MM Science Journal | www.mmscience.eu
ISSN 1803-1269 (Print) | ISSN 1805-0476 (On-line)
Special Issue | HSM 2025
18th International Conference on High Speed Machining
October 15-16, 2025, Metz, France
DOI: 10.17973/MMSJ.2025_11_2025137

HSM2025-44826

AUTOMATED EXECUTION OF ENERGY-AWARE PRODUCTION SCHEDULES BASED ON REAL INDUSTRIAL PROCESS AND ORDER DATA

J. Stock[1]*, A. Clement[1], D. Fuhrländer-Völker[1], M. Weigold[1]

[1]Technical University of Darmstadt, Institute for Production Management, Technology and Machine Tools, Darmstadt, Germany

*Corresponding author; e-mail: j.stock@ptw.tu-darmstadt.de

Abstract

The rising share of variable renewable electricity in Germany has increased price volatility, emphasizing the need for demand response. This study introduces an execution service within Energy-Aware Production Scheduling (EAPS), using OPC UA for automated schedule implementation. Based on real industrial data, machine states and energy use are modeled, enabling a cyber-physical system to optimize makespan, energy costs, and peak load. Simulations show a 6 % energy cost reduction compared to the Shortest Processing Time dispatching rule. Successful integration with four simulated OPC UA servers confirms system stability, demonstrating the enhanced EAPS architecture's potential for automated, energy-flexible, and sustainable manufacturing.

Keywords:

Industrial communication, Job-Shop-Scheduling Problem, Demand Response Measures

1 INTRODUCTION

In 2023, electricity generation from renewable energy sources surpassed that of conventional sources in Germany for the first time [AGEE-Stat 2025]. This transformation in electricity generation results in increased volatility of electricity prices in the energy market [Sauer 2019; Sauer 2022]. At the same time, the peak load costs have risen. The increase in 2024 was around 25 percent compared to the previous year in Germany [Bundesnetzagentur 2025]. This development presents novel prospects for industrial companies to reduce energy costs through demand response measures [Sauer 2022]. Demand response refers to the deliberate adjustment of electricity consumption in reaction to price signals or grid requirements [U.S. Department of Energy 2006]. This approach enables companies to reduce costs by shifting or curtailing energy use during peak demand periods [Walther 2022].

Furthermore, preliminary empirical studies indicate a positive correlation between enhanced energy efficiency and a company's financial performance [Fan 2017; Özbuğday 2020]. Biel [2016] showed that the topic of energy-aware production scheduling has become increasingly popular in recent years. EAPS refers to the development of production schedules that integrate conventional production-related objectives—such as throughput, lead times, and resource utilization—with energy-related goals [Grosch 2024]. Energy costs have proven to be particularly relevant in EAPS [Biel 2016]. Previous studies have achieved a reduction in energy costs through the implementation of production planning methodologies on individual production machines, with a focus on the volatile electricity market prices [Grosch 2022b; Fuhrländer-Völker 2023]. To the best of our knowledge, there is a lack of EAPS frameworks that can be used in a real production system. This is consistent with the literature review findings in [Gao 2020; dos Santos 2023; Grosch 2024].

Therefore, the aim of this work is to integrate an execution service into an EAPS architecture and enable the final architecture to be used in the production system.

In section 2, we present our findings of the literature research. The architecture proposed in this paper is described in section 3. Section 4 demonstrates the practical relevance of our approach through its application to an industrial use case involving real process and order data. In section 5, we conduct a simulation-based validation to assess the architecture's performance. Finally, section 6 offers a critical reflection on our findings and outlines potential directions for future research.

2 STATE OF RESEARCH

To identify the research gap, we conducted a structured literature review following the methodology proposed by vom Brocke [2009]. The key findings of the review are summarized as follows:

Only one study implements a scheduling model within an actual production environment [Gao 2024]. Four studies validated their approaches using industrial data [Li 2022; Qu 2022; Gao 2024; Ye 2025], while another four rely on

MM Science Journal | www.mmscience.eu
ISSN 1803-1269 (Print) | ISSN 1805-0476 (On-line)
Special Issue | HSM 2025
18th International Conference on High Speed Machining
October 15-16, 2025, Metz, France
DOI: 10.17973/MMSJ.2025_11_2025137

benchmark datasets available in the literature [Salido 2017; Nouiri 2018; Mota 2023; Chen 2024]. Additionally, one study tests a small instance of an abstract use case [Defersha 2022]. Gao [2024] propose a digital twin-based scheduling model for a production system and evaluate their approach through a real-world application in an automobile welding process. Ye [2025] apply a reinforcement learning approach to address the dynamic real-time scheduling problem under variable energy pricing conditions. Similarly, Mota [2023] develop a job-shop scheduling method that accounts for fluctuating energy prices by incorporating renewable energy resources into their mathematical model. Gao [2020] conduct a comprehensive literature review and identified future research directions, emphasizing the importance of directly integrating energy-efficient planning methods into production planning systems.

To summarise, the literature review points to the lack of an integrated architecture for the automated execution of energy-aware production schedules using industrial communication protocols such as OPC UA. Furthermore, there seems to be no optimisation approaches that simultaneously consider objectives such as peak loads, energy costs and makespan. In addition, there are only a few approaches with real industry and order data.

Therefore, the objective of this research is to develop an architecture that can execute planned energy-aware schedules in a production system. In order to accomplish this research goal, the present study proposes an expansion of an extant production planning tool, with the objective of optimising peak loads, energy costs and makespan.

