



HAL
open science

Exact method to optimize the total electricity cost in two-machine permutation flow shop scheduling problem under Time-of-use tariff

Minh Hung Ho, Faicel Hnaien, Frédéric Dugardin

► **To cite this version:**

Minh Hung Ho, Faicel Hnaien, Frédéric Dugardin. Exact method to optimize the total electricity cost in two-machine permutation flow shop scheduling problem under Time-of-use tariff. *Computers and Operations Research*, 2022, 144, pp.105788. 10.1016/j.cor.2022.105788 . hal-04453419

HAL Id: hal-04453419

<https://utt.hal.science/hal-04453419>

Submitted on 22 Jul 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial 4.0 International License

Exact method to optimize the total electricity cost in two-machine permutation flow shop scheduling problem under Time-of-use tariff.

Minh Hung Ho^{1,*}, Faicel Hnaien, Frederic Dugardin

*ICD-LOSI, University of Technology of Troyes
12 rue Marie Curie CS 42060 - 10004 Troyes Cedex - France*

Abstract

This study investigates the minimization of total electricity cost in a two-machine permutation flow shop scheduling problem under the most common electricity tariff, Time-of-Use (ToU). We provide a new property and solution approach to enhance existing methods in the literature. First, we provide an overview of the optimal cases for some specific ToU pricing structures that consist of only two pricing intervals. When the electricity price decreases, Johnson's rule and dynamic programming give rise to an optimal solution. On the other hand, when the electricity price increases, we provide a condition of optimality for Johnson's rule. Second, we develop a property based on Johnson's rule to determine the optimal sequence for general ToU pricing structures. Third, we propose a new mixed-integer linear programming. Then, we design an exact method based on "Logic-based Benders decomposition" to solve the problem. Finally, the numerical tests show that our proposed approach significantly improves the quality of existing results in the literature.

Keywords: Scheduling; Flow shop; Electricity cost; Time of use tariff.

1. Introduction

In 2015, 195 countries signed the first-ever universal, legally binding global climate deal at the Paris Climate Conference (COP21). The Paris Agreement aims to reduce Greenhouse Gas Emissions and keep the rise of the global average temperature below 2°C above pre-industrial levels (COP21 (2016)). This challenge is enormous and requires profound social as well as technological change. Since electricity is responsible for the prominent source of emissions, accounting for 42% of the global total in 2016, its efficiency have become the focus of today's debate (IEA (2019)). Improving electricity efficiency and reforming energy pricing play essential role in the mitigation mechanisms of CO₂ emissions (Akpan and Akpan (2012)).

Additionally, the industrial sector is responsible for 38% of carbon dioxide emissions (IEA (2019)). In recent years, there has been a growing interest in power-saving strategies in manufacturing. There are different perspectives for improving energy efficiencies such as product-level,

*Corresponding author

Email address: minh_hung.ho@utt.fr (Minh Hung Ho)

machine-level and system-level (Dai et al. (2013)). At the product-level, the researchers focus on product redesign to reduce the intrinsic energy. Meanwhile, the development or redesigning of more energy-efficient machines is an important approach at the machine-level. Finally, the system-level considers energy efficiency alongside traditional performance indicators in planning and scheduling problems. (Drake et al. (2006)) show that production operations consume only 19% of energy in the mass production environment. The energy utilization of idle machines can be considerable due to the inefficiency of scheduling. With the regulation of electricity usage via participating demand response (DR) programs, this approach may achieve non-negligible improvements without significant investment for manufacturing companies.

Moreover, with the growth of Smart Grid, Demand Side Management (DSM) is essential to improve energy efficiency and optimize the allocation of power. DSM consists of three concepts: Demand Response (DR), Energy Efficiency (EE), and Energy Conservation (Dabur et al. (2012)). From an industrial perspective, the EE and DR of DSM should work significantly to improve electricity utilization in most manufacturing companies. The EE looks for reducing energy consumption without decreasing productivity. Meanwhile, DR refers to changing customers' consumption patterns with regards to electricity price fluctuation over time (Gong et al. (2016)). "Time of use" is one of the most common pricing policies of DR (Sharma et al. (2015)). This tariff prescribes variable prices for electricity from one period to other periods. Therefore, it encourages users to manage their demand for energy consumption to minimize the electricity cost.

As a response to these circumstances, we studied a two-machine flow shop scheduling problem to minimize total electricity cost under Time-of-use (ToU) tariffs. This work aims to improve the manufacturing system in economic, ecological and social indicators. The primary contributions of this paper are threefold:

- First, we develop a property based on Johnson's rule (Johnson (1954)) to determine the optimal sequence in each pricing interval.
- Second, we propose a new mixed-integer linear model for a two-machine flow shop scheduling problem to minimize total electricity cost under ToU tariffs.
- Third, we propose an exact method to solve the problem using Logic-based Benders decomposition (Hooker and Ottosson (2003)) which exploits the property developed and dynamic programming (Wang et al. (2018b)).

The remainder of this paper is organized as follows. In Section 2, we provide a literature review regarding the optimizing of energy in the manufacturing industry. In Section 3, we describe the scheduling problem for a two-machine flow shop under energy consideration. In Section 4, we remind the reader of two results of literature that are appropriate to our study. In Section 5, we develop properties based on Johnson's rule. In Section 6, we formulate the mathematical model. In Section 7, we describe the Logic-based Benders decomposition to solve the problem. In Section 8, we provide computation experiments and results. Finally, Section 9 concludes by summarizing the contribution of this research and discusses future extensions.

2. Literature review

(Subai et al. (2006)) is one of the first studies in scheduling that considered energy consumption in manufacturing. This work studied a treatment surface line and the associated robot's moves scheduling problem. They considered the cost of energy consumption and environmental criteria with the traditional metric to characterize a nonlinear global cost function. Since (Subai et al. (2006)), researchers have developed several methods for energy optimization in the production system. Furthermore, the electricity demand management (EDM) has been developed in response to this trend. This section reviews the scientific literature relative to two principal axes of EDM for energy optimization in scheduling problems: EE and DR.

2.1. Energy efficiency

EE investigates the transparency of machine energy utilization. Following this, EE seeks frameworks to reduce energy consumption without declining the production outputs (Gong et al. (2016)). In this section, we review methodologies and techniques regarding EE in the literature.

One of the earliest techniques to reduce energy consumption in production scheduling is the “turn off/turn on” strategy of the machine. In this, non-bottleneck machines consume considerable energy as they lay idle. This method either leaves the machine idle or turns the machine off for a predetermined amount of time to reduce energy consumption. (Mouzou et al. (2007)) proposed several dispatching rules to establish an effective “turn off/turn on” plan based on the prediction of the arrival jobs to reduce energy utilization efficiency. (Dai et al. (2013)) studied this “turn off/turn on” strategy in the flexible flow shop scheduling problem. (May et al. (2015)) investigated a multi-objective energy-efficient job shop scheduling where they optimized makespan and total electricity cost. (Liu et al. (2016)) considered this method in a bi-objective optimization problem to minimize the total non-processing electricity consumption and total weighted tardiness in a job shop. (Che et al. (2017)) named the “turn off/turn on” strategy as “power-down” mechanism in their study. They simultaneously minimized total energy consumption and maximum tardiness in a single-machine scheduling problem. To enhance the quality of the Pareto front, they proposed a basic ϵ -constraint method integrated with local search and preprocessing technique. The preprocessing technique helps divide jobs into several sorted clusters based on job release time and due dates that significantly reduce the solution space.

Another technique of EE is the speed scaling framework. In some workshops, machines and appliances can process tasks at different speed levels. They consume more energy while working at a higher speed and vice versa. (Fang et al. (2013)) conducted one of the first studies on speed scaling in the manufacturing scheduling problem. They studied the problem of permutation flow shop scheduling with a restriction on peak power consumption. (Fang and Lin (2013)) studied a parallel machine scheduling where computing speeds of the machine were allowed to be adjusted. (Mansouri et al. (2016)) considered a two-machine permutation flow shop scheduling problem with sequence-dependent setup times where machines had variable speed. (Zhang and Chiong (2016)) studied the job shop scheduling problem by minimizing two objectives: total weighted tardiness and total energy consumption. They considered a machine speed scaling framework where the processing speed of each machine could be selected from a finite and discrete set. (Zhang et al.

(2019)) addressed a hybrid flow shop green scheduling problem with variable machine processing speeds.

In EE, we have other relatively less studied areas apart from the “turn off/turn on” strategy and speed scaling framework. (Li et al. (2015)) studied the trade-off between the makespan, the maximum energy consumption and the carbon footprint associated with total energy utilization in the flow shop scheduling problem. (Mokhtari and Hasani (2017)) studied a flexible job shop scheduling problem with three objective functions: minimizing total completion time, maximizing the system’s total availability, and minimizing total energy cost. They considered the total energy cost for both production and maintenance operations. (Liu et al. (2017)) considered a flow shop scheduling problem consisting of a series of processing stages and one final quality check stage. In this study, energy consumption was associated with product quality, processing speed, and equipment status. (Jiang and Wang (2019)) considered the setup and transportation time in a bi-objective energy-efficient permutation flow shop scheduling problem. (Chen et al. (2019)) studied a single machine scheduling problem with machine reliability constraints. In their study, they modeled the relationship between machine reliability and processing energy consumption. (Ji et al. (2013)) studied a uniform parallel machines scheduling problems where each machine had a different processing speed and energy charge. The objective was to determine an assignment of jobs for the machines such that the total resource consumption, including energy, could be minimized and the makespan did not exceed a certain level. They demonstrated that the problem was NP-hard. Subsequently, they developed a heuristic and particle swarm optimization to solve the problem. (Rager et al. (2015)) presented a scheduling problem in a parallel machine shop by minimizing the sum of the squared deviations of the current energy consumption with the desired level. They developed a genetic algorithm and two memetic algorithms to solve the problem.

2.2. Demand response

The growing infrastructure for advanced metering, related communication, and control technologies is motivating more and more electricity suppliers to implement variable pricing. This trend helps balance electricity supply and demand to improve the reliability and efficiency of electrical power grids (Braithwait et al. (2007)). DR for energy optimization considers the volatile electricity price. It encourages the customer to shift their electricity use from peak period to off-peak period to benefit from a lower price. In this program, we can enumerate three basic electricity pricing rates: ToU, Critical Peak Pricing (CPP) and Real-Time Pricing (RTP) (Albadi and El-Saadany (2008)).

In ToU pricing, the rates of electricity price per unit consumption differ in different blocks of time. Usually, ToU has three types of time blocks: the peak, the semi-peak and the off-peak. The electricity price during the on-peak period can be many times higher than during off-peak periods. The difference offers opportunities to users to minimize the electricity cost by assigning jobs to available cheaper periods. (Fang et al. (2016), Aghelinejad et al. (2018a), Aghelinejad et al. (2018b), Chen et al. (2018)) and (Chen and Zhang (2019)) studied single machine scheduling problems under variable ToU tariffs. Research on scheduling jobs in the shop environment under the ToU electricity tariff seems to be relatively sparse. (Ding et al. (2015)) considered an unrelated parallel machine scheduling problem. (Wang et al. (2018b), Pilerood et al. (2018)) minimized total electricity cost on a two-machine permutation flow shop scheduling problem under

ToU electricity tariffs. (Zheng et al. (2019)) studied a two-stage blocking permutation flow shop scheduling problem under ToU tariffs and variable speed machines.

CPP is similar to ToU pricing, except there is a period called the critical peak. CPP imposes a much higher rate during the critical peak period on an event day and offers discounted prices during others. The maximum number of event days per year is predetermined, but the specific dates when the events will occur are not. Consequently, it gives consumers opportunities to reduce their total electric bill by shifting electric use from CPP events to other periods (Wang and Li (2016)). (Modos et al. (2017)) studied a single machine scheduling problem with the minimization of total tardiness. In their study, the total energy consumption was limited during the specified time intervals. There is a substantial penalty fee if the manufacturer can not comply with the total energy use limits. They considered that uncertainties occurring during the execution of the schedule might lead to unexpected delays and violations of the energy consumption limits.

RTP is the most complex tariff under DR. RTP reflects the price and availability of electricity in real-time and helps shape end-use load. It requires advanced technology to communicate the price change between the utilities and the buyers. Recently, the rise of the smart grid and smart meters have promoted RTP (Gelazanskas and Gamage (2014)). (Shrouf et al. (2014)) considered a single machine scheduling problem with minimization of energy consumption costs under variable energy prices, over one day. (Gong et al. (2017)) investigated a single machine production scheduling that considered energy and labor costs under RTP. Usually, the labor wage is higher during periods with lower electricity prices. (Gong et al. (2019)) studied a many objective integrated energy and labor aware flexible job shop scheduling problem which minimized five objectives: makespan, total energy cost, total labor cost, maximal workload, and total workload.

With growing attention towards optimizing energy in the manufacturing industry, there are still many studies that we could not mention in this section. We provide three summary tables that compare relevant aspects of works on optimizing energy consumption in scheduling problems in Table (1), Table (2) and Table (3). The tables include studies that we have mentioned and have not mentioned in the previous paragraphs. The list is not exhaustive, but we hope to provide the readers an overview of this problem - Table (1) summarizes relevant aspects of works on a single machine; Table (2) summarizes relevant aspects of works on flow shop; Table (3) summarizes relevant aspects of works on job shop and parallel machines. Some notations for objective function are as follows:

$\sum C_i$: Total completion time; $\sum E_i$: Total energy consumption; TEC : Total electricity costs; TC : Total costs; $\sum w_i C_i$: Total weighted completion time. $\sum T_i$: Total tardiness; LC : Labor cost. C_{max} : The maximum completion time; $\sum w_i T_i$: Total weighted tardiness; TAS : Total availability of system; CE : Carbon emission; $\sum NPE_i$: Total non-processing energy.

3. Problem description

In this work, we study a two-machine permutation flow shop scheduling problem with electricity costs. (Wang et al. (2018b)) studied this problem and noted it as a TMPFSEC problem for short. The electricity price in this work is the ToU tariff scheme. By using Graham's notation (Graham et al. (1979)), we can refer the problem as:

$$F2|b_i, d_i, ToU|TEC$$

Table 1: Relevant aspects of works on optimizing energy consumption in scheduling problem on a single machine and on other shop environment

Reference	Objective	Energy aspect	Shop feature	Solution approach
Mouzon et al. (2007)	Min $\sum C_i$ Min $\sum E_i$	“Turn off/Turn on”	Single machine	Proposed dispatching rules based on the prediction of the arrival jobs. Mixed Integer Linear mathematical model with weighted sum of objectives
Fang et al. (2016)	Min TEC	Time-of-use tariffs	Single machine	Studied two cases: uniform and scalable machine’s speed. In both cases, non-preemptive version is NP-hard. In both cases, preemptive version is polynomial. For uniform speed problem, if all jobs have the same workload and the electricity prices are pyramidal structure, then the problem is polynomial.
Aghelinejad et al. (2018b)	Min TEC	“Turn off/Turn on” Speed scaling Time-of-use tariffs	Single machine Fixed sequence	The uniform speed problem is polynomial. The speed scalable problem is pseudo-polynomial.
Aghelinejad et al. (2018a)	Min TEC	“Turn off/Turn on” Time-of-use tariffs	Single machine	When job’s sequence is fixed: A mixed integer linear mathematical model is proposed. When job’s sequence is not predetermined: Mixed integer mathematical model is proposed. A heuristic and a genetic algorithm are proposed.
Chen et al. (2018)	Min $\sum w_i C_i + TEC$	Time-of-use tariffs	Single machine Preemptive schedule Unrelated machines	Unweighted version is polynomial. Weighted version is NP-hard. A polynomial-time approximation scheme for the weighted version. Unrelated machines scheduling version is polynomial.
Chen and Zhang (2019)	Min TEC	Time-of-use tariffs	Single machine	The feasible schedule needs to satisfy some scheduling criteria: Deadline, Bounded lateness, Bounded flow-time. Established the computational tractability of the problem: For general ToU tariffs structure, the problem is NP-hard. Identified some special ToU structures for which efficient algorithms exist
Modos et al. (2017)	Min $\sum T_i$	Energy consumption limits	Single machine Production uncertainties	Branch and Bound and Logic-based Benders decomposition. Tabu search heuristic to design robust production schedules.
Shrouf et al. (2014)	Min TEC	Real time pricing “Turn off/Turn on”	Single machine	Assumed that the jobs’ sequence is predetermined. Aimed to determine the launch times for job processing, machine’s state, etc. Mixed integer linear mathematical model. Genetic algorithm
Gong et al. (2017)	Min $TEC + LC$	Real time pricing	Single machine	Labor wage is higher during periods with lower electricity price. Case study for a blow molding process in a Belgian plastic bottle manufacturer. Mixed integer linear mathematical model Genetic algorithm
Liang et al. (2019)	Min TC : Inventory costs. Change-over costs. Energy costs.		Single machine Sequence dependent setup	Capacitated production planning and scheduling problem. Multi-products, each type of product need a different processing technique. Processing technique indicates different set of processing parameters. Each processing technique has a different energy consumption rate. A mixed integer linear programming. Proposed a fix and optimize heuristic.
Chen et al. (2019)	Min $\sum T_i + TEC$		Single machine	Modeled the relationship between reliability and processing energy consumption. New mathematical model incorporating machine’s reliability. Ant colony optimization algorithm embedded with modified Emmons rules.
Che et al. (2017)	Min $\sum T_i$ Min $\sum E_i$	“Turn of/Turn on”	Single machine	Mixed interger linear mathematical model ϵ -constraint method with a preprocessing technique that sorts jobs into different clusters and defines the precedence relationship between clusters.
Li et al. (2015)	Min CE	Speed scaling	Computer numerical control machining system	A quantitative model to evaluate cardon emission of CNC system.

Table 2: Relevant aspects of works on optimizing energy consumption in scheduling problem on flow shop.

Reference	Objective	Energy aspect	Shop feature	Solution approach
Dai et al. (2013)	Min C_{max} Min $\sum E_i$	“Turn off/Turn on” Speed scaling	Flexible flow shop	A simulated annealing algorithm in conjunction with genetic algorithm.
Fang et al. (2013)	Min C_{max}	Speed scaling Power consumption constraints	Permutation flow shop	Considered cases of discrete speeds and continuous speeds. In the cases of continuous machine’s speeds: The power consumption is an exponential function of speeds. Proposed two mixed integer linear programs: the “disjunctive formulation” and the “assignment and positional formulation”
Liu et al. (2017)	Min $\sum E_i$	Speed scaling “Turn off/Turn on”	Flow shop Reprocessing unqualified product.	Total energy consumed, $\sum E_i$, includes : Processing energy (PE), Reprocessing energy (RPE), Non-processing energy (NPE). At each processing stage, there is a set of machine’s processing speeds. A higher speed leads to shorter processing time but a higher risk of product quality degradation. Unqualified product need to be reprocessed. A three-stage decomposition approach : 1 st Determine the processing speed of each job on each machine 2 nd Determine jobs starting time to reduce NPE. 3 rd Determine machine state to futher reduce NPE.
Jiang and Wang (2018)	Min C_{max} Min $\sum E_i$ Min $\sum E_i$	Speed scaling “Turn off/Turn on”	Flow shop Sequence-dependent setup time. Controllable transportation time.	The transportation time is controllable by the speed of the transmission belt. Min $\sum E_i$ comprising: the consumption in setup, in transportation, in idle and the consumption in processing. Mixed integer linear mathematical model. An improved multi-objective evolutionary algorithm Decompose the problem into several sub-problems Dynamic strategy to adjust the mating relationship between solutions and sub-problems.
Mansouri et al. (2016)	Min C_{max} Min $\sum E_i$	Speed scaling	Two-machine permutation flow shop. Sequence dependent setups time.	Mixed integer linear mathematical model. Proposed constructive heuristic.
Pilerood et al. (2018)	Min TEC	Time-of-use tariffs	Flow shop Two machines	Continuous-time mixed integer linear mathematical model. Two-stage greedy heuristic.
Wang et al. (2018b)	Min TEC	Time-of-use tariffs	Flow shop Two machines	Mixed integer linear mathematical model Dynamic programming for fixed job’s sequence problem. Two heuristics based on Johnson’s rule and dynamic programming. Iterated local search method incorporating problem tailored procedures.
Zheng et al. (2019)	Min TEC Min C_{max}	Time-of-use tariffs Speed scaling	Two-stage permutation flow shop. Parallel batch processing machines.	Mixed integer linear mathematical model. Multi-objective hybrid ant colony optimization algorithm Max–min pheromone restriction rules and local search rule are proposed.
Zhang et al. (2019)	Min C_{max} Min $\sum E_i$	Speed scaling	Hybrid flow shop	The total energy consumption includes energy consumption when the machine is in the processing state, in the setup state and in the stand by state. A multi-objective discrete artificial bee colony algorithm.
Current paper	Min TEC	Time-of-use tariffs	Two-machine permutation flow shop	New mixed integer linear mathematical model. Developed properties based on Johnson’s rule. Logic-Based Benders Decomposition.
Luo et al. (2013)	Min TEC Min C_{max}	Time-of-use tariffs	Hybrid flow shop	Multi-objective ant colony optimization meta-heuristic. Right-shift procedure which delays operations without affecting C_{max} to reduce TEC .
Ding et al. (2021)	Min TEC Min $\sum T_i$	Time-of-use tariffs Speed scaling	Flexible flow shop	Multi-objective hybrid particle swarm optimization. Multi-objective tabu search procedure.
Badri et al. (2021)	Min TEC	Time-of-use tariffs	Flow shop	Mixed integer linear mathematical model. Transform bi-objective model into two single-objective problems by using Bellman and Zadeh fuzzy decision making principle and Zimmermann fuzzy programming method.
Cui and Lu (2021)	Min TEC Min $\sum T_i$	Time-of-use tariffs “Turn off/Turn on”	Flow shop Maintenance planning	Two-layer math-heuristic approach: 1. Outer layer optimizes the jobs’ sequence by using genetic algorithm. 2. Inner layer optimizes the maintenances’ planning by using dynamic programming.
Schulz et al. (2020)	Min TEC Min $\sum T_i$	Time-of-use tariffs Speed scaling	Hybrid flow shop	Two multi-objective mixed integer programming formulations. Eps-constraint method.
Wang et al. (2020b)	Min TEC Min C_{max}	Time-of-use tariffs	Two-stage hybrid flow shop	Bi-objective mixed integer programming formulations. Eps-constraint method. Tabu search and ant colony algorithm.

