
A MEMETIC NSGA-III FOR GREEN FLEXIBLE PRODUCTION WITH REAL-TIME ENERGY COSTS & EMISSIONS *

Sascha C Burmeister
Management Information Systems
Paderborn University
Paderborn, Germany
sascha.burmeister@upb.de

ABSTRACT

The use of renewable energies strengthens decarbonization strategies. To integrate volatile renewable sources, energy systems require grid expansion, storage capabilities, or flexible consumption. This study focuses on industries that adapt production to real-time energy markets, offering flexible consumption to the grid. Flexible production considers not only traditional goals like minimizing production time, but also minimizing energy costs and emissions, thereby enhancing the sustainability of businesses. However, existing research focuses on single goals, neglects the combination of makespan, energy costs, and emissions, or assumes constant or periodic tariffs instead of a dynamic energy market. We present a novel memetic NSGA-III to minimize makespan, energy cost, and emissions, integrating real energy market data, and allowing manufacturers to adapt energy consumption to current grid conditions. Evaluating it with benchmark instances from literature and real energy market data, we explore the trade-offs between objectives, showcasing potential savings in energy costs and emissions on estimated Pareto fronts.

Keywords OR in Energy · Sustainable Production · Green Flexible Job Shop Scheduling · Memetic NSGA-III

1 Introduction

Renewable energy is a key solution to the challenge of climate change. However, traditional energy systems face high penetration of volatile renewable energy sources and face new challenges in terms of power quality, reliability, or power system reliability (Basit et al. 2020). To support the integration of renewable energy sources, energy systems need grid expansion, storage capabilities, or flexible loads. In the context of flexible loads, consumers are encouraged to shift energy demand away from peak times to relieve grid load and toward periods of high renewable generation, promoting sustainable energy consumption. For manufacturing industries as large energy consumers, flexible loads offer a high potential for cost savings (Keller et al. 2017). Additionally, the sustainability of energy production can also be considered. Instead of pricing emitted emissions through CO₂ taxes and thus implicitly incorporating them into energy costs, it can be beneficial to explicitly account for emissions in production planning. This allows focused attention on a company's carbon footprint, which is advantageous for reasons such as anticipating or responding to regulatory pressures, attracting environmentally conscious customers, and maintaining legitimacy with external stakeholders (Dahlmann et al. 2019). This work focuses on the challenge manufacturers face in aligning their production schedule with fluctuations in energy prices and emissions of the energy market.

In the energy market, there are three classic types of energy tariffs: (1) fixed energy tariffs with constant energy prices, (2) time-of-use (TOU) tariffs, where energy prices are divided into several time periods (e.g., on-peak and off-peak prices), and (3) real-time pricing (RTP) tariffs, where the cost of electricity consumption is based on the electricity exchange and changes at least every hour. The latter allows manufacturers to align their production with the energy market, respond to price signals, adapt flexibly to the needs of the power system, and thereby reduce energy cost and emissions (Finn & Fitzpatrick 2014).

Accepted for publication in:* **Croatian Operational Research Review

In the field of operations research, classical scheduling problems prioritize economic factors such as the schedule's makespan. Green Job Shop Scheduling Problems are an extension of classical scheduling problems, incorporating resource and environmental aspects (Li & Wang 2022). As we show in Section 2, recent research places different focuses on economic goals such as the makespan or energy costs and ecological goals such as emissions. To the best of our knowledge, no research has combined the minimization of makespan, energy cost, and emissions while considering RTP tariffs. However, integrating these factors is essential for production planners to accurately calculate the potential reductions in energy costs and emissions achievable through improved production flexibility.

In this work, we focus on the effects of flexible production by exploring the research question: How do scheduling decisions that prioritize one objective — makespan, energy cost, or emissions — affect the others? Our contributions include (1) the formulation of a Flexible Job Shop Scheduling Problem (FJSP) with dynamic energy cost and emissions, (2) developing a memetic algorithm based on the Non-Dominated Sorting Algorithm III (NSGA-III), and (3) conducting computational experiments to evaluate the trade-offs among makespan, energy costs, and emissions. In our study, we continue the work of Burmeister et al. (2023), who formulate a model for the bicriteria FJSP with respect to makespan and energy cost and present a memetic NSGA-II as a solution approach. Our mathematical model extends this framework to include emissions as a third objective. Additionally, our memetic NSGA-III enhances the existing memetic NSGA-II, enabling the calculation of multi-objective schedules aimed at minimizing makespan, energy cost, and emissions. This extension allows us to evaluate trade-offs among all three objectives in our computational experiments and quantify potential savings for practitioners.

As a solution approach, we opt for memetic NSGA-III due to its ability to represent schedules in a three-dimensional Pareto front, allowing decision-makers to balance trade-offs and select solutions based on their preferences. It also leads to favorable results in similar FJSP problems (Sang & Tan 2022, Wu et al. 2021) and has already been used to optimize schedules with respect to energy-related goals (Sun et al. 2021). A notable aspect of our study is the evaluation of the scheduling problem using real energy costs and emissions data from the German energy market.

The remainder of this paper is organized as follows. Section 2 presents recent research in the area of green scheduling. Section 3 introduces the mathematical model, for which we present a memetic NSGA-III in Section 4. Section 5 shows our computational experiments and discusses our results. Section 6 summarizes our results and recommends directions for future research.

2 Recent research

In this section, we present research related to energy cost- and emission-aware scheduling. We discuss the respective objectives and solution approaches presented in the literature.

A schedule can be designed to minimize makespan, emissions, energy cost, or a combination of these. In order to take ecological and economic goals into account, one approach is to minimize makespan while limiting energy consumption (Carlucci et al. 2021). Energy consumption can also be minimized as an objective, but is still independent of the underlying energy mix and its costs and emissions (Lu et al. 2021, Sun et al. 2021). Wang et al. (2020) incorporate an economic view of energy consumption and minimize the makespan and energy costs. They divide a day into several periods with different energy prices based on a TOU tariff. A more detailed consideration of dynamic prices with RTP tariffs can be found in Abikarram et al. (2019) and Fazli Khalaf & Wang (2018). They consider hourly-changing prices but neglect a minimization of the emissions and the makespan of the production schedule. Burmeister et al. (2023) present a model for bicriteria optimization of makespan and energy cost, considering an RTP tariff.

