Energy-Efficient Flexible Job-Shop Scheduling With Type-2 Fuzzy Processing Time." *IEEE Transactions on Evolutionary Computation* 27 (3): 610–620.

Li, R., Gong, W., Wang, L., Lu, C. and Zhuang, X. 2023. "Surprisingly Popular-Based Adaptive Memetic Algorithm for Energy-Efficient Distributed Flexible Job Shop Scheduling." *IEEE Transactions on Cybernetics* 53 (12): 8013–8023.

Li, Y., Hu, R., Wu, X., Qian, B. and Zhang, Z.-Q. 2023. "Learning Variable Neighborhood Search Algorithm for Solving the Energy-Efficient Flexible Job-Shop Scheduling Problem." Paper presented at Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). https://www.scopus.com/inward/record.uri?eid = 2-s2.0-85174716785&doi = 10.1007%2f978-981-99-4755-3_21&partnerID = 40&md5 = a3514e62b93225c60cbd71c21355feae.

Liu, C.H., Cai, W., Zhang, C.X. and Wei, F.F. 2023. "Data-driven intelligent control system in remanufacturing assembly for production and resource efficiency." *INTERNATIONAL JOURNAL OF ADVANCED MANUFACTURING TECHNOLOGY* 128 (7–8): 3531–3544.

Liu, Q.H., Wang, C.Y., Li, X.Y. and Gao, L. 2023. "A multi-population co-evolutionary algorithm for green integrated process planning and scheduling considering logistics system." *ENGINEERING APPLICATIONS OF ARTIFICIAL INTELLIGENCE* 126.

Loffredo, A., May, M.C. and Matta, A. 2023. "Reinforcement learning for energy-efficient control of multi-stage production lines with parallel machine workstations." Paper presented at Materials Research Proceedings. https://www.scopus.com/inward/record.uri?eid = 2-s2.0-85176014229&doi = 10.21741%2f9781644902714-51&partnerID = 40&md5 = 2a8d800c9e92766ba1cd225837fceae7.

Luan, F., Zhao, H., Liu, S.Q., He, Y. and Tang, B. 2023. "Enhanced NSGA-II for multi-objective energy-saving flexible job shop scheduling." *Sustainable Computing: Informatics and Systems* 39.

Mokhtari-Moghadam, A., Pourhejazy, *P*. and Gupta, D. 2023. "Integrating sustainability into production scheduling in hybrid flow-shop environments." *ENVIRONMENTAL SCIENCE AND POLLUTION RESEARCH*.

Nejad, M.G., Barenji, R.V., Güleryüz, G. and Shavarani, S.M. 2023. "Energy consumption optimization for sustainable flexible robotic cells: Proposing exact and metaheuristic methods." *ENERGY & ENVIRONMENT*.

Niu, W., Li, J.Q., Jin, H., Qi, R. and Sang, H.Y. 2023. "Bi-objective optimization using an improved NSGA-II for energy-efficient scheduling of a distributed assembly blocking flow-shop." *ENGINEERING OPTIMIZATION* 55 (5): 719–740.

Sanati, H., Moslehi, G. and Reisi-Nafchi, M. 2023. "Unrelated parallel machine energy-efficient scheduling considering sequence-dependent setup times and time-of-use electricity tariffs." *EURO Journal on Computational Optimization* 11.

Tan, B.R., Karabag, O. and Khayyati, S. 2023. "Energy-efficient production control of a make-to-stock system with buffer- and time-based policies." *INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH*.

Wang, J., Liu, Y., Ren, S., Wang, C. and Ma, S.Y. 2023. "Edge computing-based real-time scheduling for digital twin flexible job shop with variable time window." *ROBOTICS AND COMPUTER-INTEGRATED MANUFACTURING* 79.

Wang, W., Zhang, B. and Jia, B. 2023. "A Multiobjective Optimization Approach for Multiobjective Hybrid Flowshop Green Scheduling with Consistent Sublots." *Sustainability (Switzerland)* 15 (3).

Wang, Y.-J., Li, J. and Wang, G.-G. 2023. "Fuzzy correlation entropy-based NSGA-II for energy-efficient hybrid flow-shop scheduling problem." *Knowledge-Based Systems* 277.

Wang, Z., Shen, L., Li, X. and Gao, L. 2023. "An improved multi-objective firefly algorithm for energy-efficient hybrid flowshop rescheduling problem." *JOURNAL OF CLEANER PRODUCTION* 385.

Wanphian, T., Rerkjirattikal, *P*. and Olapiriyakul, S. 2023. "Machine Scheduling Approach for the Sustainable Development of a Residential Industrial Area." Paper presented at Advances in Transdisciplinary Engineering. https://www.scopu

s.com/inward/record.uri?eid = 2-s2.0-85173441040&doi = 10.3233%2fATDE230104&partnerID = 40&md5 = 0c1a6d679202f3096eefb557c4165b82.

Wen, X., Sun, Y., Ma, H.-L. and Chung, S.-H. 2023. "Green smart manufacturing: energy-efficient robotic job shop scheduling models." *INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH* 61 (17): 5791–5805.

Wen, X.H., Cao, H.J., Li, H.C., Ge, W.W. and Hon, B. 2023. "Operation state-based adaptive control in response to production disturbances of energy-efficient serial production lines." *APPLIED ENERGY* 347.

Xia, T.B., Sun, H., Ding, Y.T., Han, D.Y., Qin, W., Seidelmann, J. and Xi, L.F. 2023. "Digital twin-based real-time energy optimization method for production line considering fault disturbances." *JOURNAL OF INTELLIGENT MANUFACTURING*.

Xiaolong, C. and Yu, D. 2023. "Hybrid artificial immune algorithm for energy-efficient distributed flexible job shop in semiconductor manufacturing." *Cluster Computing*.

Xu, G., Bao, Q. and Zhang, H. 2023. "Multi-objective green scheduling of integrated flexible job shop and automated guided vehicles." *ENGINEERING APPLICATIONS OF ARTIFICIAL INTELLIGENCE* 126.

Xue, L. and Wang, X. 2023. "A multi-objective discrete differential evolution algorithm for energy-efficient two-stage flow shop scheduling under time-of-use electricity tariffs." *APPLIED SOFT COMPUTING* 133.

Yue, L., Wang, H., Mumtaz, J., Rauf, M. and Li, Z. 2023. "Energy-efficient scheduling of a two-stage flexible printed circuit board flow shop using a hybrid Pareto spider monkey optimisation algorithm." *JOURNAL OF INDUSTRIAL INFORMATION INTEGRATION* 31.

Yun, L.X., Wang, D. and Li, L. 2023. "Explainable multi-agent deep reinforcement learning for real-time demand response towards sustainable manufacturing." *APPLIED ENERGY* 347.

Zhang, L., Yan, Y. and Hu, Y. 2023. "Integrated Scheduling of Flexible Job Shop and Energy-Efficient Automated Guided Vehicles." Paper presented at 2023 8th IEEE International Conference on Advanced Robotics and Mechatronics, ICARM 2023. https://www.scopus.com/inward/record.uri?eid = 2-s2.0-85171557468&doi = 10.1109%2fICARM58088.2023.10218812&partnerID = 40&md5 = f73a98ac2a3f739f277e2a3c4c8b043d.

