Green smart manufacturing: Energy-efficient robotic job shop scheduling models

Xin Wen, Yige Sun

Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong

windy.wen@polyu.edu.hk (Wen), yige-jessica.sun@connect.polyu.hk (Sun)

Hoi-Lam Ma (corresponding author)

Department of Supply Chain and Information Management, The Hang Seng University of Hong Kong Shatin, N.T., Hong Kong

helenma@hsu.edu.hk (Ma)

Sai-Ho Chung

Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong

nick.sh.chung@polyu.edu.hk (Chung)

Acknowledgements

This work was supported by a grant from the Research Committee of The Hong Kong Polytechnic University under the Project ID P0036042, and under the Account Code RH8P.

Green smart manufacturing: Energy-efficient robotic job-shop scheduling models

Abstract

Smart manufacturing has boosted the wide application of mobile robots in robotic cells for automated material delivery. However, the mismatching between machine production process and robot movement process causes extensive energy waste. Nevertheless, most existing robotic job-shop scheduling (RJSP) studies mainly focus on minimizing makespan but overlook the low energy efficiency problem faced by robotic cells. Motivated by the importance of green smart manufacturing, in this study, we innovatively propose to achieve robotic cell energy saving through coordinating the machine production process and robot movement process. Specifically, both machines and the mobile robot can flexibly adjust operating speeds with a *V-scale* speed framework. Two novel energy-efficient RJSP approaches (i.e., the RJSP-E and the RJSP-EM) are thus proposed. The RJSP-E focuses on minimizing energy consumption, while the RJSP-EM simultaneously considers makespan (i.e., productivity) and energy consumption. Through computational experiments, the RJSP-E demonstrates superior performances in reducing energy consumption (15% on average), at a loss of productivity (20% on average). On the other hand, the RJSP-EM can select the most suitable energy-saving operating speeds without much sacrifice in productivity. Notably, the RJSP-EM can reduce energy consumption by a mean of 10% even without increasing makespan. The RJSP-EM also demonstrates higher solution efficiency.

Keywords: Green production; Smart manufacturing; Robotic job-shop scheduling; Energy saving; Mixed integer linear programming.

1. Introduction

Smart manufacturing is developing fast with the wide application of automated technologies (He and Stecke, 2021; Chung, 2021). In smart robotic cells, autonomous mobile robots (AMRs) are used to pick, transport, and sort items on manufacturing shop floors without manual intervention. The AMRs market values USD 1.9 billion in 2019 and is expected to grow at a compound annual growth rate (CAGR) of

19.6% from 2020 to 2027[1], which demonstrates the promises of such a technology. However, the application of robots leads to higher energy (electricity) [2] consumption, which worsens the serious environmental concerns for the energy-intensive manufacturing industry (Zhang and Yan, 2021). Besides, the existing practice overlooks the coordination between the machinery production process and the robot movement process, leading to excessive energy waste (Gürel, et al., 2019; Liu et al., 2019). The energy waste in robotic cells come from several aspects. First, as the movement of the robot is subjected to diverse restrictions like robot availability, machines may stay idle for a long period (Koulamas and Panwalkar, 2019). Second, the robot/machines generally moves/produce at a constant speed. It is thus commonly seen that (i) products are blocked on machines for a long time (i.e., the product is blocked on the processing machine after completion, waiting for the availability of the robot, named as *machine blocking*), or (ii) the robot arrives at a machine earlier than the completion of the current operation and has to wait there before conducting the next transportation (named as *robot partial-blocking*)[3]. These circumstances imply poor coordination between the machine production process and the robot movement process, which causes nonnegligible energy waste. Considering the increasing energy costs and the growing public awareness of sustainability, the low energy efficiency in robotic cells greatly dampens the benefits brought by robotic technology (Mokhtari and Hasani, 2017). It is thus of great significance to enhance the energy efficiency of robotic cells through improving robotic job-shop scheduling decisions[4] (Jiang and Wang, 2019; Lamotte & Geroliminis, 2021; Parente et al., 2020).

According to Zhang and Chiong (2016), the energy consumption rate of machine production declines if it switches to a slower processing speed. Similarly, the robot is shown to consume less electricity at a slower moving speed (Brossog et al., 2015). Therefore, it is promising to achieve energy

---

1 Details are at: https://www.grandviewresearch.com/industry-analysis/autonomous-mobile-robots-market#:~:text=The%20global%20autonomous%20mobile%20robots%20market%20size%20was,manufacturing%20and%20distribution%20facilities%20without%20any%20manual%20intervention.

[2] In a robotic cell, the energy consumed is generally electricity. In this study, we use “energy” and “electricity” interchangeably.

[3] A *robot full-blocking* refers to the situation that the robot must wait at the machine for the whole operation process to deliver the same job (as instructed by the optimal schedule that minimizes the makespan while avoiding deadlocks).

[4] The robotic job-shop scheduling problem (RJSP) aims to identify the optimal production schedule for machines and the optimal delivery route for robots, while the makespan is usually minimized to improve productivity (Brucker et al., 2012).

conservation through speed adjustments for both machines and robots. The mechanism of energy-saving by speed adjustment is explained as follows. On one hand, to reduce machine idling and blocking periods, machines can process at a slower speed. In fact, as long as the processing finishes before the arrival of the robot, the overall makespan is not affected. In this way, the production energy consumption declines (as the processing speed is slower), while the energy waste caused by machine idling is also reduced. On the other hand, robot partial-blocking can be eliminated/reduced if the robot moves at a slower speed, thus achieving movement energy saving. Accordingly, by adjusting the processing/moving speeds, the system energy consumption of a robotic cell can be reduced significantly. Therefore, proper operating speed selection and better coordination between the machine production process and the robot movement process are crucial for the energy efficiency enhancement of robotic cells. However, two major challenges arise. First of all, the scheduling framework should identify the diverse inter-related factors leading to energy inefficiency in the production system. Second, the scheduling scheme should reduce the energy waste incurred by the speed mismatching between the two processes, while the interaction between the two processes further complicates the scheduling and speed adjustment tasks.

Motivated by the real challenges faced by robotic cells and the potential to reduce energy waste through speed adjustment, in this paper, we propose two novel energy-efficient robotic job-shop scheduling models. A *V-scale* speed framework is thus developed for both machines and the mobile robot. In this *V-scale* speed framework, the processing speed of each machine as well as the moving speed of the robot can be selected from a finite and discrete set with $|V|$ levels. First, a *robotic job-shop scheduling with energy consumption* (i.e., RJSP-E) is developed to minimize the total energy consumption. Then, we further propose a *robotic job-shop scheduling with energy consumption and makespan limitation* (i.e., RJSP-EM) which simultaneously considers energy consumption and system productivity by limiting the increase of makespan. The novelty of our models is twofold. First, the energy consumption derived by both machines and the robot is considered along with the generation of the machine production plans and the robot routes. Second, speed adjustment is applied for both machines and the robot. In this way, the production system can consume less electricity through slower operating speeds, while energy waste can be reduced through better coordination between the machine production process and the robot movement process. Through computational experiments, the RJSP-E is shown to remarkably reduce energy consumption (with an average of 15%) by selecting slower operating speeds. However, the makespan increases by 20% on average. Differently, for the RJSP-EM, as the makespan increase is restricted, it demonstrates superior ability in selecting the most proper operating speeds based on the evaluation of the production system, thus demonstrating great potential to achieve process coordination by reducing robot partial-blocking and machine blocking. Notably, the RJSP-EM is shown to reduce energy consumption by a mean of 10% compared with the traditional model even without sacrifice in productivity (i.e., when the makespan is not allowed to increase). However, it should be pointed out that the energy-saving efficacy of the RJSP-EM declines when the allowed makespan growth becomes larger. This is because (i) the energy saving is at a cost of productivity; and (ii) the saving effect is counteracted by the additional electricity consumed by the prolonged machine idling period.

To the best of our knowledge, this is the first study that integrates the energy consumption of both machines and the mobile robot into the RJSP framework, which theoretically contributes to the JSP literature by proposing a new research direction. Besides, we innovatively propose to achieve energy saving through speed selection and coordination between the machine production process and the robot movement process by reducing/eliminating machine blocking and robot partial-blocking. It is believed that our study could greatly enhance the green level of smart manufacturing.

The rest of this paper is arranged as follows. Section 2 reviews the related literature. Section 3 describes the problem studied. Section 4 then formulates the mathematical models. Section 5 demonstrates the superior performances of the proposed models through computational experiments. Finally, Section 6 concludes the study.

2. Literature Review

Our study is related to two research streams: robotic job-shop scheduling problems and job-shop scheduling problems with energy considerations.