Due to the NP-hard complexity of Job Shop Scheduling Problems (Garey et al. 1976), many studies favor metaheuristics over exact solution methods, allowing fast adaptation to fluctuating prices and emissions. Schulz et al. (2019) use a multiphase iterated local search algorithm to determine a Pareto front regarding makespan, total energy costs and peak load. Most studies on energy-efficient scheduling reviewed in Gao et al. (2020) employ swarm intelligence and evolutionary algorithms. Dong & Ye (2022) design an improved hybrid salp swarm and NSGA-III algorithm to reduce carbon emissions and energy costs under the TOU tariff. Lu et al. (2021) and Wang et al. (2020) apply a multi-objective Genetic Algorithm, while Burmeister et al. (2023) and Sun et al. (2021) use multi-objective evolutionary metaheuristics for fast computation.

Recent research has focused on several areas of green scheduling. However, the reviewed works that include energy costs and emissions neglect the real-time energy market, while studies related to real-time energy markets do not include emissions. We aim to fill this research gap by combining economic and environmental perspectives, taking into account a dynamic energy market.

3 Mathematical model

In this section, we present the mathematical optimization model for the multi-objective FJSP with the objectives of minimizing makespan, energy cost, and emissions. The model is based on the research conducted by Burmeister et al. (2023), which we extend to take into account emissions. Table 1 presents the notation for the mathematical model. The set $J = \{1, \dots, \mu\}$ contains μ jobs to be processed, each of which is divided into v_i operations $O_i = \{(i, 1), \dots, (i, v_i)\}$. The operations of a job must be processed in sequence. The set $O = \bigcup_{i \in J} O_i$ contains all operations. For each operation, τ_{ijk} specifies the duration with which an operation (i, j) can be processed on machine $k \in M$. Supplementary, $\eta_{ijk t}$ and $\zeta_{ijk t}$ contain the energy cost and emissions for processing operation (i, j) on machine $k \in M$ beginning at time $t \in T$.

Table 1: Notation for the math. formulation

Notation	Description	Notation	Description
<i>Sets</i>		<i>Variables</i>	
J	Jobs, $i \in J$	c^{max}	Maximum makespan
O	Operations, $O = \bigcup_{i \in J} O_i$, $O_i = \{(i, 1), \dots, (i, v_i)\}$	p^{sum}	Sum of all energy cost
M	Machines, $k \in M$	e^{sum}	Sum of all emissions
T	Time steps, $t \in T$	s_{ijk}	Start time of operation (i, j) on machine k
<i>Parameters</i>		c_{ijk}	End time of operation (i, j) on machine k
L	A large number	x_{ijk}	Binary indicator, 1 iff operation (i, j) is allocated on machine k
τ_{ijk}	Processing time of operation (i, j) on machine k	$y_{ij'j'k}$	Binary indicator, 1 iff operation (i, j) is predecessor of operation (i', j') on machine k
$\eta_{ijk t}$	Energy cost for processing operation (i, j) on machine k when starting at time t	$p_{ijk t}$	Binary indicator, 1 iff operation (i, j) starts on machine k at time t
$\zeta_{ijk t}$	Energy emissions for processing operation (i, j) on machine k when starting at time t		

Objective function 1 minimizes the variables c^{max} , p^{sum} , and e^{sum} , which represent the maximum makespan, the sum of all energy cost, and the sum of all emissions, respectively. Constraint 2 reflects the final completion time across all operations $(i, j) \in O$ and machines $k \in M$. Constraints 3 and 4 sum the energy cost $\eta_{ijk t}$ and emissions $\zeta_{ijk t}$, respectively, for all operations $(i, j) \in O$ on all machines $k \in M$ at all time steps $t \in T$.

Constraints 5 to 10 are based on the MILP formulation for the general FJSP of Özgüven et al. (2010). They ensure that each operation is assigned to exactly one machine (Constraint 5), that operations of a job can only start if previously required operations have been completed (Constraints 6 to 8), and that operations of a machine cannot overlap (Constraints 9 and 10).

To consider the energy consumption in the model, we add a link from the allocation of the operations to their individual energy consumption: Constraint 11 forces the sum over binary indicators $p_{ijk t}$ over all time steps t to be one, if operation (i, j) is assigned to machine k . Constraints 12 and 13 ensure that the binary indicator $p_{ijk t}$ is set to one for the time t at which the operation (i, j) starts.

The model is composed of binary and continuous variables. Together with convex and linear constraints, it is classified as an MILP.

$$\begin{aligned}
 \min (c^{max}, p^{sum}, e^{sum}) & \quad (1) \\
 \text{s.t. } c^{max} & \geq c_{ijk} \quad \forall i, j, k & (2) \\
 p^{sum} & \geq \sum_{i,j,k,t} \eta_{ijkt} p_{ijkt} & (3) \\
 e^{sum} & \geq \sum_{i,j,k,t} \zeta_{ijkt} p_{ijkt} & (4) \\
 \sum_k x_{ijk} & = 1 \quad \forall i, j & (5) \\
 s_{ijk} + c_{ijk} & \leq x_{ijk} L \quad \forall i, j, k & (6) \\
 c_{ijk} & \geq s_{ijk} + \tau_{ijk} - (1 - x_{ijk})L \quad \forall i, j, k & (7) \\
 \sum_k s_{ijk} & \geq \sum_k c_{i,j-1,k} \quad \forall i, j & (8) \\
 s_{ijk} & \geq c_{i'j'k} - y_{ij'j'k} L \quad \forall i, j, i', j', k & (9) \\
 s_{i'j'k} & \geq c_{ijk} - (1 - y_{ij'j'k})L \quad \forall i, j, i', j', k & (10) \\
 x_{ijk} & = \sum_t p_{ijkt} \quad \forall i, j, k & (11) \\
 s_{ijk} - t & \geq -(1 - p_{ijkt})L \quad \forall i, j, k, t & (12) \\
 s_{ijk} - t & \leq (1 - p_{ijkt})L \quad \forall i, j, k, t & (13) \\
 c^{max}, p^{sum}, e^{sum}, s_{ijk}, c_{ijk} & \in \mathbb{R}_+, & \\
 x_{ijk}, y_{ij'j'k}, p_{ijkt} & \in \{0, 1\} \quad \forall i, j, i', j', k, t & (14)
 \end{aligned}$$

4 The memetic NSGA-III

In this section, we present the memetic NSGA-III for solving the green multi-objective FJSP. We divide the section into two parts, first introducing the representation of solutions in genotypes and phenotypes in Subsection 4.3, before explaining the algorithm of the memetic NSGA-III in Subsection 4.4.