Table 3: Relevant aspects of works on optimizing energy consumption in scheduling problem on job shop and parallel machines.

Reference	Objective	Energy aspect	Shop feature	Solution approach
Liu et al. (2016)	Min $\sum w_i T_i$ Min $\sum NPE_i$	“Turn off/Turn on”	Job shop	As “Turn off/Turn on” strategy needs comparatively long idle period, they studied a scheduling technique to integrate fragmented short idle periods into large ones. Multi objective genetic algorithm based on NSGA-II implemented two new steps: 1) “1 to n scheduling building” aims to exploit the “Turn off/Turn on” strategy. 2) “Family creation and individual rejection” enhances the diversity of solution pool.
May et al. (2015)	Min C_{max} Min $\sum E_i$	“Turn off/Turn on”	Job shop	A green genetic algorithm combined by NSGA-II and SPEA-II Studied the performance of different machine behaviour policies: 1) All the machines are switched on when the first operation is started. 2) The machines can be individually switched on. 3) The machines can be switched on and off when they are idle . 4) The machines have the standby state which consumes energy.
Zhang and Chiong (2016)	Min $\sum w_i T_i$ Min $\sum E_i$	Speed scaling	Job shop	Mixed integer linear mathematical model Multi-objective genetic algorithm enhanced with local searches: 1) Min $\sum w_i T_i$ under fixed machine speed. 2) Min $\sum E_i$ under a fixed schedule.
Wu et al. (2019)	Min $\sum E_i$ Min C_{max}		Flexible job shop Deterioration effect	The processing time of jobs is not deterministic due to deterioration effect. The processing time is estimated based on a step-deterioration effect model. Proposed an energy consumption model for different machine’s states. Multi-objective hybrid pigeon inspired optimization Simulated annealing algorithm.
Masmoudi et al. (2019)	Min TEC	Power peak limit Time-of-use tariffs	Job shop	Two integer programming formulations: a disjunctive model and a time-indexed model.
Wang et al. (2020a)	Min TEC Min C_{max}	Time-of-use tariffs Machine selection	Flexible job shop	Mixed integer programming model. Hybrid multi-objective evolutionary algorithm based on decomposition.
Ding et al. (2015)	Min TEC	Time-of-use tariffs	Unrelated parallel machine.	Time-interval-based mixed integer linear mathematical model. Dantzig-Wolfe decomposition and column generation heuristic.
Jia et al. (2019)	Min $\sum E_i$ Min C_{max}		Parallel machines. Batch processing machines.	The energy consumption of machine is equal to its processing power multiplied by the total processing time of the batches on that machine. Bi-objective ant colony optimization algorithm (BOACO). Proposed an effective method to construct the feasible solutions. Proposed a neighborhood-based local optimization to enhance BOACO.
Leung et al. (2012)	Min TEC Min $\sum C_i$ Min C_{max}		Unrelated parallel machine. Preemptive schedule. Nonpreemptive schedule.	The machine cost is associated with the processing of a job on a given machine. The proposition of five specialized machine cost functions. The proposition of six objectives: - Two objectives based on the hierarchical structuring of $\sum C_i$ and TEC . - Two objectives are based on the hierarchical structuring of C_{max} and TEC . - Two objectives are based on the aggregation of $\sum C_i$, TEC , and C_{max} Determine the complexity status of the problems proposed.
Zhou et al. (2018)	Min TEC Min C_{max}	Time-of-use tariffs	Parallel machines Batch processing machines.	Mixed integer linear mathematical model. Multi-objective discrete differential evolution algorithm.
Ji et al. (2013)	Min $\sum E_i$	Machine selection	Uniform parallel machines. Bounded makespan	They demonstrated that the problem is NP-hard. Particle swarm optimization algorithm.
Zhang et al. (2021)	Min TEC	Time-of-use tariffs Speed scaling	Two-stage parallel machines. Bounded makespan	Continuous-time mixed-integer linear programming model. Tabu search-greedy insertion hybrid algorithm.
Kong et al. (2021)	Min TEC	Time-of-use tariffs Speed scaling “Turn off/Turn on”	Parallel machines Rescheduling. Deteriorating job.	Variable neighborhood search algorithm with three novel swapping neighborhood structures.
Wang et al. (2018a)	Min TEC Min C_{max}	Time-of-use tariffs	Parallel machines	Eps-constraint method. Constructive heuristic method with a local search strategy. NSGA-II algorithm.

Where, b_i is the unit power required of machine i when it is busy (processing some job), d_i stands for unit power required when machine i is on idle.

An instance of $F2|b_i, d_i, ToU|TEC$ consists of a set $J = 1, 2, \dots, n$ of jobs and a set $M = 1, 2$ of machines that are available to be processed at time zero. Each job $j \in J$ has an amount of processing time p_{ij} on machine $i \in M$. They must be processed non-preemptively first on machine 1, then on machine 2. Each machine i will start “working” at time instant 0. That means the machine either busy or idle since time zero and can be shut down only if all the jobs have been completed. At any time, each machine can process only one job at most, and each job can only be processed on one machine at most. Additionally, the machine i will consume b_i as the amount of energy per time unit when it processes job and d_i as the amount of energy per time unit when it is idle.

We consider the fluctuation in energy price over periods according to the ToU tariff scheme. Time horizon consists of a set $t = 1, 2, \dots, T$ of time periods. Each period has an associated electricity price $c_t > 0$ per unit of energy. Generally, we have three levels of price in a day: off-peak, on-peak and semi-peak. In this model, we divide the time horizon in set of $l \geq 2$ pricing intervals: $G = \{G_1 = [ST_1, ST_2), \dots, G_l = [ST_l, ST_{l+1})\}$.

$L_g = ST_{g+1} - ST_g > 0$ is the length of pricing period G_g for $g = 1, \dots, l$, where ST_g is starting time of pricing interval G_g and $ST_{l+1} = T$. All time periods t in the same pricing interval g have the same energy price $c_t = fc_g, \forall ST_g \leq t \leq ST_{g+1}$.

An example of ToU electric price of Pacific Gas & Electric Company’s tariff during summer months is illustrated in Figure (1) (Wang and Li (2016)).

The definition of the problem’s set, variables are given as follows:

Parameters:

- M : Set of machines.
 - the machine index i .
- J : Set of job.
 - the job index j, k .
- T : Planning horizon.
 - the time index t .
- G : Set of pricing interval
 - the pricing interval index g .
- b_i : The amount of energy consumed per unit time when machine i processes jobs.
- d_i : The amount of energy consumed per unit time when machine i is idle.
- c_t : The electricity price associated to period t .
- fc_g : The electricity price associate to pricing interval g .

- p_{ij} : Processing time of job j on machine i .
- ST_g : Starting time of pricing interval g .
- $ED_g = ST_g + L_g$: Ending time of pricing interval g .
- V : Large positive number.

Variables:

- S_{ij} : The starting time of job j on machine i .
- C_{ij} : The completion time of job j on machine i .
- F_i : The shutdown time of machine i .
- $x_{it} = 1$ if machine i is working during time period t (either busy or idle), and $= 0$ otherwise (i.e. shutdown).
- $y_{it} = 1$ if machine i is on idle during time period t , and $= 0$ otherwise.
- $z_{ijt} = 1$ if machine i processes job j during time period t , and $= 0$ otherwise.
- α_{ijg} : The amount of processing time of job j on machine i during pricing interval g .
- $\beta_{ijg} = 1$ if job j is processed on machine i during pricing interval g , and $= 0$ otherwise.
- $\delta_{jk} = 1$ if job j precedes job k , and $= 0$ otherwise.
- $\gamma_{jg} = 1$ if job j is started on machine 2 during pricing interval g , and $= 0$ otherwise.
- $\omega_{jk}^g = 1$ if job j and job k are started during the same pricing interval g , and $= 0$ otherwise.
- TEC : Total electricity cost.

4. Johnson's rule and dynamic programming

Before tackling the problem, we briefly remind the readers of two of the results in the literature that are appropriate to our solution approach. Firstly, the Johnson's rule (Johnson (1954)) can find the optimal solution in $O(n \log n)$ steps for $F2|perm|C_{max}$ problem. Secondly, the dynamic programming (DP) proposed by (Wang et al. (2018b)) optimally solves $F2|b_i, d_i, ToU|TEC$ when jobs sequence is predetermined.

Electricity charge (\$/kWh)

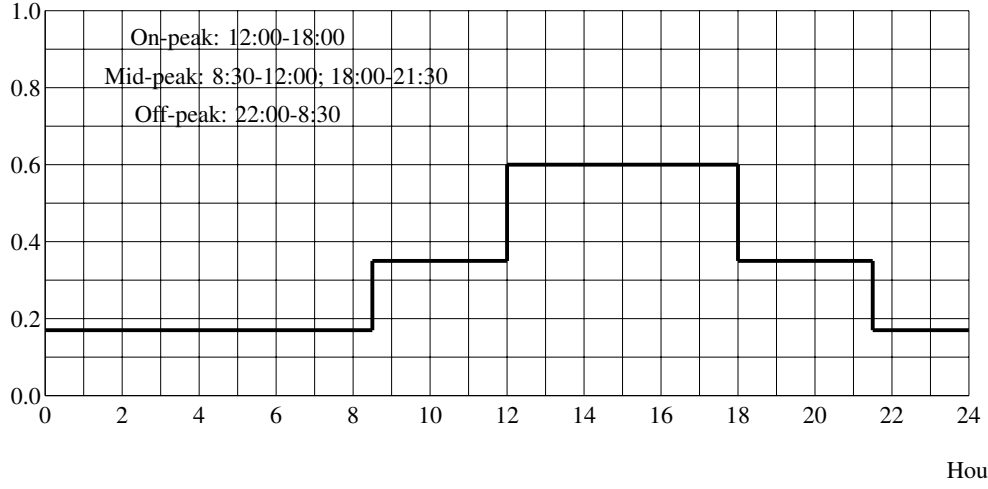


Figure 1: ToU charge profiles of PGE's tariff during summer months

4.1. Johnson's rule

For any instance of $F2|perm|C_{max}$, base on job's processing time, we distinguish two sets $\bar{J} = \{J_i \in J | p_{1i} < p_{2i}\}$ and $\underline{J} = \{J_i \in J | p_{2i} < p_{1i}\}$. Any job whose $p_{1i} = p_{2i}$ can be placed arbitrary either in \bar{J} or \underline{J} . To obtain a Johnson's sequence, we use the following algorithm:

1. Sort \bar{J} in non-decreasing order of processing time on machine 1.
2. Sort \underline{J} in non-increasing order of processing time on machine 2.
3. The optimal sequence is obtained by concatenating two sets \bar{J} and \underline{J} after sorting.

4.2. Dynamic programming (Wang et al. (2018b))

We shortly remind the dynamic programming proposed by (Wang et al. (2018b)). For any given job sequence π , DP searches for the optimal starting time of each job for $F2|b_i, d_i, ToU|TEC$. Let $\underline{C}_{i,k}$ and $\bar{C}_{i,k}$ be respectively the lower and upper bounds for the completion time of the k th job in π on machine i . Because the time horizon T and the sequence π are given, we can calculate $\underline{C}_{i,k}$ and $\bar{C}_{i,k}$ for all jobs $k \in J$. A possible completion time of job k must be satisfied: 1) $\underline{C}_{1,k} \leq C_{1,k} \leq \bar{C}_{1,k}$ and 2) $\max\{C_{1,k} + p_{2,k}, \underline{C}_{2,k}\} \leq C_{2,k} \leq \bar{C}_{2,k}$. DP enumerates the set of possible completion time for jobs in two machines in $O(T^2)$. Then, DP determines the optimal completion time of job from the set of possible completion time, which has the minimal total electricity cost. DP recurrently processes from the first position to the last position of sequence π in $O(nT^4)$ times to get the minimum total electricity cost.

5. Property based on Johnson's rule

In this section, we develop some properties based on Johnson's rule to enhance the solution approach. First of all, we present some instances of configurations for two pricing intervals problem. We show that in these configurations, Johnson's rule (Johnson (1954)) with DP gives the optimal solution. Then, we develop a property based on Johnson's rule for general instances.

5.1. Two pricing intervals

We assume that the planning horizon consists of two pricing intervals g and $g + 1$ with the associated electricity prices being fc_g and fc_{g+1} respectively; g precedes $g + 1$.

Let π be Johnson's sequence, δ_1 be the **shortest positive** idle time on machine 2 between any job j and its successor obtained when all jobs of π start as soon as possible. For each instance, we can easily calculate $A_j = \frac{p_{1j} - p_{2j}}{\min(\delta_1, p_{2j})}, \forall j \in \underline{J}$. Note that b_2 and d_2 are the amount of energy consumed when machine 2 processes jobs and idle, respectively.

Property 1. *When $F2|b_i, d_i, ToU|TEC$ problem has only two pricing intervals, the application of DP on Johnson's sequence gives the optimal solution in the following cases:*

1. $fc_g \geq fc_{g+1}$.
2. $fc_g < fc_{g+1}$ and $\frac{b_2 \times (fc_{g+1} - fc_g)}{d_2 \times fc_g} \leq A_j, \forall j \in \underline{J}$.

Proof is to be found in Appendix A.

5.2. Several pricing intervals

For more general instances, where property (1) is not valid, we present a property taken into account in our approach. In the ToU electricity tariff, we assume that each pricing interval is relatively long enough to execute not only one job but a set of jobs. For each pricing interval g with its starting time ST_g and ending time $ED_g = ST_g + L_g$, we can determine a set of jobs whose starting times on machine 2 are executed during this pricing interval. We prove that by ordering these jobs following Johnson's rule, we can get a solution as best as any other sequence in electricity cost.

Property 2. *Let $J_g \subset J$ be set of jobs whose starting times on machine 2 are executed during pricing interval g ($\forall j \in J_g, ST_g \leq S_{2j} \leq ED_g$). For any order of jobs belonging to J_g , reorder these jobs following Johnson's rule does not deteriorate the solution of $F2|b_i, d_i, ToU|TEC$.*

Proof is to be found in Appendix B.

Property (2) is illustrated by an example in Figure (2). In the example, we have three pricing interval $(g - 1)$, g and $(g + 1)$ and a set of job $J_g = \{J_1, J_2, J_3, J_4\}$ whose starting times on machine 2 are assigned to pricing interval g . The initial partial sequence $\pi = J_1 \prec J_2 \prec J_4 \prec J_3$. Reorder π according to Johnson's rule, we have partial sequence $\pi' = J_3 \prec J_4 \prec J_1 \prec J_2$. As stated by the property (2), we get $TEC_{\pi'} \leq TEC_{\pi}$.

Thus, property (2) is a rule for scheduling jobs in each pricing interval. This means the objective is to define sets of jobs to execute during each pricing interval and then schedule jobs following the order given by property (2) (Johnson's order in each mentioned set). In the following sections, we present a new mathematical model and solution approach enhanced by property (2).

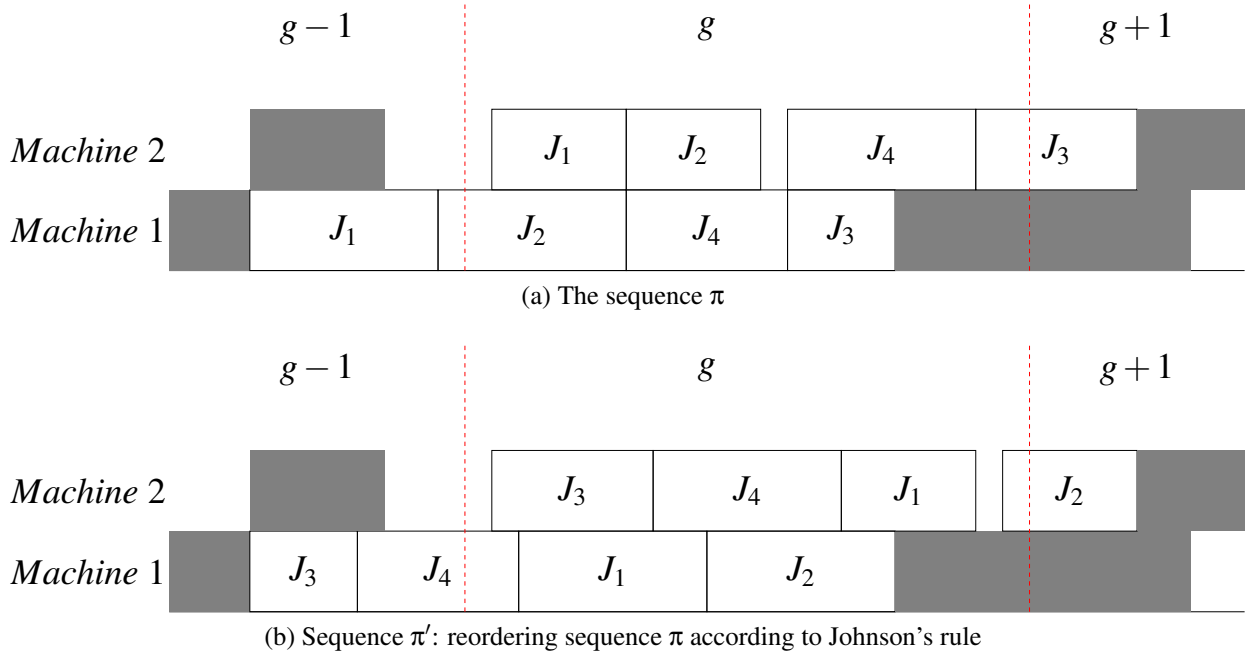


Figure 2: Sequence π and π' : reordering π according to Johnson's rule

6. New mathematical model

6.1. The mathematical model of (Wang et al. (2018b))

(Aghelinejad et al. (2019)) demonstrated that the single machine scheduling problem under the ToU tariff scheme with several machine states is NP-hard. As a single machine scheduling problem is a particular case of the flow shop scheduling problem, we can see that our problem is also NP-hard.

We remind in this section the objective function and some constraints of (Wang et al. (2018b)) that are necessary to formulate a new mathematical model. The objective function given by (Wang et al. (2018b)) and the constraints that determine the state of machine are given in equations (1) and (2).

$$\min TEC = \sum_{i \in M} \sum_{j \in J} \sum_{t \in T} c_t \times b_i \times z_{ijt} + \sum_{i \in M} \sum_{t \in T} c_t \times d_i \times y_{it}. \quad (1)$$

Subject to:

$$\sum_{j \in J} z_{ijt} + y_{it} = x_{it}, \forall i \in M, t \in T. \quad (2)$$

The objective (1) aims to minimize the total electricity costs. The total electricity cost includes two type of energy consumption cost:

- When machines are busy: $\sum_{i \in M} \sum_{j \in J} \sum_{t \in T} c_t \times b_i \times z_{ijt}$
- When machines are idle: $\sum_{i \in M} \sum_{t \in T} c_t \times d_i \times y_{it}$.

Constraints (2) define the state of the machine i after turning on, either busy or idle .