3 ARCHITECTURE

This section introduces the EAPS execution service by outlining its essential requirements, providing a description of the current EAPS architecture, and explaining the core functionalities of the service along with its integration into the existing EAPS framework.

3.1 Requirements for the EAPS execution service

The objective of this work is to deploy an existing method for EAPS in a production environment. To ensure the transferability of the presented approach to other production processes and use cases, we define success criteria.

Based on the Quality Attribute Definitions by Bansiya [2002], the following requirements are defined:

- *Reusability:* The architecture should be applicable to new problems and use cases with minimal effort.
- *Flexibility:* The architectural design should be conceived in such a manner that it is readily adaptable and transferable to alternative energy-aware production planning methodologies and implementations.
- *Understandability:* The implementation should be comprehensible and well structured. The aim of the extension is to enable the integration of an EAPS method into a production system.

This work extends an existing EAPS methodology for the automatic execution of planned jobs. We select the developed method for EAPS by Grosch [2024]. Grosch evaluated the effectiveness of the method in an industry-related use case in the ETA research factory at Technical University of Darmstadt.

3.2 Description of the existing EAPS architecture

Grosch [2024] highlights the research gap concerning the absence of conceptual frameworks that enable the application of EAPS in real-world production environments. To address this deficit, Grosch proposes a methodology involving the implementation of a cyber physical system. To this end, Grosch designs an implementation process and an EAPS architecture.

The core architecture elements are the optimisation algorithm and the virtual representation of real production systems, called environment. The production systems environment holds the energetic machine models. The environment can also maintain a constant awareness of the orders scheduled for production and the inventory levels within the production system. The Non-dominated Sorting Genetic Algorithm II solves the multi-objective optimisation problem, minimising makespan and energy costs. The implementation of this adapted algorithm is inspired by Deb [2002]. The production system configuration provides the necessary information for the instantiation of the production system environment, including machine parameters and the manufacturing steps of the products. The EAPS framework is published under open source licence [Grosch 2024].

The present development status of the framework represents the virtual area of the production system. It should be noted that a connection to a real production system has not yet been implemented. This supports our decision in using this EAPS method to implement the automated control capability of machines in the existing architecture.

3.3 Description of the EAPS execution service

The following section provides a detailed description of the developed EAPS execution service, which is based on the virtualised environment and architecture. We select the Python programming language to implement the execution service, as it offers a comprehensive selection of libraries that support object-oriented programming. Fig. 1 shows a simplified Unified Modelling Language (UML) diagram of the service.

The modular service enables energy-aware planned jobs to be executed automatically. First, the energy-aware production plan is exported as a formatted CSV file. This file contains the following column labels, which are important for the internal flow of the service:

- *starttime***:** Specifies the timestamp which the job is initiated. In cases where a setup process is required, this timestamp precedes the *working_starttime*.
- *working_starttime:* Specifies the scheduled timestamp when the machine transitions into working mode. From this moment onward, drives are actively processed.
- *endtime*: Specifies the scheduled completion timestamp of the individual machine program,

MM Science Journal | www.mmscience.eu
ISSN 1803-1269 (Print) | ISSN 1805-0476 (On-line)
Special Issue | HSM 2025
18th International Conference on High Speed Machining
October 15-16, 2025, Metz, France
DOI: 10.17973/MMSJ.2025_11_2025137

marking the end of the job's execution on the assigned machine.

- *machine*: Specifies the specific production machine to which the job is assigned.
- *job*: Refers to the unique identifier or number assigned to a production job.
- *capacity*: Represents the processing capacity of the machine, particularly relevant in scenarios where multiple components can be handled simultaneously.

The defined columns ensure that each production step is unique and can be accurately associated with its corresponding job.

We divide the service into two main functionalities and classes. One class manages the connection to the production machines, while the other implements a scheduler. The *ConnectionManager* establishes a direct connection to the machine's programmable logic controller (PLC) via configured nodes. To achieve this, it is necessary to transfer the configuration file from the *BaseExecutionEnv* to the parent class. The structure of the config is used from the *eta-nexus* package (version 0.1.1), which has been further developed from the *eta-utility* package [Grosch 2022a]. The *ConnectionManager*, also derived from this package, enables the coordination of multiple protocols (e.g., Modbus and OPC UA) across multiple machines within a unified configuration file.

For the integration of the production plan into the production system and the machine control, we propose a systematic mapping in the *NODE_TEMPLATES* variable. This approach enables the abstraction of the machine variables in a standardised, machine-independent notation. The underlying concept is based on a template-driven mapping in which the variables from the production plan (*machine, job, capacity*) are transferred to a machine-specific notation. The naming convention follows a consistent structure and internally defined standard, in which placeholders (see the *machines* variable) are used for machine identifiers in order to generate a unique assignment for each machine.

The second class is the scheduler. The scheduler approach is a concept originating from informatics that describes a component of the operating system responsible for selecting the next task to be executed [Silberschatz 2018]. The *ExecuteProductionScheduler* class gets the formatted production plan and sets the *working_starttime* for the job activation timestamp. The scheduler also stores the other information for the planned operation in the task. To maximize efficiency, asynchronous programming paradigms are utilised. The implemented *scheduler* is based on the python package *apscheduler* [Grönholm 2024], where we used the implemented asynchronous scheduler.