4.1 Representation of solutions

The NSGA-III is an evolutionary algorithm and describes solutions as individuals of a population that evolve over multiple generations. To represent solutions as individuals, we follow a decoder-based approach that encodes solutions as genotypes and uses phenotypes for decoding. We base our genotype on the work of Dai et al. (2019), who choose a bipartite gene string for the sequence of operations and their machine assignment, and Burmeister et al. (2023), who choose a tripartite gene string for the additional representation of the maximum allowable energy cost per operation. To account for emissions, we extend latter approach and add a fourth gene string to this representation to indicate the maximum allowable emissions per operation. Fig. 3 shows an example genotype for three jobs to be assigned to two machines.

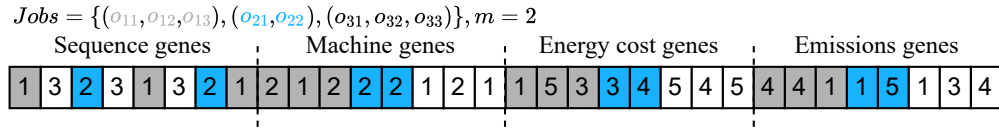


Figure 1: Example genotype for solution encoding

Fig. 4 illustrates the representation of the genotype on Fig. 3. The allocation of operations to the machines is plotted as a Gantt chart, with time steps noted on the x-axis and machines on the left y-axis. On the right y-axis are the energy cost values, shown as a dashed line, and the emissions values, shown as a dotted line. Energy cost are given in €/MWh and emissions are given in grams of carbon dioxide equivalent (gCO₂eq) per kWh. The sequence gene string specifies the order in which the jobs are placed, while the machine gene string reflects the machine to be selected. Thus, operation (1,1) is first assigned to machine 2. The operation is scheduled at the first possible time that satisfies both the associated maximum allowable energy cost of the energy cost gene string and the emissions of the emissions gene string. This means that operation (1,1) is scheduled at the first time that costs less than or equal to 1 € and less than or equal to 4 gCO₂eq, which is time step 4.

All jobs are scheduled in the order specified by the sequence gene string. If the time horizon considered is not sufficient to place an operation, e.g., because the allowed energy cost or emissions are too high, either the time horizon can

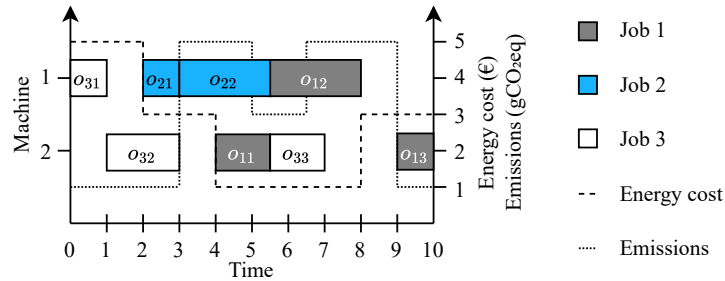


Figure 2: Representation of the example genotype as a phenotype

be extended or the energy cost and emissions can be reduced. While the former tends to produce long schedules with favorable energy cost and emissions, the latter tends to produce fast and expensive schedules. To avoid bias and promote population diversification, we alternate the approach in each generation. Overall, the genotype and phenotype can represent any sequence of operations on any combination of machines, energy cost, and emissions, resulting in a complete representation of the solution space.

4.2 Algorithm

In this section, we present the memetic NSGA-III for solving the green multi-objective FJSP. We divide the section into two parts, first introducing the representation of solutions in genotypes and phenotypes in Section 4.3, before explaining the algorithm of memetic NSGA-III in Section 4.4.

4.3 Representation of solutions

The NSGA-III is an evolutionary algorithm and represents solutions as individuals of a population that evolve over multiple generations. To represent solutions as individuals, we follow a decoder-based approach that encodes solutions as genotypes and uses phenotypes for decoding. We base our genotype on the work of Dai et al. (2019), who choose a bipartite gene string for the sequence of operations and their machine assignment, and Burmeister et al. (2023), who choose a tripartite gene string for the additional representation of the maximum allowable energy cost per operation. To account for emissions, we extend the latter approach and add a fourth gene string to this representation to indicate the maximum allowable emissions per operation.

Figure 3 shows an example genotype for three jobs to be assigned to two machines. The genotype is divided into four strings of genes, representing the sequence in which operations are allocated, the machine allocation, the tolerated energy costs, and the tolerated emissions. Genes linked to jobs 1, 2, and 3 are highlighted in gray, blue, and white, respectively. Each section has a length corresponding to the total number of operations. For decoding the genotype in Figure 3 as a schedule, Figure 4 illustrates the phenotype. The allocation of operations to the machines is plotted as a Gantt chart, with time steps noted on the x-axis and machines on the left y-axis. On the right y-axis are the energy cost values, shown as a dashed line, and the emissions values, shown as a dotted line. Energy cost is given in €/MWh and emissions are given in grams of carbon dioxide equivalent (gCO₂eq) per kWh. The sequence gene string specifies the order in which the jobs are placed, while the machine gene string reflects the machine to be selected. Thus, operation (1,1) is first assigned to machine 2. The operation is scheduled at the first possible time that satisfies both the associated maximum allowable energy cost of the energy cost gene string and the emissions of the emissions gene string. This