2.1 Robotic job-shop scheduling problems (RJSP)

The RJSP investigates the joint scheduling of machines and mobile robots. Common optimization objectives of the RJSP are set to minimize makespan or tardiness (Petrović et al., 2019). As the robotic movement process is integrated into the RJSP decision framework, various restrictions like the machine buffer capacity, pickup criteria, and movement conflict avoidance should be considered. For example, Liu et al. (2018) study a job-shop scheduling problem with four different buffer constraints (no-wait, no-buffer, limited buffer, and infinite buffer) and formulate the problem with mixed integer programming (MIP) models. The pickup criteria are determined by the equipment and job requirements. For example, Zeng and Yan (2014) study a blocking job shop with automated guided vehicles (AGVs) for delivery, where a job remains on the current machine after an operation is finished until the next machine is available. Sun et al. (2021) investigate a blocking RJSP with robot movement considerations. They model job scheduling and robot movement simultaneously with several deadlock strategies developed to avoid conflicts. Differently, Cheng et al. (2019) explore a mixed no-wait flow-shop, which allows the co-existence of no-wait machines (operations should be removed instantly after completion) and regular machines. Hurink and Knust (2000) study an RJSP following the time window rule, which requires each operation to begin within a time window. Similarly, Caumond et al. (2009) use a time window approach to schedule a flexible manufacturing system with one transportation vehicle. Differently, a few studies focus on job shops with several robots (El Khayat et al., 2006; Ham, 2020).

The incorporation of robots in RJSP problems has further increased the difficulty of the JSP problem which is already NP-hard. Besides, compared with robotic flowshop problems, studies on job-shop problems with robotic constraints are still very limited.

2.2 JSP with energy considerations

Many JSP studies have incorporated energy considerations into planning (Abedi et al., 2020). For instance, Hassani et al. (2019) model a JSP with considerations of no-buffer and energy consumption into a MILP model. Mokhtari and Hasani (2017) investigate the flexible JSP and propose a multi-objective optimization model, which minimizes the completion time, the system availability, and the energy cost. Dai et al. (2019) establish an optimization model that minimizes the energy consumption and makespan for a flexible JSP with transportation considerations.

From the literature, various energy-saving strategies can be identified. First, considering that the price of electricity varies across time, research efforts are paid to saving costs through scheduling production activities to off-peak periods (Masmoudi et al., 2019; Wang and Wang, 2019). Besides, several studies develop machine turning off/on strategies, which plan to turn off machines when they become idle for a long period to save energy (Meng et al., 2019). However, such strategies may be not that feasible due to the switching energy consumption and possible damage to devices (Zhang and Chiong, 2016). As the processing speed affects the energy consumption rate, the speed scaling mechanism is another strategy that enables energy conservation with flexible adjustment of machine operating speeds. Zhang and Chiong (2016) adopt a speed scale framework for a JSP to pursue a trade-off between energy savings and tardiness. Abedi et al. (2020) also apply a machine speed scaling mechanism to achieve energy savings where the production is affected by deteriorations.

As can be seen from the discussions above, many existing JSP studies have made an effort to save energy from the machine side (e.g., turn on/off strategies, production speed adjustment). However, in RJSP where robot movements are critical for the energy efficiency of the overall system, much less attention has been paid to the energy consumed by robots. As discussed in previous sections, in robotic cells, energy waste could be generated due to the poor coordination between the machine production process and the robot movement process (e.g., machine blocking, robot partial-blocking), which emphasizes the importance of considering the energy consumed by both machines and robots in achieving higher energy efficiency in robotic cells. To our best knowledge, there are only a few studies trying to reduce robot energy consumption in RJSP. For instance, Gürel et al. (2019) investigate the robot speed and moving sequence in a robotic cell and find that controlling robot speeds can significantly reduce energy consumption. Bukata et al. (2019) try to reduce the energy consumed by the robotic cells without deterioration in throughput by applying robot power-saving modes and adjusting robot positions. Barak et al. (2021) incorporate the energy efficiency of AGVs into flexible manufacturing systems. However, none of these studies has endeavored to achieve energy saving by coordinating the machine production process and the robot movement process through speed adjustment for both machines and robots. Our study thus fills this gap.

2.3 Research Gaps

Through the review above, several research gaps can be derived. First of all, although robotic technologies have re-shaped the manufacturing industry, the energy efficiency of robotic cells is under-explored. Second, even though several energy-saving strategies have been developed for job shops, most studies focus on reducing energy consumption from the side of machines. However, the energy consumed by the robot movement process and the energy waste caused by the poor coordination between the machine production process and the robot movement process have been rarely considered. To the best of our knowledge, no prior studies explore the coordination of these two processes to reduce energy waste. Third, little research investigates the benefits of simultaneously controlling the operating speeds of machines and the robot in enhancing the systematic sustainability performances. To bridge these critical research gaps, in this work, we propose two novel RJSP models with a speed scaling framework, in which the speeds of machines and the robot are adjustable to achieve the goal of energy saving.

3. Problem Description

The RJSP with energy considerations studied in this paper is described as follows. In the robotic cell, there are $|M|$ processing machines and one mobile robot. Each machine can perform a specific type of operation. An input depot D and an output stock S are placed at the two ends of the cell. The robotic cell aims to process sets of jobs. The job features and machine execution follow the job shop setting. Each job i consists of several ordered operations denoted by $J_i = \{O_{i1}, O_{i2}, \ldots, O_{i|Ji|}\}$. The sequence of operations for a job is named as *in-job sequence*. Each O_i is a nonstop operation that will be performed by a designated machine M_{ij} with processing time PT_{ij}. Pre-emption is not allowed. Notations used in this paper are summarized in Table 1.

Table 1. Notations

Parameters	
I	Set for jobs, $I = \{1,2, \ldots, \|I\|\}$.
$\|I\|$	Total number of jobs.
i, m, h	Indexes for jobs, where $i, m, h \in I$.

J_i	Set for operations in job *I*, $J_i = \{1,2,\dots,\lvert J_i\rvert\}$.
$J_{i'}$	Set for operations with the stock operation in job *i*, $J_{i'} = \{1,2,\dots,\lvert J_i\rvert + 1\}$.
$\lvert J_i\rvert$	Number of operations in job *i*.
$\lvert J_{i'}\rvert$	Number of operations with the stock operation in job *i*, $\lvert J_{i'}\rvert = \lvert J_i\rvert + 1$.
j, n, g	Indexes for operations, where $j \in J_{i'}, n \in J_{m'}, g \in J_{h'}$.
O_{ij}, O_{mn}, O_{hg}	Index for the *j-th, n-th,* and *g-th* operation of job *i,* job *m,* and job *h.*
M	Set for machines, $M = \{1,2,\dots,\lvert M\rvert\}$.
$\lvert M\rvert$	Number of machines.
k	Index for the *k-th* machine, $k \in M$.
M_{ij}	Index for the machine to execute O_{ij}.
PM_{ij}	Position for machine to execute O_{ij} (Positions for D and S: $PM_D = 0$, $PM_{i\lvert J_{i'}\rvert} = \lvert M\rvert + 1$).
SPT_{ij}	Processing time of O_{ij} on the machine under normal speed (stock operation: $PT_{i\lvert J_{i'}\rvert} = 0$).
tl_{ij}	Moving time for a loaded movement from M_{ij-1} to M_{ij}.
tu_{ijmn}	Moving time for an empty movement from M_{ij} to M_{mn} under normal speed.
tu_{ijD}	Moving time for an empty movement from M_{ij} to D under normal speed.
σ_{ij}	Moving distance for a loaded movement from M_{ij-1} to M_{ij}.
σ_{ijmn}	Moving distance for an empty movement from M_{ij} to M_{mn}.
σ_{ijD}	Moving distance for an empty movement from M_{ij} to *D*.
V	Set for speed scales of machines and robot, $V = \{1,2,\dots,\lvert V\rvert\}$.
$\lvert V\rvert$	Number of speed scales of machines and robot.
v_k	Normal processing speed for machine *k*.
v_{kv}	Actual processing speed for machine *k* under speed scale *v*.
v_R	Normal robot moving speed (1 unit distance/minute).
v_r	Robot moving speed under speed scale r.
μ_k	Processing power of machine *k* under normal speed (unit: w).
μ_{kv}	Processing power of machine *k* under speed scale *v*.
α_k	Operating characteristics of the machine *k*.
p (q)	Parameters denote the positive relationships between the speed and machine (robot) power.
w	Robot loaded weight.
ξ	Operating characteristics of the robot.
C_0	Minimized makespan derived by the traditional model.
EI_k	Idling energy consumption per unit time of machine *k*.
SPE_{ij}	Energy consumption to perform O_{ij} under normal speed.
ERE	Energy consumption per unit distance for robot empty movements under normal speed.
ERL	Energy consumption per unit distance for robot loaded movements.
SLE_{ij}	Energy consumption for robot loaded movement from M_{ij-1} to M_{ij}.
SEE_{ijmn}	Energy consumption for robot empty movement under normal speed from M_{ij} to M_{mn}.
β	A large positive number.
α	Makespan increase tolerance.
F	Dummy sink node that connects with the last operation in a job batch.