Zhang, W., Geng, H., Li, C., Gen, M., Zhang, G. and Deng, M. 2023. "Q-learning-based multi-objective particle swarm optimization with local search within factories for energy-efficient distributed flow-shop scheduling problem." *JOURNAL OF INTELLIGENT MANUFACTURING*.

Zhao, F., Jiang, T. and Wang, L. 2023. "A Reinforcement Learning Driven Cooperative Meta-Heuristic Algorithm for Energy-Efficient Distributed No-Wait Flow-Shop Scheduling With Sequence-Dependent Setup Time." *IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS* 19 (7): 8427–8440.

Zhao, F., Zhang, H. and Wang, L. 2023. "A Pareto-Based Discrete Jaya Algorithm for Multiobjective Carbon-Efficient Distributed Blocking Flow Shop Scheduling Problem." *IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS* 19 (8): 8588–8599.

Zhou, B.-H. and Zhang, J.-H. 2023. "An improved bi-objective salp swarm algorithm based on decomposition for green scheduling in flexible manufacturing cellular environments with multiple automated guided vehicles." *SOFT COMPUTING* 27 (22): 16717–16740.

Assou, B., Khettabi, I. and Benyoucef, L. 2022. "Operators Health and Safety Consideration in Sustainable Multi-objective Process and Production Planning for Reconfigurable Manufacturing System (RMS)." Paper presented at IEEE International Conference on Industrial Engineering and Engineering Management. https://www.scopus.com/inward/record.uri?eid = 2-s2.0-85146335747&doi = 10.1109%2fIEEM55944.2022.9989727&partnerID = 40&md5 = 1a4627be2755145a8fe28c0a9a927387.

Chang, JR, Yu, D., Zhou, Z., He, W.W. and Zhang, L.*P*. 2022. "Hierarchical Reinforcement Learning for Multi-Objective Real-Time Flexible Scheduling in a Smart Shop Floor." *MACHINES* 10 (12).

Chen, J., Ning, T., Xu, G. and Liu, Y. 2022. "A memetic algorithm for energy-efficient scheduling of integrated production and shipping." *INTERNATIONAL JOURNAL OF COMPUTER INTEGRATED MANUFACTURING* 35 (10–11): 1246–1268.

Chen, J.C., Jia, Z.Y. and Wang, X.H. 2022. "Dynamic performance prediction in flexible production lines with two geometric machines." *INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH* 60 (13): 4006–4024.

Chen, W.D., Wang, J.N., Yu, G.Y. and Hu, Y.M. 2022. "Energy-Efficient Hybrid Flow-Shop Scheduling under Time-of-Use and Ladder Electricity Tariffs." *APPLIED SCIENCES-BASEL* 12 (13).

Dong, J. and Ye, C. 2022. "Green scheduling of distributed two-stage reentrant hybrid flow shop considering distributed energy resources and energy storage system." *Computers and Industrial Engineering* 169.

Duan, J., Feng, M. and Zhang, Q. 2022. "Energy-efficient collaborative scheduling of heterogeneous multi-stage hybrid flowshop for large metallic component manufacturing." *JOURNAL OF CLEANER PRODUCTION* 375.

Eriksson, K., Ramasamy, S., Zhang, X., Wang, Z. and Danielsson, F. 2022. "Conceptual framework of scheduling applying discrete event simulation as an environment for deep reinforcement learning." Paper presented at Procedia CIRP. https://www.scopus.com/inward/record.uri?eid = 2-s2.0-85132264077&doi = 10.1016%2fj.procir.2022.05.091&partnerID = 40&md5 = 32ccbf9b3a144f7259cab6aa154477ea.

Füchtenhans, M. and Glock, C.H. 2022. "Energy-efficient job shop scheduling considering processing speed and incentive-based programs." Paper presented at IFAC-PapersOnLine. https://www.scopus.com/inward/record.uri?eid = 2-s2.0-85144562225&doi = 10.1016%2fj.ifacol.2022.09.389&partnerID = 40&md5 = 1708be41dafcbea9793427430ce2e005.

Gao, S., Liu, C. and Wang, G. (Eds.) 2022, *Multi-objective Green Flexible Workshop Scheduling Considering Multi-energy Consumption Factors* 2174.

He, L., Chiong, R., Li, W., Dhakal, S., Cao, Y. and Zhang, Y. 2022. "Multiobjective Optimization of Energy-Efficient JOB-Shop Scheduling with Dynamic Reference Point-Based Fuzzy Relative Entropy." *IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS* 18 (1): 600–610.

Jiang, T.H., Zhu, H.Q., Liu, L. and Gong, Q.T. 2022. "Energy-conscious flexible job shop scheduling problem considering transportation time and deterioration effect simultaneously."

SUSTAINABLE COMPUTING-INFORMATICS & SYSTEMS 35.

Khettabi, I., Benyoucef, L. and Boutiche, M.A. 2022. "Sustainable multi-objective process planning in reconfigurable manufacturing environment: adapted new dynamic NSGA-II vs New NSGA-III." *INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH* 60 (20): 6329–6349.

Kong, T.Q., Nhg and Shi, V.T. 2022. "Method for Optimizing Energy Consumption in Machining Manufacturing Process." *MATHEMATICAL PROBLEMS IN ENGINEERING* 2022.

Liu, M., Yang, T. and Li, Q.J. 2022. "Dynamic Production Control and Optimization under Low Carbon Economy considering Delivery Time Demand." *COMPUTATIONAL INTELLIGENCE AND NEUROSCIENCE* 2022.

Liu, Z.C., Luo, Q., Wang, L., Tang, H.T. and Li, Y.B. 2022. "The Low-Carbon Scheduling Optimization of Integrated Multispeed Flexible Manufacturing and Multi-AGV Transportation." *PROCESSES* 10 (10).

Lu, C., Huang, Y., Meng, L., Gao, L., Zhang, B. and Zhou, J. 2022. "A Pareto-based collaborative multi-objective optimization algorithm for energy-efficient scheduling of distributed permutation flow-shop with limited buffers." *ROBOTICS AND COMPUTER-INTEGRATED MANUFACTURING* 74.

Lu, C., Zhang, B., Gao, L., Yi, J. and Mou, J. 2022. "A Knowledge-Based Multiobjective Memetic Algorithm for Green Job Shop Scheduling With Variable Machining Speeds." *IEEE Systems Journal* 16 (1): 844–855.

Lv, Y., Li, C.B., Tang, Y. and 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.

Mou, J., Duan, *P*., Gao, L., Liu, X. and Li, J. 2022. "An effective hybrid collaborative algorithm for energy-efficient distributed permutation flow-shop inverse scheduling." *Future Generation Computer Systems* 128: 521–537.

Oukil, A., El-Bouri, A. and Emrouznejad, A. 2022. "Energy-aware job scheduling in a multi-objective production environment – An integrated DEA-OWA model." *Computers and Industrial Engineering* 168.