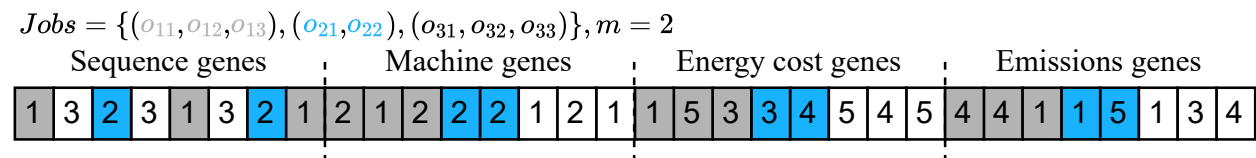


Figure 3: Example genotype for solution encoding based on Burmeister et al. (2023)

means that operation (1,1) is scheduled at the first time that costs less than or equal to 1 € and less than or equal to 4 gCO₂eq, which is time step 4.

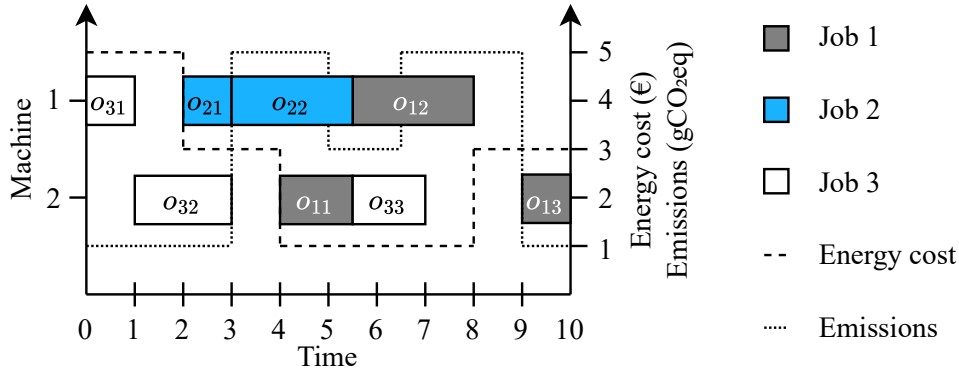


Figure 4: Representation of the example genotype as a phenotype based on Burmeister et al. (2023)

All jobs are scheduled in the order specified by the sequence gene string. If the time horizon considered is not sufficient to place an operation, e.g., because the allowed energy cost or emissions are too high, either the time horizon can be extended or the energy cost and emissions can be reduced. While the former tends to produce long schedules with favorable energy cost and emissions, the latter tends to produce fast and expensive schedules. To avoid bias and promote population diversification, we alternate the approach in each generation. Overall, the genotype and phenotype can represent any sequence of operations on any combination of machines, energy cost, and emissions, resulting in a complete representation of the solution space.

4.4 Algorithm

In this section, we explain our memetic NSGA-III. We base the algorithm on the NSGA-III (Deb & Jain 2014), which uses a similar framework to its predecessor NSGA-II (Deb et al. 2002) while providing better diversity.

The NSGA-III is outlined in Algorithm 1. For a population P_t of size N , NSGA-III generates an equally sized next generation P_{t+1} using recombination and mutation operators: For recombination, we use a two-point crossover to intensify the search in the solution space and to find better individuals based on existing ones. From two parents, the genes of the gene strings are swapped between two different random positions. The same positions for swapping are chosen for all gene strings. By swapping, the gene strings remain feasible for machine assignment, energy cost, and emissions. In the case of the sequence gene string, swapping may cause an infeasibility, e.g., if a job is no longer listed in the number of its operations. In this case, the defective children are repaired by replacing excess jobs with missing ones. For mutation, the algorithm randomly modifies gene strings by performing one of the following changes: swapping two genes within the sequence gene string, reassigning a gene in the machine gene string, or assigning new values to either the energy cost gene string or the emission gene string.

We extend NSGA-III to a memetic algorithm and adopt local refinement from Burmeister et al. (2023). In their local refinement, a greedy approach improves solutions by adjusting the values of the gene strings for energy cost, ensuring that lower energy cost are achieved without increasing the makespan. We extend this approach to include emissions as shown in Figure 5. (1) First, it sorts all operations of the parent, based on their energy consumption, in descending order into a queue L . (2) Second, it calculates the lower and upper feasible start times l_{cij} and u_{cij} for each operation $(i, j) \in O$ based on the duration of previous and subsequent operations of the job for both children $c \in \{1, 2\}$. (3) Third, the greedy procedure selects the operation with the highest energy consumption, and (4) fourth, schedules the selected operation at the time of the cheapest energy cost (4a) and emissions (4b), respectively. (5) Fifth, it adjusts the earliest possible start and end times of the operations that are still to be sorted, and (6) dequeues the current operation from L . The procedure repeats steps (3) and (4) successively for the next operation until all operations in L have been scheduled.

After recombination, mutation, and local refinement, the resulting set R_t is sorted into several non-dominated fronts as described in Deb & Jain (2014). Algorithm 1 performs a non-dominated sort of individuals on different fronts and successively adds the individuals of F_i to the set S_t until their cardinality is greater than or equal to the desired population size N . If $|S_t| = N$, the next generation P_{t+1} inherits all individuals from S_t . Otherwise, all previous fronts except the last front F_i are added to the new generation. To decide which k individuals remaining in F_i are included in

Algorithm 1 Pseudocode of the memetic NSGA-III based on Deb & Jain (2014)

Require: Population P_t , Population size N
 $S_t = \emptyset, i = 1$
 $R_t \leftarrow P_t \cup \text{RECOMBINATION+MUTATION}(P_t)$
 $R_t \leftarrow \text{LOCALREFINEMENT}(R_t)$
 $(F_1, F_2, \dots) = \text{NON-DOMINATED-SORT}(R_t)$
repeat
 $S_t = S_t \cup F_i$ and $i \leftarrow i + 1$
until $|S_t| \geq N$
if $|S_t| = N$ **then**
 $P_{t+1} \leftarrow S_t$
else
 $P_{t+1} \leftarrow \bigcup_{j=1}^{i-1} F_j$
 $k \leftarrow N - |P_{t+1}|$
 $Z^r \leftarrow \text{NORMALIZATION}(S_t)$
 $\text{ASSOCIATION}(Z^r, S_t)$
 $P_{t+1} \leftarrow P_{t+1} \cup \text{NICHING}(F_i, k)$
 \triangleright Remaining space in P_{t+1}