6.2. New optimization objective's formulation

To enhance the resolution approach, we propose a new objective function derived from the function (1). Firstly, we propose new variables α_{ijg} which are the amount of processing time of job j on machine i during pricing interval g (Figure.3). Parameters fc_g are the electricity price of pricing interval g .

$$c_t = fc_g, \forall ST_g \leq t \leq ED_g$$

$$\alpha_{ijg} = \sum_{ST_g \leq t \leq ED_g} z_{ijt}, \forall i \in M, j \in J, g \in G.$$

Then, the optimization objective proposed by (Wang et al. (2018b)) can be reformulated as follows:

$$TEC = \sum_{i \in M} \sum_{j \in J} \sum_{g \in G} fc_g \times (b_i - d_i) \times \alpha_{ijg} + \sum_{i \in M} \sum_{t \in T} c_t \times d_i \times x_{it}. \quad (3)$$

As stated in equation (3), we can estimate the total electricity cost of a sequence by the assignment of the amount processing time during each pricing interval (α_{ijg}) and by the shutdown times of two machines (x_{it}). When two solutions have the same shutdown times of two machines, they hold the same subset of x_{it} . Their second term in TEC gets the same value. Thus the difference is located in the assignment of processing time during each pricing interval.

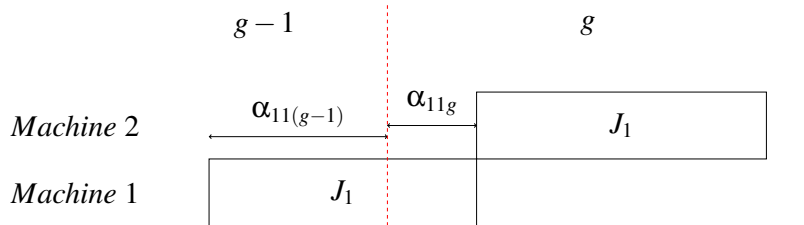


Figure 3: Example of variables α_{ijg} for processing of job J_1 on machine 1 during two pricing intervals ($g-1$) and g

6.3. New mathematical model (M1)

$$\min TEC = \sum_{i \in M} \sum_{j \in J} \sum_{g \in G} f c_g \times (b_i - d_i) \times \alpha_{ijg} + \sum_{i \in M} \sum_{t \in T} c_t \times d_i \times x_{it}. \quad (4)$$

Subject to:

$$\beta_{ijg} \leq 1 - \frac{(S_{ij} - ED_g)}{V}, \forall i \in M, j \in J, g \in G. \quad (5)$$

$$\beta_{ijg} \leq 1 + \frac{(C_{ij} - ST_g)}{V}, \forall i \in M, j \in J, g \in G. \quad (6)$$

$$\sum_{g \in G} \alpha_{ijg} = p_{ij}, \forall i \in M, j \in J. \quad (7)$$

$$\alpha_{ijg} \leq (ED_g - S_{ij}) + V \times (1 - \beta_{ijg}), \forall i \in M, j \in J, g \in G. \quad (8)$$

$$\alpha_{ijg} \leq (C_{ij} - ST_g) + V \times (1 - \beta_{ijg}), \forall i \in M, j \in J, g \in G. \quad (9)$$

$$\alpha_{ijg} \leq (ED_g - ST_g) \times \beta_{ijg}, \forall i \in M, j \in J, g \in G. \quad (10)$$

$$C_{ij} = S_{ij} + p_{ij}, \forall i \in M, j \in J. \quad (11)$$

$$C_{1j} \leq S_{2j}, \forall j \in J. \quad (12)$$

$$F_i \geq C_{ij}, \forall i \in M, j \in J. \quad (13)$$

$$F_i \leq t + V \times x_{it}, \forall i \in M, t \in T. \quad (14)$$

$$F_i \geq t + 1 - V \times (1 - x_{it}), \forall i \in M, t \in T. \quad (15)$$

$$F_i \leq T, \forall i \in M. \quad (16)$$

$$\delta_{jk} + \delta_{kj} \leq 1, \forall j, k \in J : j \neq k. \quad (17)$$

$$C_{ij} \leq S_{ik} + V \times (1 - \delta_{jk}), \forall i \in M, j, k \in J : j \neq k. \quad (18)$$

$$C_{ik} \leq S_{ij} + V \times \delta_{jk}, \forall i \in M, j, k \in J : j \neq k. \quad (19)$$

$$C_{ij}, S_{ij}, \alpha_{ijg} \geq 0, \forall i \in M, j \in J, g \in G. \quad (20)$$

$$\beta_{ijg}, x_{ij}, \delta_{jk} \in \{0, 1\}, \forall i \in M, j, k \in J, g \in G. \quad (21)$$

The objective (4) aims to minimize the total electricity cost.

Constraints (5) ensure that during pricing intervals which are before the starting time of job j on machine i , none of the processing time of j is executed. If $S_{ij} > ED_g$, then $S_{ij} - ED_g > 0$. That leads to $\frac{S_{ij} - ED_g}{V} > 0$, then $1 - \frac{S_{ij} - ED_g}{V} < 1$. So in this case, $\beta_{ijg} < 1$, the only value β_{ijg} can take is zero. That means if the starting time of job j on machine i is after the ending time of pricing interval g ($S_{ij} > ED_g$), none of the processing time of j is executed in this pricing interval $\beta_{ijg} = 0$.

Constraints (6) ensure that during pricing intervals which are after the completion time of job j on machine i , none of the processing time of j is executed. If $C_{ij} < ST_g$, then $C_{ij} - ST_g < 0$. That leads to $\frac{C_{ij} - ST_g}{V} < 0$, then $1 + \frac{C_{ij} - ST_g}{V} < 1$. So in this case, $\beta_{ijg} < 1$, the only value β_{ijg} can

take is zero. That means if the completion of job j is before the starting time of pricing interval g ($C_{ij} < ST_g$), none of the processing time of j is executed in this pricing interval $\beta_{ijg} = 0$.

Constraints (7), (8), (9) and (10) allocate processing time of job j on machine i to each pricing interval according to job's starting time and its completion time. Constraints (11) ensure that jobs are processed non-preemptively. Constraints (12) ensure that any job j must be processed on machine 1 and then on machine 2. Constraints (13), (14), (15) and (16) define the shutdown time of machine i . Constraints (17), (18) and (19) determine the sequence. Constraints (20) and (21) give variables' definition.

Table (4) gives the comparison between models MC and M1. Table (4a) shows the number of constraints, decision variables in terms of number of machines (M), number of jobs (J), time horizon (T), and number of pricing intervals(G). In Time-of-use tariffs, the number of pricing intervals is considerably small compared to the period in the time horizon. So based on the formulas given in Table (4a), the number of binary variables in M1 has been significantly reduced compared to MC. A few good examples are given in Table (4b) to illustrate the reduction of binary variables in model M1. These reductions help in improving the running time of model M1.

Table 4: Comparison of the number of constraints, the number of decision variables between the model MC and M1.

(a) Number of constraints, number of decision variables of models MC and M1

		MC	M1
Number of decision variables	Binary	$ M T (2 + J) + J J $	$ M T + (M J + 1 + J) G + J J $
	Continuous	$ M (2 J + 1) + 1$	$ M (2 J + J G + 1) + 1$
Number of constraints		$ M T (4 + 2 J) + 5 J + 1 + J J + J $	$ M [2T + J (5 G + 3) + 2] + J J + G $

(b) The difference in the number of binary variables between MC and M1 in instances of 4 pricing intervals.

Instance		Model		Difference of binary variables number between MC and M1
Number of jobs	Time horizon	MC	M1	
20	350	15800	1344	14456
30	400	26500	1564	24936
40	450	39400	1784	37616
50	500	54500	2004	52496

7. Logic-based Benders decomposition

7.1. Introduction

Benders decomposition uses a problem-solving strategy that can deal with large-scale mixed-integer programming models (Benders (1962)). It partitions the variables of a problem into two vectors x and y . Firstly, it fixes y to a trial value by optimally solving a mixed-integer Master Problem (MP). The MP is a relaxation of the global model. Then, the value of y is then used to define a sub-problem (SP) that contains only x . The solution of SP may reveal that the trial value of y is unacceptable, and the solution will be used to generate a Benders cut. The Benders cut eliminates the unacceptable value of y from the solution set. Then we obtain the next set of

trial values of y by solving the MP containing all the Benders so far generated. We solve the MP iteratively to optimality and use the solution to generate sub-problems until the MP and SP converge in value.

As a $F2|b_i, d_i, ToU|TEC$ problem requires making two different decisions, ordering jobs to form an optimal sequence and scheduling starting time of each job, a decomposition approach may be well suited. However, the classical Benders decomposition requires that MP should be a mixed integer model, and the SPs should be a linear or nonlinear programming problem. We utilize a logic-based Benders Decomposition (LBBD) approach (Hooker and Ottosson (2003)) which excluding that necessity in modeling MP and SPs. In this study, the mixed-integer MP assigns jobs to pricing intervals and then, base on Johnson's rule property previously developed, we can obtain a global jobs' sequence. Then the SP utilizes dynamic programming (Wang et al. (2018b)) to determine the optimal starting times for each job of the sequence given by the master problem solution. In the next sections, we go into the details of the Master Problem, the Sub Problem and Benders cut.

7.2. *Jobs assignment Master Problem (MP)*

As stated in the previous section, we estimate the total electricity cost with respect to the shutdown time of each machine and of workload assigned to each pricing interval. The MP assigns jobs to pricing intervals. According to the jobs assignment, MP can estimate workloads in each pricing interval and approximate shutdown times of each machine. Before formulating the MP as a MIP model, we represent some interesting results that can enhance the model's performance.

Each pricing interval has a certain duration and can contain at most a certain number of jobs. Let Cap_g^i be the maximum number of jobs that can be started on machine i during interval g . On each machine i , we order jobs according to shortest processing time rule (SPT). For each job according to SPT's order, Cap_g^i is increased by one when the total processing time on machine i is not superior to the length of the interval g . By considering machine 1 and machine 2 separately, we can obtain the maximum number of jobs assigned to each pricing interval on each machine.

As we know, Johnson's rule provides an optimal solution for $F2||C_{max}$ in polynomial time. Let π be Johnson's sequence for $F2||C_{max}$, for any sequences, their completion times on machine 1, and on machine 2 are superior to that of π . To formulate the MP, we need some new variables and parameters as follows:

Parameters :

- Cap_g^i : maximum number of jobs that can be started on machine i during interval g according to SPT.
- $F_i^{Johnson}$: shutdown time of machine i according to Jonhson's rule for $F2||C_{max}$ problem.

Variables:

- $\gamma_{jg}^i = 1$ if job j is started on machine i during pricing interval g , and 0 otherwise.
- TEC_1^h is the total electricity cost found by MP at h^{th} iteration.

Objective function:

$$\min TEC_1^h = \sum_{i \in M} \sum_{j \in J} \sum_{g \in G} fc_g \times (b_i - d_i) \times \alpha_{ijg} + \sum_{i \in M} \sum_{t \in T} c_t \times d_i \times x_{it}. \quad (22)$$

Subject to:

$$\sum_{g \in G} \gamma_{jg}^i = 1, \forall i \in M, j \in J. \quad (23)$$

$$\sum_{j \in J} \sum_{g \in G} \gamma_{jg}^i \leq Cap_g^i, \forall i \in M. \quad (24)$$

$$\beta_{ijg} \leq 1 - \frac{\sum_{g1 \in G} \gamma_{jg1}^i \times g1 - g}{l}, \forall i \in M, j \in J, g \in G. \quad (25)$$

$$\beta_{ijg} \leq 1 + \frac{\sum_{g1 \in G} \gamma_{jg1}^i \times ED_{g1} + p_{ij} - ST_g}{T}, \forall i \in M, j \in J, g \in G. \quad (26)$$

$$\beta_{1jg} \times g \leq \sum_{g1 \in G, g1 > g} \gamma_{jg1}^2 \times g1, \forall j \in J, g \in G. \quad (27)$$

$$\sum_{g \in G} \alpha_{ijg} = p_{ij}, \forall i \in M, j \in J. \quad (28)$$

$$\sum_{j \in J} \alpha_{ijg} \leq ED_g - ST_g, \forall i \in M, g \in G. \quad (29)$$

$$\alpha_{ijg} \leq (ED_g - ST_g) \times \beta_{ijg}, \forall i \in M, j \in J, g \in G. \quad (30)$$

$$\gamma_{jg}^i \leq \alpha_{ijg}, \forall i \in M, j \in J, g \in G. \quad (31)$$

$$F_i \geq \beta_{ijg} \times ST_g + \sum_{k \in J} \alpha_{ikg}, \forall i \in M, j \in J, g \in G. \quad (32)$$

$$F_1 \leq F_2. \quad (33)$$

$$F_i \geq F_i^{Johnson}, \forall i \in M. \quad (34)$$

$$\text{Equations(14), (15), (16)}. \quad (35)$$

$$\text{MIP cuts.} \quad (36)$$

The objective (22) aims to minimize the total electricity costs.

Constraints (23) ensure that job j is started on machine i during at most one pricing interval g .

Constraints (24) ensure that the number of jobs j which are started on machine i during interval g respect its capacity.

Constraints (25) ensure that for any job j , if it starts on machine i during interval g so it can not be processed on that machine during intervals g locating before g , $\beta_{ijg} = 0$.

Constraints (26) determine for any job j whether it is processed on machine i during pricing interval g ($\beta_{ijg} = 1$) or not ($\beta_{ijg} = 0$) in function of the time period where job j is started on machine i and its processing time. If a job j is started on machine i during pricing interval g ($\gamma_{jg}^i = 1$), there is a possibility that job j started at the last time period of interval g (ED_{g1}). Hence, $\sum_{g1 \in G} \gamma_{jg1}^i \times ED_{g1} = ED_{g1}$. If the processing time of job j (p_{ij}) is long enough, then it's processing will be continued and finished during next pricing interval ($g+1$) ($ST_{g1+1} < ED_{g1} + p_{ij} < ST_{g1+2}$). Then ($ED_{g1} + p_{ij} - ST_{g1+1} > 0$) and ($ED_{g1} + p_{ij} - ST_{g1+2} < 0$). We will have: $\beta_{ij(g1+1)} \leq 1 + \frac{\sum_{g1 \in G} \gamma_{jg1}^i \times ED_{g1} + p_{ij} - ST_{g1+1}}{T} = 1 + \frac{ED_{g1} + p_{ij} - ST_{g1+1}}{T} \leq 1$. So $\beta_{ij(g1+1)} = 0$ or $\beta_{ij(g1+1)} = 1$. Contrarily, $\beta_{ij(g1+2)} \leq 1 + \frac{\sum_{g1 \in G} \gamma_{jg1}^i \times ED_{g1} + p_{ij} - ST_{g1+2}}{T} = 1 + \frac{ED_{g1} + p_{ij} - ST_{g1+2}}{T} < 1$. So $\beta_{ij(g1+2)} = 0$.

Constraints (27) ensure that for any job j , its starting time on machine 2 located after its starting time on machine 1.

Constraints (28) ensure that jobs are processed entirely.

Constraints (29) ensure that total processing time during each pricing interval respects its duration.

Constraints (30) ensure that job can only be processed during pricing intervals assigned.

Constraints (31) ensure that job j will be processed during the pricing interval where it started.

Constraints (32) ensure that the shutdown time of machine i must superior to the starting time of all pricing intervals plus it's processing time.

Constraints (33) ensure that machine 1 shut down before machine 2.

Constraints (34) ensure that shutdown time on each machine must be superior or equal to the shutdown time of Johnson's sequence for $F2||C_{max}$.

Constraints (35) consist of equations (14)-(16) of model M1 that determine the state of each machine in function of its shutdown time and ensure that all jobs will be processed during the planning horizon.

Constraints (36) are Benders cuts. The Benders cuts are presented in section (7.3).

7.3. Starting time assignment Sub Problem and Bender's cut

The solution of MP gives a set of jobs assigned to each pricing interval. Let W^h be the set of solutions at the iteration h of the MP, $\gamma_{jg}^2 = 1$. Thanks to the property (2), by knowing the jobs assignment, we can get a global sequence. The SP determines the optimal starting time of sequence deduced by MP. As proposed in (Wang et al. (2018b)), when the job sequence is fixed, a dynamic algorithm can determine the optimal starting time of each on each machine in $O(nT^4)$ time. So we utilize the dynamic algorithm proposed by (Wang et al. (2018b)) for SPs to obtain solutions with total electricity cost TEC_2^h . The solution obtained by SP at the iteration h helps to generate a Benders cut added to the MP. A valid Benders cut in a given iteration must exclude the current globally infeasible assignment in the MP without removing any globally optimal assignment. We

propose a Benders cut based on the study of (Hooker and Ottosson (2003)) as follows:

$$TEC \geq TEC_2^h - \sum_{\gamma_{jg}^2 \in W^h} (1 - \gamma_{jg}^2) \times V \quad (37)$$

Where W^h is set of $\gamma_{jg}^2 = 1$: the assignment of jobs' starting time on machine 2 to pricing intervals g .

The procedure of our solution approach is illustrated in Figure. (4) and Figure. (5). Figure. (5) represents the solution approach of LBBDD for an instance of 8 jobs. It shows how the Master Problem and Sub Problem update the upper bound and lower bound of the global problem.

1. First, LBBDD solves the Master Problem (MP) to optimality. Because Master Problem is a relaxed version of our problem, so it's solution is a lower bound.
2. Then, based on MP's solution, LBBDD solves the Sub Problem (SP). If SP finds that MP's solution is feasible, then it is our globally optimal solution. Otherwise, SP will generate a Bender's cut. We use SP's solution to update the upper bound.
3. LBBDD repeats this process until the MP and SP converge in value. In this example, MP and SP converge to the optimal solution whose $TEC = 16,8\$$ after 39 iterations.

8. Computational experiments and results

In this section, we investigate the performance of the approach we proposed in section 7 through computational study. In detail, we compare the performance of our proposed MILP model M1 and the Logic-based Benders decomposition (LBBDD) with the performance of solution approaches of the MILP model of (Wang et al. (2018b)). We named the MILP model of (Wang et al. (2018b)) as MC.

We solve the mathematical models M1, MC with ILOG CPLEX. These MILP model and LBBDD are implemented on g++ (Ubuntu 7.4.0-lubuntu 18.04.1) 7.4.0 linked with Cplex Studio IDE 12.9.0. The computational experiment was executed on an IntelCore i7-6820HQ CPU @2.7 GHz 2.7 GHz PC with 8 GB RAM and the Ubuntu 18.04.2 LTS operating system.

8.1. Data generation

The performance of the proposed approaches was evaluated by solving a set of experimental test instances. The test instances were randomly generated whose characteristics are based on (Wang et al. (2018b)). Firstly, we randomly generate the processing time of job p_{ij} from the uniform distribution on $\{1, 2, \dots, 10\}$. Then to complete the set of instances, we generate the following parameters:

- The planning horizon is determined by $T = \lceil 0.8 \times \sum_{i=1}^2 \sum_{j \in J} p_{ij} \rceil$.
- For fluctuation of electricity prices, we divide the planning horizon into 4 pricing intervals: off-peak, peak, off-peak, semi-peak. The pricing intervals have approximately equal duration $T/4$. To denote the stable and unstable electricity markets, the levels $\max\{c_t\}/\min\{c_t\} = 3$ and $\max\{c_t\}/\min\{c_t\} = 6$ are tested respectively.