Peng, Z., Zhang, H., Tang, H., Feng, Y. and Yin, W. 2022. "Research on flexible job-shop scheduling problem in green sustainable manufacturing based on learning effect." *JOURNAL OF INTELLIGENT MANUFACTURING* 33 (6): 1725–1746.

Sang, Y. and Tan, J. 2022. "Intelligent factory many-objective distributed flexible job shop collaborative scheduling method." *Computers and Industrial Engineering* 164.

Scholz, S. and Meisel, F. 2022. "Coordination of heterogeneous production equipment under an external signal for sustainable energy." *JOURNAL OF CLEANER PRODUCTION* 338, N.PAG-N.PAG.

Schulz, S., Schönheit, M. and Neufeld, J.S. 2022. "Multi-objective carbon-efficient scheduling in distributed permutation flow shops under consideration of transportation efforts." *JOURNAL OF CLEANER PRODUCTION* 365.

Utama, D.M., Ibrahim, M.F., Wijaya, D.S., Widodo, D.S. and Primayesti, M.D. (Eds.) 2022, *A Novel Hybrid Multi-Verse Optimizer Algorithm for Energy-Efficient Permutation Flow Shop Scheduling Problem* 2394.

Utama, D.M., Salim, A. and Widodo, D.S. 2022. "A novel hybrid archimedes optimization algorithm for energy-efficient hybrid flow shop scheduling." *International Journal of Advances in Intelligent Informatics* 8 (2): 237–250.

Wang, J.-J. and Wang, L. 2022. "A Cooperative Memetic Algorithm with Learning-Based Agent for Energy-Aware Distributed Hybrid Flow-Shop Scheduling." *IEEE Transactions on Evolutionary Computation* 26 (3): 461–475.

Wei, Z.Z., Liao, W.Z. and Zhang, L.Y. 2022. "Hybrid energy-efficient scheduling measures for flexible job-shop problem with variable machining speeds." *EXPERT SYSTEMS WITH APPLICATIONS* 197.

Zhang, Z.-Q., Hu, R., Qian, B., Jin, H.-*P*., Wang, L. and Yang, J.-B. 2022. "A matrix cube-based estimation of distribution algorithm for the energy-efficient distributed assembly permutation flow-shop scheduling problem." *EXPERT SYSTEMS WITH APPLICATIONS* 194.

Barak, S., Moghdani, R. and Maghsoudlou, H. 2021. "Energy-efficient multi-objective flexible manufacturing scheduling." *JOURNAL OF CLEANER PRODUCTION* 283.

Cheng, L., Tang, Q., Zhang, L. and Meng, K. 2021. "Mathematical model and enhanced cooperative co-evolutionary algorithm for scheduling energy-efficient manufacturing cell." *JOURNAL OF CLEANER PRODUCTION* 326.

Cota, L.*P*., Coelho, V.N., Guimaraes, F.G. and Souza, M.J. 2021. "Bi-criteria formulation for green scheduling with unrelated parallel machines with sequence-dependent setup times." *INTERNATIONAL TRANSACTIONS IN OPERATIONAL RESEARCH* 28 (2): 996–1017.

Cui, *P*.H., Wang, J.Q., Li, Y. and Yan, F.Y. 2021. "Energy-efficient control in serial production lines: Modeling, analysis and improvement∗." *JOURNAL OF MANUFACTURING SYSTEMS* 60: 11–21.

Danishvar, M., Danishvar, S., Katsou, E., Mansouri, S.A. and Mousavi, A. 2021. "Energy-Aware Flowshop Scheduling: A Case for AI-Driven Sustainable Manufacturing." *IEEE ACCESS* 9: 141678–141692.

Diaz, J.L. and Ocampo-Martinez, C. 2021. "Non-centralised control strategies for energy-efficient and flexible manufacturing systems." *JOURNAL OF MANUFACTURING SYSTEMS* 59: 386–397.

Frigerio, N., Marzano, L. and Matta, A. 2021. "An Online Policy for Energy-Efficient State Control of Manufacturing Equipment." *IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING* 18 (2): 705–716.

Gao, S.N., Daaboul, J. and Le Duigou, J. 2021. "Process Planning, Scheduling, and Layout Optimization for Multi-Unit Mass-Customized Products in Sustainable Reconfigurable Manufacturing System." *SUSTAINABILITY* 13 (23).

Gong, G.L., Chiong, R., Deng, Q.W., Han, W.W., Zhang, L.K. and Huang, D. 2021. "Energy-efficient production scheduling through machine on/off control during preventive maintenance." *ENGINEERING APPLICATIONS OF ARTIFICIAL INTELLIGENCE* 104.

Ham, A., Park, M.-J. and Kim, K.M. 2021. "Energy-Aware Flexible Job Shop Scheduling Using Mixed Integer Programming and Constraint Programming." *MATHEMATICAL PROBLEMS IN ENGINEERING* 2021.

Ho, M.H., Hnaien, F. and Dugardin, F. 2021. "Electricity cost minimisation for optimal makespan solution in flow shop scheduling under time-of-use tariffs." *INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH* 59 (4): 1041–1067.

Kampa, A. and Paprocka, I. 2021. "Analysis of Energy Efficient Scheduling of the Manufacturing Line with Finite Buffer

Capacity and Machine Setup and Shutdown Times." *ENERGIES* 14 (21).

Li, H., Duan, J. and Zhang, Q. 2021. "Multi-objective integrated scheduling optimization of semi-combined marine crankshaft structure production workshop for green manufacturing." *TRANSACTIONS OF THE INSTITUTE OF MEASUREMENT AND CONTROL* 43 (3): 579–596.

Li, Y.-Z., Pan, Q.-K., Gao, K.-Z., Tasgetiren, M.F., Zhang, B. and Li, J.-Q. 2021. "A green scheduling algorithm for the distributed flowshop problem." *APPLIED SOFT COMPUTING* 109.

Lu, C., Gao, L., Gong, W., Hu, C., Yan, X. and Li, X. 2021. "Sustainable scheduling of distributed permutation flow-shop with non-identical factory using a knowledge-based multi-objective memetic optimization algorithm." *SWARM AND EVOLUTIONARY COMPUTATION* 60.

Pei, Z., Wan, M., Jiang, Z.-Z., Wang, Z. and Dai, X. 2021. "An Approximation Algorithm for Unrelated Parallel Machine Scheduling under TOU Electricity Tariffs." *IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING* 18 (2): 743–756.

Qin, H., Wang, Y., Han, Y., Chen, Q. and Li, J. 2021. "Adapting a reinforcement learning method for the distributed blocking hybrid flow shop scheduling problem." Paper presented at Proceedings of 2021 5th Asian Conference on Artificial Intelligence Technology, ACAIT 2021. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85127641540&doi=10.1109%2fACAIT53529.2021.9731228&partnerID=40&md5=26878d010f844b4de36e7654a5a4624e.