Decision Variables

X_{ijmn}	Binary decision variable. It equals 1 when the robot leaves for O_{mn} after O_{ij} starts, where *i* and *m* are two different jobs; 0 otherwise.
$Y_{ij(j+1)}$	Binary decision variable. It equals 1 when the robot waits for the entire processing time of the current operation O_{ij} and goes to $M_{i(j+1)}$; 0 otherwise.
Z_{ijhg}	Binary decision variable. It equals 1 when both O_{ij} and O_{hg} are executed on the same machine, and O_{ij} precedes O_{hg} (not necessarily the immediate predecessor); 0 otherwise.
VM_{ijv}	Binary decision variable. It equals 1 when machine executes O_{ij} with speed scale *v*.
VR_{ijmnr}	Binary decision variable. It equals 1 when the robot selects speed scale *r* to execute the empty movement from M_{ij} to $M_{m(n-1)}$.
SM_{ij}	Starting time of O_{ij} on the assigned machine.
SM_F	Time to reach the sink node *F*.
RM_{ij}	Removing time of O_{ij} from machine after completion.
APE_{ij}	Energy consumption of O_{ij} under actual processing speed.
APT_{ij}	Time consumption of O_{ij} under actual processing speed.
AEE_{ijmn}	Energy consumption for moving from M_{ij} to M_{mn} under actual robot speed.
TI_k	Total idling time of machine *k*.
$Cmax$	Makespan.
LE	Total energy consumption for loaded movements.
EE	Total energy consumption for empty movements.
TE	Total energy consumption for movements.
PE	Total machine processing energy consumption.
IE	Total machine idling energy consumption.
AE	Total auxiliary energy consumption.

A single-gripper robot is involved for the in-facility movements of goods. For each job, the robot should first pick the initialized job up at *D* before moving it to the first machine M_{i1}. Also, the robot should deliver the job to *S* after all operations within it are completed. As there is no precedence restriction between operations from different jobs, the robot can flexibly turn to handle another job (after an upload action) if all *in-job sequences* are not violated. Two types of robot movement exist: *loaded movement* implies moving a job to a machine for uploading and processing, while *empty movement* indicates a deadhead movement that relocates the robot for job picking-up. Due to the linear layout, the moving time between any pair of machines is symmetrically determined by the absolute distance between their locations. Similar to Sun et al. (2021), we consider the situation that machines have no buffer and the mobile robot has the capacity to hold one product each time. Therefore, machines and robot can be occupied by only one job at any time. Besides, a job will be blocked on a machine after completion until the robot comes for release. Such periods are called as *machine blocking*. The setup time of both the machines and the robots are incorporated into the processing times and movement times.

The problem aims to simultaneously determine the job schedules and the robot route. Following the above settings, two scenarios can be extracted after a loaded movement that places job i ($i \in I$) on one machine: the robot can (i) wait at the machine for the entire processing of the operation and then transport job i to its next handling machine (defined as a *robot full-blocking*); or (ii) turn to another job m ($m \in I;\ m \neq i$). In the second scenario, the robot first moves emptily to the machine currently holding job m (or to D). Then, three sub-scenarios may appear: (a) the robot arrives later than the completion time of job m's current operation (i.e., job m should experience a blocking period in this sub-scenario); (b) the robot arrives before the completion of that operation and the robot should experience a *robot partial-blocking (RPB)* before it can conduct the transport; and (c) the robot arrives exactly when the operation is finished and could pick up job m directly for the next move. Sub-scenario (c) is a synchronized process, where machine blocking and robot partial-blocking are avoided.

In the machine speed scaling framework, the processing speed of each machine can be selected from a finite and discrete set (Abedi et al., 2020; Hassani et al., 2019; Zhang and Chiong, 2016). In this paper, we propose to improve the coordination of machines and robot with a *V-scale* speed framework that enables speed adjustment for both machines and the robot. Generally, machines and robot operate at normal speeds (denoted as v_k ($k \in M$) for machine k, and v_R for the robot). For productivity, companies usually set machines to work at a high speed. Thus, in our study, we consider the normal speed as the highest speed. While with the *V-scale* speed framework, for each operation, the actual processing speed level v_{kv} ($k \in M, v \in V$) is a value selected from $|V|$ levels ($v_{kv} \leq v_k$). Also. The moving energy can be saved by adjusting the robot speed. As the robot partial-blocking only occurs after empty movements, the empty movements can be decelerated to eliminate the original robot partial-blocking periods under normal speeds. Thus, for each empty movement, the robot selects a speed v_r ($r \in V$) from the *V-scale* levels. Considering there is no robot partial-blocking after loaded movements or movements to D and S, these movements are conducted at the highest speeds to ensure productivity.

Here we specify the energy calculation method for machines and the robot. Following Zhang and Chiong (2016), the machine processing energy equals the actual processing time (APT) multiplying the power under this speed. If the normal speed is applied for O_{ij}, the APT_{ij} equals to the normal processing time SPT_{ij}. While if a lower speed v_{kv} is selected, the APT_{ij} is proportionally changed to $\frac{v_k}{v_{kv}} SPT_{ij}$. μ_k denotes the power of machine k with normal speed, and μ_{kv} represents the power of Machine k under the selected speed v_k. $\mu_k > \mu_{kv}$ because the machine power is positively related to the processing speed. Besides, similar to Zhang and Chiong (2016), we consider cases where the energy consumption decreases with the reduced speed despite the longer processing time. Therefore, we have $\mu_k \times SPT_{ij} > \mu_{kv} \times APT_{ij}$. Equations (3.1) - (3.2) derive the machine power under the normal speed and the selected speed. Parameter *p* denotes the positive relationship between the speed and the power, which should be larger than 1 to guarantee that the speed growth leads to an increasing energy consumption rate. Note that α_k represents the operation characteristics of machine *k*. Equations (3.3) calculate the actual energy consumed for processing O_{ij} and derives the relationship between the actual energy consumption APE_{ij} and the normal consumption SPE_{ij}.

$$\mu_{ks} = \alpha_k {v_k}^p \tag{3.1}$$

$$\mu_{kv} = \alpha_k {v_{kv}}^p \tag{3.2}$$

$$APE_{ij} = \mu_{kv} \times APT_{ij} = SPT_{ij} \times (\frac{v_{\mathrm{kv}}}{v_{\mathrm{k}}})^{p-1} \times \mu_{ks} = SPE_{ij} \times (\frac{v_{\mathrm{kv}}}{v_{\mathrm{k}}})^{p-1} \tag{3.3}$$

Following Gürel et al. (2019), we consider that the robot movement energy consumption depends on the robot moving speed, the traveled distance, the carrying load, and the operating characteristics of the robot. Equations (3.4) calculate the *loaded energy consumption per unit distan*ce (ERL), which is jointly determined by robot operating characteristics ξ, loaded weight *w*, and robot normal speed v_R. The exponential parameter *q* (*q*>1) forms the positive relationship between the moving speed and energy consumption, which indicates that higher speeds lead to larger energy consumption (but not a linear relationship). Equations (3.5) obtain the energy for loaded movement from $M_{i(j-1)}$ to M_{ij} (the traveling distance is σ_{ij}). Similarly, Equations (3.6) computes the unit consumption of empty movements under normal speed, based on which Equations (3.7) calculates the energy consumption of empty movements under normal speed for a distance σ_{ijmn}. Equations (3.8) derive the actual energy needed for an empty movement from M_{ij} to M_{mn} by applying the *V-scale* speed scales.

$$ERL = \xi w {v_R}^q \tag{3.4}$$
$$SLE_{ij} = \sigma_{ij} \times ERL \tag{3.5}$$
$$ERE = \xi {v_R}^q \tag{3.6}$$
$$SEE_{ijmn} = \sigma_{ijmn} \times \xi {v_R}^q \tag{3.7}$$
$$AEE_{ijmn} = \sigma_{ijmn} \times \xi {v_r}^q = \sigma_{ijmn} \times ERE \times (\frac{v_r}{v_R})^q \tag{3.8}$$

4. Model Development

As introduced, this study proposes two novel robotic job-shop scheduling models to enhance energy efficiency. In this section, we first present the novel RJSP-E in Section 4.1. Then, the RJSP-EM is constructed in Section 4.2.

4.1 RJSP-E

We first develop the model named as *robotic job-shop scheduling with energy consumption* (i.e., RJSP-E) formulated with Equation (0) to Equations (44) in Table 2. Following Sun et al. (2021), we apply the network-based modelling approach to build the new models. The optimization objective is to minimize the total energy consumed for the robotic cell. The overall energy consumption (as shown in Equation (0)) consists of four parts: the machine processing energy, the machine idling energy, the robot movement energy, and the auxiliary energy consumption. In the following, we first explain the energy consumption constraints in Section 4.1.1 to Section 4.1.4. Then, Section 4.1.5 briefly introduces the other traditional RJSP constraints.