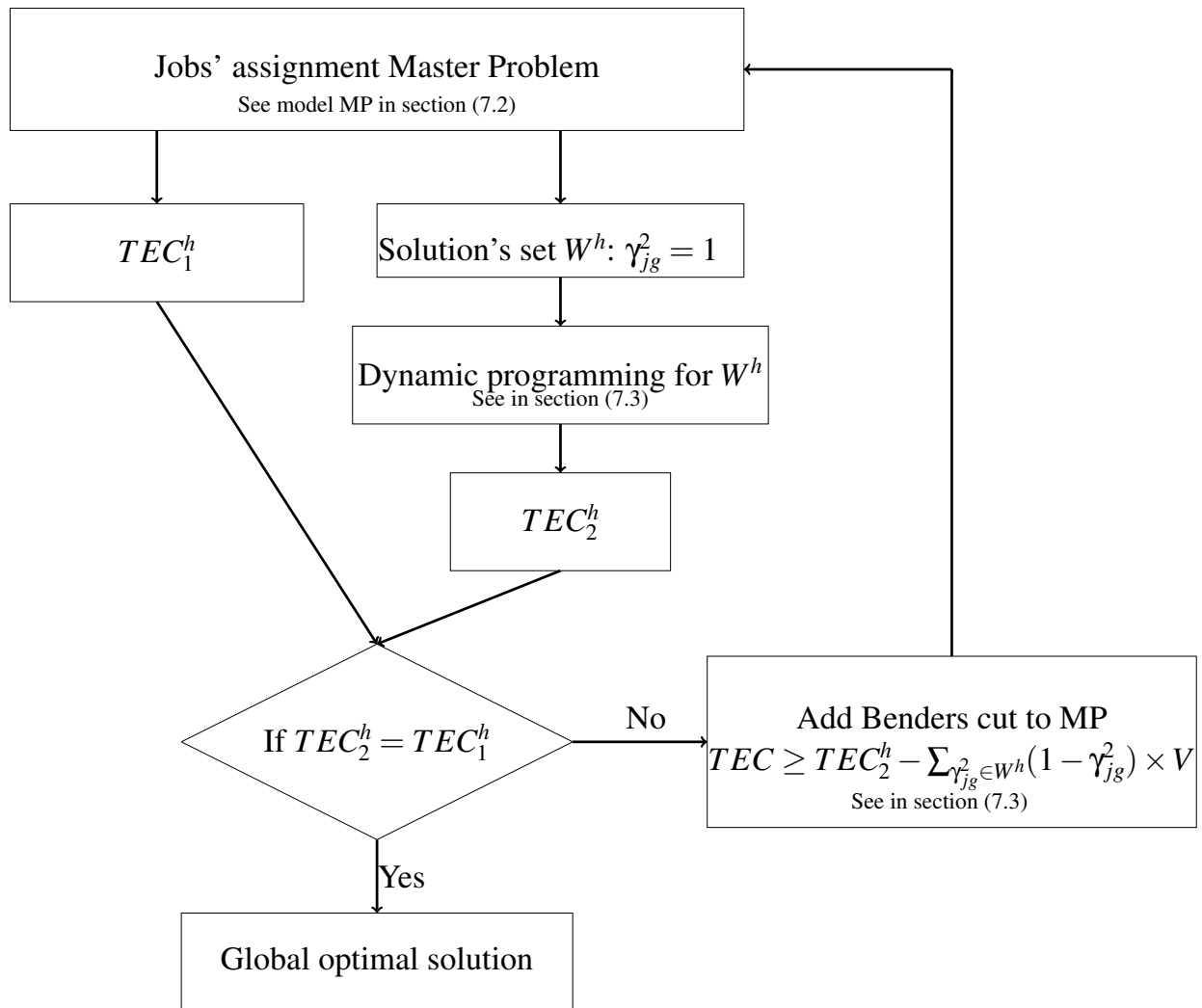


Figure 4: Logic-based Benders decomposition for $F2|b_i, d_i, ToU|TEC$

Electricity cost (\$)

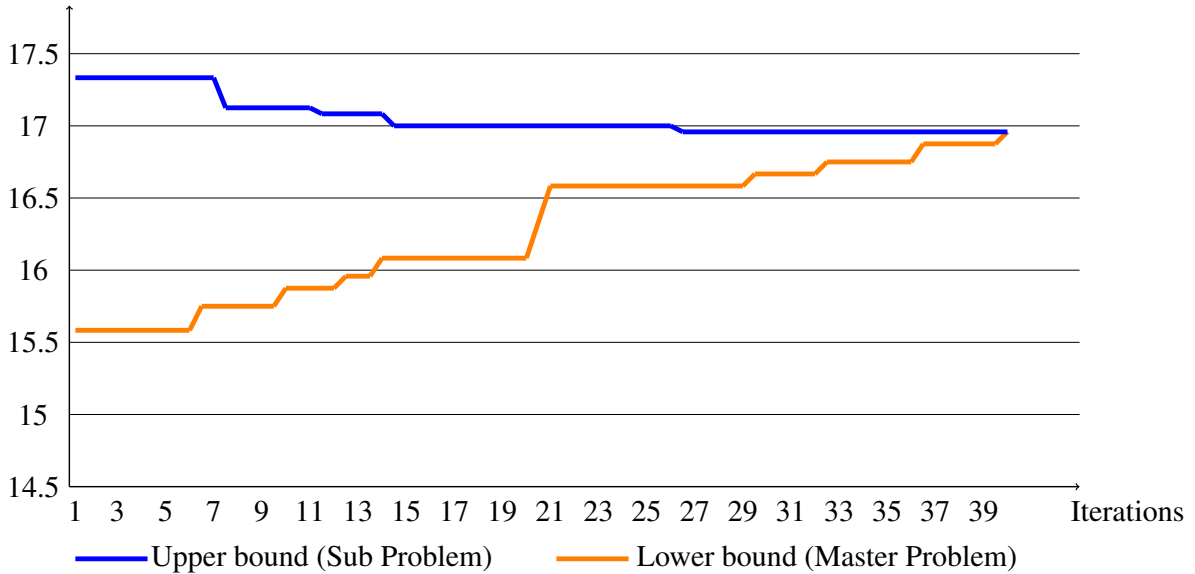


Figure 5: An example of LBBD for solving $F2|b_i, d_i, ToU|TEC$ with instance of 8 jobs

- For energy consumption rates, we consider three scenarios of energy consumption rates on machine: $R1 : b_1 = 2, d_1 = 1, b_2 = 2, d_2 = 1$, two machines are identical; $R2 : b_1 = 2, d_1 = 1, b_2 = 6, d_2 = 2$, machine 2 is more energy-intensive; and $R3 : b_1 = 6, d_1 = 2, b_2 = 2, d_2 = 1$, machine 1 is more energy-intensive. We generate 30 instances for each scenarios $R1, R2$ and $R3$.

Firstly, we test on small-size instances, $n = 6, 8$ jobs, and then on large-size instances $n = 20, 30, 40$ and 50 jobs. We set a 30 minutes computational time limit on each instance.

8.2. Numerical results

To test the performance of our solution approaches, we compare our MILP (M1), LBBD with the mathematical model of (Wang et al. (2018b)) (MC). (Wang et al. (2018b)) also proposed several heuristics to solve this problem. We cite the two most efficient heuristics of (Wang et al. (2018b)): ED11 and ILSDP.

- ED11 generates randomly ten sequences, which are difference from Johnson’s sequence and then use the dynamic programming on these sequences and Johnson’s sequence.
- ILSDP is an iterated local search algorithm to solve the problem with problem-tailored procedures and move operators. Dynamic programming is then applied to the final solution to improve the electricity cost.

We also proposed a new heuristic JohnsonDP. JohnsonDP applies dynamic programming on Johnson’s sequence to reduce the electricity cost. So, for small-size instances ($n = 6, 8$ jobs), and medium-size instances ($n = 20, 30$ jobs), we compared our MILP (M1), LBBD with MILP (MC)

of (Wang et al. (2018b)). For large-size instances ($n = 40, 50$ jobs), we compared LBBB with several heuristics, JohnsonDP, ED11 and ILSDP. The results are reported in Tables (5) - (10) and in Figure.(6) and Figure.(7).

For small-size instances ($n = 6, 8$ jobs), all approaches provide the same TEC within the time limit. So for these instances, we are interested only in the comparison between computational time. On the other hand, for medium and large-size instances ($n = 20, 30, 40, 50$ jobs), all approaches can not provide the optimal solution within the time limit. All three approaches, (MC, M1, LBBB) reached the same computational time of 30 minutes. Therefore, we only present the comparison between all solution approaches with the “Best Known Solution” (BKS) obtained by all solution approaches (M1, MC and LBBB, Johnson DP, ED11 and ILSDP). The Gap_i represents the gap between the solution of approach i and the best solution: $Gap_i = \frac{TEC_i - TEC^{BKS}}{TEC^{BKS}}$.

Tables (5)-(7) summarize the computational results. In each table, Max, Average, and Min stand respectively for maximum value, average value and minimum value of the solution’s set. Table (5) reports the computational results of the entire instance’s set for each configuration $n = 6, 8, 20, 30, 40$ and 50 jobs. That gives an overview of the performance of the solution approaches for all scenarios. Then Tables (6) - (7) give more details of numerical tests for each scenario. That provides the behavior of solution approaches in dealing with the stable and unstable electricity markets or with the different energy-intensity of machines. In each table, Max, Average, Min and σ stand respectively for maximum value, average value, minimum value and standard deviation of the solution’s set.

First of all, we focus on small-size instances, $n = 6, 8$ jobs, to evaluate the performance of our MILP M1, and LBBB. As all MILP give optimal solution in very short computational time, we do not need heuristics for these problems. As shown in Table (5a) and Table (6), we obtain the following observations:

- Table (5a) shows that the average computational time of M1 and LBBB are better than MC’s. We have significantly reduced the computational time. In particular, for $n = 8$ jobs, M1 and LBBB outperform strongly MC. For example, the maximum value of computational time of all approaches are as follows: 26.0 seconds for LBBB, 304.0 seconds for M1 compared to 1626.0 seconds for MC.
- In Table (6), we can find that the performance of MC is relatively better on scenario $R1$ meaning on identical machines. However, the average computational time of new MILP M1 is better than MC’s on other scenarios. In particular, for $n = 8$ jobs and the level of price equal to 6 (unstable electricity markets), the new MILP M1 outperform MC strongly. For example, for scenario $R2$ where the machine 2 is more energy-intensive, the average computational time of M1 is 15.8 seconds compared to 116.5 seconds for MC. Regarding LBBB, this approach outperforms other approaches on all scenarios for $n = 8$ jobs.

Then, for medium-size instances, we provide in Table (10) the gap to optimality of LBBB given by CPLEX. From Table (5b), Table (7) and Table (10) we have following observations:

- For medium-size instances, the performances of MC deteriorate rapidly when n increases. For some instances, MC gets a gap with BKS until 15.8% for $n = 20$ jobs and 60.76% for

$n = 30$ jobs. Especially, we are interesting in instances of $n = 30$ jobs because the size is close to industrial problem's. In Table (7), we can find that the performance of M1 significantly dominates MC for unstable electricity markets (Level of price equals 6). In addition, in the scenario $R2$ where the machine 2 is more energy-intensive, the average gap to “Best Known Solution” and the standard deviation of M1 in $R2$ is smaller than in $R1$ and $R3$. That means M1 works better in scenario $R2$ than in other scenarios.

- Regarding LBBD, this approach outperforms all other existing solution approaches in terms of TEC . LBBD provides the best solution for all instances. In addition, in Table (10), we can see that the maximum gap to optimality given by CPLEX for LBBD is under 6.2% for $n = 20$ and is under 3.3% for $n = 30$ within the time limit. Its average gap to optimality is under 3%. By comparing the performances of LBBD with M1's, we can see that the property 2 suits very well the decomposition approach and considerably improves the quality of the solutions.

For large-size instances, we compare the performance of LBBD with heuristics JohnsonDP, ED11 and ILSDP. Two MILP M1 and MC are not designed to solve a large-size problem, their performance has deteriorated rapidly for medium-size instances. From Table(5b), Table(8), Table(9), Table(10) and two figures Figure.(6), Figure.(7), we have the following observations:

- For $n = 20, 30$ jobs, we can see that LBBD outperforms significantly JohnsonDP and ILSDP (for $n = 20$ jobs, Gap of JohnsonDP and ILSDP are 6.6% and 12.65% respectively compared with LBBDs' solutions in some instances). ED11 has a similar performance with LBBD in these cases (Gap of ED11 is smaller than 2%). However, the computational time of ED11 is not significantly different from that of LBBD (for some instances, ED11 needs until 1790 seconds, compared with 1800 seconds of LBBD). In addition, from Table(10), the average gap to optimality given by CPLEX of LBBD is small (under 5%). So for medium-size instances ($n = 20, 30$ jobs), LBBD can assure the best solution in terms of electricity cost within a reasonable computational time (30 minutes). ED11 is a good alternative method to have a shorter running time and a small gap compared to “Best Known Solution”.
- For $n = 40, 50$ jobs, LBBD still outperforms all the heuristics. It provides the best solutions among all the approaches. However, the gap between heuristics and LBBD gets smaller when n increases. First of all, the computational time of ED11 rapidly increases and surpasses that of LBBD (for some instances of $n = 50$, ED11 needs 2331 seconds compared to 1800 seconds of LBBD). JohnsonDP and ILSDP give approximate performance as ED11 within a shorter time. Hence, ED11 is not favorable for large-size instances. Secondly, in Figure.(6), we can see that JohnsonDP and ILSDP work well in the case of a stable electricity market (Level of price equal to 3). The average gap of JohnsonDP and ILSDP to “Best Known Solution” is under 2%. On the other hand, in Figure. (7), for unstable electricity market (level of price equal to 6) and energy-intensity of two machines are not identical (scenario $R2$ and $R3$), the maximum gap of JohnsonDP and ILSDP are significant (until 6.37% and 10.10% respectively). In these cases, we consider a trade-off between electricity cost and computational time. LBBD assure the best solution in almost all instances

within 30 minutes with an average gap to optimality under 5%. JohnsonDP and ILSDP provide good solutions within a shorter computational time (around 10 minutes) but can have a maximum gap until 10% compared with “Best Known Solution”.

Hence, for large-size instances, we evaluate our approach (LBBD) in terms of the objective function value (electricity cost) and in terms of computational time.

- In terms of electricity cost, LBBD outperforms all heuristics. It provides the best solution for all instances. Especially, when the electricity market is unstable and two machines are not energy-intensive identical, LBBD outperforms significantly other approaches.
- In terms of computational time, JohnsonDP and ILSDP provide good solutions within a short computational time. However, in the worst case, their solution cost up to 10% more expensive than LBBD’s solution. For $n = 50$ jobs, the difference in terms of computational time between LBBD (30 minutes) with JohnsonDP and ILSDP (around 10 minutes) is reasonable for an operational problem as a scheduling problem.
- In addition, LBBD provides good solutions. The average gap to optimality of LBBD given by CPLEX is under 5%. That helps assure the solution’s quality.

We have Figure. (8) and Table (11) to evaluate the performance of LBBD. These two demonstrate the computational efforts of the LBBD method decomposed on its two main components: the master problem (MP) and the subproblem (SP). Figure. (8) shows the breakdown of the average solution times by both the MP and SP in percentages, for all instances, in two scenarios: the level of price equals 3, and the level of price equals 6. The breakdown of the average solution times in each scenario is then detailed in Table (11). From Figure. (8) and Table (11), we have some observations. First, MP takes a large part of the computational efforts (more than 90% for small-size instances, and more than 80% for large-size instances). Second, the more the number of jobs, the greater the computational efforts allocated to SP. Last, for unstable electricity market instances, less computational efforts are allocated to MP (in average 70%) compared to the stable electricity market (MP takes in average 80% of computational times). We can explain these observations as follows:

- SP utilizes the dynamic programming of (Wang et al. (2018b)). The complexity of the dynamic programming is $O(nT^4)$. When the number of jobs increases, the planning horizon T also increases to cover all jobs. Therefore, the computational time required is longer when the number of jobs increases.
- The complexity of MP is still an open question that we will address in our future research. However, we can see that the stable electricity market instances required more computational efforts to solve MP than the unstable electricity cases. One reason may be that the stable electricity market cases contain more equivalent solutions. Therefore, it requires more global search efforts of the solution space in MP for stable electricity cases. This also explains why LBBD slightly outperforms other approaches in stable electricity markets but strongly dominates in unstable electricity markets.

Based on previous observations, when the production plant steps into an environment with TOU tariff from an environment without TOU tariff, we need to consider whether the electricity market is stable or unstable.

- If the electricity market is unstable, LBBD strongly outperforms other solution approaches in terms of electricity cost. Especially, when the electricity market is unstable and two machines are not energy-intensive identical, LBBD is favorable for all instances (small-, medium- and large-size instances). It provides good quality solutions within reasonable computational time.
- If the electricity market is stable, we distinguish between small-, medium- and large-size instances.
 - For small-size instances, LBBD, M1, MC and ED11 deliver approximate performance, both in terms of electricity cost and computational time. The manager can use any solution approaches above for small-size instances.
 - For medium-size instances, LBBD provides best solutions in terms of electricity cost. However, ED11 is a good alternative method with a shorter running time and a small gap compared to the “Best Known Solution”.
 - For large-size instances, LBBD still outperforms all the heuristics. LBBD provides the best solutions in all instances. However, the gap between heuristics and LBBD shrinks when the number of jobs increases. The average gap of JohnsonDP and ILSDP from the “Best Known Solution” is under 2%. Hence, JohnsonDP and ILSDP are favorable in this configuration.

9. Conclusion

This work studies a two-machine flow shop scheduling problem with minimization of total electricity cost under Time-of-use tariffs. $F2|b_i, d_i, ToU|TEC$. The planning horizon is divided into pricing intervals. Firstly, to enhance the solution approach, we proposed a property based on Johnson’s rule. The property determines the optimal sequence for a given pricing interval’s job assignment. Secondly, we developed a new mixed-integer linear model with a new objective function. Thirdly, we proposed an exact method, “Logic-based Benders Decomposition” to solve the problem. Finally, we tested the performance of our proposed solution approaches with the mathematical model of (Wang et al. (2018b)).

Our new mixed-integer linear models M1 outperforms the MILP of (Wang et al. (2018b)) for small size and large size instance. The new MILP works better for unstable electricity market and when machine 2 is more energy-intensive. In addition, we proposed an exact method “Logic-based Benders Decomposition” to solve the problem. Within the time limit, LBBD provides the best solution comparing with solution approaches in the literature for almost all instances. For large-size instances, LBBD gives solutions with an average gap to lower bound under 2.8%. The gap to lower bound can also be utilized to evaluate the performance of other heuristics.

Table 5: The overview of numerical results

(a) Average computational time (second).

Number of jobs		Approches					
		<i>LBBD</i>	<i>MI</i>	<i>MC</i>	<i>JohnsonDP</i>	<i>EDII</i>	<i>ILSDP</i>
6	Max	16.0	9.0	35.0	—	—	—
	Average	2.0	2.2	6.5	—	—	—
	Min	1.0	1.0	1.0	—	—	—
8	Max	26.0	304.0	1626.0	—	—	—
	Average	1.2	20.8	66.1	—	—	—
	Min	1.0	1.0	3.0	—	—	—
20	Max	***	***	***	51.30	1501.00	52.63
	Average	***	***	***	30.45	766.19	31.65
	Min	***	***	***	10.94	15.00	10.94
30	Max	***	***	***	261.77	1790.00	318.57
	Average	***	***	***	161.15	994.18	228.73
	Min	***	***	***	74.78	246.00	125.61
40	Max	***	***	***	473.09	1967.22	533.73
	Average	***	***	***	370.31	1156.21	373.56
	Min	***	***	***	206.72	337.11	206.72
50	Max	***	***	***	626.02	2331.23	586.07
	Average	***	***	***	475.46	1514.27	476.23
	Min	***	***	***	377.84	475.02	342.75

*** : reach the limit computational time (1800s)

— : Test is not performed

(b) Average gap (%) to BKS for large size instances

Number of jobs		Approches					
		<i>LBBD</i>	<i>MI</i>	<i>MC</i>	<i>JohnsonDP</i>	<i>EDII</i>	<i>ILSDP</i>
20	Max	0.00	3.66	15.80	7.68	2.68	12.65
	Average	0.00	0.26	4.26	3.57	0.77	4.60
	Min	0.00	0.00	0.00	0.00	0.00	0.00
30	Max	0.00	18.02	60.76	4.57	1.33	7.91
	Average	0.00	0.87	23.53	2.12	0.39	2.43
	Min	0.00	0.00	0.22	0.00	0.00	0.00
40	Max	0.00	—	—	6.37	6.37	10.10
	Average	0.00	—	—	1.89	1.86	2.97
	Min	0.00	—	—	0.00	0.00	0.11
50	Max	0.00	—	—	4.83	4.83	6.93
	Average	0.00	—	—	1.23	1.22	1.93
	Min	0.00	—	—	0.00	0.00	0.00

— : Test is not performed

Table 6: Comparison of computational time between solution approaches for instances of 6.8 jobs.