Qin, H.-X., Han, Y., Li, J., Sang, H., Chen, Q., Meng, L. and Zhang, B. 2021. "A quick and effective iterated greedy algorithm for energy-efficient hybrid flow shop scheduling problem with blocking constraint." Paper presented at 9440648. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85107976012&doi=10.1109%2fICIST52614.2021.9440648&partnerID=40&md5=ee6b90ec0450ce8ede3cec7c7dc48c24.

Sang, Y.W., Tan, J.*P*. and Liu, W. 2021. "A new many-objective green dynamic scheduling disruption management approach for machining workshop based on green manufacturing." *JOURNAL OF CLEANER PRODUCTION* 297.

Shen, K., Pessemier, T. de, Martens, L. and Joseph, W. 2021. "A parallel genetic algorithm for multi-objective flexible flowshop scheduling in pasta manufacturing." *Computers and Industrial Engineering* 161.

Shi, L., Guo, G. and Song, X. 2021. "Multi-agent based dynamic scheduling optimisation of the sustainable hybrid flow shop in a ubiquitous environment." *INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH* 59 (2): 576–597.

Song, C. 2021. "A Hybrid Multi-Objective Teaching-Learning Based Optimization for Scheduling Problem of Hybrid Flow Shop with Unrelated Parallel Machine." *IEEE ACCESS* 9: 56822–56835.

Wang, J., Liu, Y., Ren, S., Wang, C. and Wang, W. 2021. "Evolutionary game based real-time scheduling for energy-efficient distributed and flexible job shop." *JOURNAL OF CLEANER PRODUCTION* 293.

Wu, M.L., Yang, D.S., Zhou, B.W., Yang, Z.L., Liu, T.Y., Li, L.G., Wang, Z.F. and Hu, K.Y. 2021. "Adaptive Population NSGA-III with Dual Control Strategy for Flexible Job Shop Scheduling Problem with the Consideration of Energy Consumption and Weight." *MACHINES* 9 (12).

Xiao, Y., Zheng, Y., Yu, Y., Zhang, L., Lin, X. and Li, B. 2021. "A branch and bound algorithm for a parallel machine scheduling problem in green manufacturing industry considering time cost and power consumption." *JOURNAL OF CLEANER PRODUCTION* 320.

Zhang, H., Xu, G., Pan, R. and Ge, H. 2021. "A novel heuristic method for the energy-efficient flexible job-shop scheduling problem with sequence-dependent set-up and transportation time." *ENGINEERING OPTIMIZATION*.

Zhang, Y.C., Sun, Z.Y., Qin, R.W. and Xiong, H.Y. 2021. "Idle Duration Prediction for Manufacturing System Using a Gaussian Mixture Model Integrated Neural Network for Energy Efficiency Improvement." *IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING* 18 (1): 47–55.

Zhao, F., He, X. and Wang, L. 2021. "A two-stage cooperative evolutionary algorithm with problem-specific knowledge for energy-efficient scheduling of no-wait flow-shop problem." *IEEE Transactions on Cybernetics* 51 (11): 5291–5303.

Zhou, B.H. and Lei, Y.R. 2021. "Bi-objective grey wolf optimization algorithm combined Levy flight mechanism for the FMC green scheduling problem." *APPLIED SOFT COMPUTING* 111.

Ambrogio, G., Guido, R., Palaia, D. and Filice, L. 2020. "Job shop scheduling model for a sustainable manufacturing." Paper presented at Procedia Manufacturing. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85084213558&doi=10.1016%2fj.promfg.2020.02.034&partnerID=40&md5=8502060682c81bd5821f0f5d0a96e2d9.

An, Y.J., Chen, X.H., Zhang, J. and Li, Y.H. 2020. "A hybrid multi-objective evolutionary algorithm to integrate optimization of the production scheduling and imperfect cutting tool maintenance considering total energy consumption." *JOURNAL OF CLEANER PRODUCTION* 268.

Caldeira, R.H., Gnanavelbabu, A. and Vaidyanathan, T. 2020. "An effective backtracking search algorithm for multi-objective flexible job shop scheduling considering new job arrivals and energy consumption." *Computers and Industrial Engineering* 149.

Chen, T.-L., Cheng, C.-Y. and Chou, Y.-H. 2020. "Multi-objective genetic algorithm for energy-efficient hybrid flow shop scheduling with lot streaming." *ANNALS OF OPERATIONS RESEARCH* 290 (1–2): 813–836.

Cui, W.W., Sun, H.L. and Xia, B.X. 2020. "Integrating production scheduling, maintenance planning and energy controlling for the sustainable manufacturing systems under TOU tariff." *JOURNAL OF THE OPERATIONAL RESEARCH SOCIETY* 71 (11): 1760–1779.

Ebrahimi, A., Jeon, H.W., Lee, S. and Wang, C. 2020. "Minimizing total energy cost and tardiness penalty for a scheduling-layout problem in a flexible job shop system: A comparison of four metaheuristic algorithms." *COMPUTERS & INDUSTRIAL ENGINEERING* 141.

Faraji Amiri, M. and Behnamian, J. 2020. "Multi-objective green flowshop scheduling problem under uncertainty: Estimation of distribution algorithm." *JOURNAL OF CLEANER PRODUCTION* 251.

Feng, Y.X., Hong, Z.X., Li, Z.W., Zheng, H. and Tan, JR 2020. "Integrated intelligent green scheduling of sustainable flexible workshop with edge computing considering uncertain machine state." *JOURNAL OF CLEANER PRODUCTION* 246.

Geng, K.F., Ye, C.M., Dai, Z.H. and Liu, L. 2020. "Bi-Objective Re-Entrant Hybrid Flow Shop Scheduling considering

Energy Consumption Cost under Time-of-Use Electricity Tariffs." *COMPLEXITY* 2020.

Gong, G., Chiong, R., Deng, Q., Han, W., Zhang, L., Lin, W. and Li, K. 2020. "Energy-efficient flexible flow shop scheduling with worker flexibility." *EXPERT SYSTEMS WITH APPLICATIONS* 141.

Homayouni, S.M. and Fontes, D. 2020. "Optimization of Sustainable Single-Machine Scheduling Problem Short Research Paper, CSCI-ISCI." Paper presented at 9458088. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85113379914&doi=10.1109%2fCSCI51800.2020.00280&partnerID=40&md5=724597e00acc5e1c5204763037f82289.

Huo, D.X., Xiao, X.J. and Pan, Y.J. 2020. "Multi-objective energy-saving job-shop scheduling based on improved NSGA-II." *INTERNATIONAL JOURNAL OF SIMULATION MODELLING* 19 (3): 494–504.

Li, H., Zhu, H. and Jiang, T. 2020. "Modified migrating birds optimization for energy-aware flexible job shop scheduling problem." *Algorithms* 13 (2).

Li, L.L., Li, C.B., Tang, Y., Li, L. and Chen, X.Z. 2020. "An Integrated Solution to Minimize the Energy Consumption of a Resource-Constrained Machining System." *IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING* 17 (3): 1158–1175.

Liu, N., Zhang, Y.F. and Lu, W.F. 2020. "Energy-efficient integration of process planning and scheduling in discrete parts manufacturing with a heuristic-based two-stage approach." *INTERNATIONAL JOURNAL OF ADVANCED MANUFACTURING TECHNOLOGY* 106 (5–6): 2415–2432.