Moreover, the *ExecuteProductionScheduler* class inherits from the *ExecutionHandler*, which is responsible for saving the executed orders (*schedule_data_store*) and comparing the execution status with the original production plan.

The function *execute_production_process* in the *BaseExecutionEnv* class combines the capseled classes and ensures the functionality of the enhanced EAPS architecture.

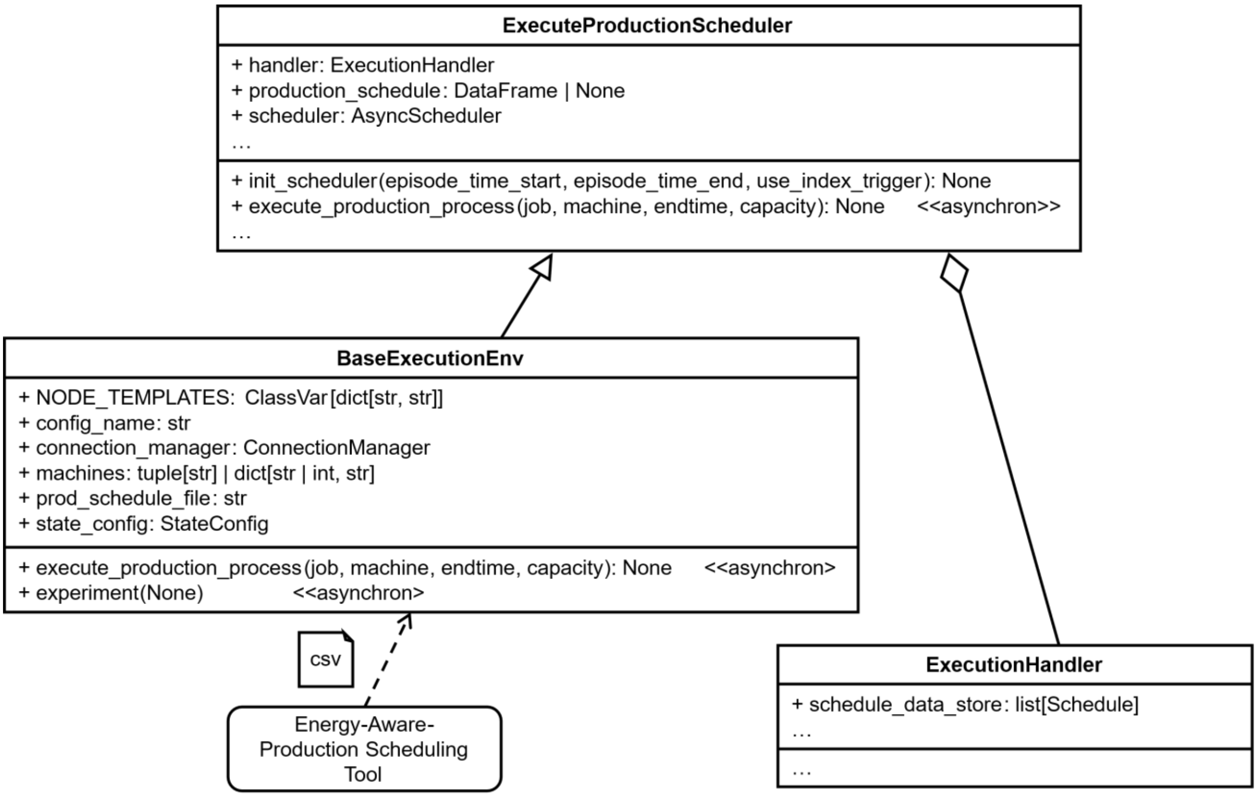

Fig. 1: Simplified UML diagram of the EAPS execution service.

MM Science Journal | www.mmscience.eu
ISSN 1803-1269 (Print) | ISSN 1805-0476 (On-line)
Special Issue | HSM 2025
18th International Conference on High Speed Machining
October 15-16, 2025, Metz, France
DOI: 10.17973/MMSJ.2025_11_2025137

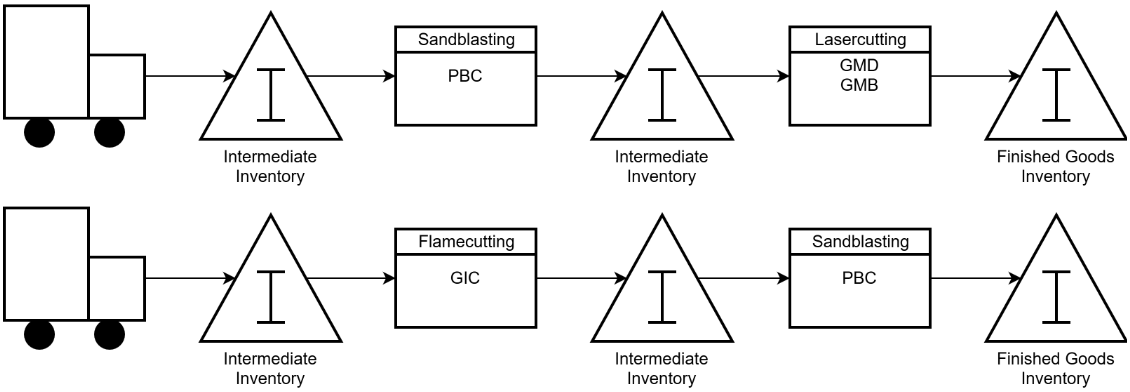

Fig. 2: Value stream of the company from the German mechanical engineering sector.