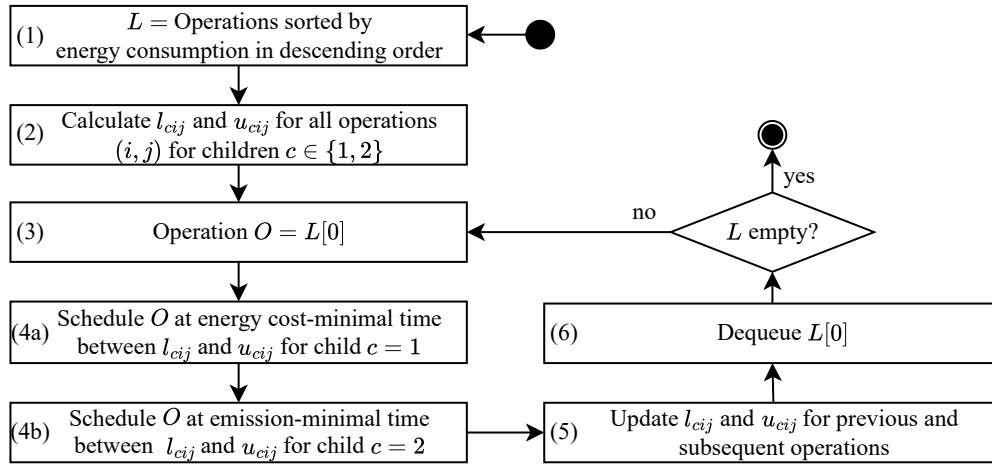


Figure 5: Greedy local refinement

P_{t+1} , the algorithm performs (1) normalization, (2) association, and (3) niching. The three procedures are explained in detail in Deb & Jain (2014) and are briefly outlined below: (1) First, the three values of the objective function of all individuals in S_t are normalized. (2) Then, the algorithm creates reference points Z^f and associates each individual from S_t with its nearest reference point. (3) Finally, the niching procedure successively adds the individuals from F_i to P_{t+1} with which the fewest individuals are associated. Niching ends when k individuals have been added to the new generation P_{t+1} , so $|P_{t+1}| = N$ holds. The memetic NSGA-III iterates until a termination criterion (e.g., generation or runtime limit) is satisfied. The first front F_1 of the last generation is then the estimate of the Pareto front.

5 Computational experiments

In this section, we present the computational experiments. Section 5.1 describes the experimental setup of scheduling problems in the presence of real-world energy market data. Sections 5.2 to 5.4 present the results and discuss the pairwise relationship of the objective function values, i.e., the relationship between makespan and energy cost, makespan and emissions, and energy cost and emissions. Section 5.5 formulates implications.

5.1 Instances and experimental setting

For the computational experiments, we use the benchmark set of Brandimarte (1993). Table 2 shows the set that contains 15 instances for the FJSP with jobs and their respective operations as well as machines, making it suitable for

emulating production schedules. We select a 15-minute duration for each time step. Thus, a schedule covers multiple periods with varying energy prices and emissions.

Table 2: Benchmark instances by Brandimarte (1993)

Instance	Jobs	Machines	Operations per job	Operations in total	Time steps per operation
mk01	10	6	5-7	55	1-7
mk02	10	6	5-7	58	1-7
mk03	15	8	10	150	1-20
mk04	15	8	3-10	90	1-10
mk05	15	4	5-10	106	5-10
mk06	10	10	15	150	1-10
mk07	20	5	5	100	1-20
mk08	20	10	5-10	225	5-20
mk09	20	10	10-15	240	5-20
mk10	20	15	10-15	240	5-20
mk11	30	5	5-8	179	10-30
mk12	30	10	5-10	193	10-30
mk13	30	10	5-10	231	10-30
mk14	30	15	8-12	277	10-30
mk15	30	15	8-12	284	10-30

For consideration of energy consumption, we add an energy demand of $500 \frac{i}{|J|}$ kW for each operation of job $i \in J$, this means the demands of the operations range up to 500 kW. We enrich the instances with real data from the German energy market published by the Federal Network Agency Germany (2024). Figure 6 shows both electricity prices from the German wholesale market and emission values in hourly resolution from February 1st to June 30th, 2022. Energy prices are determined by the European Energy Exchange. Emission values are based on lifecycle emissions per generation technology, as outlined in Schlömer et al. (2014). The Pearson correlation coefficient for energy prices and energy emissions is 0.72.

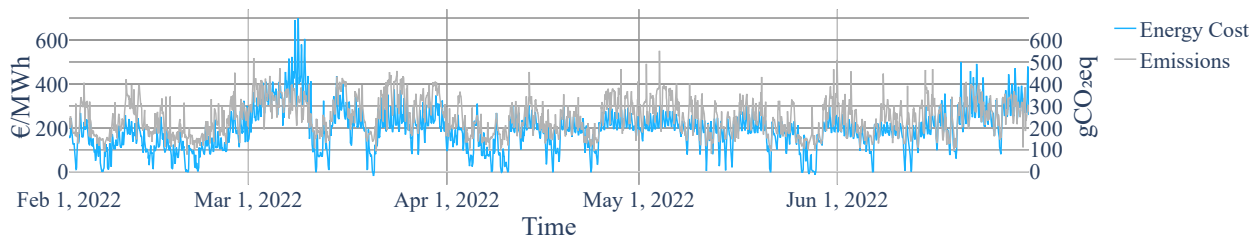


Figure 6: Electricity prices and emissions of the German energy market Federal Network Agency Germany (2024)

For parameterization, we follow the settings of Burmeister et al. (2023). We limit the runtime to 45 minutes to reflect the flexibility to respond to hourly price changes in the energy market. The memetic NSGA-III is implemented in C# 10 within the .NET 6 software framework. The problem is solved on a Red Hat Enterprise Linux 8.5 (Oopta) operating system with an Intel Xeon Gold 6148 CPU, 20x2.4GHz, and 190 GByte main memory.