Liu, Z., Yan, J., Cheng, Q., Yang, C., Sun, S. and Xue, D. 2020. "The mixed production mode considering continuous and intermittent processing for an energy-efficient hybrid flow shop scheduling." *JOURNAL OF CLEANER PRODUCTION* 246.

Lu, H. and Qiao, F. 2020. "A Sustainable Parallel-machine Scheduling Problem with Time Constraint Based on Hybrid Metaheuristic Algorithm." Paper presented at 9327164. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85100926643&doi=10.1109%2fCAC51589.2020.9327164&partnerID=40&md5=138858166001a874a7cf5fa96b834b38.

Qi, Y., Fu, Y., Sun, C. and Guo, X. 2020. "Multi-objective Parallel-machine Scheduling Problem Considering Energy Consumption and Deteriorating Effect." Paper presented at 9327353. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85100929005&doi=10.1109%2fCAC51589.2020.9327353&partnerID=40&md5=9a15b1cd39494169228072032 4a64fea.

Soares, J., Canizes, B., Faria, *P*., Vale, Z. and Ramos, C. 2020. "Production scheduling considering dynamic electricity price in energy-efficient factories." Paper presented at IFAC-PapersOnLine. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85105102268&doi=10.1016%2fj.ifacol.2020.12.1823&partnerID=40&md5=8a1e352817f9bfa2f04cfb869fa3c219.

Sun, Z.Y., Dababneh, F. and Li, L. 2020. "Joint Energy, Maintenance, and Throughput Modeling for Sustainable Manufacturing Systems." *IEEE TRANSACTIONS ON SYSTEMS MAN CYBERNETICS-SYSTEMS* 50 (6): 2101–2112.

Tsao, Y.C., Thanh, V.V. and Hwang, F.J. 2020. "Energy-efficient single-machine scheduling problem with controllable job processing times under differential electricity pricing." *RESOURCES CONSERVATION AND RECYCLING* 161.

Wang, J.-J. and Wang, L. 2020. "A Knowledge-Based Cooperative Algorithm for Energy-Efficient Scheduling of Distributed Flow-Shop." *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 50 (5): 1805–1819.

Wang, S., Wang, X., Chu, F. and Yu, J. 2020. "An energy-efficient two-stage hybrid flow shop scheduling problem in a glass production." *INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH* 58 (8): 2283–2314.

Wen, X., Wang, K., Sun, H., Li, H. and Zhang, W. 2020. "An Effective Two-Stage Optimization Method Based on NSGA-II for Green Multi-objective Integrated Process Planning and Scheduling Problem." Paper presented at Communications in Computer and Information Science. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85083980974&doi=10.1007%2f978-981-15-3425-6_50&partnerID=40&md5=d8d5a671da6f61604cc5e479b59e5bfb.

Xu, Z.J., Zheng, Z. and Gao, X.Q. 2020. "Energy-efficient steelmaking-continuous casting scheduling problem with temperature constraints and its solution using a multi-objective hybrid genetic algorithm with local search." *APPLIED SOFT COMPUTING* 95.

Zandi, A., Ramezanian, R. and Monplaisir, L. 2020. "Green parallel machines scheduling problem: A bi-objective model and a heuristic algorithm to obtain Pareto frontier." *JOURNAL OF THE OPERATIONAL RESEARCH SOCIETY* 71 (6): 967–978.

Zhang, X., Zhang, H. and Yao, J. 2020. "Multi-Objective Optimization of Integrated Process Planning and Scheduling Considering Energy Savings." *ENERGIES* 13 (23).

Zheng, X., Zhou, S., Xu, R. and Chen, H. 2020. "Energy-efficient scheduling for multi-objective two-stage flow shop using a hybrid ant colony optimisation algorithm." *INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH* 58 (13): 4103–4120.

Zhou, S., Jin, M. and Du, N. 2020. "Energy-efficient scheduling of a single batch processing machine with dynamic job arrival times." *ENERGY* 209.

Caro-Ruiz, C., Lombardi, *P*., Richter, M., Pelzer, A., Komarnicki, *P*., Pavas, A. and Mojica-Nava, E. 2019. "Coordination of optimal sizing of energy storage systems and production buffer stocks in a net zero energy factory." *APPLIED ENERGY* 238: 851–862.

Chen, J.-F., Wang, L. and Peng, Z.-*P*. 2019. "A collaborative optimization algorithm for energy-efficient multi-objective distributed no-idle flow-shop scheduling." *SWARM AND EVOLUTIONARY COMPUTATION* 50.

Dai, M., Tang, D.B., Adriana, G. and Miguel, A.S. 2019. "Multi-objective optimization for energy-efficient flexible job shop scheduling problem with transportation constraints." *ROBOTICS AND COMPUTER-INTEGRATED MANUFACTURING* 59: 143–157.

Fu, Y.*P*., Tian, G.D., am Fathollahi-Fard, Ahmadi, A. and Zhang, C.Y. 2019. "Stochastic multi-objective modelling and optimization of an energy-conscious distributed permutation flow shop scheduling problem with the total tardiness constraint." *JOURNAL OF CLEANER PRODUCTION* 226: 515–525.

Huang, J., Chang, Q., Arinez, J. and Xiao, G. 2019. "A maintenance and energy saving joint control scheme for sustainable manufacturing systems." Paper presented at Procedia CIRP. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85067183212&doi=10.1016%2fj.procir.2019.01.073&partnerID=40&md5=530d70b38db81a17c040e186cb7cb2b0.

Jiang, E.-D. and Wang, L. 2019. "An improved multi-objective evolutionary algorithm based on decomposition for energy-efficient permutation flow shop scheduling problem with sequence-dependent setup time." *INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH* 57 (6): 1756–1771.

Jiang, S.-L. and Zhang, L. 2019. "Energy-Oriented Scheduling for Hybrid Flow Shop with Limited Buffers Through Efficient Multi-Objective Optimization." *IEEE ACCESS* 7: 34477–34487.

Li, Y., Wang, J.Q. and Chang, Q. 2019. "Event-Based Production Control for Energy Efficiency Improvement in Sustainable Multistage Manufacturing Systems." *JOURNAL OF MANUFACTURING SCIENCE AND ENGINEERING-TRANSACTIONS OF THE ASME* 141 (2).

Liao, W. and Wang, T. 2019. "A novel collaborative optimization model for job shop production-delivery considering time window and carbon emission." *Sustainability (Switzerland)* 11 (10).

Liu, N., Zhang, Y.F. and Lu, W.F. 2019. "Improving Energy Efficiency in Discrete Parts Manufacturing System Using an Ultra-Flexible Job Shop Scheduling Algorithm." *INTERNATIONAL JOURNAL OF PRECISION ENGINEERING AND MANUFACTURING-GREEN TECHNOLOGY* 6 (2): 349–365.

Liu, Y., Liao, X. and Zhang, R. 2019. "An enhanced MOPSO algorithm for energy-efficient single-machine production scheduling." *Sustainability (Switzerland)* 11 (19).