Level of price		Number of jobs																
		6 jobs			8 jobs													
Scenario	Level of price	Max	Average	Min	σ	LBBDD	MI	MC	Scenario	Level of price	Max	Average	Min	σ	LBBDD	MI	MC	
1	3	16.0	9.0	7.0	16.0	3.6	4.2	2.4	1	3	26.0	1.8	30.5	48.4	304.0	948.0	1	1
		1.0	1.0	1.0	1.0	3.6	2.6	1.9	1.0		2.0	3.0	58.7	172.7				
		5.0	2.5	3.5	5.0	2.5	2.0	2.5	5.0		1.0	1.0	88.0	303.0				
2	3	5.0	2.3	3.5	5.0	2.3	5.7	1.0	2	3	4.0	1.1	21.4	38.9	21.4	38.9	2	2
		1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0		1.0	1.0	5.0	5.0				
		1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0		1.0	1.0	22.57	69.47				
3	3	5.0	5.0	14.0	5.0	1.5	2.3	5.7	3	3	2.0	2.0	27.4	42.4	27.4	42.4	3	3
		1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0		1.0	7.0	4.0	4.0				
		0.97	1.12	4.37	0.97	1.12	4.37	1.12	4.37		0.18	29.44	54.99					
1	6	8.0	8.0	35.0	8.0	8.0	35.0	8.0	1	6	2.0	2.0	159.9	110.4	62.0	1597.0	1	1
		2.0	2.0	10.8	2.0	2.0	10.8	2.0	2.0		2.0	5.0	5.0					
		1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0		1.0	16.1	311.9					
2	6	4.0	4.0	27.0	4.0	4.0	27.0	4.0	2	6	2.0	2.0	1626.0	116.5	87.0	1626.0	2	2
		1.5	1.3	7.8	1.5	1.3	7.8	1.5	1.3		6.0	6.0	15.8	116.5				
		1.0	1.0	2.0	1.0	1.0	2.0	1.0	1.0		1.0	16.9	333.1					
3	6	2.00	2.00	29.00	2.00	2.00	29.00	2.00	3	6	1.0	1.0	47.0	39.7	47.0	453.0	3	3
		1.07	1.07	8.50	1.07	1.07	8.50	1.07	1.07		6.0	6.0	14.0	39.7				
		1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00		1.00	2.0	6.0	6.0				
		0.2	0.2	6.1	0.2	0.2	6.1	0.2			0.0	13.2	83.2					

Table 7: Comparison of gap (%) to “BKS” between solution approaches *LBBD.M1* and *MC* for instances of 20.30 jobs.

		Number of jobs															
		20 jobs					30 jobs										
Level of price	Scenario	Max	Average	Min	σ	LBBD	M1	MC	Level of price	Scenario	Max	Average	Min	σ	LBBD	M1	MC
3	1	0.87	0.11	0.00	0.24	0.00	0.11	4.94	3	1	0.00	0.00	0.00	2.85	0.00	1.29	38.89
	2	1.97	0.19	0.00	0.41	0.00	3.96	2		0.00	0.00	0.00	0.00	5.63	0.00	0.00	8.23
	3	1.86	0.32	0.00	0.48	0.00	2.67	3		0.00	0.00	0.00	0.00	1.41	0.00	0.00	0.49
6	1	1.20	0.16	0.00	0.27	0.00	5.07	12.42	6	1	0.00	0.00	0.00	3.38	0.00	0.00	2.36
	2	3.66	0.53	0.00	0.84	0.00	6.72	14.58		2	0.00	0.00	0.00	1.15	0.00	0.00	6.48
	3	1.49	0.24	0.00	0.40	0.00	5.02	15.80		3	0.00	0.00	0.00	15.63	0.00	0.00	34.88
								3.71									16.61

Table 8: Comparison of gap (%) to “BKS” between solution approaches *LBB*, *JohnsonDP*, *ED11* and *ILSDP* for instances of 20, 30 jobs.

Level of price		Number of jobs																		
		20 jobs					30 jobs													
Scenario	Max	Average	Min	σ	LBB	JohnsonDP	ED11	ILSDP	Level of price	Scenario	Max	Average	Min	σ	LBB	JohnsonDP	ED11	ILSDP		
1	0.00	6.64	0.67	8.43	0.00	0.00	0.00	0.00	3	1	0.00	4.38	0.55	4.59	0.00	4.38	0.55	4.59		
	0.00	3.31	0.17	3.58	0.00	0.00	0.17	3.58			0.00	2.15	0.14	2.20	0.00	0.00	0.00	0.00	0.00	0.00
	0.00	2.26	0.24	2.39	0.00	0.00	0.24	2.39			0.00	0.00	0.16	1.35	1.35	0.00	0.00	1.35	0.16	1.35
2	0.00	7.68	1.67	9.90	0.00	0.00	1.67	9.90	3	2	0.00	4.48	1.31	5.21	0.00	4.48	1.31	5.21		
	0.00	4.27	0.78	4.84	0.00	0.00	0.78	4.84			0.00	2.42	0.33	2.53	0.00	0.00	0.00	0.00	0.00	
	0.00	0.77	0.00	0.35	0.00	0.00	0.00	0.35			0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	0.00	2.16	0.45	2.41	0.00	0.00	0.45	2.41	3	3	0.00	1.27	0.29	1.34	0.00	1.27	0.29	1.34		
	0.00	6.98	1.44	8.47	0.00	0.00	1.44	8.47			0.00	4.57	0.72	4.64	0.00	0.00	0.00	0.00	0.00	
	0.00	3.93	0.58	4.47	0.00	0.00	0.58	4.47			0.00	2.00	0.31	2.05	0.00	0.00	0.00	0.00	0.00	0.00
1	0.00	1.99	0.42	2.03	0.00	0.00	0.42	2.03	6	1	0.00	1.43	0.17	1.45	0.00	1.43	0.17	1.45		
	0.00	5.26	1.94	8.93	0.00	0.00	1.94	8.93			0.00	3.88	0.94	4.66	0.00	0.00	0.00	0.00	0.00	
	0.00	2.95	0.77	4.16	0.00	0.00	0.77	4.16			0.00	1.86	0.35	1.89	0.00	0.00	0.00	0.00	0.00	0.00
2	0.00	0.31	0.00	0.50	0.00	0.00	0.00	0.50	6	2	0.00	0.09	0.00	0.34	0.00	0.09	0.00	0.34		
	0.00	1.37	0.54	2.07	0.00	0.00	0.54	2.07			0.00	1.16	0.26	1.17	0.00	0.00	0.00	0.00	0.00	
	0.00	6.60	2.68	12.65	0.00	0.00	2.68	12.65			0.00	4.57	1.33	7.91	0.00	0.00	0.00	0.00	0.00	0.00
3	0.00	3.77	1.21	6.40	0.00	0.00	1.21	6.40	6	3	0.00	2.21	0.62	3.36	0.00	2.21	0.62	3.36		
	0.00	0.21	0.00	0.84	0.00	0.00	0.00	0.84			0.00	0.36	0.24	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	0.00	1.94	0.66	3.08	0.00	0.00	0.66	3.08			0.00	1.23	0.31	2.05	0.00	0.00	0.00	0.00	0.00	0.00
6	0.00	6.86	2.26	8.67	0.00	0.00	2.26	8.67	6	3	0.00	4.08	1.06	4.16	0.00	4.08	1.06	4.16		
	0.00	3.17	1.12	4.15	0.00	0.00	1.12	4.15			0.00	2.11	0.60	2.56	0.00	0.00	0.00	0.00	0.00	0.00
	0.00	0.44	0.00	0.44	0.00	0.00	0.00	0.44			0.00	0.39	0.03	0.39	0.00	0.00	0.00	0.03	0.39	
		1.66	0.66	1.99	0.00	0.00	0.66	1.99							0.00	1.08	0.27	1.17		

Table 9: Comparison of gap (%) to “BKS” between solution approaches *LBB*, *JohnsonDP*, *ED11* and *ILSDP* for instances of 40, 50 jobs.

Level of price		Number of jobs									
		40 jobs					50 jobs				
Scenario	Max Average Min σ	LBB	JohnsonDP	ED11	ILSDP	Scenario	Level of price	LBB	JohnsonDP	ED11	ILSDP
3	1	0.00	3.29	3.29	3.33	1	3	0.00	2.18	2.18	2.21
		0.00	1.76	1.72	1.73			0.00	1.09	1.09	1.14
		0.00	0.00	0.00	0.11			0.00	0.00	0.00	0.00
3	2	0.00	1.12	1.12	1.13	2	3	0.00	0.84	0.84	0.85
		0.00	5.39	5.39	6.14			0.00	4.38	4.38	5.43
		0.00	1.91	1.89	2.83			0.00	1.06	1.05	1.23
3	3	0.00	0.00	0.00	0.12	3	3	0.00	0.00	0.00	0.00
		0.00	1.22	1.21	1.71			0.00	1.08	1.08	1.25
		0.00	3.81	3.73	5.10			0.00	2.57	2.48	3.92
6	1	0.00	1.90	1.84	2.56	1	6	0.00	1.16	1.15	1.68
		0.00	0.03	0.03	0.59			0.00	0.00	0.00	0.29
		0.00	1.23	1.26	1.29			0.00	0.84	0.83	0.92
6	2	0.00	3.66	3.66	5.20	2	6	0.00	2.32	2.32	4.67
		0.00	1.79	1.77	2.88			0.00	1.21	1.21	1.78
		0.00	0.00	0.00	0.87			0.00	0.03	0.03	0.14
6	3	0.00	1.07	1.07	1.17	3	6	0.00	0.82	0.82	1.06
		0.00	6.37	6.37	10.10			0.00	4.83	4.83	6.93
		0.00	2.08	2.04	4.73			0.00	1.24	1.22	3.11
6	3	0.00	0.02	0.02	0.79	3	6	0.00	0.00	0.00	0.93
		0.00	1.29	1.30	2.38			0.00	1.03	1.03	1.43
		0.00	4.10	4.10	5.81			0.00	2.91	2.61	4.31
6	3	0.00	1.90	1.90	3.11	3	6	0.00	1.59	1.57	2.62
		0.00	0.00	0.00	0.76			0.00	0.10	0.10	0.88
		0.00	1.41	1.41	1.43			0.00	0.82	0.80	0.81

Table 10: Gap (%) to optimality given by CPLEX of *LBB*D for instance of 20.30.40 and 50 jobs

Level of price	Scenario	Gap(%) to optimality											
		20 jobs			30 jobs			40 jobs			50 jobs		
		Max	Average	Min	Max	Average	Min	Max	Average	Min	Max	Average	Min
3	1	6.2	2.8	0.1	2.7	0.7	0.1	7.4	5.3	1.0	9.7	5.1	4.6
	2	4.4	2.4	0.3	3.3	1.5	0.1	3.7	3.1	2.2	9.3	4.8	2.6
	3	3.4	1.3	0.1	2.1	0.7	0.1	2.9	0.4	0.1	8.3	3.6	0.1
6	1	4	1.8	0.0	2.2	0.9	0.1	4.3	2.1	0.2	8.6	4.8	0.5
	2	4.1	2.2	0.2	2.2	0.9	0.1	3.2	2.3	0.3	9.2	3.3	1.8
	3	3.2	1.3	0.1	2.1	0.4	0.1	0.4	0.2	0.1	7.2	1.5	0.1

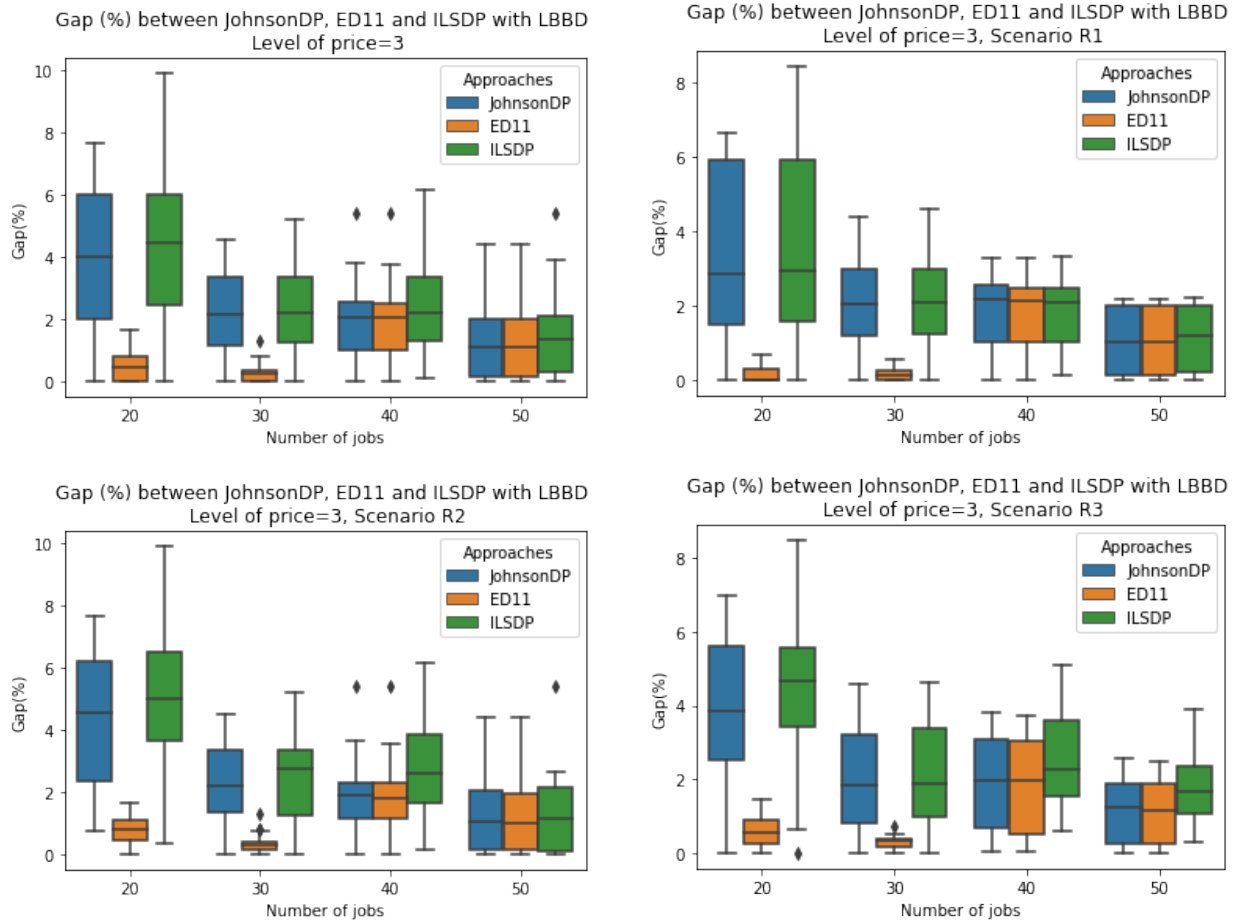


Figure 6: Gap (%) between JohnsonDP, ED11 and ILSDP with LBB for instances of stable electricity markets (level of price=3)

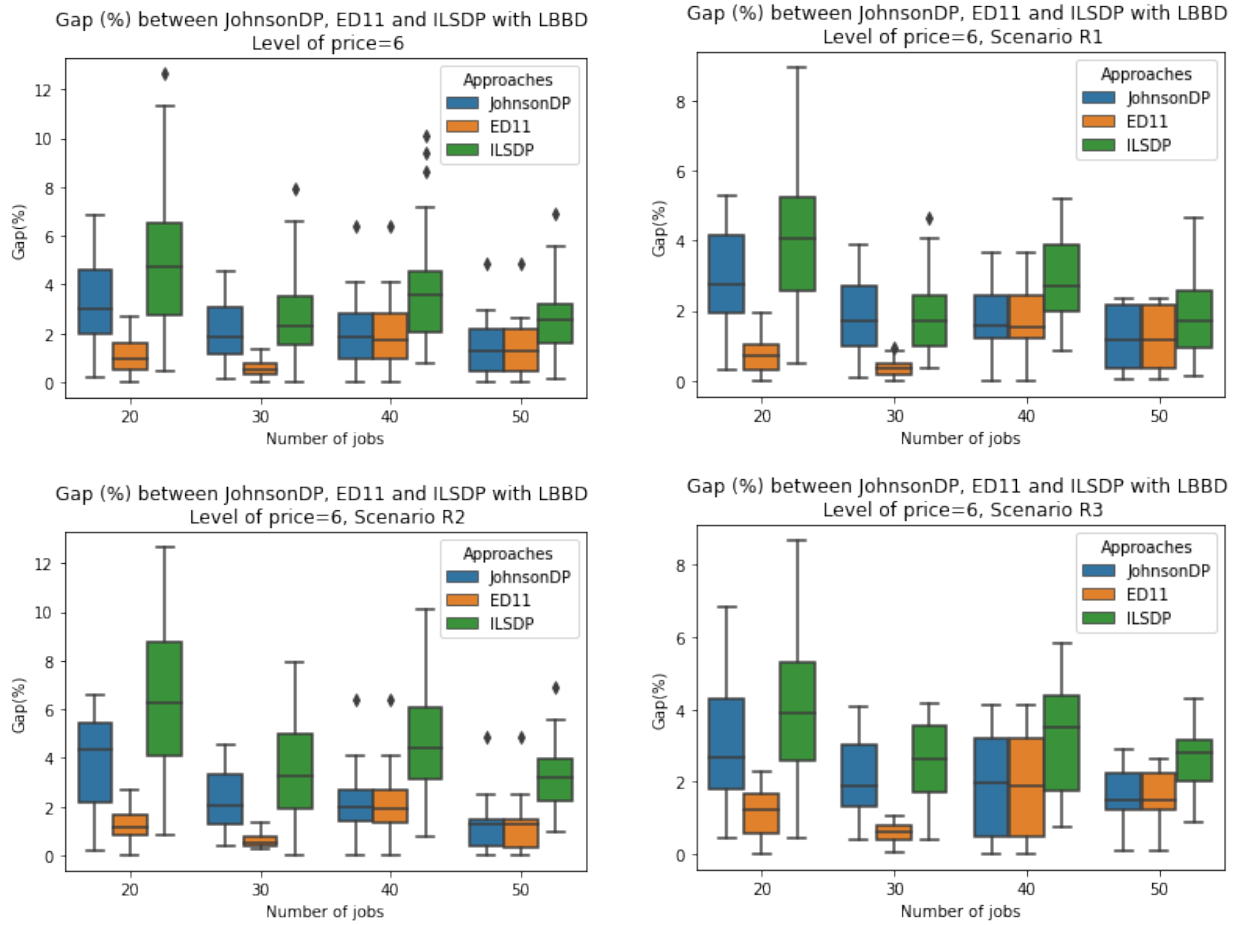


Figure 7: Gap (%) between JohnsonDP, ED11 and ILSDP with LBBD for instances of unstable electricity markets (level of price =6)

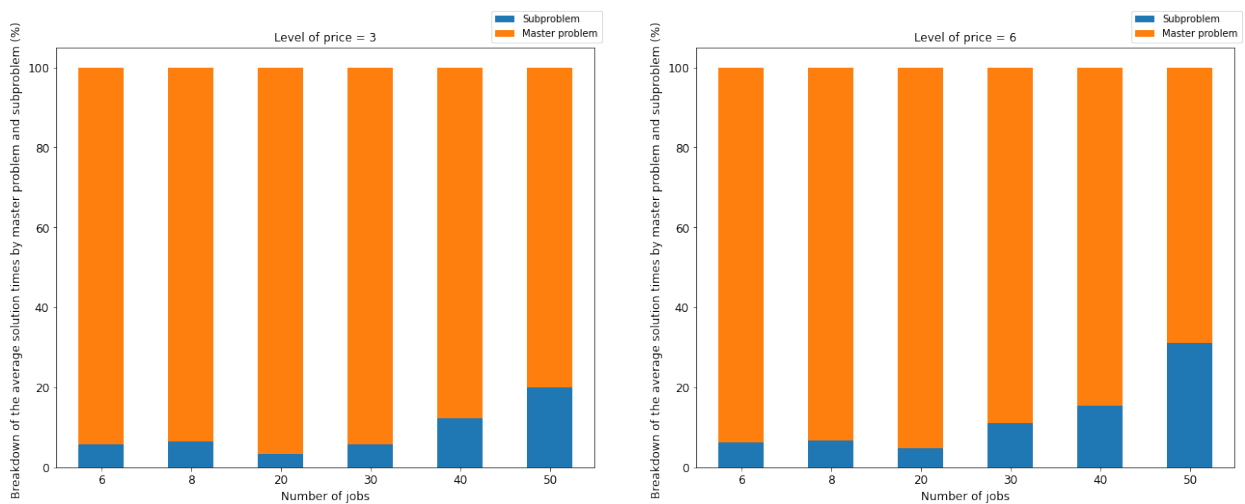


Figure 8: Breakdown of the average solution times (%) by MP and SP for the level of price = 3 and the level of price = 6.

Table 11: Breakdown of the average solution times by MP and SP (%) for each scenario.