4 APPLICATION

This chapter presents the use case, introduces the general terminology for energy models in the existing EAPS architecture, applies these models to the use case, and outlines the implementation of the defined objectives.

4.1 Introduction of the use case

In this study, we utilise a dataset derived from a mid-sized company in Germany, that operates in the mechanical engineering industry. The production includes the manufacture of steel components. This research focuses on the cutting of steel components for the application of the EAPS. It intentionally excludes other machinery within the value chain, such as machining centres or welding robots, from consideration.

The machinery employed for the implementation of the EAPS method in the cutting area comprises two laser cutting machines (GMB and GMD), a flame cutting machine (GIA), and a sandblasting machine (PBC). The individual production steps of the use case are depicted in Fig. 2.

Initially, sheets are retrieved from the sheet metal inventory. For tasks that require the use of a laser cutting machine, the sheets undergo a preliminary cleaning process in the sandblasting machine to eliminate any corrosion. Following this, the sheets are fed into the laser cutting machine, where the specific process program is activated. In the case of orders involving the flame cutting machine, corroded sheets are inserted directly into the machine, whereupon the specific processing programme is initiated. The sandblasting process is characterised by its consistent processing time. Additionally, we make the following assumptions regarding the use case:

- The sheet thicknesses are constant.
- One product is permanently one sheet. The differently cut steel components and the offcuts are negligible.
- The production program of the sandblasting system is permanently constant, while the production program of the other machines varies depending on the respective job.
- A set-up time is required on all machines before the start of the production process

4.2 General terminology of the energy models in the existing EAPS architecture

To apply the developed architecture in this work, it is necessary to introduce the corresponding general energy models. The EAPS architecture according to Grosch [2024] integrates discrete event models of the job shop scheduling problem with discrete-time linear machine models of the production machines. The connection between the two models takes place over the time t. Grosch assumes the following four energy modes for the EAPS modeling:

- Off (no power consumption),
- Standby (a_t^{st}),
- Operational (a_t^{op}),
- Working (a_t^{wk}).

At any given timestamp, a machine can be in only one state. This state is characterised by the specific binary variable a_t. The regression parameters beta $\beta^{\mathrm{el}} \in \mathbb{R}^{+}$ quantify the change in power demand from a lower to a higher energy mode. These coefficients are estimated using the method of least squares, based on observational data collected over a three-day period.

The underlying logic in the architecture also supports energy efficiency. The parameter $t_{\mathrm{wait\_standby}}$ specifies the duration that a production machine remains in the operational energy state before being scheduled into the standby state for energy-saving purposes. The variable $c_{m,h'}$ denotes the endtime point for the scheduled job h on the machine m, while $s_{m,h''}$ indicates the startpoint of the subsequent event on the same machine.

4.3 Energy models in the use case

In comparison with the use case in the original EAPS architecture, the machines in the present use case have not been energy-optimised. During analysis of the laser cutting machines' power consumption, we identified an additional inefficient standby state $a_t^{\mathrm{st*}}$.

The electrical power consumption for the laser cutting machine may be calculated as follows:

$$\begin{aligned} P_t^{\mathrm{el}} = & \left(a_t^{\mathrm{st}^*} + a_t^{\mathrm{st}} + a_t^{\mathrm{op}} + a_t^{\mathrm{wk}}\right) \cdot \beta_{\mathrm{st}^*}^{\mathrm{el}} \\ & + \left(a_t^{\mathrm{st}} + a_t^{\mathrm{op}} + a_t^{\mathrm{wk}}\right) \cdot \beta_{\mathrm{st}}^{\mathrm{el}} \\ & + \left(a_t^{\mathrm{op}} + a_t^{\mathrm{wk}}\right) \cdot \beta_{\mathrm{op}}^{\mathrm{el}} \\ & + a_t^{\mathrm{wk}} \cdot \beta_t^{\mathrm{wk}} \end{aligned} \quad (1)$$

MM Science Journal | www.mmscience.eu
ISSN 1803-1269 (Print) | ISSN 1805-0476 (On-line)
Special Issue | HSM 2025
18th International Conference on High Speed Machining
October 15-16, 2025, Metz, France
DOI: 10.17973/MMSJ.2025_11_2025137

$$a_t^{\mathrm{st}} = \begin{cases} 1, \text{if } s_{m,h''} - c_{m,h'} > t_{\text{wait\_standby}} \text{ and } t \in [c_{m,h'}, s_{m,h''}] \\ 0, \text{otherwise} \end{cases} \quad (2)$$

$$a_t^{\mathrm{op}} = \begin{cases} 1, \text{if } s_{m,h''} - c_{m,h'} \leq t_{\text{wait\_standby}} \text{ and } t \in [c_{m,h'}, s_{m,h''}] \\ 0, \text{otherwise} \end{cases} \quad (3)$$

$$a_t^{\mathrm{wk}} = \begin{cases} 1, \text{if } \left(s_{m,h''} - c_{m,h'} > t_{wait\_standby} \text{ and } t \in (s_{m,h''} + t_{\text{peak}}, c_{m,h''})\right) \text{ or} \\ \left(s_{m,h''} - c_{m,h'} \leq t_{wait\_standby} \text{ and } t \in (s_{m,h''}, c_{m,h''})\right) \\ 0, \text{otherwise} \end{cases} \quad (4)$$

$$a_t^{\mathrm{pk}} = \begin{cases} 1, \text{if } s_{m,h''} - c_{m,h'} > t_{wait\_standby} \text{ and } t \in (s_{m,h''}, s_{m,h''} + t_{\text{peak}}] \\ 0, \text{otherwise} \end{cases} \quad (5)$$

while we assume that the inefficient state is not incorporated into the production planning. The integration of the energy model with the production planning model is achieved through equations (2) and (3), which correspond to the standby and operational energy states, respectively. The following equation :

$$a_t^{\mathrm{wk}} = \begin{cases} 1, t \notin [c_{m,h'}, s_{m,h''}] \\ 0, \text{otherwise} \end{cases} \quad (6)$$

applies to the working state.

