

Research Article

A Branch-and-Bound Algorithm for Minimizing the Energy Consumption in the PFS Problem

Guo-Sheng Liu,¹ Bi-Xi Zhang,¹ Hai-Dong Yang,² Xin Chen,² and George Q. Huang³

¹ School of Management, Guangdong University of Technology, Longdong Campus 161 Yinglong Road, Guangzhou 510520, China

² Department of Mechatronics Engineering, Guangdong University of Technology, 100 Waihuan Xi Road, Higher Education Mega Center, Guangzhou 510006, China

³ Department of Industrial and Manufacturing Systems Engineering, The University of Hong Kong, Pokfulam, Hong Kong

Correspondence should be addressed to Guo-Sheng Liu; armstrongliu@yahoo.cn

Received 22 November 2012; Accepted 24 January 2013

Academic Editor: Wei-Chiang Hong

Copyright © 2013 Guo-Sheng Liu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper considers the energy consumption minimization in permutation flow shop (PFS) scheduling problem. The energy consumption of each machine is decomposed into two parts: useful part which completes the operation at current stage and wasted part which is consumed during idle period. The objective considered here is to minimize the total wasted energy consumption which is a weighted summation of the idle time of each machine. To solve this new problem, a branch-and-bound algorithm is developed. Two lower bounds are proposed, and an initial upper bound by using a variant of NEH heuristic algorithm is applied. Compared with the makespan minimization criterion, this model deduces more energy-saving solutions. Experimental results also validate the efficiency of the proposed algorithm for problems with job number not larger than 15.

1. Introduction

In 2011, world primary energy consumption grew by 2.5%, roughly in line with the ten-year average [1]. It is projected to increase by approximately 30% until 2030 [2]. Mankind is faced with more and more adverse conditions of energy resource exhaustion and environmental pollution. Manufacturing industry uses a relatively high share of the amount of total energy consumption of each year. This is especially true in developing countries, such as the BRIC: Brazil, Russia, India, and China. In recent years, scientists and engineers have paid more and more concerns on how to save energy and reduce green gas emission in manufacturing process. There arise several different directions being pursued in academia and industry, which are categorized by the implementation strategies for saving energy by management, technology, and policy [3].

Scheduling is one of the main manners in production and operation management. It deals with the allocation of scarce resources to tasks during manufacturing process. The researches on this topic began to be taken seriously in manufacturing at the beginning of the last century with

the work of Gantt and other pioneers [4, 5]. More interest had been paid after the first scheduling publication appeared in the early 1950s by Johnson [6]. In the past six decades, most scientists considered the scheduling problems with particular emphasis on the time requirement. They adopted various objective functions of the completion time of the involved jobs, such as makespan, weighted completion time, lateness, and tardiness [4, 5, 7, 8]. We refer to them as job-based objectives. In manufacturing processes, energy consumption is mainly characterized by the power, process time, and state of all involved machines, which deduces a machine-based criterion. Mouzon and Yildirim [9] first addressed the energy consumption combined with job-based objective (total completion time and total tardiness) on single machine. They measured the total energy consumption by summation of idle power and machine-dependent setup power. Actually, for single machine case, the idle time is controlled by the release time of each job. The key for saving energy in this case is to determine if the machine should be turned off during idle time. Liu et al. [10] gave the mathematical model for hybrid PFS problem to minimize the energy consumption. A conventional genetic algorithm is adopted. Fang et al. [11]

presented a multiobjective model to minimize the makespan, the energy consumption, and the carbon emission. A simple case study is provided for the production scheduling of cast iron plates on two machines.

To the best of our knowledge, there were very limited literatures which considered the energy requirement in scheduling problems. Previous few papers focused on the studies of framework modeling for specific industrial implications. In this paper, we begin the research of energy-saving scheduling within the basic permutation flow shop (PFS) problem. The energy consumption of each machine is decomposed into two parts: the useful part which completes the operation at current stage and the wasted part consumed during idling period. Hence, the objective in this model is to minimize the total idle energy consumption. It is formulated by a function of weighted idle time on each machine, as the weight is the machine power. For solving this model, we develop a branch-and-bound algorithm in which two lower bounds and an upper bound are proposed and applied.

After this introduction, we describe the problem dealt with and detail the energy-saving PFS model in Section 2. The branch and bound algorithm is set out in Section 3, in which two lower bounds and a heuristic procedure are proposed to delineate the total energy consumption of PFS model. Computational results and comparative analysis on Car's instances and 2400 randomly generated instances are shown in Section 4. Finally, the main conclusions are included in Section 5.