4.1.1 Total machine processing energy consumption

The energy consumed by machine production is the product of the power of machines (in Watts, w) and the processing time (in seconds). Constraints (18-20) calculate the total machine processing energy consumption by summing the actual energy consumed by each operation. Specially, Constraints (18) ensure that each operation is assigned with one speed from the *V-scale* framework. Constraints (19) derive the actual processing energy (APE_{ij}) under the selected speed with the relationship between APE_{ij} and the normal energy consumption SPE_{ij} (refer to Equation (3.3) in Section 3). Constraints (20) then add up all APE_{ij} to obtain the total machine energy consumption PE.

4.1.2 Total robot movement energy consumption

The energy consumed by robot movement is the product of the electricity consumed for a unit distance[5] movement (in KJ) and the moving distance. Constraints (21-25) formulate the total robot movement energy consumption by summing up the robot energy consumed by loaded movements and empty movements. Constraints (21) make sure that the empty movement from O_{ij} to $O_{m(n-1)}$ will be assigned with a speed level if arc X_{ijmn} is selected. Constraints (22-24) derive the total energy consumed by empty movements (*EE*), which is further divided into empty movements to machines (Constraints (22)) and empty movements to the *D* (Constraints (23)). Constraints (22) apply the *V-scale* speed framework to empty movements and derive the energy consumption (refer to Equations (3.8) in Section 3). Constraints (23) add up all empty movements to the input depot for picking up the initialized jobs, which are conducted with the normal speed. Constraints (25) obtain the energy consumed for loaded movements (*LE*) by summing up the electricity used for every loaded movement, as calculated with Equations (3.5) in Section 3.

4.1.3 Total machine idling energy consumption

The total machine idling energy consumption is the electricity used during machine idling periods throughout the entire manufacturing process. Constraints (26) calculate the length of idling time encountered by each machine by subtracting the processing time of that machine from the lasting time *Cmax*. Constraints (27) obtain the total machine idling energy consumption by adding up the idling energy of each machine, and the latter is calculated by multiplying the individual idling power and the idling time.

[5] A unit distance is the distance moved in a minute of the robot.

4.1.4 Auxiliary energy consumption

The auxiliary energy is consumed by supporting activities in the robotic cell not directly related to production, such as for keeping temperature and humidity. Following Meng et al. (2019), we model it as proportional to the total processing time (i.e., the makespan) by an auxiliary energy consumption coefficient *s* (Constraint (28)).

Table 2. The formulation of RJSP-E.

Obj. Min $PE + IE + LE + EE + AE$		(0)
s.t.		
$C_{max} \geq SM_F,$		(1)
$SM_F \geq SM_{ij}$	$\forall i, j \in \{1,2,\dots,\lvert J_{i'}\rvert\},$	(2)
$SM_{11} = \lvert PM_{11} - PM_D\rvert / v_R,$		(3)
$\sum_{m\in I}\sum_{n\in J_{m'}} X_{ijmn} + Y_{ij(j+1)} = 1,$	$\forall i, i \neq m, j \in \{1,2,\dots,\lvert J_i\rvert\},$	(4)
$\sum_{m\in I}\sum_{n\in J_{m'}} X_{mnij} + Y_{i(j-1)j} = 1,$	$\forall i, i \neq m, j \in \{2,3,\dots,\lvert J_{i'}\rvert\},$	(5)
$\sum_{m\in I}\sum_{n\in J_{m'}} X_{i\lvert J_{i'}\rvert mn} + X_{i\lvert J_{i'}\rvert F} = 1,$	$\forall i, i \neq m,$	(6)
$\sum_{m\in I}\sum_{n\in J_{m'}} X_{mni1} = 1,$	$\forall i, i \neq m, i \neq 1,$	(7)
$SM_{i1} \geq SM_{mn} + tu_{mnD} + tl_{i1} - (1 - X_{mni1}) \times \beta,$	$\forall i, m, i \neq m, n \in \{1,2,\dots,\lvert J_{m'}\rvert\},$	(8)
$RM_{ij} \geq SM_{mn} + \sum_{r\in V} VR_{mni(j+1)r} \times tu_{mnij} \times (v_R/v_r) - \left(1 - X_{mni(J+1)}\right) \times \beta,$	$\forall i, m, i \neq m, j \in \{1,2,\dots,\lvert J_i\rvert\},, n \in \{1,2,\dots,\lvert J_{m'}\rvert\},$	(9)
$RM_{ij} \geq SM_{ij} + APT_{ij}$	$\forall i, j \in \{1,2,\dots,\lvert J_i\rvert\},$	(10)
$APT_{ij} = \sum_{v\in V} VM_{ijv} \times PT_{ij} \times (v_k/v_{kv}),$	$\forall i, j \in \{1,2,\dots,\lvert J_i\rvert\},$	(11)
$SM_{ij+1} \geq RM_{ij} + tl_{ij+1}$	$\forall i, j \in \{1,2,\dots,\lvert J_i\rvert\},$	(12)
$Z_{ijhg} + Z_{hgij} = 1,$	$\forall i, h, j \in \{1,2,\dots,\lvert J_i\rvert\}, g \in \{1,2,\dots,\lvert J_h\rvert\}, M_{ij} = M_{hg},$	(13)
$SM_{ij} \geq SM_{h(g+1)} + tu_{h(g+1)i(j-1)} + tl_{ij} - Z_{ijhg} \times \beta,$	$\forall i, h, j \in \{2,3,\dots,\lvert J_i\rvert\}, g \in \{1,2,\dots,\lvert J_h\rvert\}, M_{ij} = M_{hg,},$	(14)
$SM_{hg} \geq SM_{i(j+1)} + tu_{i(j+1)h(g-1)} + tl_{hg} - (1 - Z_{ijhg}) \times \beta,$	$\forall i, h, j \in \{1,2,\dots,\lvert J_i\rvert\}, g \in \{2,3,\dots,\lvert J_h\rvert\}, M_{ij} = M_{hg,},$	(15)
$SM_{i1} \geq SM_{h(g+1)} + tu_{h(g+1)D} + tl_{i1} - Z_{i1hg} \times \beta,$	$\forall i, h, g \in \{1,2,\dots,\lvert J_h\rvert\}, M_{i1} = M_{hg},$	(16)
$SM_{h1} \geq SM_{i(j+1)} + tu_{i(j+1)D} + tl_{h1} - (1 - Z_{ijh1}) \times \beta,$	$\forall i, h, j \in \{1,2,\dots,\lvert J_i\rvert\}, M_{ij} = M_{h1},$	(17)
Processing		
$\sum_{v\in V} VM_{ijv} = 1,$	$\forall i, j \in \{1,2,\dots,\lvert J_i\rvert\},$	(18)
$APE_{ij} = \sum_{v\in V}[VM_{ijv} \times SPE_{ij} \times (v_{kv}/v_k)^{p-1}],$	$\forall i, j \in \{1,2,\dots,\lvert J_i\rvert\},$	(19)
$PE = \sum_{i\in I}\sum_{j\in J_i} APE_{ij},$		(20)
Transportation		
$\sum_{r\in V} VR_{ijmnr} = X_{ijmn},$	$\forall i, m, m \neq i, j \in \{1,2,\dots,\lvert J_{i'}\rvert\}, n \in \{2,3,\dots,\lvert J_{m'}\rvert\},$	(21)
$AEE_{ijmn} \geq \sum_{r\in V}[VR_{ijmnr} \times \sigma_{ijm(n-1)} \times (v_r/v_R)^q \times ERE],$	$\forall i, m, m \neq i, j \in \{1,2,\dots,\lvert J_{i'}\rvert\}, n \in \{2,3,\dots,\lvert J_{m'}\rvert\},$	(22)
$AEE_{ijm1} \geq \sigma_{ijD} * ERE - (1 - X_{ijm1}) \times \beta,$	$\forall i, m, m \neq i, j \in \{1,2,\dots,\lvert J_{i'}\rvert\},$	(23)
$EE \geq \sum_{i\in I}\sum_{j\in J_{i'}}\sum_{m\in I}\sum_{n\in J_{m'}} AEE_{ijmn},$		(24)
$LE \geq \sum_{i\in I}\sum_{j\in\{2,3,\dots,\lvert J_{i'}\rvert\}}\left(\sigma_{ij} \times ERL\right) + \sum_i \sigma_{i1} \times ERL,$		(25)
Idling		
$TI_k \geq Cmax - \sum_{i\in I}\sum_{j\in J_{i'}} APT_{ij},$	$\forall k \in \{1,2,\dots,\lvert M\rvert\}, M_{ij} = k;$	(26)