The sandblasting machine demonstrates distinct behavior during the transition from standby to operational mode. During turbine start-up and prior to the commencement of the programmed process, an elevated power demand is observed over a specified time period. We classify this state the peak state, occurring over a defined time interval t_{peak}.

In contrast to the laser cutting machines, no additional standby state is present. Consequently, the following equation is utilized to calculate its electrical power consumption:

$$\begin{aligned} P_t^{\mathrm{el}} = &\left(a_t^{\mathrm{st}} + a_t^{\mathrm{op}} + a_t^{\mathrm{wk}} + a_t^{\mathrm{pk}}\right) \cdot \beta_{\mathrm{st}}^{\mathrm{el}} \\ &+ \left(a_t^{\mathrm{op}} + a_t^{\mathrm{wk}} + a_t^{\mathrm{pk}}\right) \cdot \beta_{\mathrm{op}}^{\mathrm{el}} \\ &+ \left(a_t^{\mathrm{wk}} + a_t^{\mathrm{pk}}\right) \cdot \beta_{\mathrm{wk}}^{\mathrm{el}} \\ &+ a_t^{\mathrm{pk}} \cdot \beta_{\mathrm{pk}}^{\mathrm{el}} \end{aligned} \quad (7)$$

The equations (2) and (3) for the standby and operational energy states remain applicable. The binary variable a_t^{wk} representing the working state is dependent on the preceding state of the machine. If the machine was previously in the operational state, the binary variable in equation (4) is assigned a value of 1. Otherwise, the peak load period is first determined using equation (5).

Finally, it is important to note that the flame cutting machine makes use of the pre-existing electrical energy models stored within the system.

4.4 Implementation of the objectives

For the application of the architecture, it is necessary to implement the objectives. For the production related objective, we use the makespan. With the equation (8), the makespan $MSKP$ is calculated for all the planned jobs:

$$MSKP = \max_{m \in M, h \in e_m} c_{m,h} \quad (8)$$

For the energy related objective we use the energy related cost ERC calculated by equation (9) where $P_{m,t}^{\mathrm{el}}$ is the power of a machine m and C_t is the energy cost at time t.

$$ERC = \sum_{t=0}^{t_{\max}} \sum_{m \epsilon M} P_{m,t}^{\mathrm{el}} \cdot C_t \quad (9)$$

Both makespan and energy costs have already been incorporated into the existing EAPS architecture and are well-established objectives in the literature [Biel 2016].

The characteristic start-up behaviour of the PBC after standby mode must be taken into account, as this leads to a significant increase in peak loads. As a result, we make an additional consideration of the maximum load peak within a production plan, which serves as an additional objective for EAPS optimisation. Equation (10) calucalutes the maximum power peak consumption PPC of the production plan, where P_m^{el} is the power of a machine m:

$$PPC = \max_{t \epsilon [0, t_{\max}]} \left(\sum_{m \epsilon M} P_m^{\mathrm{el}} \right)_t \quad (10)$$

The implementation of the developed execution service, together with the corresponding use case, has been published under an open-source licence [Stock 2025].

5 EVALUATION

The measurement of electrical power demand for the machines commenced on 29 January 2024 at 00:00 and concluded on 1 February 2024 at 12:00. This timeframe encompasses three full production days, during which job data were concurrently collected. For the implementation of the complete EAPS architecture, only specific job-related information—namely, the number of sheets, processing time, and setup time—is considered relevant. Throughout the measurement period, a total of 16 jobs were recorded for the GMB, 15 for the GIC, and 13 for the GMD.

The use case under consideration pertains to a medium-sized manufacturing company located in Germany. The direct implementation of the extended EAPS architecture and the deployment of the EAPS execution service on the production machinery is currently unfeasible, primarily due to organisational constraints and insufficient technological infrastructure. To test the service in a live production environment, it is necessary to use an isolated, controlled setting. Moreover, it is crucial to guarantee that ongoing production activities remain uninterrupted and that financial losses are avoided. Nevertheless, it should be noted that the provision of these conditions cannot be fully guaranteed during the testing and evaluation phases of the EAPS execution service.

To address this limitation, we establish a dedicated test environment within the scope of this work. In this work, the

MM Science Journal | www.mmscience.eu
ISSN 1803-1269 (Print) | ISSN 1805-0476 (On-line)
Special Issue | HSM 2025
18th International Conference on High Speed Machining
October 15-16, 2025, Metz, France
DOI: 10.17973/MMSJ.2025_11_2025137

OPC UA communication protocol is used as the basis for communication between the implemented execution service and the machines. OPC UA is a manufacturer-independent protocol from the OPC Foundation and has established itself as a standard to ensure interoperability [Zezulka 2018]. This test environment consists of four independent server instances, each simulating a server corresponding to a production machine within the context of the use case.