2. Problem Description

In the PFS, there are a set $N = \{1, 2, \dots, n\}$ of jobs and a set $M = \{1, 2, \dots, m\}$ of successive machines. Each job j consists of m operations $\{O_{1j}, O_{2j}, \dots, O_{mj}\}$, where each operation O_{ij} for job j is to be processed on machine i for t_{ij} time units, and O_{rj} must precede O_{sj} if $r < s$. We assume that the rated output power of machine i is p_i . The total energy consumption is decomposed into two parts as follows:

$$E = U + W, \quad (1)$$

where U denotes the useful energy consumption and W denotes the wasted energy consumption. In the PFS problem, the useful part U is the minimum energy consumption required for completing all the jobs independently. It is the summation of the theoretical value of each job; that is,

$$U = \sum_{j \in N} U_j = \sum_{j \in N} \sum_{i \in M} p_i t_{ij}. \quad (2)$$

To characterize the wasted part of energy consumption W in the basic PFS problem, we first introduce the following assumptions.

- (A1) Machines are turned on as soon as the first job arrives and turned off when the last job leaves.
- (A2) Machines will keep being idle during two successive jobs.

Assumption (A1) indicates the start time and complete time of each time. Assumption (A2) indicates that the waste part

involved arises only from being idle on each machine. It is the weighted summation of the idle time of each machine; that is,

$$W = \sum_{i \in M} W_i = \sum_{i \in M} p_i I_i, \quad (3)$$

where I_i is the total idle time of machine i during the whole process. Therefore, the objective of the energy-saving PFS problem is to minimize the total idle energy consumption W . We denote this model by $F_m | prmu | W$. For a given job sequence (j_1, j_2, \dots, j_n) , the mathematical expression of $F_m | prmu | W$ is listed as follows:

$$\begin{aligned} \text{Min } W &= \sum_{i=1}^m p_i \sum_{k=1}^n I_{i,[k]}, \\ \text{s.t. } C_{1,[k]} &= \sum_{s=1}^k t_{1,[s]}, \\ C_{i,[1]} &= \sum_{r=1}^i t_{r,[1]}, \\ C_{i,[k]} &= \max \{C_{i-1,[k]}, C_{i,[k-1]}\} + t_{i,[k]}, \\ I_{1,[k]} &= 0, \quad I_{i,[1]} = 0, \\ I_{i,[k]} &= \max \{C_{i-1,[k]} - C_{i,[k-1]}, 0\}, \end{aligned} \quad (4)$$

where $I_{i,[k]}$ is the idle time of machine i before starting operation $O_{i,[k]}$, and $C_{i,[k]}$ is the completion time of job j_k on machine i . For solving the proposed $F_m | prmu | W$ problem, we derive a branch and bound algorithm to seek the optimal solution for energy-saving objective in the next section.

3. Branch and Bound Algorithm

The branch-and-bound algorithm is a good choice for solving the small job-size PFS problems as a result of its NP-hard property [5, 7, 8]. In this section, we first provide a heuristic method for generating the initial upper bound, then followed by two lower bounds and a branching scheme to reduce the size of the set of feasible solutions, and finally the description of the proposed branch-and-bound algorithm.

3.1. An Initial Upper Bound. To construct an efficient branch and bound algorithm, an initial solution should be adopted to generate an upper bound as a benchmark. In this subsection, we extend the well-known NEH heuristic method into $F_m | prmu | W$ problem with a correction procedure as the final step. Two operation notions are introduced for the sake of clearness. Given a sequence π of n elements, define the inserting operator as

$$\rho(\pi, j, k) = (\pi_1, \pi_2, \dots, \pi_{j-1}, k, \pi_j, \dots, \pi_n) \quad (5)$$

and the extracting operator as

$$\gamma(\pi, j) = (\pi_1, \pi_2, \dots, \pi_{j-1}, \pi_{j+1}, \dots, \pi_n). \quad (6)$$

Then, the procedure of the proposed heuristic algorithm is listed as follows.

Step 1. Compute the useful energy consumption U_j for each job, and sequence all jobs by their nonincreasing order of U_j to obtain a priority sequence τ ; set $k = 1$ and $\pi = \emptyset$.