5.2 Relation of makespan to energy cost

In this section, we focus on the relationship between makespan and energy cost without considering emissions. The first three columns of Table 3 show the instance, the minimum makespan found by the memetic NSGA-III, and the associated energy cost. The remaining columns show the percentage of energy cost that can be saved by increasing the makespan.

A 5% makespan increase saves between 0.1% (mk02) to 14.6% (mk07) of energy cost across all instances. For a 20% makespan increase, the savings range from 2.2% (mk02 and mk14) to 30.4% (mk13). For increases in makespan of 50% and 75%, the energy cost remain stagnant for instances mk03, mk05, mk08, mk11, mk12, and mk14, showing no further improvements. For other instances, energy cost continues to decrease, achieving savings of up to 45.0% (mk06).

At a 75% makespan increase, instances mk06 and mk04 show the highest energy cost savings with 49.3% and 51.7%, while instances mk03 and mk14 show the lowest savings (2.5% and 2.2%).

The average savings in energy cost are 5.86%, 12.31%, 18.89%, and 22.35% for makespan increases of 5%, 20%, 50%, and 75%, respectively. The results indicate that small increases in makespan are more efficient for energy cost savings, as the relative increase in savings surpasses the increase in makespan for 9 out of 15 instances. However, efficiency decreases with larger increases in makespan. With a 20% increase in makespan, the savings in energy cost exceed the 20% mark in 2 out of 15 instances. Savings with an increase in makespan of 50% and 75% are below 50% and 75%, respectively, for all instances.

5.3 Relation of makespan to emissions

Table 4 shows the results of the reduction in emissions with increasing makespan and follows the structure of Table 3. A 5% makespan increase saves between 0.4% (mk02) and 10.5% (mk13) of the emissions. With a 20% makespan increase, the savings range from 1.0% (mk14) to 19.1% (mk13). For makespan increases of 50% and 75%, the emissions remain constant for the instances mk03, mk05, mk07, mk08, mk11, and mk14. Other instances show emissions savings of up to 23.7% (mk15) and 24.1% (mk09) with makespan increases of 50% and 75%, respectively.

The average emissions savings are 4.08%, 8.11%, 11.05%, and 12.59% for relative makespan increases of 5%, 20%, 50%, and 75%, respectively. Again, the most efficient savings are observed for smaller instances. For a 5% makespan increase, the relative increase in savings exceeds the increase in makespan for 6 of 15 instances.

The emissions savings are lower than the energy cost savings in Table 3. This can be attributed to the fact that the market values of the emissions have a positive value range, while the energy cost may have negative values. As a result, the algorithm has fewer opportunities to schedule jobs with high energy demands at favorable times when considering emissions.

5.4 Relation of energy cost to emissions

Table 5 shows the results of the emissions savings as energy cost increase without considering the makespan. It is structured similarly to Table 3 and 4. For instance mk02, the absolute value of the energy cost was used because the instance has few jobs that are processed for negative energy cost.

A 5% energy cost increase saves between 0.7% (mk08) and 5.6% (mk06) of emissions. For increases in energy costs of 20, 50 and 75%, the maximum savings are 8.6 (mk06), 12.1 (mk06), and 14.2% (mk02), respectively, and the minimum savings are 1.4% (mk08). Except for the 5.6% reduction in emissions for a 5% deterioration in energy cost for the mk07 instance, the relative savings are less than the relative deterioration in all other cases. The average emissions savings are 1.89%, 4.52%, 5.92%, and 6.74% for relative energy cost increases of 5%, 20%, 50%, and 75%, respectively.

5.5 Implications

In this section, we draw implications from the results. Figure 7 summarizes the medians of the percentage savings in energy costs when increasing the makespan (solid line), emissions when increasing the makespan (dashed line), and emissions when increasing the energy cost (dotted line). The error bars show the minimum and maximum values.

Table 3: Energy cost savings (in %) with increase in makespan

Inst.	min		Increase of ms^1 (%)				Inst.	min		Increase of ms^1 (%)			
	ms^1	ec^2	5	20	50	75		ms^1	ec^2	5	20	50	75
mk01	42	3965	2.0	6.5	17.0	22.3	mk09	341	45041	5.5	16.7	32.5	38.6
mk02	29	3204	0.1	2.2	2.3	3.1	mk10	263	40158	7.0	10.2	16.1	26.3
mk03	204	13954	1.7	2.5	2.5	2.5	mk11	621	41038	8.6	10.1	10.1	10.1
mk04	66	6943	0.7	7.7	26.5	51.7	mk12	524	49137	7.7	18.6	18.6	18.6
mk05	174	11791	5.0	6.9	6.9	6.9	mk13	457	71059	11.8	30.4	36.7	36.7
mk06	77	7988	3.7	15.9	45.0	49.3	mk14	694	63382	2.2	2.2	2.2	2.2
mk07	144	10658	14.6	15.9	15.9	15.9	mk15	429	89821	8.4	24.3	36.6	36.6
mk08	523	37481	8.9	14.5	14.5	14.5							

¹ Makespan (time steps), ² Energy cost (€)

Table 4: Emissions savings (in %) with increase in makespan

Inst.	min		Increase of ms^1 (%)				Inst.	min		Increase of ms^1 (%)			
	ms^1	em^2	5	20	50	75		ms^1	em^2	5	20	50	75
mk01	42	5.215	1.3	5.4	12.7	17.7	mk09	341	68.240	3.4	11.8	20.4	24.1
mk02	29	4.627	0.4	2.2	3.7	5.3	mk10	263	61.328	5.0	7.6	13.4	21.2
mk03	204	24.257	2.9	2.9	2.9	2.9	mk11	621	73.575	4.7	5.6	5.6	5.6
mk04	66	9.370	1.0	7.9	14.9	17.0	mk12	524	83.272	5.7	10.4	10.7	10.7
mk05	174	18.303	3.5	3.6	3.6	3.6	mk13	457	109.633	10.5	19.1	23.1	23.1
mk06	77	10.436	3.2	11.2	14.7	17.6	mk14	694	118.605	1.0	1.0	1.0	1.0
mk07	144	17.517	6.3	7.1	7.1	7.1	mk15	429	139.310	6.9	17.6	23.7	23.7
mk08	523	64.146	5.4	8.3	8.3	8.3							