The structure for initializing a server is based on the *eta-nexus* package (version 0.1.1). Each server is uniquely defined by its name, IP address, username, and password. In contrast, an OPC UA node is described by a unique name, the name of the corresponding server, the NodeID, and the associated data type. The following example illustrates the initialization for the PBC simulation server using a TOML configuration file:

```
[[system]]
name = "EAPS"

[system.servers.PBC]
url = "opc.tcp://localhost:4840"
protocol = "opcua"
usr = "admin"
pwd = "admin"

[[system.nodes]]
name= "bPBCWorkingState"
server= "PBC"
opc_id =
"ns=2;s=Application.Plc_Main.PBC.localState
.bPBCWorkingState"
dtype = "bool"
```

In the context of the present use case, we implement all OPC UA nodes as variables. The address space of each OPC UA server includes the nodes *bOperatingState* and *sJob*. During the execution of a job, the *bOperatingState* variable is set to 'True', and the current job, identified by its job number, is written to the *sJob* node.

The evaluation phase comprises the generation of an energy-aware production schedule and the validation of the extended EAPS-architecture. For scheduling, we utilize the recorded job data from January 30, 2024, and the day-ahead electricity prices from the EPEX Spot Market corresponding to the same date. A 24-hour production window is assumed, excluding breaks and shift changes. The resulting schedule serves as input for validating the execution service within a test environment. We define a simulation time of 10 minutes, during which the simulated progression corresponds to the real-time execution of the production schedule. The evaluation was performed on a system with a 13th-generation Intel Core i7-1370P processor.

The Fig. 3 illustrates the production schedule generated for the use case presented in this study. Identical colors across different machines indicate operations belonging to the same job, while numerical indices at the GIC represent the parallel processing of two sheets. Black bars denote algorithm-determined pause times, and grey segments indicate job-specific setup times. The results corresponding to the minimum energy related cost objective reveal that the algorithm predominantly schedules operations in the early morning hours, while deliberately avoiding task allocation on the PBC and GMB machines during periods of elevated electricity prices between 07:00 and 10:00. However, we also observe that the algorithm generates a high frequency of job changes across the machines.

Tab. 1 presents, for each objective criterion of the multi-objective optimisation, the solution in which this specific objective reaches its minimum value, along with the corresponding values of the remaining objective criteria for that solution.

As a reference benchmark, the Shortest Processing Time (SPT) rule is incorporated into the analysis. The results indicate that the proposed architecture, as applied to the presented use case, consistently outperforms the benchmark across the defined objectives, *ERC* and *PPC*. We achive a 6 % reduction in the energy related costs with the architecture.

Tab. 1: Final results for the optimised objectives compared to the SPT dispatching rule.

	$MKSP$	ERC	PPC
$min\,MKSP$	$41\,872\ s$		
$min\,ERC$		146.94 €	
$min\,PPC$			$213\,780\ W$
SPT	$40\,453\ s$	156.35 €	$327\,233\ W$

In the second evaluation phase, we executed the production plan optimized for minimum energy costs using the developed execution service and recorded the system's behavior in a log file for subsequent analysis. Throughout four test runs, the execution service operated without errors, demonstrating consistent and stable performance.

6 SUMMARY

This paper identified a gap in the existing literature concerning EAPS architectures that are directly applicable within production environments. In response, we develop an execution service by extending an existing EAPS architecture. The extension comprises two core functions: one establishes the interface with production machinery, while the other executes scheduled jobs using a scheduler-based approach. We applied the enhanced architecture to a use case derived from a medium-sized manufacturing company in Germany. Our evaluation demonstrates the applicability of the extended architecture within a simulated environment. In this simulation, four OPC UA servers emulated production machines, and standardized interfaces enabled the transfer of planned order data to the OPC UA servers.

We assessed the requirements by evaluating the *reusability*, the *flexibility* and the *understandability* of the developed execution service. Owing to its two encapsulated functions and the use of standardized interfaces, the service can be readily integrated with diverse production planning methods and system implementations. We introduced a custom notation and enable compatibility with various communication protocols. Nevertheless, it remains necessary to apply the proposed architecture to additional problem domains and production systems. Future work should also include validation in real-world test environments to confirm its practical applicability.

MM Science Journal | www.mmscience.eu
ISSN 1803-1269 (Print) | ISSN 1805-0476 (On-line)
Special Issue | HSM 2025
18th International Conference on High Speed Machining
October 15-16, 2025, Metz, France
DOI: 10.17973/MMSJ.2025_11_2025137