Step 2. Execute $\rho(\pi, j^*, \tau_k)$ in which $j^* = \arg \min_{1 \leq j \leq k} W(\rho(\pi, j, \tau_k))$.

Step 3. Set $k = k + 1$, and repeat Step 2 while $k \leq n$.

Step 4. Find the best sequence to $\min_{1 \leq j \leq n} W(\rho(\gamma(\pi, j), 1, \pi_j))$.

The last step in the proposed algorithm is a correction procedure which adjusts the arrangement at the first position. Experimental results show that the correction procedure plays an important role in improving the solution.

3.2. Lower Bounds. For the PFS scheduling problem including n jobs, assume a sequenced job order σ and the remaining subset S of unsequenced jobs; the completion time on machine i has

$$C_{i,[n]} \geq \max_{1 \leq r \leq i} \left\{ C_{r,[q]} + \sum_{j \in S} t_{r,j} + \sum_{h=r+1}^i t_{h,[n]} \right\}, \quad (7)$$

where $q = |n|$ is the length of sequence σ . Applying (3), the total idle energy consumption on machine i is evaluated as follows:

$$\begin{aligned} W_i &= W(\sigma, i) + p_i \sum_{j=k+1}^n I_{i,[j]} \\ &= W(\sigma, i) + p_i \left\{ C_{i,[n]} - C_{i,[q]} - \sum_{j \in S} t_{ij} \right\} \\ &\geq W(\sigma, i) + p_i \left\{ C_{r,[q]} + \sum_{j \in S} t_{r,j} + \sum_{h=r+1}^i t_{h,[n]} - C_{i,[q]} - \sum_{j \in S} t_{ij} \right\} \\ &\geq W(\sigma, i) + p_i \left\{ (C_{r,[q]} - C_{i,[q]}) + \sum_{j \in S} (t_{rj} - t_{ij}) + \sum_{h=r+1}^i t_{h,[n]} \right\}, \end{aligned} \quad (8)$$

in which $W(\sigma, i)$ denotes the idle energy consumption of subsequence σ on machine i . Hence, the first lower bound for total idle energy consumption is

$$\begin{aligned} LB_1 &= \min_{k \in S} \sum_{i=1}^n \left\{ W(\sigma, i) + p_i \left\{ (C_{r,[q]} - C_{i,[q]}) \right. \right. \\ &\quad \left. \left. + \sum_{j \in S} (t_{rj} - t_{ij}) + \sum_{h=r+1}^i t_{h,[n]} \right\} \right\}. \end{aligned} \quad (9)$$

McMahon and Burton [12] introduced (for $m = 3$) a job-based bound

$$C_{i,[n]} \geq \max_{1 \leq r \leq i} \left\{ C_{r,[q]} + \max_{k \in S} \left\{ \sum_{j \in S - \{k\}} \min \{t_{rj}, t_{ij}\} + \sum_{h=r}^i t_{hk} \right\} \right\} \quad (10)$$

based on which a lower bound of the idle energy consumption can be derived as follows:

$$\begin{aligned} W_i &\geq W(\sigma, i) + p_i \left\{ \max_{1 \leq r \leq i} \left\{ \max_{k \in S} \left\{ \sum_{j \in S - \{k\}} \min \{t_{rj}, t_{ij}\} + \sum_{h=r}^i t_{hk} \right\} \right. \right. \\ &\quad \left. \left. + C_{r,[q]} \right\} - C_{i,[q]} - \sum_{j \in S} t_{ij} \right\} \\ &\geq W(\sigma, i) + p_i \left\{ (C_{r,[q]} - C_{i,[q]}) + \sum_{j \in S} \min \{t_{rj} - t_{ij}, 0\} \right. \\ &\quad \left. + \max_{k \in S} \left\{ \sum_{h=r}^i t_{hk} - \min \{t_{rk}, t_{ik}\} \right\} \right\}. \end{aligned} \quad (11)$$

Hence, the second lower bound for total idle energy consumption is

$$\begin{aligned} LB_2 &= \sum_{i=1}^n \left\{ W(\sigma, i) + p_i \left\{ (C_{r,[q]} - C_{i,[q]}) + \sum_{j \in S} \min \{t_{rj} - t_{ij}, 0\} \right. \right. \\ &\quad \left. \left. + \max_{k \in S} \left\{ \sum_{h=r}^i t_{hk} - \min \{t_{rk}, t_{ik}\} \right\} \right\} \right\}. \end{aligned} \quad (12)$$

3.3. Branching Scheme. A branching scheme indicated how all the nodes in a spanning tree are traversed. The route of accessing nodes from top to down represents a feasible candidate sequence. For a branching scheme to be truly effective, it needs to select such a job as current node that the deduced subproblem, consisting of remaining jobs, has better performance. In this paper, we select the current node based on its bound value; that is, the nodes with lower bounds have relatively high priority. Figure 1 depicts the branching procedure and the inchoate tree structure of a PFS problem with four jobs. In each level of the tree, all candidate nodes are sequenced by their lower bound with nondecreasing order. Depth first search is applied then sequentially.