$$IE = \sum_{k \in \mathrm{M}} (EI_k \times TI_k), \tag{27}$$

Auxiliary

$$AE = s \times Cmax, \tag{28}$$

$$X_{iJmn} \in (0,1), \qquad \forall i, m \in \{1,2,\dots,|I|+1\}, i \neq m, j \in \{1,2,\dots,|J_{i'}|\}, \mathrm{n} \in \{1,2,\dots,|J_{m'}|\}, \tag{29}$$

$$Y_{ij(j+1)} \in (0,1), \qquad \forall i, j \in \{1,2,\dots,|J_i|\}, \tag{30}$$

$$Z_{iJhg} \in (0,1), \qquad \forall i, h, i \neq h, j \in \{1,2,\dots,|J_i|\}, g \in \{1,2,\dots,|J_h|\}, \tag{31}$$

$$VM_{ijv} \in (0,1), \qquad \forall i, j \in \{1,2,\dots,|J_i|\}, v \in \{1,2, \dots |\mathrm{V}|\}, \tag{32}$$

$$VR_{ijmnr} \in (0,1) \qquad \forall i, m, j \in \{1,2,\dots,|J_{i'}|\}, n \in \{1,2,\dots,|J_{m'}|\}, r \in \{1,2, \dots |\mathrm{V}|\}, \tag{33}$$

$$SM_{ij} > 0, \qquad \forall i, j \in \{1,2,\dots,|J_{i'}|\}, \tag{34}$$

$$RM_{ij} > 0, \qquad \forall i, j \in \{1,2,\dots,|J_{i'}|\}, \tag{35}$$

$$APE_{ij} > 0, \qquad \forall i, j \in \{1,2,\dots,|J_i|\}, \tag{36}$$

$$APT_{ij} > 0, \qquad \forall i, j \in \{1,2,\dots,|J_i|\}, \tag{37}$$

$$AEE_{ijmn} > 0, \qquad \forall i, \mathrm{m}, j \in \{1,2,\dots,|J_{i'}|\}, n \in \{1,2,\dots,|J_{m'}|\}, \tag{38}$$

$$TI_k > 0, \qquad \forall k \in \{1,2,\dots,|M|\}, \tag{39}$$

$$tu_{ijmn} = |PM_{ij} - PM_{mn}|/v_R, \qquad \forall i, m, j \in \{2,3,\dots,|J_{i'}|\}, n \in \{1,2,\dots,|J_{m'}|\}, \tag{40}$$

$$tl_{ij} = |PM_{ij} - PM_{i(j-1)}|/v_R, \qquad \forall i, j \in \{2,3,\dots,|J_{i'}|\}, \tag{41}$$

$$tl_{i1} = |PM_{i1} - PM_D|/v_R, \qquad \forall i, \tag{42}$$

$$tu_{ijD} = |PM_{ij} - PM_D|/v_R \qquad \forall i, j \in \{1,2,\dots,|J_{i'}|\}. \tag{43}$$

4.1.5 Other constraints

Although the makespan minimization is not the optimization objective of the RJSP-E, Constraints (1-2) calculate the value of makespan to obtain the length of processing and idling time for machines. Specifically, makespan is no less than the time when the algorithm reaches the dummy sink node *F*. Constraint (3) provides the entry of the model by specifying O_{11} as the first operation to execute. Constraints (4-7) formulate the transportation network. Constraints (8) specify that the first operation of job i (i≠1) should be later than the starting time of O_{mn} plus the traveling time of (i) the empty movement from O_{mn} to *D* and (ii) the loaded movement from *D* to M_{i1}, as long as O_{mn} is linked to O_{i1} by an *X* arc. Constraints (9-10) integrate the speed scaling framework into the robot transportation and machine scheduling processes, which guarantee that the removing time of O_{ij} should satisfy two criteria: (i) if $O_{i(j+1)}$ is the next operation to be executed after O_{mn}, the removing action of O_{ij} can happen after the empty movement of the robot from M_{mn} to M_{ij} (the removing time here is denoted by T_1); (ii) the removing action of O_{ij} can take place after the operation is completed on the dedicated machine with the actual speed (the removing time here is denoted by T_2). Note that if $T_1 > T_2$, a machine blocking appears (the length of the blocking period is $T_1 - T_2$); if $T_1 < T_2$, a partial-blocking of the robot occurs (the length of the robot partial-blocking period is $T_2 - T_1$); and if $T_1 = T_2$, no blocking or robot partial-blocking happens since O_{ij} is finished exactly when the robot arrives, which is a synchronized situation. Constraints (11) obtain the actual processing time APT_{ij} of each operation under the selected speed. Constraints (12) regulate the operation execution sequence within each job. Constraints (13) make sure that there is only one execution sequence for two operations assigned to the same machine, while Constraints (14-17) forbid possible deadlock situations in the production process. Constraints (29-43) specify the value scope of variables and the calculation methods of parameters. Due to page limits, the details of these traditional RJSP constraints are placed in Online Appendix I.

4.2 RJSP-EM

In the RJSP-E developed in the previous section, the makespan of the production system may increase significantly in order to save energy, which might not be welcomed by the industry. Therefore, in this section, we further incorporate the makespan considerations into the optimization framework, and thus formulate the model named *robotic job-shop scheduling with energy consumption and makespan limitation* (RJSP-EM). Specifically, we are interested in examining the impact of company's makespan growth tolerance on energy saving. Therefore, the RJSP-EM involves an additional Constraint (44), which restricts that the increase in makespan should not exceed an upper limit. Note that C_0 is the makespan obtained by the traditional model without energy considerations, while α represents the tolerance of the decision-maker on the increase in makespan. This way, it is clear for the company to identify how much energy they could save if they encounter a certain makespan growth.

$$Cmax \leq C_0 \times (1 + \alpha) \tag{44}$$

The logic behind this constraint can be explained as follows. If there is no makespan restriction, both machine operation processing and robot empty movements will be carried out at a low speed to minimize energy consumption. However, as productivity is another important evaluator for the industry, it is crucial to ensure that the makespan will not be compromised much when we try to save energy.

In Online Appendix II, we use a numerical example to demonstrate the efficacy of the proposed models[6]. Briefly, the RJSP-E saves the most energy by selecting slow production/moving speeds by sacrificing makespan. Differently, the RJSP-EM is able to reduce energy consumption by selecting the most appropriate speeds for both machines and the robot to realize coordination, thus achieving energy saving without much compromise in productivity. Prominently, even when the makespan is not allowed to increase (α=0), the RJSP-EM can reduce the system energy consumption. Please refer to Online Appendix II for more details.

5. Computational Experiments

In this section, we conduct computational experiments to examine the performances of the proposed models. The traditional RJSP model without energy considerations (see Sun et al. (2021)), the RJSP-E, and the RJSP-EM are coded in OPL and solved in the IBM commercial solver CPLEX Studio IDE 12.10 on a desktop MacBook Pro with 1.4 GHz Intel Core i5 processor and 8 GB of RAM. The running time limit is set as 3600s. Ten job instances are tested, which are generated based on the classical work in Bilge and Ulusoy (1995). The problem scales are presented in Table 3, denoted by i×j×k (the number of jobs, the number of operations in the job, and the number of machines). Details of the tested data are provided in Online Appendix III. Table 4 and Table 5 show energy-related parameters for machines and robot movements. Specially, a three-scale speed framework is applied to machines and robot. Specifically, the normal speed v_k is defined as level 3, which is the fastest. Machines can turn to slower speeds $\frac{5}{6}v_k$ and $\frac{2}{3}v_k$. While the robot can change to $\frac{2}{3}v_R$ and $\frac{1}{3}v_R$.

Table 3. Instances

Instance Code	Problem Scale	Instance Code	Problem Scale
1	5×3×4	6	6×3×4
2	6×3×4	7	8×3×4
3	6×4×4	8	6×4×4
4	5×5×4	9	5×4×4

[6] Due to the word limit imposed by the journal, the numerical illustration example is moved to online appendix.

5	5×3×4	10	6×4×4

Table 4. Parameters for machine energy consumption.

	Processing power (w)			Idling power (w)
	Level 3 (Normal speed v_k)	Level 2 ($\frac{5}{6}v_k$)	Level 1 ($\frac{2}{3}v_k$)	
machine 1	2270	1665	1139	370
machine 2	1820	1335	914	350
machine 3	1880	1379	944	350
machine 4	2340	1717	1175	383

Table 5. Parameters for robot movement energy consumption.