Level of price	Scenario	Component	Number of jobs						
			6 jobs	8 jobs	20 jobs	30 jobs	40 jobs	50 jobs	
3	1	Master problem	94,68	93,34	98,96	98,16	89,23	92,85	
		Subproblem	5,32	6,66	1,04	1,84	10,77	7,15	
	2	Master problem	94,34	93,00	97,46	95,90	90,65	76,30	
		Subproblem	5,66	7,00	2,54	4,10	9,35	23,70	
	3	3	Master problem	94,02	94,39	98,59	97,65	83,60	71,04
			Subproblem	5,98	5,61	1,41	2,35	16,40	28,96
6	1	Master problem	93,35	93,92	98,75	94,70	88,37	73,59	
		Subproblem	6,65	6,08	1,25	5,30	11,63	26,41	
	2	Master problem	93,30	92,38	91,85	87,51	87,44	70,07	
		Subproblem	6,70	7,62	8,15	12,49	12,56	29,93	
	3	3	Master problem	94,77	94,46	94,82	84,30	78,11	63,16
			Subproblem	5,23	5,54	5,18	15,70	21,89	36,84

We also found that by changing the variables and reformulating the objective function, we significantly improved the performances of mixed-integer linear models. For large-size instances, we reduced the maximum gap to “BKS” from 60.76% of MC to 18.02% of M1. The manager or decision-makers can use our proposed approaches LBBD or the mathematical model M1 to improve the solution’s quality. For future work, it might be interesting to design and analyze a lower bound more performance for the problem in order to evaluate or improve the solution approaches. To better exploit our developed property, it is interesting to build a heuristic or a meta-heuristic integrating the property to reduce computation time. We can also consider the case of several machine states, several operating speeds that consume different amounts of energy to extend our mathematical model. A multi-objective optimization problem could be good to study the trade-off between productivity and the cost of electricity. A bi-objective study of the scheduling of a two-machine flow shop with the minimization of electricity costs and makespan offers managers more possibilities according to the Pareto front.

10. Acknowledgment

This work is co-funded by the French Ministry of Higher Education, Research and Innovation of France and European FEDER, started on October 2017.

Appendix A. Two pricing intervals

Johnson's solution has an interesting characteristic that plays an important role in this study: for any sequence π , we can always delay some jobs in Johnson's sequence to obtain $C_{2Johnson}^* = C_{2\pi}$ and $C_{1Johnson}^* = C_{1\pi}$. ($C_{2Johnson}^*$ and $C_{1Johnson}^*$ are the completion time on machine 2 and on machine 1 respectively of Johnson's sequence after being delayed).

We can demonstrate this characteristic as follows:

1. First of all, Johnson's rule provides the optimal solution for $F2|perm|C_{max}$ whose $C_{1Johnson} \leq C_{1\pi}$ and $C_{2Johnson} \leq C_{2\pi}$ for any sequence π . We can recognize that with two constant quantities $Q_{Johnson} = (C_{2Johnson} - C_{1Johnson})$ and $Q_{\pi} = (C_{2\pi} - C_{1\pi})$, we always have $Q_{Johnson} \leq Q_{\pi}$.
2. Then, we can always delay some jobs on machine 1 in Johnson's sequence to obtain $C_{1Johnson}^* = C_{1\pi}$. So we obtain new makespan value for Johnson's sequence

$$C_{2Johnson}^* = C_{1Johnson}^* + Q_{Johnson} \leq C_{1Johnson}^* + Q_{\pi} = C_{1\pi} + Q_{\pi} = C_{2\pi}$$

3. As $C_{2Johnson}^* \leq C_{2\pi}$, we can delay jobs on machine 2 to obtain $C_{2Johnson}^* = C_{2\pi}$.

Appendix A.1. The electricity prices decrease $fc_g > fc_{g+1}$

In this section, the planning horizon consists of two pricing intervals g and $(g+1)$. The interval g corresponds to peak period with the electricity price associated fc_g , the interval $(g+1)$ corresponds to off-peak period with the electricity price associated fc_{g+1} , $fc_g > fc_{g+1}$.

Let π be an optimal solution that does not follow the Johnson's rule. As demonstrated previously, we can always delay some jobs in Johnson's sequence to obtain $C_{2Johnson}^* = C_{2\pi}$ and $C_{1Johnson}^* = C_{1\pi}$.

Let Δ_1 and Δ_2 be "idle" time on machine 1 and on machine 2 respectively. We have $\Delta_1 = C_{1\pi} - \sum_{j \in J} p_{1j}$ and $\Delta_2 = C_{2\pi} - \sum_{j \in J} p_{2j}$. For any sequence which has the same $C_{2\pi}$ and $C_{1\pi}$ gets the same Δ_1 and Δ_2 . If $fc_g = fc_{g+1}$, we can get $TEC_{Johnson} = TEC_{\pi} = (\sum_{j \in J} p_{1j} \times b_1 + \sum_{j \in J} p_{2j} \times b_2) \times fc_g + (\Delta_1 \times d_1 + \Delta_2 \times d_2) \times fc_g$. In case of $fc_g \neq fc_{g+1}$, the difference of TEC arises on the difference of assignment of jobs processing workload on machine 2 in each pricing interval.

For instance, we consider the electricity cost on machine 1 of a sequence π : TEC_{π}^1 . Let $WL_{1,g}^{\pi}$ and $WL_{1,g+1}^{\pi}$ be jobs processing workload of machine 1 of sequence π in pricing interval g and $(g+1)$ respectively. Let $\Delta_{1,g}^{\pi}$ and $\Delta_{1,g+1}^{\pi}$ be the "idle" time of machine 1 of sequence π in pricing interval g and $(g+1)$ respectively.

$$TEC_{\pi}^1 = (WL_{1,g}^{\pi} \times b_1 + \Delta_{1,g}^{\pi} \times d_1) \times fc_g + (WL_{1,g+1}^{\pi} \times b_1 + \Delta_{1,g+1}^{\pi} \times d_1) \times fc_{g+1}$$

With:

- $WL_{1,g}^{\pi} + WL_{1,g+1}^{\pi} = \sum_{j \in J} p_{1j}$.
- $\Delta_{1,g}^{\pi} + \Delta_{1,g+1}^{\pi} = \Delta_1$.
- $WL_{1,g}^{\pi} + WL_{1,g+1}^{\pi} + \Delta_{1,g}^{\pi} + \Delta_{1,g+1}^{\pi} = \sum_{j \in J} p_{1j} + \Delta_1 = C_{1\pi}$

We can also obtain the electricity cost on machine 1 of any sequence π' whose $C_{1\pi'} = C_{1\pi}$ in a similar way.

$$TEC_{\pi'}^1 = (WL_{1,g}^{\pi'} \times b_1 + \Delta_{1,g}^{\pi'} \times d_1) \times fc_g + (WL_{1,g+1}^{\pi'} \times b_1 + \Delta_{1,g+1}^{\pi'} \times d_1) \times fc_{g+1}$$

With:

- $WL_{1,g}^{\pi'} + WL_{1,g+1}^{\pi'} = \sum_{j \in J} p_{1j}$.
- $\Delta_{1,g}^{\pi'} + \Delta_{1,g+1}^{\pi'} = \Delta_1$.
- $WL_{1,g}^{\pi'} + WL_{1,g+1}^{\pi'} + \Delta_{1,g}^{\pi'} + \Delta_{1,g+1}^{\pi'} = \sum_{j \in J} p_{1j} + \Delta_1 = C_{1\pi}$

By comparing the difference of electricity cost on machine 1 between π and π' , we can get :

$$\begin{aligned} TEC_{\pi}^1 - TEC_{\pi'}^1 &= [(WL_{1,g}^{\pi} - WL_{1,g}^{\pi'}) \times b_1 + (\Delta_{1,g}^{\pi} - \Delta_{1,g}^{\pi'}) \times d_1] \times fc_g \\ &\quad + [(WL_{1,g+1}^{\pi} - WL_{1,g+1}^{\pi'}) \times b_1 + (\Delta_{1,g+1}^{\pi} - \Delta_{1,g+1}^{\pi'}) \times d_1] \times fc_{g+1} \\ &= b_1 \times [(WL_{1,g}^{\pi} - WL_{1,g}^{\pi'}) \times fc_g + (WL_{1,g+1}^{\pi} - WL_{1,g+1}^{\pi'}) \times fc_{g+1}] \\ &\quad + d_1 \times [(\Delta_{1,g}^{\pi} - \Delta_{1,g}^{\pi'}) \times fc_g + (\Delta_{1,g+1}^{\pi} - \Delta_{1,g+1}^{\pi'}) \times fc_{g+1}] \end{aligned}$$

With:

- $(WL_{1,g}^{\pi} - WL_{1,g}^{\pi'}) + (WL_{1,g+1}^{\pi} - WL_{1,g+1}^{\pi'}) = 0$
- $(\Delta_{1,g}^{\pi} - \Delta_{1,g}^{\pi'}) + (\Delta_{1,g+1}^{\pi} - \Delta_{1,g+1}^{\pi'}) = 0$
- $(WL_{1,g}^{\pi} - WL_{1,g}^{\pi'}) + (WL_{1,g+1}^{\pi} - WL_{1,g+1}^{\pi'}) + (\Delta_{1,g}^{\pi} - \Delta_{1,g}^{\pi'}) + (\Delta_{1,g+1}^{\pi} - \Delta_{1,g+1}^{\pi'}) = 0$

Let $\gamma'_1 = (WL_{1,g+1}^{\pi} - WL_{1,g+1}^{\pi'})$ be the difference in term of processing time workload on machine 1 during off-peak pricing interval between the sequence π and π' . We can find that $-\gamma'_1 = (WL_{1,g}^{\pi} - WL_{1,g}^{\pi'})$.

Let $\gamma'_2 = (\Delta_{1,g}^{\pi} - \Delta_{1,g}^{\pi'})$ be the difference in term of ‘‘idle’’ time during peak pricing interval on machine 1 between the sequence π and π' . We can find that $-\gamma'_2 = (\Delta_{1,g+1}^{\pi} - \Delta_{1,g+1}^{\pi'})$.

Thus, the difference of electricity cost on machine 1 between π and π' can be formulated as follows:

$$\begin{aligned} TEC_{\pi}^1 - TEC_{\pi'}^1 &= [-\gamma'_1 \times b_1 + \gamma'_2 \times d_1] \times fc_g \\ &\quad + [\gamma'_1 \times b_1 + (-\gamma'_2) \times d_1] \times fc_{g+1} \\ &= -\gamma'_1 \times b_1 \times (fc_g - fc_{g+1}) + \gamma'_2 \times d_1 \times (fc_g - fc_{g+1}) \end{aligned}$$

Because $C_{1\pi'} = C_{1\pi}$, we can always delay or advance some jobs of π' on machine 1 to get $\gamma'_1 = 0$ and $\gamma'_2 = 0$. Then $TEC_{\pi}^1 = TEC_{\pi'}^1$ for any sequence $\pi' | C_{1\pi'} = C_{1\pi}$.

Because $TEC_{\pi}^1 = TEC_{\pi'}^1$ for any sequence $\pi' | C_{1\pi'} = C_{1\pi}$, so the difference of TEC arises only on the difference of assignment of jobs processing workload on machine 2 in each pricing interval.

We compare now the difference of TEC between π and Johnson's solution whose $C_{2Johnson}^* = C_{2\pi}$ and $C_{1Johnson}^* = C_{1\pi}$. As demonstrated previously, $TEC_{\pi}^1 - TEC_{Johnson}^1 = 0$.

$$\begin{aligned} TEC_{\pi} - TEC_{Johnson} &= TEC_{\pi}^2 - TEC_{Johnson}^2 + TEC_{\pi}^1 - TEC_{Johnson}^1 \\ &= TEC_{\pi}^2 - TEC_{Johnson}^2 \\ &= -\gamma_1 \times b_2 \times (fc_g - fc_{g+1}) + \gamma_2 \times d_2 \times (fc_g - fc_{g+1}) \\ \Leftrightarrow TEC_{Johnson} - TEC_{\pi} &= \gamma_1 \times b_2 \times (fc_g - fc_{g+1}) - \gamma_2 \times d_2 \times (fc_g - fc_{g+1}) \end{aligned}$$

Where:

- γ_1 is the difference of processing time workload on machine 2 during off-peak pricing interval between π and Johnson's sequence. $\gamma_1 = (WL_{2,g+1}^{\pi} - WL_{2,g+1}^{Johnson})$.
- γ_2 is the difference of "idle" time during peak pricing interval between π and Johnson's sequence. $\gamma_2 = (\Delta_{2,g}^{\pi} - \Delta_{2,g}^{Johnson})$

As Johnson's solution has the smallest total "idle" time on machine 2 comparing to any other sequences (Johnson (1954)), then $\gamma_2 = (\Delta_{2,g}^{\pi} - \Delta_{2,g}^{Johnson}) \geq 0$. We can always delay some jobs of Johnson's sequence to obtain $\gamma_1 = 0$, which means the processing times in off-peak and peak period are the same between Johnson's sequence and π . As the electricity prices are decreasing ($fc_g - fc_{g+1} \geq 0$), that leads to $TEC_{Johnson} - TEC_{\pi} \leq 0$. So Johnson's rule gives the optimal solution in terms of total electricity cost.

Appendix A.2. The electricity prices increase $fc_g < fc_{g+1}$

In this section, the planning horizon consists of two pricing intervals g and $(g+1)$. The interval g corresponds to off-peak period with the electricity price associated fc_g , the interval $(g+1)$ corresponds to peak period with the electricity price associated fc_{g+1} , $fc_g < fc_{g+1}$.

We compare the total electricity cost between the Johnson sequence ($TEC_{Johnson}$) and a sequence π (TEC_{π}). The sequence π is obtained by interchanging any two jobs J_k and J_j in Johnson sequence. We consider two possible cases as follows:

- 1st case: $J_k, J_j \in \bar{J}$ or $J_k, J_j \in \underline{J}$. We have $TEC_{Johnson} \leq TEC_{\pi}$. Johnson's rule gives the optimal sequence because any interchange between J_k and J_j leads to more "idle time" or less exploiting the off-peak periods.
- 2nd case, $J_k \in \bar{J}$ and $J_j \in \underline{J}$.
 - If $p_{1J_j} > p_{1J_k}$, Johnson's rule dominates because any interchange between J_k et J_j leads to more "idle time" in off-peak interval.

- If $p_{1J_j} \leq p_{1J_k}$, we can find in figures A.9a and A.9b an example for interchanging J_k and J_j of sequence π . We note the inactive period immediately before J_k on the machine 1 and on the machine 2 are Δ_1 and Δ_2 respectively. Let δ_1 be the “idle time” on machine 2 of Johnson’s sequence and δ_2 be the “idle time” on machine 2 of sequence π .

We have:

$$\delta_1 = \Delta_1 + p_{1J_j} - \Delta_2$$

To estimate δ_2 , let J_i be the job succeeding directly the job J_k on Johnson’s sequence. We consider two configurations.

If $J_i \in \bar{J}$, according to Johnson’s rule $p_{1J_i} \geq p_{1J_k}$. In this case, we have:

$$\begin{aligned} \delta_2 &= \Delta_1 + p_{1J_j} + p_{1J_i} - (\Delta_2 + p_{2J_j}) \\ &\geq \Delta_1 + p_{1J_j} + p_{1J_k} - (\Delta_2 + p_{2J_j}) \end{aligned}$$

If $J_i \in \underline{J}$, J_i precedes J_j , so $p_{2J_k} \geq p_{2J_j}$ conforming to Johnson’s rule. In this case, if $p_{1J_i} \leq p_{1J_k}$ so the interchange between J_i and J_k leads to less “idle time” than the interchange between J_k and J_j . Thus, it is better to study the interchange between J_i and J_k . We interchange J_i with J_k . With a change of index from i to j , we have:

$$\delta_2 = \Delta_1 + p_{1J_j} + p_{1J_k} - (\Delta_2 + p_{2J_j})$$

So in all the cases, we have:

$$\delta_2 \geq \Delta_1 + p_{1J_j} + p_{1J_k} - (\Delta_2 + p_{2J_j})$$

We have:

$$\begin{aligned} \delta_2 - \delta_1 &\geq \Delta_1 + p_{1J_j} + p_{1J_k} - (\Delta_2 + p_{2J_j}) - (\Delta_1 + p_{1J_k} - \Delta_2) \\ &\geq p_{1J_j} - p_{2J_j} \end{aligned}$$

Let TEC_π and $TEC_{Johnson}$ be the total electricity cost of sequence π and *Johnson* respectively. Let “Gain” be the best gain by interchanging J_j and J_k :

$$Gain = \min(\delta_1, p_{2J_j}) \times b_2 \times (fc_{g+1} - fc_g)$$

where b_2 is the amount of energy consumption per unit of time of machine 2 when it processes jobs.

“Gain” represents the gain in electricity cost by processing a quantity of job equal to $\min(\delta_1, p_{2J_j})$ during off-peak periods instead of processing them during peak periods.

Let “Loss” be the minimal loss, that means the minimal rise in electricity cost comparing to Johnson’s sequence:

$$\begin{aligned} Loss &\geq (\delta_2 - \delta_1) \times d_2 \times fc_g \\ &\geq (p_{1J_j} - p_{2J_j}) \times d_2 \times fc_g \end{aligned}$$

where d_2 is the amount of energy consumption per unit time of machine 2 when it is on “idle”.

“Loss” represents the rise in electricity cost. This rise is in the function of the difference in “idle” times by interchanging J_j and J_k . Additionally, we assume that these “idle” time increased are located on off-peak interval.

We have:

$$\begin{aligned} TEC_{Johnson} - TEC_{\pi} &\leq Gain - Loss \\ &\leq \min(\delta_1, p_{2J_j}) \times b_2 \times (fc_{g+1} - fc_g) - (p_{1J_j} - p_{2J_j}) \times d_2 \times fc_g \end{aligned}$$

Let $A = Gain - Loss$. If $A \leq 0$, Johnson’s sequence dominates any others sequences in term of total electricity cost:

$$\begin{aligned} &\min(\delta_1, p_{2J_j}) \times b_2 \times (fc_{g+1} - fc_g) - (p_{1J_j} - p_{2J_j}) \times d_2 \times fc_g \leq 0 \\ \Leftrightarrow &\min(\delta_1, p_{2J_j}) \times b_2 \times (fc_{g+1} - fc_g) \leq (p_{1J_j} - p_{2J_j}) \times d_2 \times fc_g \\ \Leftrightarrow &\frac{b_2 \times (fc_{g+1} - fc_g)}{d_2 \times fc_g} \leq \frac{p_{1J_j} - p_{2J_j}}{\min(\delta_1, p_{2J_j})} \end{aligned}$$

We denote job J_j by j to simplify the notation. Then in the cases of two pricing intervals and the electricity prices are increasing, the Johnson’s rule gives the optimal solution if

$$\frac{b_2 \times (fc_{g+1} - fc_g)}{d_2 \times fc_g} \leq \frac{p_{1j} - p_{2j}}{\min(\delta_1, p_{2j})}, \forall j \in \underline{\mathbf{J}}$$

Appendix B. Property based on Johnson’s rule

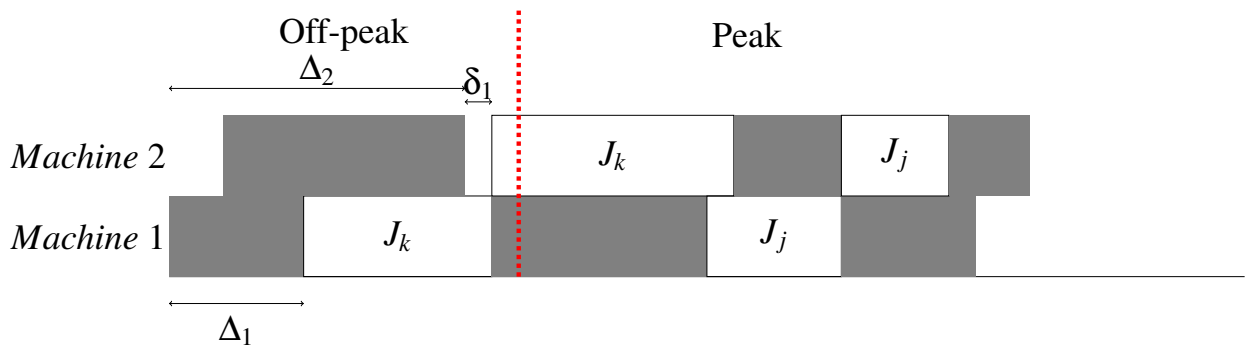
Appendix B.1. Two pricing intervals g and $g + 1$

We study a sequence π' whose $J_g \subset J$ and $J_{g+1} \subset J$. We prove that reorder jobs of the subset J_g and jobs of the subset J_{g+1} according to Johnson’s rule do not increase the total electricity cost.