3.4. Algorithm Description

Step 1 (initialization). Use the proposed variant of NEH algorithm to generate an initial solution, and calculate its idle energy consumption as the upper bound. Initialize a sequence σ of elements representing the tree nodes with size

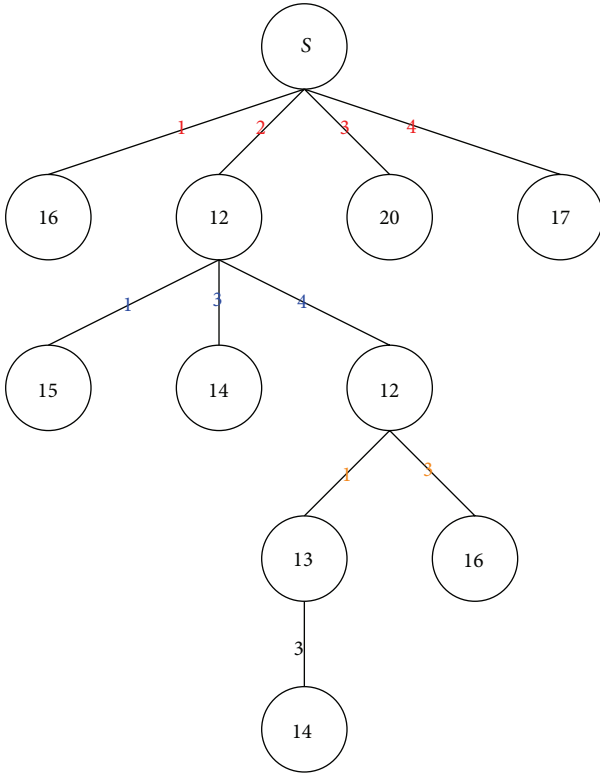


FIGURE 1: The branching tree in the proposed algorithm, in which each branch is represented by the current selected jobs, while the deduced node is characterized by its lower bound.

of $n(n+1)/2$ and a set S representing all the involved jobs with size of n .

Step 2 (branching). Calculate the lower bounds of all the nodes in S by calling the bounding procedure, and add them into σ with non-decreasing order of lower bounds. Check the termination condition. Traverse all the new nodes; if the lower bound of current new node is smaller than the global upper bound, then remove the current node from S and call the branching procedure.

Step 3 (bounding). Calculate the lower bound of current node by using (9) and (12). If S is empty and the upper bound is larger than the lower bound, replace the upper bound with the lower bound of current node.

Step 4 (termination). All branching procedures are completed, or the upper bound is zero.

It can be seen that the procedure of the branch and bound algorithm proposed earlier only requires computer memory with size of $O(n^2/2)$.

4. Computational Results

We conducted computational experiments to evaluate the performance and the efficiency of the proposed model and algorithm. The programme is coded in VC++6.0. All the test

TABLE 1: Performance test of the proposed algorithm for problems with different sizes, each including 100 randomly generated instances.

Problem size		Node number		CPU time (s)	
		Mean	Std	Mean	Std
$n = 8$	$m = 5$	122	170	0.00	0.00
	$m = 10$	203	149	0.00	0.00
	$m = 20$	485	365	0.05	0.03
$n = 9$	$m = 5$	235	425	0.00	0.00
	$m = 10$	642	919	0.01	0.01
	$m = 20$	1552	1246	0.23	0.13
$n = 10$	$m = 5$	687	1418	0.00	0.00
	$m = 10$	1427	2100	0.05	0.06
	$m = 20$	4760	3899	0.86	0.55
$n = 11$	$m = 5$	2818	7495	0.01	0.03
	$m = 10$	4931	16608	0.20	0.56
	$m = 20$	15663	13793	3.40	2.42
$n = 12$	$m = 5$	17828	57474	0.11	0.30
	$m = 10$	11145	19341	0.56	0.82
	$m = 20$	50040	53305	12.91	11.34
$n = 13$	$m = 5$	148403	787570	0.80	3.87
	$m = 10$	75550	292675	3.19	9.63
	$m = 20$	186429	236595	56.09	58.79
$n = 14$	$m = 5$	1831908	7847959	8.95	35.98
	$m = 10$	172948	439658	9.43	19.09
	$m = 20$	513043	642645	186.96	191.61
$n = 15$	$m = 5$	9165825	37677728	47.76	193.38
	$m = 10$	1374782	5754004	71.97	274.53
	$m = 20$	2748107	10862267	991.37	2540.82