Liu, Z., Guo, S. and 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.

Lu, Y. and Jiang, T. 2019. "Bi-Population Based Discrete Bat Algorithm for the Low-Carbon Job Shop Scheduling Problem." *IEEE ACCESS* 7: 14513–14522.

Lu, Y., Lu, J. and Jiang, T. 2019. "Energy-conscious scheduling problem in a flexible job shop using a discrete water wave optimization algorithm." *IEEE ACCESS* 7: 101561–101574.

Luo, J., Fujimura, S., El Baz, D. and Plazolles, B. 2019. "GPU based parallel genetic algorithm for solving an energy efficient dynamic flexible flow shop scheduling problem." *Journal of Parallel and Distributed Computing* 133: 244–257.

Luo, S., Zhang, L.X. and Fan, Y.S. 2019. "Energy-efficient scheduling for multi-objective flexible job shops with variable processing speeds by grey wolf optimization." *JOURNAL OF CLEANER PRODUCTION* 234: 1365–1384.

Luo, Y., Lu, C., Li, X., Wang, L. and Gao, L. 2019. "Green job shop scheduling problem with machine at different speeds using a multi-objective grey Wolf optimization algorithm." Paper presented at 8843132. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85072956703&doi=10.1109%2fCOASE.2019.8843132&partnerID=40&md5=aae91b0a7cd0d38f90fa51e976740929.

Mirahmadi, N. and Taghipour, S. (Eds.) 2019, *Energy-efficient optimization of flexible job shop scheduling and preventive maintenance*, 2019-January.

Nicolò, G., Ferrer, S., Salido, M.A., Giret, A. and Barber, F. 2019. "A multi-agent framework to solve energy-aware unrelated parallel machine scheduling problems with machine-dependent energy consumption and sequence-dependent setup time." Paper presented at Proceedings International Conference on Automated Planning and Scheduling, ICAPS. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85085600472&partnerID=40&md5=2b238fa1ccd437455781fc6852df312c.

Nouiri, M., Trentesaux, D. and Bekrar, A. 2019. "Towards Energy Efficient Scheduling of Manufacturing Systems through Collaboration between Cyber Physical Production and Energy Systems." *ENERGIES* 12 (23).

Ramezanian, R., Vali-Siar, M.M. and Jalalian, M. 2019. "Green permutation flowshop scheduling problem with sequence-dependent setup times: a case study." *INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH* 57 (10): 3311–3333.

Rubaiee, S. and Yildirim, M.B. 2019. "An energy-aware multiobjective ant colony algorithm to minimize total completion time and energy cost on a single-machine preemptive scheduling." *COMPUTERS & INDUSTRIAL ENGINEERING* 127: 240–252.

Safarzadeh, H. and Niaki, S.T. 2019. "Bi-objective green scheduling in uniform parallel machine environments." *JOURNAL OF CLEANER PRODUCTION* 217: 559–572.

Tian, S.L., Wang, T.Y., Zhang, L. and Wu, X.Q. 2019. "An Energy-Efficient Scheduling Approach for Flexible Job Shop Problem in an Internet of Manufacturing Things Environment." *IEEE ACCESS* 7: 62695–62704.

Tigane, M., Dahane, M. and Boudhar, M. 2019. "Multiobjective approach for deteriorating jobs scheduling for a sustainable manufacturing system." *INTERNATIONAL JOURNAL OF ADVANCED MANUFACTURING TECHNOLOGY* 101 (5–8): 1939–1957.

Wang, J., Yao, S., Sheng, J. and Yang, H. 2019. "Minimizing total carbon emissions in an integrated machine scheduling and vehicle routing problem." *JOURNAL OF CLEANER PRODUCTION* 229: 1004–1017.

Wang, J.F., Fei, Z.C., Chang, Q. and Li, S.Q. 2019. "Energy Saving Operation of Manufacturing System Based on Dynamic Adaptive Fuzzy Reasoning Petri Net." *ENERGIES* 12 (11).

Wang, J.F., Fei, Z.C., Chang, Q., Li, S.Q. and Fu, Y. 2019. "Multi-state decision of unreliable machines for energy-efficient production considering work-in-process inventory." *INTERNATIONAL JOURNAL OF ADVANCED MANUFACTURING TECHNOLOGY* 102 (1–4): 1009–1021.

Wang, J.F., Huang, Y.Q., Chang, Q. and Li, S.Q. 2019. "Event-Driven Online Machine State Decision for Energy-Efficient Manufacturing System Based on Digital Twin Using Max-Plus Algebra." *SUSTAINABILITY* 11 (18).

Wu, X., Shen, X. and Li, C. 2019. "The flexible job-shop scheduling problem considering deterioration effect and energy consumption simultaneously." *Computers and Industrial Engineering* 135: 1004–1024.

Xu, W.G. and Cao, L. 2019. "Optimal maintenance control of machine tools for energy efficient manufacturing." *INTERNATIONAL JOURNAL OF ADVANCED MANUFACTURING TECHNOLOGY* 104 (9–12): 3303–3311.

Xu, W.X. and Guo, S.S. 2019. "A Multi-Objective and Multi-Dimensional Optimization Scheduling Method Using a Hybrid

Evolutionary Algorithms with a Sectional Encoding Mode." *SUSTAINABILITY* 11 (5).

Zhang, B., Pan, Q.-K., Gao, L., Li, X.-Y., Meng, L.-L. and Peng, K.-K. 2019. "A multiobjective evolutionary algorithm based on decomposition for hybrid flowshop green scheduling problem." *Computers and Industrial Engineering* 136: 325–344.

Zhang, Z.W., Wu, L.H., Peng, T. and Jia, S. 2019. "An Improved Scheduling Approach for Minimizing Total Energy Consumption and Makespan in a Flexible Job Shop Environment." *SUSTAINABILITY* 11 (1).

Zhao, J.L., Peng, S.T., Li, T., Lv, S.P., Li, M.Y. and Zhang, H.C. 2019. "Energy-aware fuzzy job-shop scheduling for engine remanufacturing at the multi-machine level." *FRONTIERS OF MECHANICAL ENGINEERING* 14 (4): 474–488.

Dai, M. and Zhang, Y. 2018. "An improved hybrid algorithm for minimizing energy consumption and cycle time in flexible process planning." Paper presented at Proceedings of the 30th Chinese Control and Decision Conference, CCDC 2018. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85050871265&doi=10.1109%2fCCDC.2018.8407472&partnerID=40&md5=a26c4e02214136e6fc0b78f7adc61fd0.

Fazli Khalaf, A. and Wang, Y. 2018. "Energy-cost-aware flow shop scheduling considering intermittent renewables, energy storage, and real-time electricity pricing." *INTERNATIONAL JOURNAL OF ENERGY RESEARCH* 42 (12): 3928–3942.

Fei, Z., Li, S., Chang, Q., Wang, J. and Huang, Y. 2018. "Fuzzy Petri Net Based Intelligent Machine Operation of Energy Efficient Manufacturing System." Paper presented at 8560366. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85059972697&doi=10.1109%2fCOASE.2018.8560366&partnerID=40&md5=e8ab035ba9ebfc16c812f392c443b904.