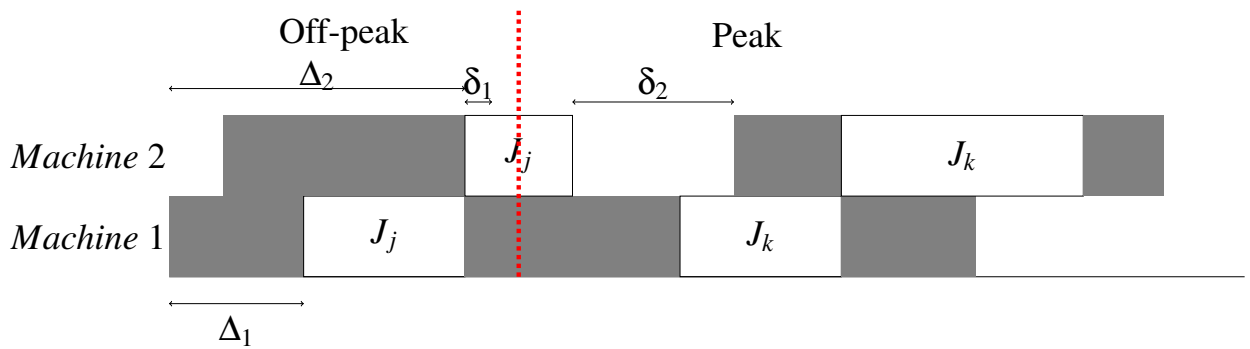
Appendix B.1.1. The set J_g

We have:

- J_g : set of jobs whose starting times on machine 2 are executed during pricing interval g .
- $S_{i\pi}$: starting time on machine i of sequence π .
- $C_{i\pi}$: completion time on machine i of sequence π .
- $\Delta_{i,g}^{\pi}$: total idle time of machine i of sequence π during pricing interval g .
- $WL_{i,g}^{\pi}$: total processing time workload of machine i of sequence π during pricing interval g .



(a) Johnson's sequence



(b) The sequence π

Figure A.9: The comparison of interchange between J_k et J_j

Let π^* be the optimal order for jobs belonging to E_p .

As all jobs j of J_g start on machine 2 during pricing period g , we have:

$$\Delta_{1,g+1}^{\pi^*} = 0. \quad (\text{B.1})$$

$$WL_{1,g}^{\pi^*} = \sum_{j \in J_g} p_{1j}. \quad (\text{B.2})$$

$$\Delta_{1,g}^{\pi^*} + WL_{1,g}^{\pi^*} = C_{1\pi^*}. \quad (\text{B.3})$$

$$WL_{1,g+1}^{\pi^*} = 0. \quad (\text{B.4})$$

$$\Delta_{2,g+1}^{\pi^*} = 0. \quad (\text{B.5})$$

$$WL_{2,g+1}^{\pi^*} = \max(0, C_{2\pi^*} - ED_g). \quad (\text{B.6})$$

$$WL_{2,g}^{\pi^*} + WL_{2,g+1}^{\pi^*} = \sum_{j \in J_g} p_{2j}. \quad (\text{B.7})$$

$$ST_g + \Delta_{2,g}^{\pi^*} + WL_{2,g}^{\pi^*} + WL_{2,g+1}^{\pi^*} = C_{2\pi^*}. \quad (\text{B.8})$$

Jobs belonging to J_g start on machine 2 during pricing interval g , that is why their processing on machine 1 must be finished before the ending time of interval g . Equations (B.1), (B.2) and (B.4) represent that constraint.

The last job of sequence π^* starts on machine 2 during pricing interval g so the job can be finished either on interval g ($WL_{2,g+1}^{\pi^*} = 0$) either on interval $g+1$ ($WL_{2,g+1}^{\pi^*} = C_{2\pi^*} - ED_g$). We always have $\Delta_{2,g+1}^{\pi^*} = 0$. Equations (B.5) and (B.6) represent that constraint.

We have the total electricity cost of sequence π^* :

$$\begin{aligned} TEC_{\pi^*} &= \sum_{i \in M} \Delta_{i,g}^{\pi^*} \times d_i \times fc_g + \sum_{i \in M} WL_{i,g}^{\pi^*} \times b_i \times fc_g \\ &+ \sum_{i \in M} \Delta_{i,g+1}^{\pi^*} \times d_i \times fc_{g+1} + \sum_{i \in M} WL_{i,g+1}^{\pi^*} \times b_i \times fc_{g+1} \end{aligned}$$

Then we consider a sequence π of J_g that order of jobs follows Johnson's rule.

Equations (B.1)-(B.8) are still valid for sequence π . As π follows Johnson's rule, we also have:

$$C_{1\pi} \leq C_{1\pi^*}. \quad (\text{B.9})$$

$$C_{2\pi} \leq C_{2\pi^*}. \quad (\text{B.10})$$

From equation (B.10), by delaying some jobs on machine 2, we are able to obtain $C_{2\pi} = C_{2\pi^*}$. Then, from equation (B.6), we get: $WL_{2,g+1}^{\pi} = WL_{2,g+1}^{\pi^*} = \max(0, C_{2\pi^*} - ED_g)$. With equations (B.7) and (B.8), we obtain also: $\Delta_{2,g}^{\pi} = \Delta_{2,g}^{\pi^*}$ and $WL_{2,g}^{\pi} = WL_{2,g}^{\pi^*}$.

Equations (B.2), (B.3) and (B.9), imply $WL_{1,g}^{\pi} = WL_{1,g}^{\pi^*} = \sum_{j \in J_g} p_{1j}$ and $\Delta_{1,g}^{\pi} \leq \Delta_{1,g}^{\pi^*}$.

So with any sequence π following Johnson's rule, we can obtain:

$$\begin{aligned} TEC_{\pi} &= \sum_{i \in M} \Delta_{i,g}^{\pi} \times d_i \times fc_g + \sum_{i \in M} WL_{i,g}^{\pi} \times b_i \times fc_g \\ &\quad + \sum_{i \in M} \Delta_{i,g+1}^{\pi} \times d_i \times fc_{g+1} + \sum_{i \in M} WL_{i,g+1}^{\pi} \times b_i \times fc_{g+1}. \\ &\leq TEC_{\pi^*}. \end{aligned}$$

We take an example illustrated in figure (A.9). We see that total idle time and total processing time on each machine during each pricing interval are the same between π^* and π , so $TEC_{\pi} = TEC_{\pi^*}$

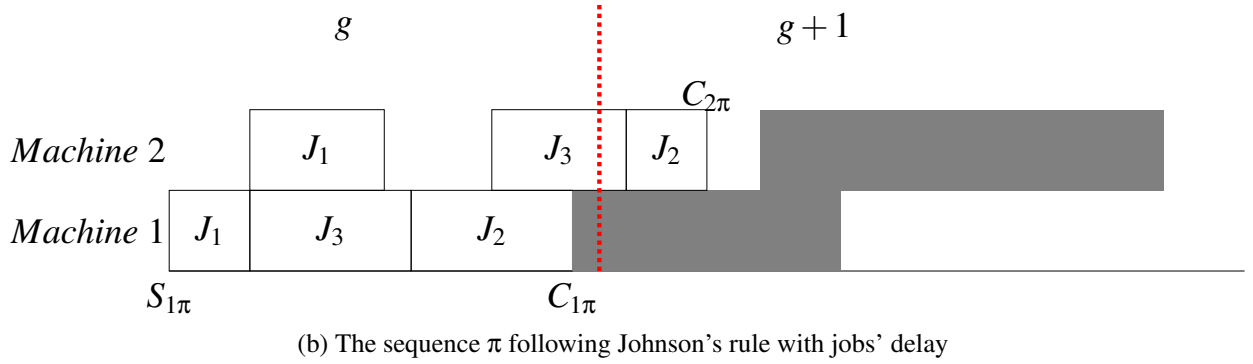
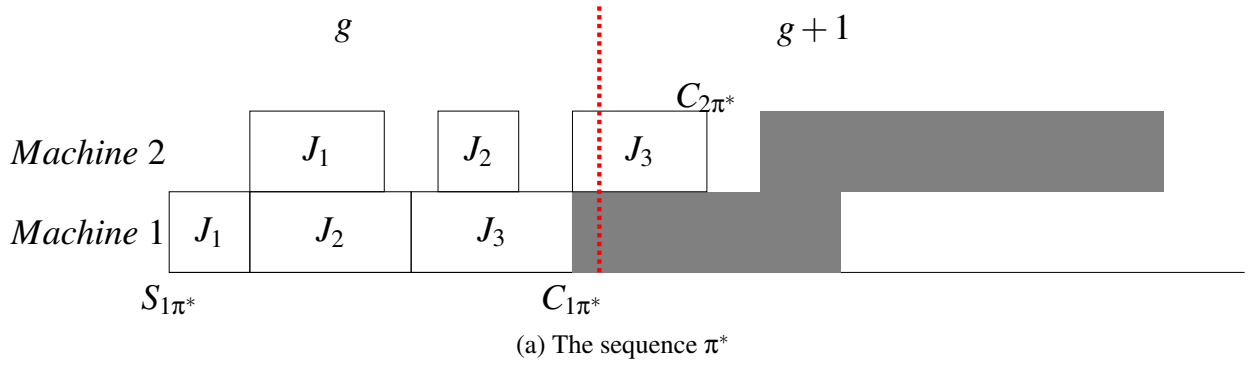


Figure B.10: The comparison between π^* and Johnson's sequence π

Appendix B.1.2. The set J_{g+1}

Let π^* be an optimal sequence for J_{g+1} . As all jobs of J_{g+1} start on machine 2 during pricing period $g + 1$, we have:

$$\Delta_{2,g}^{\pi^*} = 0. \quad (\text{B.11})$$

$$WL_{2,g}^{\pi^*} = 0. \quad (\text{B.12})$$

$$WL_{2,g+1}^{\pi^*} = \sum_{j \in J_{g+1}} p_{2j}. \quad (\text{B.13})$$

$$WL_{1,g}^{\pi^*} = \max(0, ED_g - S_{1\pi^*} - \Delta_{1,g}^{\pi^*}). \quad (\text{B.14})$$

$$WL_{1,g}^{\pi^*} + WL_{1,g+1}^{\pi^*} = \sum_{j \in J_{g+1}} p_{1j}. \quad (\text{B.15})$$

$$S_{1\pi^*} + \Delta_{1,g}^{\pi^*} + \Delta_{1,g+1}^{\pi^*} + WL_{1,g}^{\pi^*} + WL_{1,g+1}^{\pi^*} = C_{1\pi^*}. \quad (\text{B.16})$$

As all jobs of J_{g+1} start on machine 2 during pricing interval $g + 1$, the machine 2 does not process any job of J_{g+1} during interval g . Equations (B.11), (B.12) and (B.13) represent this constraint.

As all jobs of J_{g+1} start on machine 2 during pricing interval $g + 1$, thus on machine 1, either all jobs start during interval $g + 1$ ($WL_{1,g}^{\pi^*} = 0$) either at least one job start during interval g ($WL_{1,g}^{\pi^*} = ED_g - S_{1\pi^*} - \Delta_{1,g}^{\pi^*}$).

Equations (B.15) and (B.16) represent the workload on machine 1.

In equation (B.14), if $\Delta_{1,g}^{\pi^*} \neq 0$, there are at least one job start and finish during pricing interval g . Delay these jobs to obtain $\Delta_{1,g}^{\pi^*} = 0$ will have no impact neither on $C_{1\pi^*}$ nor on $C_{2\pi^*}$. Let $S_{1\pi^*}^* = S_{1\pi^*} + \Delta_{1,g}^{\pi^*}$.

So we have:

$$\Delta_{2,g}^{\pi^*} = 0. \quad (\text{B.17})$$

$$WL_{2,g}^{\pi^*} = 0. \quad (\text{B.18})$$

$$WL_{2,g+1}^{\pi^*} = \sum_{j \in E_{g+1}} p_{2j}. \quad (\text{B.19})$$

$$WL_{1,g}^{\pi^*} = \max(0, ED_g - S_{1\pi^*}^*). \quad (\text{B.20})$$

$$\Delta_{1,g}^{\pi^*} = A. \quad (\text{B.21})$$

With A is constant.

$$WL_{1,g}^{\pi^*} + WL_{1,g+1}^{\pi^*} = \sum_{j \in J_{g+1}} p_{1j}. \quad (\text{B.22})$$

$$S_{1\pi^*}^* + \Delta_{1,g+1}^{\pi^*} + WL_{1,g}^{\pi^*} + WL_{1,g+1}^{\pi^*} = C_{1\pi^*}. \quad (\text{B.23})$$

Let π be a sequence following Johnson's rule and $S_{1\pi^*}^*$ be the starting time on machine 1 of this sequence.

Equations (B.17)-(B.23) are still valid for sequence π . As π follows Johnson's rule, we have also:

$$C_{1\pi} \leq C_{1\pi^*}. \quad (\text{B.24})$$

$$C_{2\pi} \leq C_{2\pi^*}. \quad (\text{B.25})$$

From equation (B.25), by delaying some jobs on machine 1, we can obtain $C_{1\pi} = C_{1\pi^*}$ and $S_{1\pi}^* = S_{1\pi^*}$.

Thus, we have:

$$\Delta_{2,g}^\pi = \Delta_{2,g}^{\pi^*} = 0. \quad (\text{B.26})$$

$$WL_{2,g}^\pi = WL_{2,g}^{\pi^*} = 0. \quad (\text{B.27})$$

$$WL_{2,g+1}^\pi = WL_{2,g+1}^{\pi^*} = \sum_{j \in J_{g+1}} p_{2j}. \quad (\text{B.28})$$

$$\Delta_{2,g+1}^\pi \leq \Delta_{2,g+1}^{\pi^*}. \quad (\text{B.29})$$

$$WL_{1,g}^\pi = WL_{1,g}^{\pi^*} = \max(0, ED_g - S_{1\pi^*}^*). \quad (\text{B.30})$$

$$\Delta_{1,g}^\pi = \Delta_{1,g}^{\pi^*} = A. \quad (\text{B.31})$$

Where A is constant.

$$WL_{1,g}^\pi + WL_{1,g+1}^\pi = \sum_{j \in J_{g+1}} p_{1j}. \quad (\text{B.32})$$

$$S_{1\pi^*}^* + \Delta_{1,g+1}^\pi + WL_{1,g}^\pi + WL_{1,g+1}^\pi = C_{1\pi^*}. \quad (\text{B.33})$$

So with any sequence π following Johnson's rule, we can obtain:

$$\begin{aligned} TEC_\pi &= \sum_{i \in M} \Delta_{i,g}^\pi \times d_i \times fc_g + \sum_{i \in M} WL_{i,g}^\pi \times b_i \times fc_g \\ &+ \sum_{i \in M} \Delta_{i,g+1}^\pi \times d_i \times fc_{g+1} + \sum_{i \in M} WL_{i,g+1}^\pi \times b_i \times fc_{g+1}. \\ &\leq TEC_{\pi^*}. \end{aligned}$$

We take an example illustrated in figure (B.10). We see that the total idle time and the total processing time on each machine during each pricing interval are the same between π^* and π , so $TEC_\pi = TEC_{\pi^*}$.

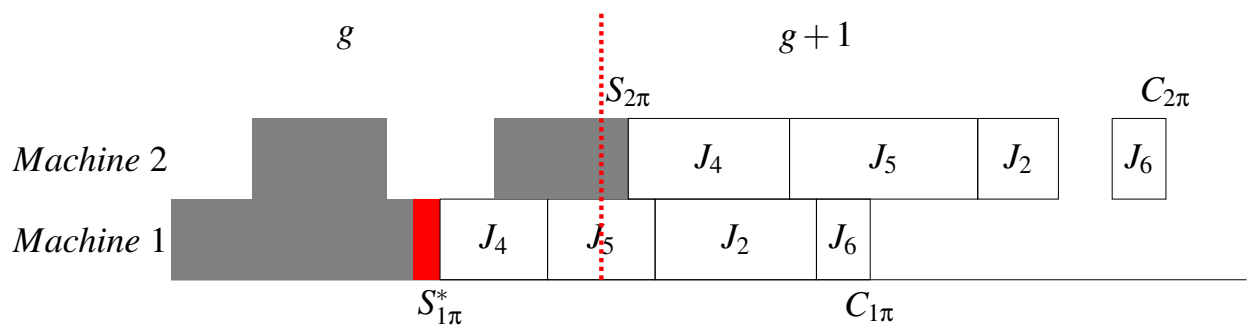
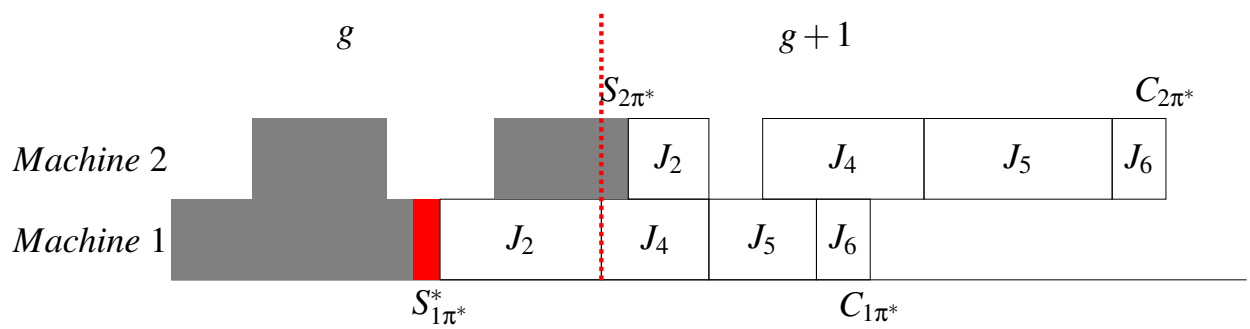
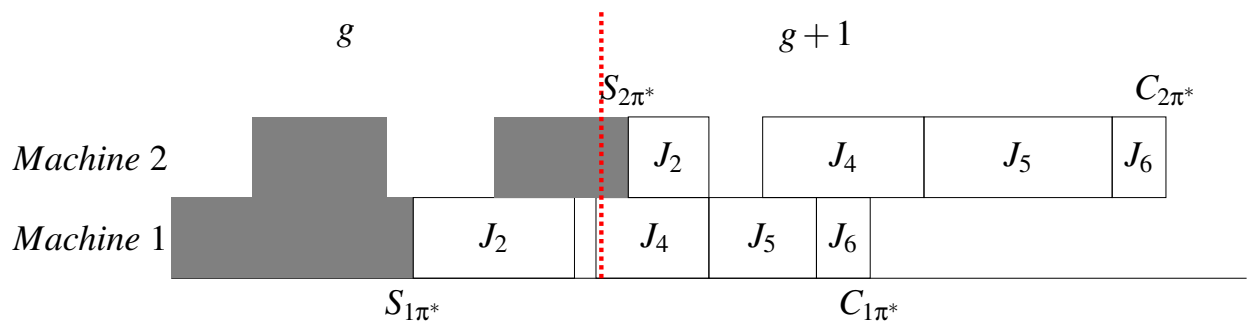


Figure B.11: The comparison between the sequence π^* and Johnson's sequence π

Appendix B.2. Three pricing intervals $(g-1), g$ and $(g+1)$

We consider in this section a sequence π^* of the ensemble $J_g \subset J$. Jobs in J_g start on machine 2 during pricing interval g . The first job of π^* starts on machine 1 during interval $(g-1)$ and the last job of π^* finishes during interval $(g+1)$.

We prove that for any sequence π^* , reorder jobs according to Johnson's rule will not increase the total electricity cost.