	Loaded movement	Empty movement		
		Level 3 (Normal speed v_R)	Level 2 ($\frac{2}{3}v_R$)	Level 1 ($\frac{1}{3}v_R$)
Energy consumption (KJ/unit distance)	47	28	18	8

We test and compare the performance of the traditional model, the RJSP-E, and the RJSP-EM with four different tolerance increasing levels (α=0, 5%, 10%, 15%). The models are evaluated from various perspectives, including the total energy consumption, the energy consumed by machine and transport processes, the makespan, and the CPU time. Major test results are presented in Table 6, which lists the comparison of the above models from three perspectives: overall energy, makespan, and CPU time. Full results are available in Online Appendix IV. The following sections unveil the impact of incorporating energy considerations in the RJSP decision framework (the performance of RJSP-E) and the performance of RJSP-EM compared with RJSP-E and the traditional model.

Table 6. Experiment results.

Metrics	Instance	RJSP-E (KJ)	RJSP-EM (KJ)				Traditional model (KJ)
			α =0	α = 5%	α = 10%	α = 15%	
Overall Energy	1	25402	26338	25727	25540	25422	30131
	2	26823	29383	28213	27514	27160	32062
	3	28943	31673	30459	30036	29494	34882

	4	27023	28188	27305	27029	27023	31201
	5	20186	20474	20333	20284	20196	23151
	6	31885	33911	33075	32349	31919	37857
	7	Uns.	29669	28407	27596	27073	32860
	8	38689	38984	38689	38689	38689	44836
	9	31795	33924	33142	32524	32182	36935
	10	37750	41008	39953	39077	38425	44766
Makespan	1	122	103	108	111	118	103
	2	131	103	108	113	118	103
	3	134	107	112	117	123	107
	4	139	123	129	135	139	123
	5	112	88	91	94	101	88
	6	155	130	136	143	148	130
	7	Uns.	108	113	118	124	108
	8	172	166	172	172	172	166
	9	166	133	139	146	152	133
	10	197	161	169	177	185	161
CPU Time	1	5	0.55	0.74	0.91	1.36	1.36
	2	124	0.91	3.21	4.19	2.26	2.26
	3	440	0.75	4.63	5.93	5.13	5.13
	4	140	8.34	6.82	11.23	3.82	3.82
	5	51	1.27	2.43	2.04	0.88	0.88
	6	33	3.02	7.73	6.4	2.04	2.04
	7	N/A	56.22	85	300	32.75	32.75
	8	540	108	317	694	3.81	3.81
	9	20	2.12	3.86	4.08	2.63	2.63
	10	227	5.69	8.59	12.56	6.08	6.08

5.1 Performance of RJSP-E in energy saving

First, we analyze the performances of the RJSP-E and the traditional model. Compared with the traditional model, RJSP-E can achieve a remarkable 15% (on average) saving in energy. However, such an achievement is at a cost of the increase in makespan and CPU time. Since the machines and robots tend to select the moderate speed, the makespan averagely grows by 20% based on the traditional model. Even though RJSP-E is superior in saving the overall energy consumption, the CPU time required to reach optimality is 48 times longer than that of using the traditional model. Besides, Instance 7 is unsolvable for the RJSP-E within the given time limit. Therefore, the productivity of the manufacturing system is impaired due to a sacrifice in makespan, and the solution efficiency is much lower.

5.2 Performance of RJSP-EM in energy and productivity

To test the performance of RJSP-EM, four makespan increase tolerance levels, 0, 5%, 10%, and 15% are examined. Table 7 summarizes the main metrics comparison results between RJSP-EM, EJSP-E and the traditional model.

5.2.1 RJSP-EM vs. RJSP-E

The RJSP-EM alleviates the disadvantages of the RJSP-E in productivity loss with the makespan increase restriction. It is reasonable that a tighter makespan increase tolerance level (i.e., a smaller α) leads to a shorter average makespan (Column MSDE). Besides, the makespan increase constraint shows the potential to accelerate the solution process. Compared with RJSP-E (Column CRE), the RJSP-EM consumes much less CPU time. When α=0, the RJSP-EM even reduces the CPU time by an average of 96%. However, when α=15%, the figure decreases to 77%, which means the advantage in computational time is impaired along with the increase of α.

Obviously, the RJSP-EM consumes more energy than the RJSP-E due to the compressed makespan. Comparing the overall energy consumption between RJSP-EM and RJSP-E (Column TECDE), 6% more energy is witnessed in RJSP-EM when the makespan is not allowed to increase. While the saving discrepancy is narrowed along with the increase in α. When α equals 15%, the average difference in energy consumption between RJSP-EM and RJSP-E is reduced to 1%, demonstrating the energy saving efficacy of RJSP-EM approximates the RJSP-E when α increases to 15%.

Table 7. Main metrics comparison.

Tolerance α	Energy					Makespan		CPU Time	
	TECDE	TECDT	PECDT	IECDT	TECDT	MSDE	MSIT	CRE	CRT
α = 0	6%	10%	10%	13%	7%	16%	0%	96%	7%
α = 5%	3%	12%	14%	11%	8%	13%	5%	93%	-279%
α = 10%	2%	14%	16%	8%	9%	9%	9%	85%	-899%
α = 15%	1%	15%	18%	3%	10%	6%	13%	77%	-1970%

TECDE: total energy consumption discrepancy compared with RJSP-E; TECDT: total energy consumption discrepancy compared with the traditional model; PECDT: processing energy consumption discrepancy compared with the traditional model; IECDT: idle energy consumption discrepancy compared with the traditional model; TECDT: transportation energy consumption discrepancy compared with the traditional model; MSIT: makespan increase compared with the traditional model; MSDE: makespan decrease compared with the RJSP-E; CRE: CPU time reduction based on the RJSP-E.

5.2.2 RJSP-EM vs. traditional model

We further examine the performance of the RJSP-EM over the traditional model to illustrate the significance of the proposed model in facilitating processing and transport collaboration. From column MSIT, it is obvious that the makespan obtained by the RJSP-EM equals to that of the traditional model when α=0. For the RJSP-EM with the other three α, the average makespan increases are prone to reach the given upper bound (i.e., 5%, 9%, and 13% under α=5%, 10%, and 15%). This shows that the RJSP-EM is efficient in adjusting the operating speed for operations or empty movements by fully utilizing the allowed makespan relaxation.

From the perspective of energy saving, the amount of energy saved by the RJSP-EM increases along with the growth of α (Column TECDT), which is reasonable because the relaxation in makesapn leaves more space for energy-saving solutions. It is valuable to note that when α=0, the RJSP-EM is much greener than the traditional model with a significant average energy saving of 10%, demonstrating the merits of the EJSP-EM in speed coordination for saving energy. However, with the rise in α, even though more energy can be saved, the saving efficacy declines. For example, the energy is saved by 12% when α is set as 5%, while the figure only grows to 15% when α is 15%.

We thus take a closer look into the decomposed energy consumption (i.e., the machine processing energy consumption (PE), the machine idling consumption (IE), and the robot movement energy consumption (TE)). The RJSP-EM is shown to consume less PE than the traditional model in all instances by switching to lower production speeds (Column PECDT). Besides, along with the increase in the allowed production time, more energy saving from PE is witnessed, while the saving rate is slowed down. For IE (Column IECDT)., the largest saving by RJSP-EM is achieved when α=0. This saving efficacy is also weakened with the increased α. Moreover, the TE savings achieved by the RJSP-EM overall witness a slight growth along with the increase in α (Column TECDT). But it does not show a necessary growing trend in individual instances, because under different α the robot can re-design the delivery route or accelerate the movement when necessary to better coordinate with the machine production process. Thus, the increase in TE can be counteracted by the reduced PE to achieve overall energy saving.

By comparing the CPU time with the traditional model (Column CRT), we see that the RJSP-EM with the tightest makespan upper bound shows higher solution efficiency at an average of 7%. However, along with the growth in α, much longer CPU time is required for the RJSP-EM.

5.3 Sustainability analysis

The RJSP-EM and RJSP-E can facilitate the coordination between machines and robot with the *V-scale* speed framework. However, in a more general view, the coordination of machines and robot depends on many factors. Basically, it relates to the number of jobs (batch size), the number of operations in each job (processes), and the number of machines. From the machine perspective, it also depends on the processing time of operations and machine speed. While from the transport perspective, it relies on the layout of machines and the speed of the robot. Therefore, to explore the sustainability of the robotic cell, in this section, we further conduct a sustainability analysis for the above covariates.

To evaluate the sustainability of the entire system (i.e., the coordination between machines and robot in performing batches of jobs), machine blocking and robot blocking (both full-blocking and partial-blocking) can be adopted as measurements. As the experiments vary in the number of machines and the optimal makespan, we use the average machine blocking rate and robot blocking rate as metrics. First, Figure 1(a-c) show the optimal schedules of three different scenarios with a variation in the number of jobs and machines (operations) at the normal processing speed. In case 1, there are three jobs (each job has eight operations) and eight involved machines. Case 2 oppositely schedules eight jobs (each job has three operations) on three machines. In the more balanced case 3, five jobs (each job has five operations) are planned on five machines.