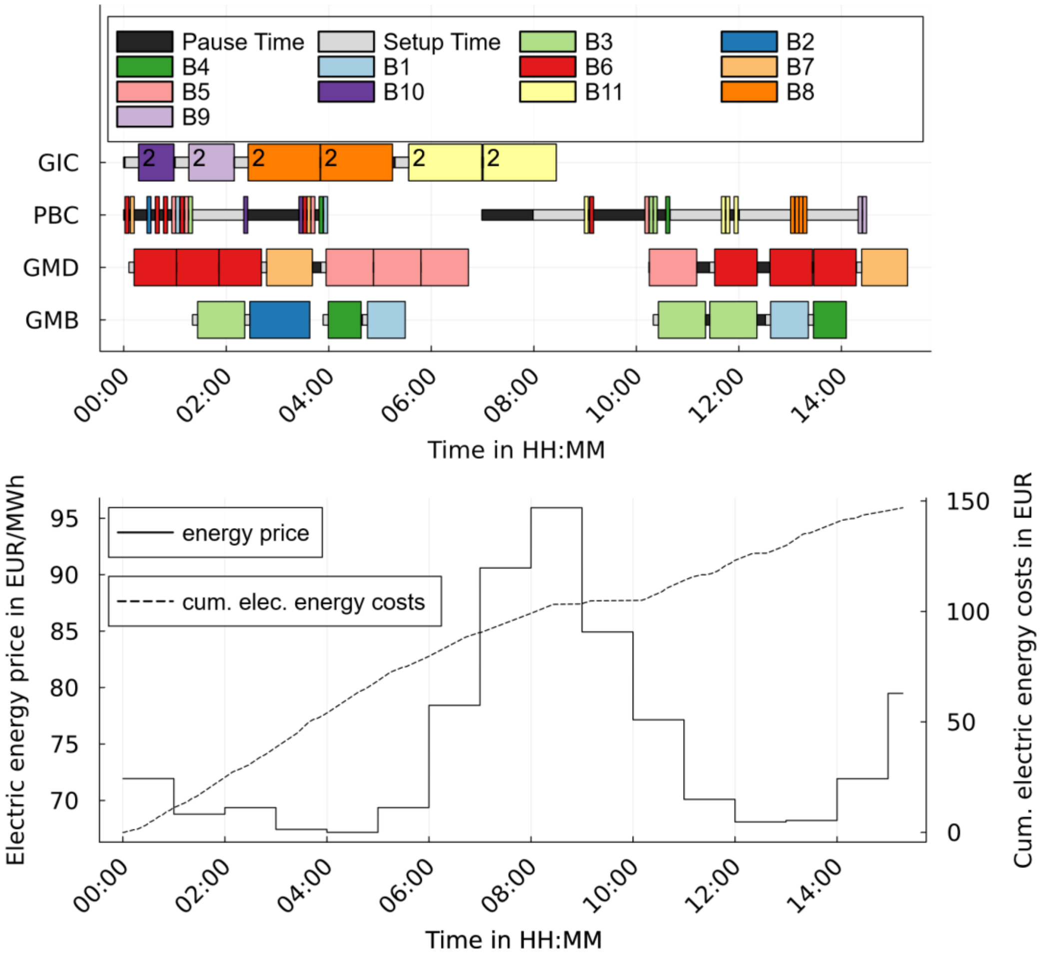

Fig. 3: Final solution of the EAPS architecture applied to the use case.

7 ACKNOWLEDGMENTS

The authors gratefully acknowledge the financial support of the Kopernikus-Project "SynErgie" (Grant Number 03SFK3A0-3) by the Federal Ministry of Research, Technology and Space of Germany (BMFTR) and thank Projektträger Jülich (PtJ) for the project supervision.

8 CREDIT AUTHOR STATEMENT

Jerome Stock: Conceptualization, Software, Writing– Original Draft. **Andreas Clement:** Software, Writing–Review & Editing. **Daniel Fuhrländer-Völker:** Writing–Review & Editing. **Matthias Weigold:** Supervision, Resources, Funding acquisition.

9 REFERENCES

Paper in a journal:

[Bansiya 2002] Bansiya, J., and Davis, C.G. A hierarchical model for object-oriented design quality assessment. IEEE Transactions on Software Engineering, January 2002, Vol.28, No.1, pp.4–17.

[Biel 2016] Biel, K., and Glock, C.H. Systematic literature review of decision support models for energy-efficient production planning. Computers & Industrial Engineering, November 2016, Vol.101, pp.243–259.

[Chen 2024] Chen, R., et al. A Q-Learning based NSGA-II for dynamic flexible job shop scheduling with limited transportation resources. Swarm and Evolutionary Computation, October 2024, Vol.90, pp.1–16.

[Deb 2002] Deb, K., et al. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation, April 2002, Vol.6, No.2, pp.182–197.

[Defersha 2022] Defersha, F.M., et al. Mathematical model and simulated annealing algorithm for setup operator constrained flexible job shop scheduling problem. Computers & Industrial Engineering, September 2022, Vol.171, pp.1–22.

[dos Santos 2023] dos Santos, S.A.B., et al. Demand response application in industrial scenarios: A systematic mapping of practical implementation. Expert Systems with Applications, April 2023, Vol.215, pp.1–19.

[Fan 2017] Fan, L.W., et al. Does energy efficiency affect financial performance? Evidence from Chinese energy-intensive firms. Journal of Cleaner Production, May 2017, Vol.151, pp.53–59.

[Gao 2020] Gao, K., et al. A review of energy-efficient scheduling in intelligent production systems. Complex &

MM Science Journal | www.mmscience.eu
ISSN 1803-1269 (Print) | ISSN 1805-0476 (On-line)
Special Issue | HSM 2025
18th International Conference on High Speed Machining
October 15-16, 2025, Metz, France
DOI: 10.17973/MMSJ.2025_11_2025137

Intelligent Systems, September 2020, Vol.6, No.2, pp.237–249.

[Gao 2024] Gao, Q., et al. A framework of cloud-edge collaborated digital twin for flexible job shop scheduling with conflict-free routing. Robotics and Computer-Integrated Manufacturing, April 2024, Vol.86, pp.1–23.

[Grosch 2022a] Grosch, B., et al. A framework for researching energy optimization of factory operations. Energy Informatics, September 2022, Vol.5, pp.1–13.