Base on previous sections (Appendix B.1.1) and (Appendix B.1.2), we have total workload and total idle time allocated to each pricing interval as follows:

$$\Delta_{2,g-1}^{\pi^*} = 0. \quad (\text{B.34})$$

$$\Delta_{2,g+1}^{\pi^*} = 0. \quad (\text{B.35})$$

$$WL_{2,g-1}^{\pi^*} = 0. \quad (\text{B.36})$$

$$WL_{2,g+1}^{\pi^*} = \max(0, C_{2\pi^*} - ED_g). \quad (\text{B.37})$$

$$WL_{2,g}^{\pi^*} + WL_{2,g+1}^{\pi^*} = \sum_{j \in J_g} p_{2j}. \quad (\text{B.38})$$

$$ST_g + \Delta_{2,g}^{\pi^*} + WL_{2,g}^{\pi^*} + WL_{2,g+1}^{\pi^*} = C_{2\pi^*}. \quad (\text{B.39})$$

$$\Delta_{1,g+1}^{\pi^*} = 0. \quad (\text{B.40})$$

$$\Delta_{1,g-1}^{\pi^*} = A. \quad (\text{B.41})$$

Where A is constant.

$$WL_{1,g+1}^{\pi^*} = 0. \quad (\text{B.42})$$

$$WL_{1,g-1}^{\pi^*} = \max(0, ED_{g-1} - S_{1\pi^*}^*). \quad (\text{B.43})$$

$$WL_{1,g-1}^{\pi^*} + WL_{1,g}^{\pi^*} = \sum_{j \in J_g} p_{1j}. \quad (\text{B.44})$$

$$S_{1\pi^*} + \Delta_{1,g-1}^{\pi^*} + \Delta_{1,g}^{\pi^*} + WL_{1,g-1}^{\pi^*} + WL_{1,g}^{\pi^*} = C_{1\pi^*}. \quad (\text{B.45})$$

We can estimate total electricity cost of sequence π^* as follows:

$$\begin{aligned}
TEC_{\pi^*} &= \sum_{i \in M} \Delta_{i,g-1}^{\pi^*} \times d_i \times fc_{(g-1)} + \sum_{i \in M} WL_{i,g-1}^{\pi^*} \times b_i \times fc_{(g-1)} \\
&+ \sum_{i \in M} \Delta_{i,g}^{\pi^*} \times d_i \times fc_g + \sum_{i \in M} WL_{i,g}^{\pi^*} \times b_i \times fc_g. \\
&+ \sum_{i \in M} \Delta_{i,g+1}^{\pi^*} \times d_i \times fc_{(g+1)} + \sum_{i \in M} WL_{i,g+1}^{\pi^*} \times b_i \times fc_{(g+1)}.
\end{aligned}$$

Let π be the sequence of J_g following Johnson's rule and $S_{1\pi^*}$ be the starting time on machine 1 of sequence π .

Equations (B.34)-(B.45) are still valid for sequence π . Additionally, as π follows Johnson's rule, we also have:

$$C_{1\pi} \leq C_{1\pi^*}. \quad (\text{B.46})$$

$$C_{2\pi} \leq C_{2\pi^*}. \quad (\text{B.47})$$

As representing in equations (B.46) and (B.47), with π , we can delay jobs in this sequence to obtain $C_{1\pi} = C_{1\pi^*}$ and $C_{2\pi} = C_{2\pi^*}$. Thus, we can get:

$$\Delta_{2,g-1}^{\pi} = \Delta_{2,g-1}^{\pi^*} = 0. \quad (\text{B.48})$$

$$\Delta_{2,g+1}^{\pi} = \Delta_{2,g+1}^{\pi^*} = 0. \quad (\text{B.49})$$

$$WL_{2,g-1}^{\pi} = WL_{2,g-1}^{\pi^*} = 0. \quad (\text{B.50})$$

$$WL_{2,g+1}^{\pi} = WL_{2,g+1}^{\pi^*} = \max(0, C_{2\pi^*} - ED_g). \quad (\text{B.51})$$

$$WL_{2,g}^{\pi} = WL_{2,g}^{\pi^*} = \sum_{j \in J_g} p_{2j} - \max(0, C_{2\pi^*} - ED_g). \quad (\text{B.52})$$

$$\Delta_{2,g}^{\pi} = \Delta_{2,g}^{\pi^*} = C_{2\pi^*} - WL_{2,g+1}^{\pi^*} - WL_{2,g}^{\pi^*} - ST_g. \quad (\text{B.53})$$

$$\Delta_{1,g+1}^{\pi} = \Delta_{1,g+1}^{\pi^*} = 0. \quad (\text{B.54})$$

$$\Delta_{1,g-1}^{\pi} = \Delta_{1,g-1}^{\pi^*} = A. \quad (\text{B.55})$$

where A is constant.

$$WL_{1,g+1}^{\pi} = WL_{1,g+1}^{\pi^*} = 0. \quad (\text{B.56})$$

$$WL_{1,g-1}^\pi = WL_{1,g-1}^{\pi^*} = \max(0, ED_{g-1} - S_{1\pi^*}^*). \quad (\text{B.57})$$

$$WL_{1,g}^\pi = WL_{1,g}^{\pi^*} = \sum_{j \in J_g} p_{1j} - \max(0, ED_{g-1} - S_{1\pi^*}^*). \quad (\text{B.58})$$

$$\Delta_{1,g}^\pi = \Delta_{1,g}^{\pi^*} = C_{1\pi^*} - S_{1\pi^*}^* - \Delta_{1,g-1}^{\pi^*} - WL_{1,g-1}^{\pi^*} - WL_{1,g}^{\pi^*}. \quad (\text{B.59})$$

With equations (B.48)-(B.59), we can obtain $TEC_\pi = TEC_{\pi^*}$.

Thus, for any sequence π^* whose jobs' starting time on machine 2 begin during a same pricing interval, we can get a sequence π following Johnson's rule and possesses a same total electricity cost.

We provide an example illustrated on figure (B.12). We see that the total idle time and the total processing time on each machine during each pricing interval $g-1$, g and $g+1$ are the same between π^* and π , so $TEC_\pi = TEC_{\pi^*}$.

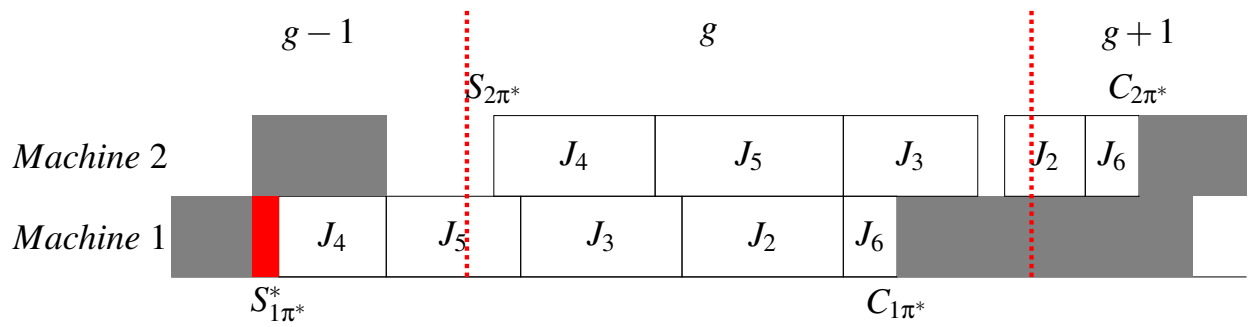
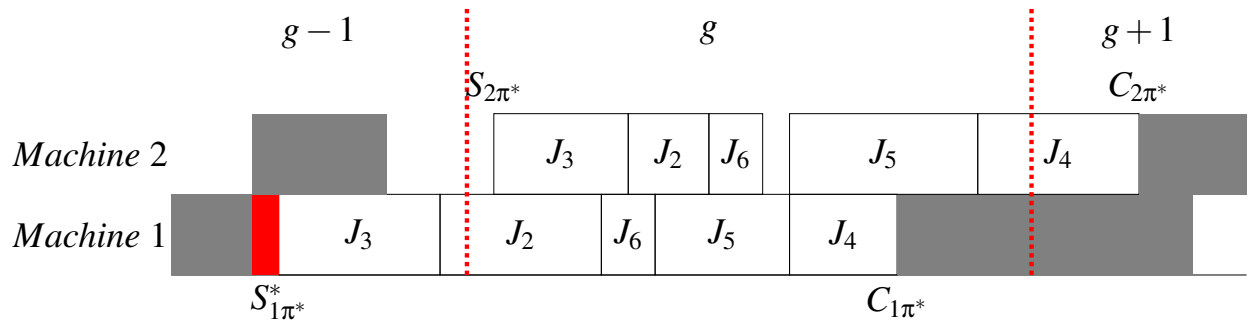
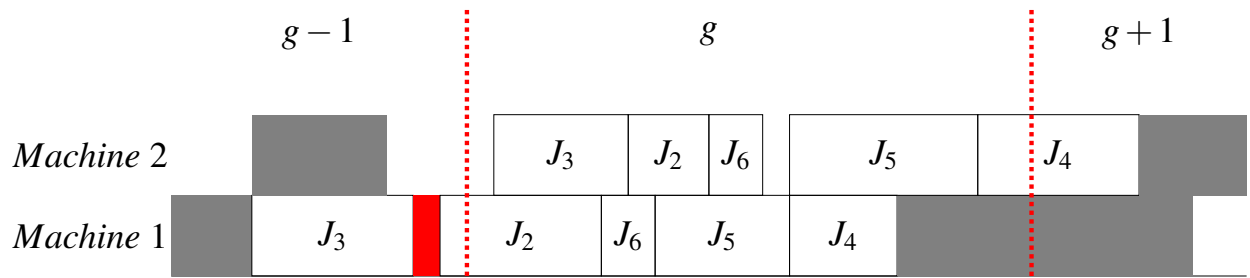


Figure B.12: The comparison between π^* and Johnson's sequence π

References

- Aghelinejad, M., Ouazene, Y., Yalaoui, A., 2018a. Production scheduling optimisation with machine state and time-dependent energy costs. *International Journal of Production Research* 56, 5558–5575.
- Aghelinejad, M., Ouazene, Y., Yalaoui, A., 2019. Complexity analysis of energy-efficient single machine scheduling problems. *Operations Research Perspectives* 6, 100105.
- Aghelinejad, M.M., Ouazene, Y., Yalaoui, A., 2018b. Energy efficient scheduling problems under time-of-use tariffs with different energy consumption of the jobs. *IFAC-PapersOnLine* 51, 1053–1058.
- Akpan, U.F., Akpan, G.E., 2012. The contribution of energy consumption to climate change: A feasible policy direction. *International Journal of Energy Economics and Policy* 2, 21–33.
- Albadi, M.H., El-Saadany, E.F., 2008. A summary of demand response in electricity markets. *Electric power systems research* 78, 1989–1996.
- Badri, S.A., Daghdandan, A., Aghabeiginiyay Fatalaki, Z., Mirzazadeh, M., 2021. Flow shop scheduling under time-of-use electricity tariffs using fuzzy multi-objective linear programming approach. *Journal of Mathematical Modeling* 9, 215–227.
- Benders, J.F., 1962. Partitioning procedures for solving mixed-variables programming problems. *Numerische mathematik* 4, 238–252.
- Braithwait, S., Hansen, D., O’Sheasy, M., 2007. Retail electricity pricing and rate design in evolving markets. *Edison Electric Institute* , 1–57.
- Che, A., Wu, X., Peng, J., Yan, P., 2017. Energy-efficient bi-objective single-machine scheduling with power-down mechanism. *Computers & Operations Research* 85, 172–183.
- Chen, B., Zhang, X., 2019. Scheduling with time-of-use costs. *European Journal of Operational Research* 274, 900–908.
- Chen, L., Megow, N., Rischke, R., Stougie, L., Verschae, J., 2018. Optimal algorithms for scheduling under time-of-use tariffs. *arXiv preprint arXiv:1811.12657* .
- Chen, L., Wang, J., Xu, X., 2019. An energy-efficient single machine scheduling problem with machine reliability constraints. *Computers & Industrial Engineering* 137, 106072.
- COP21, 2016. Paris agreement. https://ec.europa.eu/clima/policies/international/negotiations/paris_entab-0-0.
- Cui, W., Lu, B., 2021. Energy-aware operations management for flow shops under tou electricity tariff. *Computers & Industrial Engineering* 151, 106942.
- Dabur, P., Singh, G., Yadav, N.K., 2012. Electricity demand side management: Various concept and prospects. *International Journal of Recent Technology and Engineering (IJRTE)* 1, 1–6.
- Dai, M., Tang, D., Giret, A., Salido, M.A., Li, W.D., 2013. Energy-efficient scheduling for a flexible flow shop using an improved genetic-simulated annealing algorithm. *Robotics and Computer-Integrated Manufacturing* 29, 418–429.
- Ding, J., Schulz, S., Shen, L., Buscher, U., Lü, Z., 2021. Energy aware scheduling in flexible flow shops with hybrid particle swarm optimization. *Computers & Operations Research* 125, 105088.
- Ding, J.Y., Song, S., Zhang, R., Chiong, R., Wu, C., 2015. Parallel machine scheduling under time-of-use electricity prices: New models and optimization approaches. *IEEE Transactions on Automation Science and Engineering* 13, 1138–1154.
- Drake, R., Yildirim, M.B., Twomey, J.M., Whitman, L.E., Ahmad, J.S., Lodhia, P., 2006. Data collection framework on energy consumption in manufacturing .
- Fang, K., Uhan, N.A., Zhao, F., Sutherland, J.W., 2013. Flow shop scheduling with peak power consumption constraints. *Annals of Operations Research* 206, 115–145.
- Fang, K., Uhan, N.A., Zhao, F., Sutherland, J.W., 2016. Scheduling on a single machine under time-of-use electricity tariffs. *Annals of Operations Research* 238, 199–227.
- Fang, K.T., Lin, B.M., 2013. Parallel-machine scheduling to minimize tardiness penalty and power cost. *Computers & Industrial Engineering* 64, 224–234.
- Gelazanskas, L., Gamage, K.A., 2014. Demand side management in smart grid: A review and proposals for future direction. *Sustainable Cities and Society* 11, 22–30.

- Gong, X., De Pessemer, T., Joseph, W., Martens, L., 2016. A generic method for energy-efficient and energy-cost-effective production at the unit process level. *Journal of Cleaner Production* 113, 508–522.
- Gong, X., De Pessemer, T., Martens, L., Joseph, W., 2019. Energy-and labor-aware flexible job shop scheduling under dynamic electricity pricing: A many-objective optimization investigation. *Journal of cleaner production* 209, 1078–1094.
- Gong, X., Van der Wee, M., De Pessemer, T., Verbrugge, S., Colle, D., Martens, L., Joseph, W., 2017. Energy-and labor-aware production scheduling for sustainable manufacturing: A case study on plastic bottle manufacturing. *Procedia CIRP* 61, 387–392.
- Graham, R.L., Lawler, E.L., Lenstra, J.K., Kan, A.R., 1979. Optimization and approximation in deterministic sequencing and scheduling: a survey, in: *Annals of discrete mathematics*. Elsevier. volume 5, pp. 287–326.
- Hooker, J.N., Ottosson, G., 2003. Logic-based benders decomposition. *Mathematical Programming* 96, 33–60.
- IEA, 2019. CO2 emissions from fuel combustion 2018 overview. <https://www.iea.org/statistics/co2emissions/>.
- Ji, M., Wang, J.Y., Lee, W.C., 2013. Minimizing resource consumption on uniform parallel machines with a bound on makespan. *Computers & operations research* 40, 2970–2974.
- Jia, Z.h., Wang, Y., Wu, C., Yang, Y., Zhang, X.y., Chen, H.p., 2019. Multi-objective energy-aware batch scheduling using ant colony optimization algorithm. *Computers & Industrial Engineering* 131, 41–56.
- Jiang, E.d., Wang, L., 2018. 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* , 1–16.
- 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, 1756–1771.
- Johnson, S.M., 1954. Optimal two-and three-stage production schedules with setup times included. *Naval research logistics quarterly* 1, 61–68.
- Kong, M., Xu, J., Zhang, T., Lu, S., Fang, C., Mladenovic, N., 2021. Energy-efficient rescheduling with time-of-use energy cost: Application of variable neighborhood search algorithm. *Computers & Industrial Engineering* 156, 107286.
- Leung, J.Y.T., Lee, K., Pinedo, M.L., 2012. Bi-criteria scheduling with machine assignment costs. *International Journal of Production Economics* 139, 321–329.
- Li, C., Tang, Y., Cui, L., Li, P., 2015. A quantitative approach to analyze carbon emissions of cnc-based machining systems. *Journal of Intelligent Manufacturing* 26, 911–922.
- Liang, J., Wang, Y., Zhang, Z.H., Sun, Y., 2019. Energy efficient production planning and scheduling problem with processing technology selection. *Computers & Industrial Engineering* 132, 260–270.
- Liu, G.S., Yang, H.D., Cheng, M.B., 2017. A three-stage decomposition approach for energy-aware scheduling with processing-time-dependent product quality. *International Journal of Production Research* 55, 3073–3091.
- Liu, Y., Dong, H., Lohse, N., Petrovic, S., 2016. A multi-objective genetic algorithm for optimisation of energy consumption and shop floor production performance. *International Journal of Production Economics* 179, 259–272.
- Luo, H., Du, B., Huang, G.Q., Chen, H., Li, X., 2013. Hybrid flow shop scheduling considering machine electricity consumption cost. *International Journal of Production Economics* 146, 423–439.
- Mansouri, S.A., Aktas, E., Besikci, U., 2016. Green scheduling of a two-machine flowshop: Trade-off between makespan and energy consumption. *European Journal of Operational Research* 248, 772–788.
- Masmoudi, O., Delorme, X., Gianessi, P., 2019. Job-shop scheduling problem with energy consideration. *International Journal of Production Economics* 216, 12–22.
- May, G., Stahl, B., Taisch, M., Prabhu, V., 2015. Multi-objective genetic algorithm for energy-efficient job shop scheduling. *International Journal of Production Research* 53, 7071–7089.
- Modos, I., Sucha, P., Hanzalek, Z., 2017. Algorithms for robust production scheduling with energy consumption limits. *Computers & Industrial Engineering* 112, 391–408.
- Mokhtari, H., Hasani, A., 2017. An energy-efficient multi-objective optimization for flexible job-shop scheduling problem. *Computers & Chemical Engineering* 104, 339–352.

- Mouzon, G., Yildirim, M.B., Twomey, J., 2007. Operational methods for minimization of energy consumption of manufacturing equipment. *International Journal of production research* 45, 4247–4271.
- Pilerood, A.E., Heydari, M., Mazdeh, M.M., 2018. A two-stage greedy heuristic for a flowshop scheduling problem under time-of-use electricity tariffs. *The South African Journal of Industrial Engineering* 29, 143–154.
- Rager, M., Gahm, C., Denz, F., 2015. Energy-oriented scheduling based on evolutionary algorithms. *Computers & Operations Research* 54, 218–231.
- Schulz, S., Udo, B., Liji, S., 2020. Multi-objective hybrid flow shop scheduling with variable discrete production speed levels and time-of-use energy prices. *Zeitschrift für Betriebswirtschaft* 90, 1315–1343.
- Sharma, A., Zhao, F., Sutherland, J.W., 2015. Econological scheduling of a manufacturing enterprise operating under a time-of-use electricity tariff. *Journal of Cleaner Production* 108, 256–270.
- Shrouf, F., Ordieres-Meré, J., García-Sánchez, A., Ortega-Mier, M., 2014. Optimizing the production scheduling of a single machine to minimize total energy consumption costs. *Journal of Cleaner Production* 67, 197–207.
- Subai, C., Baptiste, P., Niel, E., 2006. Scheduling issues for environmentally responsible manufacturing: The case of hoist scheduling in an electroplating line. *International Journal of Production Economics* 99, 74–87.
- Wang, L., et al., 2020a. Multi-objective optimization based on decomposition for flexible job shop scheduling under time-of-use electricity prices. *Knowledge-Based Systems* 204, 106177.
- Wang, S., Wang, X., Chu, F., Yu, J., 2020b. An energy-efficient two-stage hybrid flow shop scheduling problem in a glass production. *International Journal of Production Research* 58, 2283–2314.
- Wang, S., Wang, X., Yu, J., Ma, S., Liu, M., 2018a. Bi-objective identical parallel machine scheduling to minimize total energy consumption and makespan. *Journal of cleaner production* 193, 424–440.
- Wang, S., Zhu, Z., Fang, K., Chu, F., Chu, C., 2018b. Scheduling on a two-machine permutation flow shop under time-of-use electricity tariffs. *International Journal of Production Research* 56, 3173–3187.
- Wang, Y., Li, L., 2016. Critical peak electricity pricing for sustainable manufacturing: Modeling and case studies. *Applied energy* 175, 40–53.
- Wu, X., Shen, X., Li, C., 2019. The flexible job-shop scheduling problem considering deterioration effect and energy consumption simultaneously. *Computers & Industrial Engineering* 135, 1004–1024.
- Zhang, B., Pan, Q.k., Gao, L., Li, X.y., Meng, L.l., Peng, K.k., 2019. A multiobjective evolutionary algorithm based on decomposition for hybrid flowshop green scheduling problem. *Computers & Industrial Engineering* 136, 325–344.
- Zhang, H., Wu, Y., Pan, R., Xu, G., 2021. Two-stage parallel speed-scaling machine scheduling under time-of-use tariffs. *Journal of Intelligent Manufacturing* 32, 91–112.
- 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.
- Zheng, X., Zhou, S., Xu, R., Chen, H., 2019. Energy-efficient scheduling for multi-objective two-stage flow shop using a hybrid ant colony optimisation algorithm. *International Journal of Production Research* , 1–18.
- Zhou, S., Li, X., Du, N., Pang, Y., Chen, H., 2018. A multi-objective differential evolution algorithm for parallel batch processing machine scheduling considering electricity consumption cost. *Computers & operations research* 96, 55–68.