Feng, Y., Wang, Q., Gao, Y., Cheng, J. and Tan, J. 2018. "Energy-Efficient Job-Shop Dynamic Scheduling System Based on the Cyber-Physical Energy-Monitoring System." *IEEE ACCESS* 6: 52238–52247.

Golpîra, H., Khan, S.A. and Zhang, Y. 2018. "Robust Smart Energy Efficient Production Planning for a general Job-Shop Manufacturing System under combined demand and supply uncertainty in the presence of grid-connected microgrid." *JOURNAL OF CLEANER PRODUCTION* 202: 649–665.

Gong, G., Deng, Q., Gong, X., Liu, W. and Ren, Q. 2018. "A new double flexible job-shop scheduling problem integrating processing time, green production, and human factor indicators." *JOURNAL OF CLEANER PRODUCTION* 174: 560–576.

Iqbal, A. and Al-Ghamdi, K.A. 2018. "Energy-efficient cellular manufacturing system: Eco-friendly revamping of machine shop configuration." *ENERGY* 163: 863–872.

Jiang, T.H., Zhang, C., Zhu, H.Q., Gu, J.C. and Deng, G.L. 2018. "Energy-Efficient Scheduling for a Job Shop Using an Improved Whale Optimization Algorithm." *MATHEMATICS* 6 (11).

Lei, D., Gao, L. and Zheng, Y. 2018. "A Novel Teaching-Learning-Based Optimization Algorithm for Energy-Efficient Scheduling in Hybrid Flow Shop." *IEEE TRANSACTIONS ON ENGINEERING MANAGEMENT* 65 (2): 330–340.

Li, L., Li, C. and Tang, Y. 2018. "Energy Efficient Process Planning for Resource-Constrained Machining Systems." Paper presented at 8616239. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85062216931&doi=10.1109%2fSMC.2018.00243&partnerID=40&md5=c1a4272953eece3c87ff206907a1d173.

Li, M., Li, T., Peng, S. and Guo, Y. 2018. "Batch Scheduling of Remanufacturing Flexible Job Shop for Minimal Electricity- and Time-Cost." Paper presented at Communications in Computer and Information Science. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85053892681&doi=10.1007%2f978-981-13-2396-6_28&partnerID=40&md5=78aef80c17647afd5b1d5f32f71af57a.

Li, Y., Chang, Q., Ni, J. and Brundage, M.P. 2018. "Event-Based Supervisory Control for Energy Efficient Manufacturing Systems." *IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING* 15 (1): 92–103.

Liao, W. and Wang, T. 2018. "Promoting green and sustainability: A multi- objective optimization method for the job-shop scheduling problem." *Sustainability (Switzerland)* 10 (11).

Nouiri, M., Bekrar, A. and Trentesaux, D. 2018. "Towards Energy Efficient Scheduling and Rescheduling for Dynamic Flexible Job Shop Problem." available at: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85052900787&doi=10.1016%2fj.ifacol.2018.08.357&partnerID=40&md5=5831316f7bbf2df16fbc8ce3a94065e8.

Wang, S., Liang, Y.C., Li, W.D. and Cai, X.T. 2018. "Big Data enabled Intelligent Immune System for energy efficient manufacturing management." *JOURNAL OF CLEANER PRODUCTION* 195: 507–520.

Wang, W.B., Yang, H.D., Zhang, Y.F. and Xu, J.X. 2018. "IoT-enabled real-time energy efficiency optimisation method for energy-intensive manufacturing enterprises." *INTERNATIONAL JOURNAL OF COMPUTER INTEGRATED MANUFACTURING* 31 (4–5): 362–379.

Wu, X.L., Shen, X.L. and Cui, Q. 2018. "Multi-Objective Flexible Flow Shop Scheduling Problem Considering Variable Processing Time due to Renewable Energy." *SUSTAINABILITY* 10 (3).

Xia, T., Xi, L., Du, S., Xiao, L. and Pan, E. 2018. "Energy-oriented maintenance decision-making for sustainable manufacturing based on energy saving window." *Journal of Manufacturing Science and Engineering, Transactions of the ASME* 140 (5).

Zhang, H., Ge, H., Pan, R. and Wu, Y. 2018. "Multi-objective Bi-level programming for the energy-aware integration of flexible job shop scheduling and multi-row layout." *Algorithms* 11 (12).

Zhong, L.-C., Qian, B., Hu, R. and Zhang, C.-S. 2018. "The hybrid shuffle frog leaping algorithm based on cuckoo search for flow shop scheduling with the consideration of energy consumption." Paper presented at Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). https://www.scopus.com/inward/record.uri?eid=2-s2.0-85051872188&doi=10.1007%2f978-3-319-95930-6_65&partnerID=40&md5=1fefe96f2a9061f06fdd02d270eb2fe4.

Giglio, D., Paolucci, M. and Roshani, A. 2017. "Integrated lot sizing and energy-efficient job shop scheduling problem in manufacturing/remanufacturing systems." *JOURNAL OF CLEANER PRODUCTION* 148: 624–641.

Gong, X., van der Wee, M., Pessemier, T. de, Verbrugge, S., Colle, D., Martens, L. and Joseph, W. 2017. "Energy- and Labor-aware Production Scheduling for Sustainable Manufacturing: A Case Study on Plastic Bottle Manufacturing." Paper presented at Procedia CIRP. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85020046504&doi=10.1016%2fj.procir.2016.11.13

6&partnerID = 40&md5 = 22df2727d080d3d2312c0314bfb1286b.

Gu, W., Li, Y. and Wang, Y. 2017. "Energy-efficient job shop scheduling problem using an improved particle swarm algorithm." Paper presented at Proceedings of 2017 IEEE 3rd Information Technology and Mechatronics Engineering Conference, ITOEC 2017. https://www.scopus.com/inward/record.uri?eid = 2-s2.0-85043467714&doi = 10.1109%2fITOEC.2017.8122469&partnerID = 40&md5 = 2ab614b8a793027be4273e4538b49261.

Iqbal, N., Aziz, M.H., Jahanzaib, M., Ahmad, W. and Hussain, S. 2017. "Integration of cell formation and job sequencing to minimize energy consumption with minimum make-span." *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 231 (14): 2636–2651.

Lee, S., Do Chung, B., Jeon, H.W. and Chang, J. 2017. "A dynamic control approach for energy-efficient production scheduling on a single machine under time-varying electricity pricing." *JOURNAL OF CLEANER PRODUCTION* 165: 552–563.

Lei, D., Zheng, Y. and Guo, X. 2017. "A shuffled frog-leaping algorithm for flexible job shop scheduling with the consideration of energy consumption." *INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH* 55 (11): 3126–3140.

Lu, C., Gao, L., Li, X., Pan, Q. and Wang, Q. 2017. "Energy-efficient permutation flow shop scheduling problem using a hybrid multi-objective backtracking search algorithm." *JOURNAL OF CLEANER PRODUCTION* 144: 228–238.