Table 8 summarizes the sustainability indicators. As can be seen, in case 1 when machines are in a large number while the number of jobs is small (but each job has a large number of operations), the machine blocking rate and robot blocking rate are not very high, as the robot can readily handle the products. On the other hand, when more jobs are scheduled on a few machines in case 2, the machine blocking rate is reduced while the robot's blocking time increases. This is because jobs should always wait for the availability of machines and the robot is often blocked by machines (machines are usually occupied). In case 3, the growth in the number of jobs enables parallel processing. The increase in the number of machines facilitates the reduction of robot waiting caused by the high machine occupation rate and transport restrictions. Thus, the robot is more occupied. Nevertheless, the machine blocking rate becomes larger, showing the struggle of the robot to handle larger workloads but still maintain the system efficiency. Consequently, the number of machines and robots should be matched to ensure the sustainability of the system.

Table 8. Summary of sustainability indicators.

	Avg_machine_blocking_rate	**Robot_blocking_rate**
Case 1	0.08	0.15
Case 2	0.05	0.45
Case 3	0.15	0.12

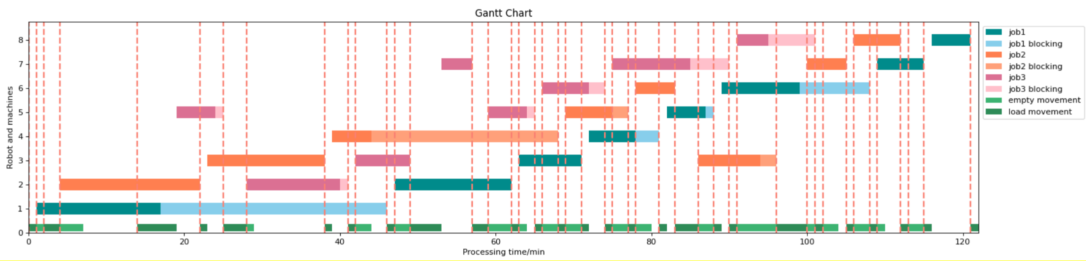

Figure 1(a). Gantt Chart for Case 1: 3 jobs, 8 operations, and 8 machines.

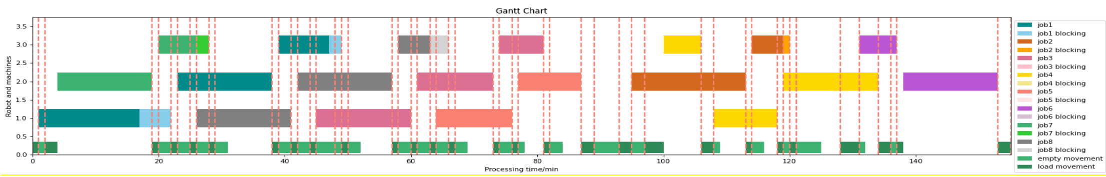

Figure 1(b). Gantt Chart for Case 2: 8 jobs, 3 operations, and 3 machines.

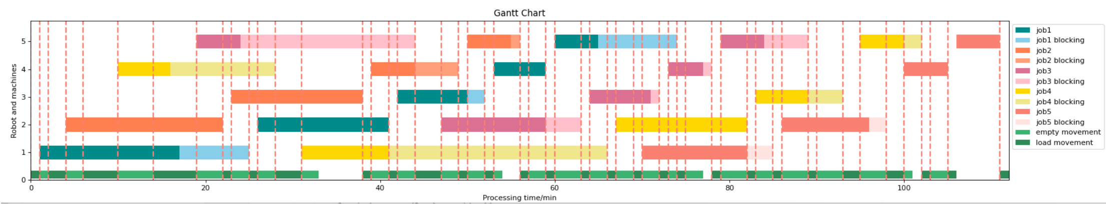

Figure 1(c). Gantt Chart for Case 3: 5 jobs, 5 operations, and 5 machines.

Figure 1 Alt Text: The schedules of operations on machines and robot movement are plotted. The X-axis is the processing time. Y-axis includes the robot (bottom green bar) and the code of machines. In the bottom green bar, dark green is the loaded robot movement, and light green is the empty robot movement. Each job is represented by two colors: the dark one denotes the processing, and the light one denotes the machine blocking.

We further explore how the speed changes of machines and robot will affect the production process. Figure 2 presents the changes in two metrics along with the four factors: (a) controls the number of machines, (b) controls the number of jobs, (c) controls the machine speed, and (d) controls the robot speed. Similar to the above analysis, when the number of jobs increases, the robot blocking time is likely to be reduced (Figure 2 (a)). When the number of machines (also operations in jobs) increases, the robot blocking time decreases, while the machine blocking time increases (Figure 2 (b)).

Figure 2 (c) reflects the changes in two metrics with the machine speed adjustment. Decreasing the machine speed has a slight impact on machine blocking time but tends to increase the robot blocking rate. This is because the robot needs to wait longer in both full-blocking and partial-blocking. While in Figure 2 (d), decreasing the robot speed will increase the robot blocking rate as the robot is less capable of processing more operations simultaneously. Instead of turning to other operations, the robot will often be blocked by the current operation.

Therefore, the speed scale also has an impact on the system sustainability. However, by using the proposed models, instead of uniformly changing the speed levels, the machines and robot speeds are changed for separate operations or transport, which can be seen as system fine-tuning for sustainability improvement under the premise of maintaining productivity.

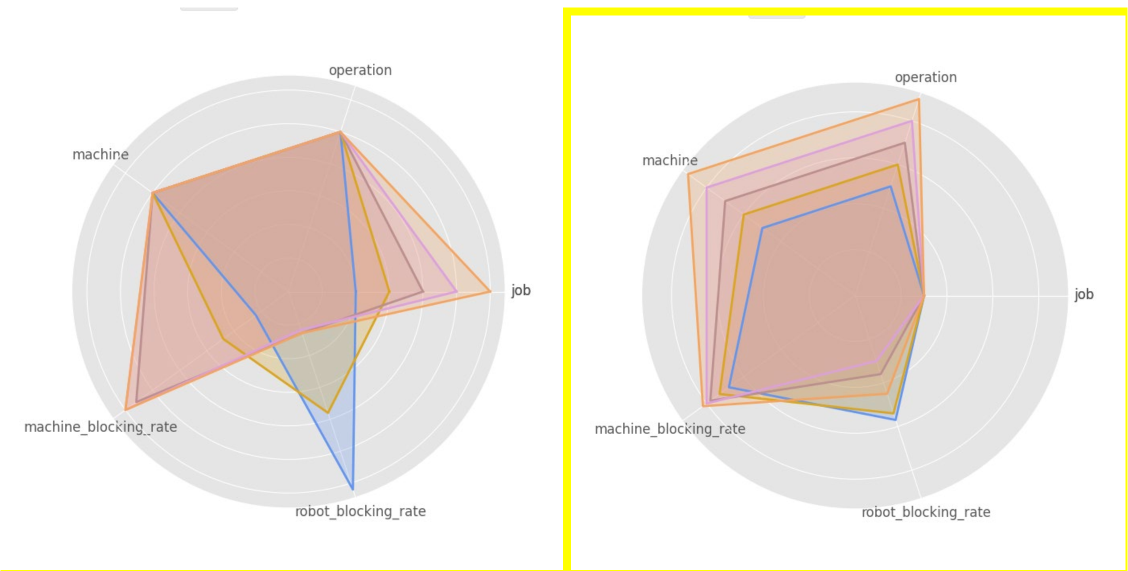

Figure 2 (a) Figure 2 (b)

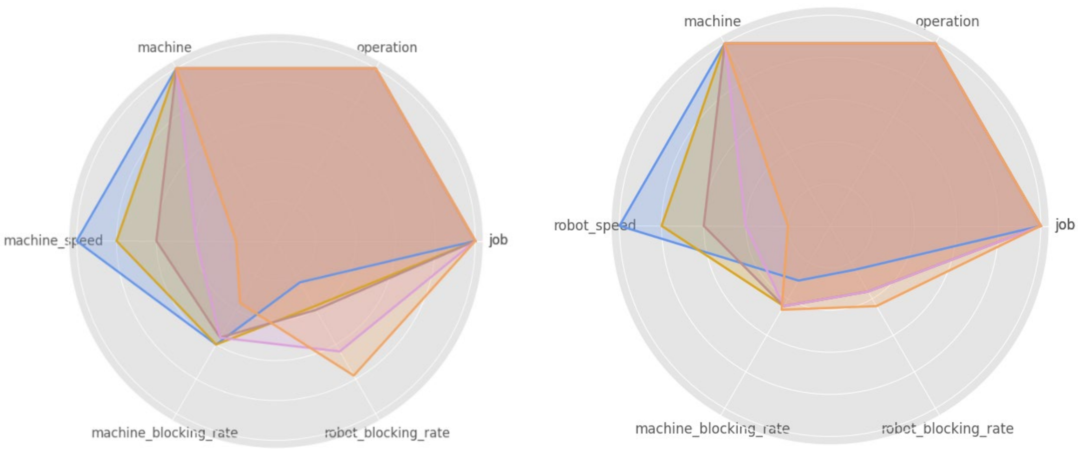

Figure 2 (c) Figure 2 (d)

Figure 2. Sustainability measurement and general factors.