problems were run on a Pentium 4-3.10 GHz desktop with RAM size of 2 Gbytes.

4.1. Performance Test. In order to test the performance of the proposed branch and bound algorithm, eight different numbers of jobs ($n = 8, 9, \dots, 15$) and three different numbers of machines ($m = 5, 10, 20$) were used. For each combination of problems with different sizes, 100 instances with randomly generated integral processing time in $[1, 100]$ were tested. All involved machines have identical output rated power. The number of nodes and CPU time for solving each instance by using the proposed algorithm were reported. The mean value and standard deviation are given in Table 1. It can be seen that the node numbers and CPU time increase dramatically with job number n . For $n = 8, 9, 10, 11$, the node numbers and cpu time increase steadily with machine number m ; however, for $n = 12, 13, 14, 15$, the node numbers with $m = 10$ are obviously less than those with $m = 5$ and 20, while the cpu time is still larger than the others. This indicates that the branching scheme is more effective when m and n are close, and the CPU time is mainly determined by the computational consumption for lower bound. Table 1 also shows that the

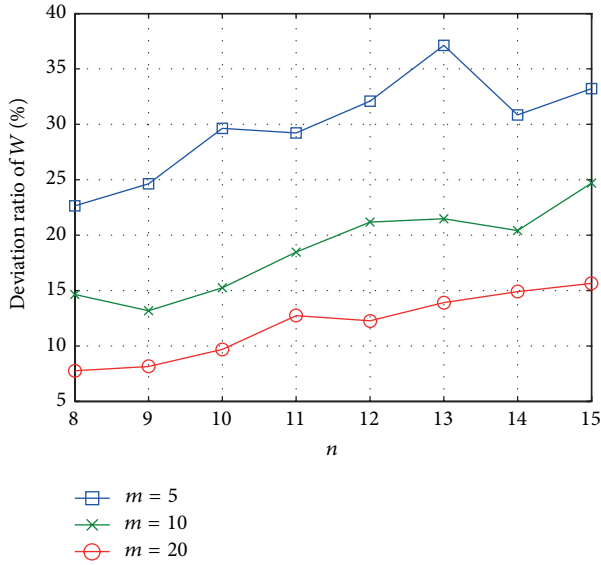


FIGURE 2: The deviation ratio of W (WDR) varies against the job number n and machine number m .

proposed algorithm has good performance for problems with n not larger than 15.

To demonstrate the performance of the proposed NEH heuristic algorithm for the energy-saving PFS problem, the deviation between the initial solution and the optimal solution for the tested instances was evaluated. Similar within [7, 8, 13], the deviation ratio of W is defined as follows:

$$\text{WDR} = \frac{1}{|\Gamma_o|} \sum_{\kappa \in \Gamma_o} \frac{\overline{W}_\kappa - W_\kappa^*}{\overline{W}_\kappa} \times 100\%, \quad (13)$$

in which \overline{W}_κ is the energy consumption of the initial solution for instance κ generated by the proposed NEH algorithm, while W_κ^* is the optimal (minimum) energy consumption calculated by the proposed branch and bound algorithm. The evaluated instance set is $\Gamma_o = \{\kappa \mid \kappa \in \Gamma, \overline{W}_\kappa \neq 0\}$, where Γ is the test set consisting of 100 randomly generated instances. Figure 2 depicts the computational results of WDR for the problems with different sizes. It can be seen that the proposed heuristic algorithm generating the initial solution has better performance when m is small. The WDR mainly increases linearly with job number n . However, the maximum deviation is not beyond 40%, which indicates that the proposed heuristic NEH algorithm has good performance.