¹ Makespan (time steps), ² Emissions (tons of CO₂eq)

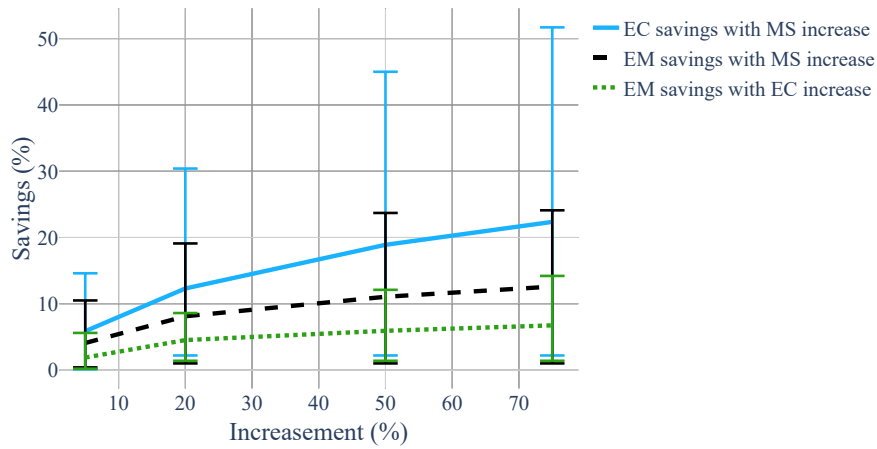


Figure 7: Minimum, mean and maximum savings

First, extending the makespan can lead to savings in energy costs and emissions. The percentage values observed in Figure 7 indicate disproportionately high average savings in energy costs when the makespan increases by 5%. In contrast, increases in makespan beyond 5% and reductions in emissions show lower percentage savings compared to the initial increase in makespan. Energy costs show a higher savings potential than emissions, as the energy market can have negative prices, while emissions cannot be negative. Although an increase in makespan results in a disproportionately low increase in the percentage of savings in energy costs and emissions, the absolute savings remain considerable. This is because the underlying absolute energy costs range between €3,204 and €89,821, and emission values range from 4.63 to 139.31 tons of CO₂eq. We advise decision-makers to consider the benefits of enhancing production flexibility for potential savings.

Table 5: Emissions savings (in %) with increase in energy cost

Inst.	min		Increase of ec^1 (%)				Inst.	min		Increase of ec^1 (%)			
	ec^1	em^2	5	20	50	75		ec^1	em^2	5	20	50	75
mk01	5	3.249	0.3	1.7	5.4	6.2	mk09	10170	45.452	2.0	5.7	6.1	6.1
mk02 ³	-2	3.098	2.1	6.6	9.4	14.2	mk10	7182	40.082	1.8	5.0	6.7	7.3
mk03	1282	17.464	1.3	6.6	8.1	9.7	mk11	18933	60.929	1.4	2.5	2.5	2.5
mk04	71	6.343	1.6	5.7	8.0	8.9	mk12	16571	62.976	1.0	2.7	2.8	2.8
mk05	838	13.940	2.2	4.0	5.2	6.6	mk13	22309	74.076	3.4	3.6	3.6	3.6
mk06	132	7.506	0.2	8.6	12.1	13.5	mk14	32770	100.836	1.4	2.2	2.2	2.2
mk07	668	13.517	5.6	6.9	10.7	11.5	mk15	29438	93.441	3.3	4.6	4.6	4.6
mk08	13019	48.015	0.7	1.4	1.4	1.4							

¹ Energy cost (€), ² Emissions (tons of CO₂eq), ³ Since the value is negative, we use the absolute value for the percentage increase.

Second, opting for an increase in energy cost to reduce emissions is less effective than increasing makespan. The average emission reductions achieved by creating time flexibility with a longer makespan is higher than that resulting from the acceptance of higher energy cost for the purchase of renewable energy. If decision-makers are unable to consider an increase in makespan (e.g., due to deadlines), it may still be attractive for manufacturers to balance higher energy costs against emissions savings. Additional costs for reducing or avoiding emissions could be incorporated into adjusted product pricing, thereby promoting sustainable production without extending the makespan.

6 Conclusion

Our study addresses the challenge of managing flexible loads in manufacturing industries, with the aim of optimizing schedules with respect to makespan, energy cost, and emissions. Drawing from existing literature, we introduce a linear optimization model for a Green FJSP and develop a memetic NSGA-III approach as a three-objective solution method. Evaluating our method on benchmark instances enriched with real energy market data from Germany, we analyze potential savings in energy costs and emissions. This helps production planners assess the benefits of flexible production planning adapted to the energy market. We find that even a small extension of the makespan can lead to substantial savings in energy costs and emissions, with potential savings in energy costs being higher than those in emissions. In addition, emissions can be reduced at the expense of higher energy costs. However, while the potential savings from increased makespan are significant, they do not match the extent of savings achievable through flexible production. Our findings underscore the importance of considering the interplay between the considered objectives to strengthen sustainable production planning in manufacturing companies.

Based on the limitations and findings of this study, we advise avenues for further research. First, we recommend comparing the solutions obtained from our approach with those of other state-of-the-art methods. Given that this work focuses on exploring trade-offs between objectives, our results currently serve as lower bounds for potential savings, suggesting that improved solutions may also be possible. In future statistical analysis, solution quality could be examined by assessing differences in efficiency and solution quality, as well as calculating an optimality gap.