[Li 2022] Li, J.-Q., et al. A Hybrid Iterated Greedy Algorithm for a Crane Transportation Flexible Job Shop Problem. IEEE Transactions on Automation Science and Engineering, July 2022, Vol.19, No.3, pp.2153–2170.

[Özbuğday 2020] Özbuğday, F.C., et al. Resource efficiency investments and firm performance: Evidence from European SMEs. Journal of Cleaner Production, April 2020, Vol.252, pp.1–11.

[Qu 2022] Qu, M., et al. An improved electromagnetism-like mechanism algorithm for energy-aware many-objective flexible job shop scheduling. The International Journal of Advanced Manufacturing Technology, January 2022, Vol.119, pp.4265–4275.

[Salido 2017] Salido, M.A., et al. Rescheduling in job-shop problems for sustainable manufacturing systems. Journal of Cleaner Production, September 2017, Vol.162, pp.121–132.

[Walther 2022] Walther, J., et al. A Methodology for the Classification and Characterisation of Industrial Demand-Side Integration Measures. Energies, January 2022, Vol.15, No.3, 923, pp.1–26.

[Ye 2025] Ye, X., et al. Real-Time Price-Based Demand Response for Industrial Manufacturing Process via Safe Reinforcement Learning. IEEE Transactions on Industrial Informatics, January 2025, Vol.21, No.4, pp.2937–2946.

Paper in proceedings:

[Grosch 2022b] Grosch, B., et al. Cyber-physical production system for energy-flexible control of production machines. In: Procedia CIRP, Vol.107, 55th CIRP Conference on Manufacturing Systems, Lugano, June 2022, pp.221–226.

[Mota 2023] Mota, B., et al. Production Line Energy Cost Optimization with Renewable Energy Resource Usage in a Flexible Job Shop Configuration. In: IFAC-PapersOnLine, Vol.56, No.2, 22nd IFAC World Congress, Yokohama, July, 2023, pp.7972–7977.

[Nouiri 2018] Nouiri, M., et al. Towards Energy Efficient Scheduling and Rescheduling for Dynamic Flexible Job Shop Problem. In: IFAC-PapersOnLine, Vol.51, No.11, 16th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2018, Bergamo, June 2018, pp.1275–1280.

[vom Brocke 2009] vom Brocke, J., et al. Reconstructing the Giant: On the Importance of Rigour in Documenting the Literature Search Process. In: ECIS 2009 Proceedings, 17th ECIS 2009, Verona, June 2019, pp.2206-2217.

[Zezulka 2018] Zezulka, F., et al. Communication Systems for Industry 4.0 and the IIoT. In: IFAC-PapersOnLine, Vol.51, No.6, 15th IFAC Conference in Programmable Devices and Embedded Systems PDeS 2018, Ostrava, May 2018, pp.150–155.

Technical reports or thesis:

[AGEE-Stat 2025] AGEE-Stat. Erneuerbare Energien in Zahlen. Dessau-Roßlau: Umweltbundesamt, 2025. https://www.umweltbundesamt.de/themen/klima-energie/erneuerbare-energien/erneuerbare-energien-in-zahlen#uberblick. Accessed 7 May 2025.

[Bundesnetzagentur 2025] Bundesnetzagentur für Elektrizität, Gas, Telekommunikation, Post und Eisenbahnen, and Bundeskartellamt. Monitoringbericht 202. Bonn: Bundesnetzagentur, 2025. https://data.bundesnetzagentur.de/Bundesnetzagentur/SharedDocs/Mediathek/Monitoringberichte/MonitoringberichtEnergie2024.pdf. Accessed 7 May 2025.

[Fuhrländer-Völker 2023] Fuhrländer-Völker, D. Automation Architecture for Demand Response on Aqueous Parts Cleaning Machines. Technical University of Darmstadt, 2023.

[Grosch 2024] Grosch, B. Simplified Implementation of Energy-Aware Production Scheduling in Job Shops. Düren: Shaker Verlag, 2024.

[Sauer 2019] Sauer, A., et al. Energieflexibilität in der deutschen Industrie. Ergebnisse aus dem Kopernikus-Projekt - Synchronisierte und energieadaptive Produktionstechnik zur flexiblen Ausrichtung von Industrieprozessen auf eine fluktuierende Energieversorgung (SynErgie). Stuttgart: Fraunhofer Verlag, 2019.

[Sauer 2022] Sauer, A., et al. Energieflexibilität in der deutschen Industrie. Band 2. Markt- und Stromsystem, Managementsysteme und Technologien energieflexibler Fabriken. Stuttgart: Fraunhofer Verlag, 2022.

[Silberschatz 2018] Silberschatz, A., et al. Operating System Concepts. Hoboken, NJ: Wiley, 2018.

[U.S. Department of Energy 2006] U.S. Department of Energy. Benefits of Demand Response in Electricity Markets and Recommendations for Achieving Them. Washington, DC: U.S. Department of Energy, 2006. https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=a6a4ea518799c16a26f76597182df1cd0555c78e. Accessed 8 May 2025.

Software:

[Grönholm 2024] Grönholm, A. apscheduler, version 4.0.0a5, 2024. https://github.com/agronholm/apscheduler. Accessed 7 May 2025.

[Stock 2025] Stock, J. EAPS Execution (1.0.0), 2025. Zenodo. https://doi.org/10.5281/zenodo.16962147. Accessed 27 August 2025.