From the results in Table 1, we can see that the branch-and-bound algorithm is timeconsuming while it can find the exact optimum. For large job-number problem, heuristic algorithm might be a good alternative. We test the cpu time of the proposed heuristic algorithm. Figure 3 shows the results for problems with 5, 10, 20 machines and 10, 20, 30, ..., 100 jobs, respectively. It can be seen that there are some gaps between the results of $m = (5, 10)$ and $m = (10, 20)$, which indicates that the cpu time consumed is proportional to machine number m . On the other hand, the cpu time is approximately polynomial with respect to job number n .

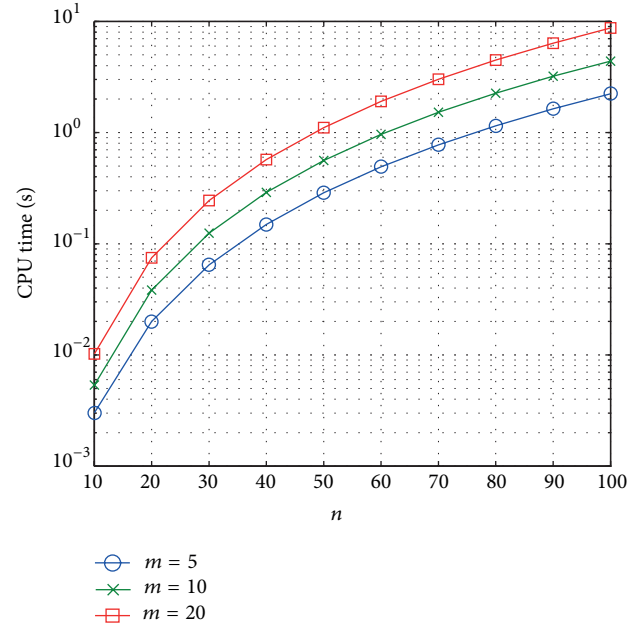


FIGURE 3: The CPU time consumed by the proposed heuristic algorithm for solving large size PFS problem.

It can be concluded from numerical results that the proposed heuristic algorithm has good performance for solving large job-size PFS problem.

4.2. Efficiency Validation. In this subsection, we validate the efficiency of saving energy of the proposed model by comparison with makespan minimization model in the PFS problem. The open-accessing instances presented by Carlier [14] were selected for the test instances. The efficiency ratio of energy consumption and completion time is defined as follows:

$$\begin{aligned} \text{EER} &= \frac{W^* - W}{W} \times 100\%, \\ \text{TER} &= \frac{C_{\max} - C_{\max}^*}{C_{\max}^*} \times 100\%, \end{aligned} \quad (14)$$

in which W^* and C_{\max} are the wasted energy consumption and, completion, respectively of the solution calculated by the proposed energy-saving model, while W and C_{\max}^* are calculated by the makespan minimization model. For each machine involved in Car's instances, the rated output powers are set equal to one. Table 2 shows the detailed results of eight Car's instances. The comparative results are depicted in Figure 4 for each instance. It can be seen that the solution of the proposed model can reduce the wasted energy dramatically while keeping completion time increase in a small range.

5. Conclusions

This study has investigated the $F_m|pmu|W$ problem by using a branch and bound algorithm. The energy consumption

TABLE 2: Results of Car's instances for makespan minimization and energy consumption minimization.

Instance	Problem size ($m \times n$)	$F prmu C_{\max}$		$F prmu W$	
		C_{\max}^*	W	C_{\max}	W^*
CAR1	5×11	7038	4135	8263	415
CAR2	4×13	7166	2524	9163	0
CAR3	5×12	7312	3684	9010	61
CAR4	4×14	8003	2272	8952	222
CAR5	6×10	7720	6979	8803	2148
CAR6	9×8	8505	8411	9814	5493
CAR7	7×7	6590	3496	7264	1089
CAR8	8×8	8366	8251	9082	6433