Second, we recommend extending our approach to account for data uncertainty, as the current model assumes deterministic knowledge of energy costs and emissions. In a dynamic variant, schedules could be developed using energy market data and forecasts, allowing real-time adjustments to accommodate evolving grid conditions and renewable energy availability. Investigating a dynamic model would provide insights into how decision-makers can adapt to fluctuations in the energy market.

Third, we recommend that our approach be analyzed on real production data. While we use real energy market data combined with scheduling problems from a benchmark set to analyze savings potentials for different-sized problem cases, we also recommend evaluating our approach with production orders from a real manufacturer. In this way, our goal is to achieve a better understanding of the trade-off between makespan, energy cost, and emissions.

Acknowledgments

The authors gratefully acknowledge the financial support provided by the state of North Rhine-Westphalia, Germany, as part of the *progres.nrw* program area, in the framework of Re²Pli (project number EFO 0127A) and the funding of this project by computing time provided by the Paderborn Center for Parallel Computing (PC²).

References

- Abikarram, J. B., McConky, K. & Proano, R. (2019), 'Energy cost minimization for unrelated parallel machine scheduling under real time and demand charge pricing', *Journal of cleaner production* **208**, 232–242.
- Basit, M. A., Dilshad, S., Badar, R. & Sami ur Rehman, S. M. (2020), 'Limitations, challenges, and solution approaches in grid-connected renewable energy systems', *International Journal of Energy Research* **44**(6), 4132–4162.
- Brandimarte, P. (1993), 'Routing and scheduling in a flexible job shop by tabu search', *Annals of Operations research* **41**(3), 157–183.
- Burmeister, S. C., Guericke, D. & Schryen, G. (2023), 'A memetic nsga-ii for the multi-objective flexible job shop scheduling problem with real-time energy tariffs', *Flex Serv Manuf J*.
- Carlucci, D., Renna, P. & Materi, S. (2021), 'A job-shop scheduling decision-making model for sustainable production planning with power constraint', *IEEE Transactions on Engineering Management*.
- Dahlmann, F., Branicki, L. & Brammer, S. (2019), 'Managing carbon aspirations: The influence of corporate climate change targets on environmental performance', *Journal of business ethics* **158**, 1–24.

- Dai, M., Tang, D., Giret, A. & Salido, M. A. (2019), ‘Multi-objective optimization for energy-efficient flexible job shop scheduling problem with transportation constraints’, *Robotics and Computer-Integrated Manufacturing* **59**, 143–157.
- Deb, K. & Jain, H. (2014), ‘An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: Solving problems with box constraints’, *IEEE Transactions on Evolutionary Computation* **18**(4), 577–601.
- Deb, K., Pratap, A., Agarwal, S. & Meyarivan, T. (2002), ‘A fast and elitist multiobjective genetic algorithm: Nsga-ii’, *IEEE Transactions on Evolutionary Computation* **6**(2), 182–197.
- Dong, J. & Ye, C. (2022), ‘Green scheduling of distributed two-stage reentrant hybrid flow shop considering distributed energy resources and energy storage system’, *Computers & Industrial Engineering* **169**, 108146.
- Fazli Khalaf, A. & 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.
- Federal Network Agency Germany (2024), ‘SMARD market data’, online.
URL: <https://www.smard.de/en/downloadcenter/download-market-data>
- Finn, P. & Fitzpatrick, C. (2014), ‘Demand side management of industrial electricity consumption: Promoting the use of renewable energy through real-time pricing’, *Applied Energy* **113**, 11–21.
- Gao, K., Huang, Y., Sadollah, A. & Wang, L. (2020), ‘A review of energy-efficient scheduling in intelligent production systems’, *Complex & Intelligent Systems* **6**, 237–249.
- Garey, M. R., Johnson, D. S. & Sethi, R. (1976), ‘The complexity of flowshop and jobshop scheduling’, *Mathematics of operations research* **1**(2), 117–129.
- Keller, F., Schultz, C., Simon, P., Braunreuther, S., Glasschröder, J. & Reinhart, G. (2017), ‘Integration and interaction of energy flexible manufacturing systems within a smart grid’, *Procedia CIRP* **61**, 416–421.
- Li, M. & Wang, G.-G. (2022), ‘A review of green shop scheduling problem’, *Information Sciences* **589**, 478–496.
- Lu, C., Zhang, B., Gao, L., Yi, J. & Mou, J. (2021), ‘A knowledge-based multiobjective memetic algorithm for green job shop scheduling with variable machining speeds’, *IEEE Systems Journal* **16**(1), 844–855.
- Özgüven, C., Özbakır, L. & Yavuz, Y. (2010), ‘Mathematical models for job-shop scheduling problems with routing and process plan flexibility’, *Applied Mathematical Modelling* **34**(6), 1539–1548.
- Sang, Y. & Tan, J. (2022), ‘Intelligent factory many-objective distributed flexible job shop collaborative scheduling method’, *Computers & Industrial Engineering* **164**, 107884.
- Schlömer, S., Bruckner, T., Fulton, L., Hertwich, E., McKinnon, A., Perczyk, D., Roy, J., Schaeffer, R., Sims, R., Smith, P. et al. (2014), Annex iii: Technology-specific cost and performance parameters, in ‘Climate change 2014: Mitigation of climate change: Contribution of working group III to the fifth assessment report of the Intergovernmental Panel on Climate Change’, Cambridge University Press, pp. 1329–1356.
- Schulz, S., Neufeld, J. S. & Buscher, U. (2019), ‘A multi-objective iterated local search algorithm for comprehensive energy-aware hybrid flow shop scheduling’, *Journal of cleaner production* **224**, 421–434.
- Sun, X., Wang, Y., Kang, H., Shen, Y., Chen, Q. & Wang, D. (2021), ‘Modified multi-crossover operator nsga-iii for solving low carbon flexible job shop scheduling problem’, *Processes* **9**(1), 62.
- Wang, L. et al. (2020), ‘Multi-objective optimization based on decomposition for flexible job shop scheduling under time-of-use electricity prices’, *Knowledge-Based Systems* **204**, 106177.
- Wu, M., Yang, D., Zhou, B., Yang, Z., Liu, T., Li, L., Wang, Z. & Hu, K. (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), 344.