Figure 2 Alt Text: Metrics (machine blocking rate and robot blocking rate) comparison by controlling (a) the number of jobs, (b) the number of machines (operations in jobs), (c) machine speed, and (d) robot speed.

6. Conclusions

Smart manufacturing has boosted the wide application of mobile robots in robotic cells. However, the mismatching between machine production and robot movement causes extensive energy waste. In this paper, we innovatively propose to achieve energy saving by enhancing the process coordination between machine production and robot movement. Two MILP models are developed with the application of a *V-scale* speed adjustment framework. The RJSP-E minimizes the overall energy consumption, while RJSP-EM simultaneously considers makespan and energy consumption. Computational experiments are conducted to verify the model performance. The RJSP-E demonstrates superior performances in reducing overall energy consumption (with an average of 15%) but at a loss of makespan (20% on average) due to the slow operating speeds. On the other hand, the RJSP-EM is able to select the most suitable operating speeds to achieve energy saving without much sacrifice in productivity. Notably, the RJSP-EM reduces energy consumption by a mean of 10% with no compromise in makespan. However, the energy saving efficacy of the RJSP-EM declines with the enlarged permitted makespan duration, as the energy saved from machine processing is counteracted by the additional prolonged idling consumption. It is believed that the novel RJSP approaches developed in this work can enhance the energy efficiency of modern robotic cells, thus promoting the healthy and sustainable development of smart manufacturing. Specifically, robotic cells with configurations similar to our problem setting can directly apply our model and fine-tune the parameters (like the distances among machines, machine production speeds, robot movement speeds) to determine their own job assignments, job sequences, machine processing speeds, and robot moving speeds. On the other hand, job shops with more robots can take our model as a benchmark to adjust their production schedules.

Future research

In this study, to examine the impact of company's tolerance against the increase in makespan on energy saving, we apply an α tolerance in makespan growth (as in Constraint (44)). One interesting future research direction is thus to apply the epsilon-constraint approach to identify the trade-off between energy and makespan. Besides, the current robotic cell only considers one robot transporting products. It will be valuable to involve more mobile robots which can better simulate the reality. Moreover, it is valuable to separately consider machine setup times and robot reconfiguration times. However, considering that the modelling of the current system is already very complicated, it will be extremely challenging to formulate the assignment and sequencing decisions, as well as to depict the deadlock and blocking situations when more mobile robots exist.

References

Abedi, M., Chiong, R., Noman, N., & Zhang, R. (2020). A multi-population, multi-objective memetic algorithm for energy-efficient job-shop scheduling with deteriorating machines. *Expert Systems with Applications*, *157*, 113348.

Barak, S., Moghdani, R., & Maghsoudlou, H. (2021). Energy-efficient multi-objective flexible manufacturing scheduling. *Journal of Cleaner Production, 283*, 124610.

Bilge, Ü., & Ulusoy, G. (1995). A time window approach to simultaneous scheduling of machines and material handling system in an FMS. *Operations Research*, *43*(6), 1058-1070.

Brossog, M., Bornschlegl, M., & Franke, J. (2015). Reducing the energy consumption of industrial robots in manufacturing systems. *The International Journal of Advanced Manufacturing Technology, 78*(5-8), 1315-1328.

Brucker, P., Burke, E. K., & Groenemeyer, S. (2012). A mixed integer programming model for the cyclic job-shop problem with transportation. *Discrete Applied Mathematics*, *160*(13-14), 1924-1935.

Brucker, P., & Kampmeyer, T. (2008). Cyclic job shop scheduling problems with blocking. *Annals of Operations Research*, *159*(1), 161-181.

Bukata, L., Šůcha, P., & Hanzálek, Z. (2019). Optimizing energy consumption of robotic cells by a Branch & Bound algorithm. *Computers & Operations Research*, *102*, 52-66.

Caumond, A., Lacomme, P., Moukrim, A., & Tchernev, N. (2009). An MILP for scheduling problems in an FMS with one vehicle. *European Journal of Operational Research*, *199*(3), 706-722.

Cheng, C. Y., Ying, K. C., Li, S. F., & Hsieh, Y. C. (2019). Minimizing makespan in mixed no-wait flowshops with sequence-dependent setup times. *Computers & Industrial Engineering*, *130*, 338-347.

Chung, S. H. (2021). Applications of smart technologies in logistics and transport: A review. *Transportation Research Part E: Logistics and Transportation Review, 153*, 102455.

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.

El Khayat, G., Langevin, A., & Riopel, D. (2006). Integrated production and material handling scheduling using mathematical programming and constraint programming. *European Journal of Operational Research*, *175*(3), 1818-1832.

Gürel, S., Gultekin, H., & Akhlaghi, V. E. (2019). Energy conscious scheduling of a material handling robot in a manufacturing cell. *Robotics and Computer-Integrated Manufacturing*, *58*, 97-108.

Ham, A. (2021). Transfer-robot task scheduling in job shop. *International Journal of Production Research, 59*(3), 813-823.

Hassani, Z. I. M., Barkany, A. E., Abbassi, I. E., Jabri, A., & Darcherif, A. M. (2019). New model of planning and scheduling for job-shop production system with energy consideration. *Management and Production Engineering Review*, *10* (1), 89-97.

He, Y., & Stecke, K. E. (2021). Simultaneous part input sequencing and robot scheduling for mass customisation. *International Journal of Production Research,* in press.

Hurink, J., & Knust, S. (2002). A tabu search algorithm for scheduling a single robot in a job-shop environment. *Discrete Applied Mathematics*, *119*(1-2), 181-203.

Jiang, E. D., & 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.

Koulamas, C., & Panwalkar, S. S. (2019). The two-stage no-wait/blocking proportionate super shop scheduling problem. *International Journal of Production Research*, *57*(10), 2956-2965.

Lamotte, R., & Geroliminis, N. (2021). Monotonicity in the trip scheduling problem. *Transportation Research Part B: Methodological, 146*, 14-25.

Liu, S. Q., Kozan, E., Masoud, M., Zhang, Y., & Chan, F. T. (2018). Job shop scheduling with a combination of four buffering constraints. *International Journal of Production Research*, *56*(9), 3274-3293.

Liu, Z., Guo, S., & Wang, L. (2019). Integrated green scheduling optimization of flexible job shop and crane transportation considering comprehensive energy consumption. *Journal of Cleaner Production, 211,* 765-786.

Masmoudi, O., Delorme, X., & Gianessi, P. (2019). Job-shop scheduling problem with energy consideration. *International Journal of Production Economics*, *216*, 12-22.

Meng, L., Zhang, C., Shao, X., & Ren, Y. (2019). MILP models for energy-aware flexible job shop scheduling problem. *Journal of Cleaner Production*, *210*, 710-723.

Mokhtari, H., & Hasani, A. (2017). An energy-efficient multi-objective optimization for flexible job-shop scheduling problem. *Computers & Chemical Engineering*, *104*, 339-352.

Parente, M., Figueira, G., Amorim, P., & Marques, A. (2020). Production scheduling in the context of Industry 4.0: Review and trends. *International Journal of Production Research, 58*(17), 5401-5431.

Petrović, M., Miljković, Z., & Jokić, A. (2019). A novel methodology for optimal single mobile robot scheduling using whale optimization algorithm. *Applied Soft Computing*, *81*, 105520.

Sun, Y., Chung, S. H., Wen, X., & Ma, H. L. (2021). Novel robotic job-shop scheduling models with deadlock and robot movement considerations. *Transportation Research Part E: Logistics and Transportation Review, 149*, 102273.

Wang, J. J., & Wang, L. (2019). Decoding methods for the flow shop scheduling with peak power consumption constraints. *International Journal of Production Research, 57*(10), 3200-3218.

Zeng, C., Tang, J., & Yan, C. (2014). Scheduling of no buffer job shop cells with blocking constraints and automated guided vehicles. *Applied Soft Computing*, *24*, 1033-1046.

Zhang, R., & Chiong, R. (2016). Solving the energy-efficient job shop scheduling problem: A multi-objective genetic algorithm with enhanced local search for minimizing the total weighted tardiness and total energy consumption. *Journal of Cleaner Production*, *112*, 3361-3375.

Zhang, M., & Yan, J. (2021). A data-driven method for optimizing the energy consumption of industrial robots. *Journal of Cleaner Production, 285*, 124862.