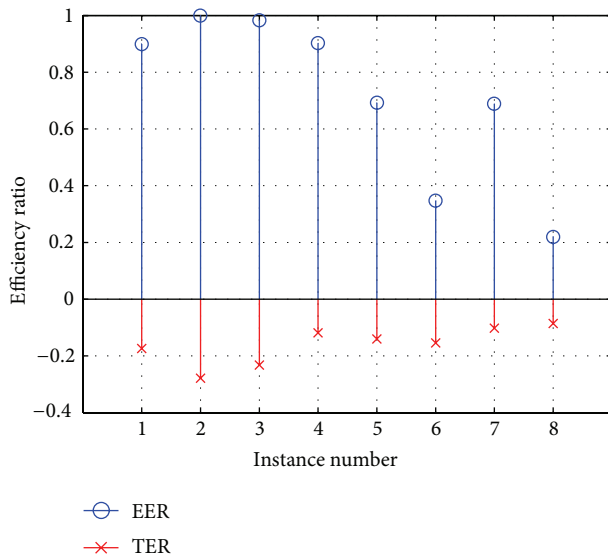


FIGURE 4: The efficiency ratio of energy and time by comparison of the proposed energy-saving model and makespan minimization model for Car's instances.

in proposed model is decomposed into useful part and wasted part. The wasted energy consumption is assumed to arise from the idle of machine between two successive jobs. It is a weighted function of the idle time of all involved machines. To solve the proposed $F_m|prmu|W$ problem, a branch and bound algorithm is proposed. Two lower bound, and an improved NEH heuristic algorithm are applied. Numerical experiments show the good performance and high efficiency of the proposed model and algorithm.

To the best of our knowledge, this is the attempt to give a branch and bound algorithm for dealing with the energy-saving optimization in PFS problem. However, this research topic has a wide need in practice. A further discussion on $F_m|prmu|W$ problem with setup time, deteriorating jobs, and limited buffer will be reported in the future work.

Acknowledgment

This work was supported in part by the Humanity and Society Science Program of the Ministry of Education (12YJCZH129) and the National Natural Science Foundation of China (70971026 and 71271060).

References

- [1] "Statistical review of world energy," Tech. Rep., BP corporation, 2012.
- [2] "International energy outlook 2009: World energy and economic outlook," Tech. Rep., U.S. Energy Information Administration, 2009.
- [3] E. A. Abdelaziz, R. Saidur, and S. Mekhilef, "A review on energy saving strategies in industrial sector," *Renewable and Sustainable Energy Reviews*, vol. 15, no. 1, pp. 150–168, 2011.
- [4] M. L. Pinedo, *Scheduling: Theory, Algorithms, and Systems*, Prentice Hall, 2nd edition, 2000.
- [5] C. N. Potts and V. A. Strusevich, "Fifty years of scheduling: a survey of milestones," *Journal of the Operational Research Society*, vol. 60, supplement 1, pp. S41–S68, 2009.
- [6] S. M. Johnson, "Optimal two- and three-stage production schedules with setup times," *Naval Research Logistics Quarterly*, vol. 1, pp. 61–68, 1954.
- [7] W.-C. Lee, C.-C. Wu, Y.-H. Chung, and H.-C. Liu, "Minimizing the total completion time in permutation flow shop with machine-dependent job deterioration rates," *Computers and Operations Research*, vol. 36, no. 6, pp. 2111–2121, 2009.
- [8] W.-C. Lee and Y.-H. Chung, "Permutation flowshop scheduling to minimize the total tardiness with learning effects," *International Journal of Production Economics*, vol. 141, pp. 327–334, 2013.
- [9] G. C. Mouzon and M. B. Yildirim, "A framework to minimize total energy consumption and total tardiness on a single machine," in *Proceedings of the 4th Annual Symposium on Graduate Research and Scholarly Projects (GRASP '08)*, Wichita State University, 2008.
- [10] X. Liu, F. Zou, and X. Zhang, "Mathematical model and genetic optimization for hybrid flow shop scheduling problem based on energy consumption," in *Proceedings of the Chinese Control and Decision Conference (CCDC '08)*, pp. 1002–1007, July 2008.
- [11] K. Fang, N. Uhana, F. Zhao, and J. W. Sutherland, "A new approach to scheduling in manufacturing for power consumption and carbon foot- print reduction," *Journal of Manufacturing Systems*, vol. 30, pp. 234–240, 2011.
- [12] G. B. McMahon and P. G. Burton, "Flow-shop scheduling with the branch- and-bound method," *Operation Research*, vol. 15, pp. 473–481, 1967.
- [13] C. T. Ng, J.-B. Wang, T. C. E. Cheng, and L. L. Liu, "A branch-and-bound algorithm for solving a two-machine flow shop problem with deteriorating jobs," *Computers and Operations Research*, vol. 37, no. 1, pp. 83–90, 2010.
- [14] J. Carlier, "Or-library," <http://people.brunel.ac.uk/~mastjjb/jeb/info.html>.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