Raileanu, S., Anton, F., Iatan, A., Borangiu, T., Anton, S. and Morariu, O. 2017. "Resource scheduling based on energy consumption for sustainable manufacturing." *JOURNAL OF INTELLIGENT MANUFACTURING* 28 (7): 1519–1530.

Salido, M.A., Escamilla, J., Barber, F. and Giret, A. 2017. "Rescheduling in job-shop problems for sustainable manufacturing systems." *JOURNAL OF CLEANER PRODUCTION* 162, S121-S132.

Sundstrom, N., Wigstrom, O. and Lennartson, B. 2017. "Conflict between Energy, Stability, and Robustness in Production Schedules." *IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING* 14 (2): 658–668.

Wang, J., Feng, Y., Fei, Z., Li, S. and Chang, Q. 2017. "Markov chain based idle status control of stochastic machines for energy saving operation." Paper presented at IEEE International Conference on Automation Science and Engineering. https://www.scopus.com/inward/record.uri?eid = 2-s2.0-85044961506&doi = 10.1109%2fCOASE.2017.8256236&partnerID = 40&md5 = 818bbef9f97ff7e98a044d74638637a3.

Wang, J., Xue, J., Duque, E.T., Li, S. and Chang, Q. 2017. "Fuzzy decision of machine switch on-off for energy efficient operation of manufacturing system." Paper presented at IEEE International Conference on Automation Science and Engineering. https://www.scopus.com/inward/record.uri?eid = 2-s2.0-85044966387&doi = 10.1109%2fCOASE.2017.8256259&partnerID = 40&md5 = bc20a68bbcfdf72d861aa110d5df394c.

Zhang, R. 2017. "Sustainable scheduling of cloth production processes by multi-objective genetic algorithm with tabu-enhanced local search." *Sustainability (Switzerland)* 9 (10).

Zhang, Y., Wang, J. and Liu, Y. 2017. "Game theory based real-time multi-objective flexible job shop scheduling considering environmental impact." *JOURNAL OF CLEANER PRODUCTION* 167: 665–679.

Zou, J., Chang, Q., Arinez, J. and Xiao, G.X. 2017. "Data-driven modeling and real-time distributed control for energy efficient manufacturing systems." *ENERGY* 127: 247–257.

Biel, K. and Glock, C.H. 2016. "On the use of waste heat in a two-stage production system with controllable production rates." *INTERNATIONAL JOURNAL OF PRODUCTION ECONOMICS* 181: 174–190.

Brundage, M.*P*., Chang, Q., Li, Y., Arinez, J. and Xiao, G.X. 2016. "Implementing a Real-Time, Energy-Efficient Control Methodology to Maximize Manufacturing Profits." *IEEE TRANSACTIONS ON SYSTEMS MAN CYBERNETICS-SYSTEMS* 46 (6): 855–866.

Dai, M., Tang, D., Huang, Z. and Yang, J. 2016. "Energy-efficient process planning using improved genetic algorithm." *Transactions of Nanjing University of Aeronautics and Astronautics* 33 (5): 602–609.

Gong, X., Pessemier, T. de, Joseph, W. and Martens, L. (Eds.) 2016, *A power data driven energy-cost-aware production scheduling method for sustainable manufacturing at the unit process level*, 2016-November.

Gong, X., Pessemier, T. de, Joseph, W. and Martens, L. 2016. "A Stochasticity Handling Heuristic in Energy-cost-aware Scheduling for Sustainable Production." Paper presented at Procedia CIRP. https://www.scopus.com/inward/record.uri?eid = 2-s2.0-84986003683&doi = 10.1016%2fj.procir.2016.03.028&partnerID = 40&md5 = 98ca9f7dd6291b31870ff3a8b91512a2.

González, M.A., Oddi, A. and Rasconi, R. 2016. "A multi-objective memetic algorithm for solving job shops with a non-regular energy cost." Paper presented at Proceedings of the 11th Workshop on Constraint Satisfaction Techniques for Planning and Scheduling Problems, COPLAS 2016. https://www.scopus.com/inward/record.uri?eid = 2-s2.0-85067017496&partnerID = 40&md5 = f0a5088ac7f5c088e7ff36a2e28be7c3.

Jimenez, J.-F., Bekrar, A., Giret, A., Leitão, *P*. and Trentesaux, D. 2016. "A dynamic hybrid control architecture for sustainable manufacturing control." Paper presented at IFAC-PapersOnLine. https://www.scopus.com/inward/record.uri?eid = 2-s2.0-85012904326&doi = 10.1016%2fj.ifacol.2016.12.171&partnerID = 40&md5 = 8cd5660934a2ed11bb2511d7b8f77741.

Mansouri, S.A. and Aktas, E. 2016. "Minimizing Energy consumption and makespan in a two-machine flowshop scheduling problem." *JOURNAL OF THE OPERATIONAL RESEARCH SOCIETY* 67 (11): 1382–1394.

Palasciano, C. and Taisch, M. 2016. "Autonomous energy-aware production systems control." Paper presented at Proceedings of the Summer School Francesco Turco. https://www.scopus.com/inward/record.uri?eid = 2-s2.0-85006056313&partnerID = 40&md5 = 608bb67080678434649a593f605fba1e.

Samukawa, T. and Suwa, H. 2016. "An optimization of energy-efficiency in machining manufacturing systems based on a framework of multi-mode RCPSP." *International Journal of Automation Technology* 10 (6): 985–992.

Tang, D., Dai, M., Salido, M.A. and Giret, A. 2016. "Energy-efficient dynamic scheduling for a flexible flow shop using an improved particle swarm optimization." *COMPUTERS IN INDUSTRY* 81: 82–95.

Wang, S., Liu, M., Chu, F. and Chu, C. 2016. "Bi-objective optimization of a single machine batch scheduling problem

with energy cost consideration." *JOURNAL OF CLEANER PRODUCTION* 137: 1205–1215.

Wei, W. and Wang, Z. 2016. "Integrated optimal production and energy control of a single machine and single product-type manufacturing system." Paper presented at 7497870. https://www.scopus.com/inward/record.uri?eid=2-s2.0-84981318257&doi=10.1109%2fWODES.2016.7497870&partnerID=40&md5=2398cb9633fc09ba4a72d106bf76142b.

Xu, W.J., Shao, L.Y., Yao, B.T., Zhou, Z.D. and Pham, D.T. 2016. "Perception data-driven optimization of manufacturing equipment service scheduling in sustainable manufacturing." *JOURNAL OF MANUFACTURING SYSTEMS* 41: 86–101.

Zhang, Z., Tang, R., Peng, T., Tao, L. and Jia, S. 2016. "A method for minimizing the energy consumption of machining system: integration of process planning and scheduling." *JOURNAL OF CLEANER PRODUCTION* 137: 1647–1662.

Appendix 3. Utilised benchmark instances

Brandimarte (1993): Mk01 – Mk10

Hurink, Jurisch, and Thole (1994):

- edata: la04, la06, la19, la22, orb10
- rdata: la31, mt06, mt10, mt20, orb6
- vdata: abz7, car4, la03, la05, la38