American Journal of Engineering Research (AJER)
e-ISSN: 2320-0847 p-ISSN : 2320-0936
Volume-5, Issue-9, pp-189-193
www.ajer.org
Research Paper Open Access

Exact and heuristic algorithms for parallel-machine scheduling with learning effect

Mahdi Jemmali[1,3], Lotfi Hidri[2]
[1]Majmaaah University,College of science, Zulfi, Saudi Arabia, Assistant professor, Computerscience
[2]KingSaoudUniversity, Associate professor, Computer science
[3]MARS Research Unit

ABSTRACT: *In this paper we focus on parallel-machine scheduling*

g problem. The objective is to minimize the makespan. This problem is an NP-hard problem. We propose three heuristics and a branch-and-bound exact algorithm. An extensive experimental study has been conducted to proof the efficiency of the proposed procedures. Finally, we use the exact algorithms to applicate results for scheduling algorithms for the problem.

Keywords: *Parallel-machine, Scheduling, Learning effect, Branch-and-bound, Algorithms, Heuristics.*

I. INTRODUCTION

Scheduling problems is applied on many domains and environments. Several scheduling is based in classic theory [1]. Human requirements oblige to use more and more scheduling. Thus, new scheduling models such asscheduling tasks with time-dependent-processing times [2], multiprocessor task scheduling [3] and scheduling with availability. A so famous group of new scheduling methodsbased on scheduling models with the learning effect [4] that we consider in our research.

The majority of scheduling problems due withfixed and known processing times and over the entire process. However, in reality the job processingtimes in many circumstances, like that in the industry field,decreases due to the continuous improvements as firms produce more of a product and gain experience or knowledge.

As a result of experience and knowledge, the production time (processing times) of a given product isshorter if it is processed later. This aspect is known in literature as the ‘‘learning effect’’. Bachman and Janiak [6], Biskup[7], and Janiak and Rudek [8] have detailed some comprehensive reviews of scheduling models and problems with learningeffects.

Some interesting works was intended to develop exact and heuristic algorithms for parallel-machine scheduling with DeJong's learning effect in 2010 [5].

A general learning effect is modelled in scheduling theory by assumption that the processing time of a job is a function of the job position in a schedule.

In the literature on scheduling with the learning effect, it is assumed that the job processing time is a known constantwhich will be shortened due to the learning effect.

Develop scheduling algorithms for parallel machines and execute with applying program to elaborate upper bounds and exact algorithms. An experimental case must be done to make simulation of bounds and exact methods.

The paper, we will consider the following schedulingproblem with a learning effect.
There is given a set of jobs$J_1, J_2, \dots, J_n$ which have to be processed on machines$M_1, M_2, \dots, M_m$.
Job preemption is not allowed and all jobs are available for processing at time 0. The processing time of job J_j processed at position r ($1 \le r \le n$) is given by DeJong’s learning curve as follows:

$p_{j,r} = p_j \times (M + (1 - M)r^a)$Where$p_j$ is the initial job processingtime,

- $a \le 0$ is the learning index,
- M is the incompressibility factor,

- *r* is the current position of a job in a given schedule
- *1* $\leq$ *j*
- *r*$\leq$ *n.*

Function $p_{j,r}$ introduced by DeJong [9], meaning theimpact of a learning effect on job processing times.

The main advantage ofDeJong's model follows from the fact that parameter *M* representsthe part of job processing time that is limited by some conditionsand cannot be shortened. Different values of *M* are recommendedin literature.

For example, DeJong suggests *M = 0.25* for laborintensivejobs and *M = 0.5* for machine-intensive jobs [10].

Throughout the paper, we assume that $M \in [0,1]$.

The objective function of schedule optimality in our problem is to minimizethe makespan,$C_{max} := max\{C_j : 1 \leq j \leq n\}$, where C_j is the completiontime of job J_j. Extending the three fields notation [11].

We will denote the problemas:

$$Pm|p_{j,r} = p_j \times (M + (1-M)r^a)|C_{max}.$$

II. LITERATURE REVIEW

We briefly present the most recent literature concerningmulti-machine scheduling problems with a learning effectand branch-and-bound algorithms proposed for problems of thistype.

Mosheiov [14] was the first author to study the learning effect on parallel machines. Heshowed that the total completion time problem on parallel identical machines is polynomially solvable.Some of researches fields are discussed in Biskup[4].

Eren [13] studied a bi-criterion identical parallel machines scheduling problem witha learning effect of setup times and removal times. The objective was to minimize the weighted sum of total completion timeand total tardiness. He provided a mathematical programming model to solve problems with up to 15 jobs and five machinesand three heuristic approaches to solve problems with large numbers of jobs.

Toksarı and Güner [12] considered parallel-machine Earliness/ Tardiness (ET) scheduling problem with simultaneous effects oflearning and linear deterioration, sequence dependent setups anda common due-date for all jobs, and gave a mixed non-linear integerprogramming formulation of the problem.

They showed that the optimal solution for the ET scheduling problem under effects oflearning and deterioration is a V-shaped schedule under certain agreeable conditions. Furthermore, they developed a mathematicalmodel, an algorithm and a lower bound procedure for problems with large numbers of jobs.

Okołowski and Gawiejnowicz [5] considered a parallel-machine problem under the general DeJong's learning curve with minimization of the makespan. For this NP-hard problem, they proposed two exact algorithms, a sequential branch-and-bound algorithm and a parallel branch-and-bound algorithm.

Yanget al. [15] considered the parallel-machine scheduling problem with aging effects and multiple-maintenance activities. Theobjective was to find the optimal maintenance frequencies, the optimal positions of the maintenance activities, and the optimaljob sequences. They provided an efficient algorithm to solve the problem when the maintenance frequencies on the machineswere given.

III. Problem properties

In this section, we present some properties of the consideredproblem, which will be used in subsequent sections of the paper [5].

Problem$Pm|p_{j,r} = p_j \times (M + (1-M)r^a)|C_{max}$ is a generalization ofthe ordinary *NP*-hard problem $Pm| \ |C_{max}$.

1.1 Property 1

Problem$Pm|p_{j,r} = p_j \times (M + (1-M)r^a)|C_{max}$ is ordinary *NP*-hard.[5].

1.2 Property 2

Problem$\mathrm{Pm}|\mathrm{p}_{\mathrm{j,r}} = \mathrm{p}_\mathrm{j} \times (\mathrm{M} + (1-\mathrm{M})\mathrm{r}^\mathrm{a})|\mathrm{C}_{\max}$ is strongly *NP*-hard. [5].

1.3 Property 3

For problem$Pm|p_{j,r} = p_j \times (M + (1-M)r^a)|C_{max}$ there existoptimal schedules that do not include artificial idle times. [5].

IV. HEURISTICS

1.4 Longest processing timeheuristic: H1

The longest job are handled first and completed. We denote by H_{LPT} this heuristic.

Example 1:

We consider the instance 1 with:

- number of jobs *n=5*
- number of machines *m=1*
- *M=0.25*
- *a=0.1*

Table 1. Instances 1

Jobs	Processing Time
1	6
2	2
3	3
4	8
5	9

We execute in this order:

$p_5 p_4 p_1 p_3 p_2$

The schedule for this instance is giving by the following figure:

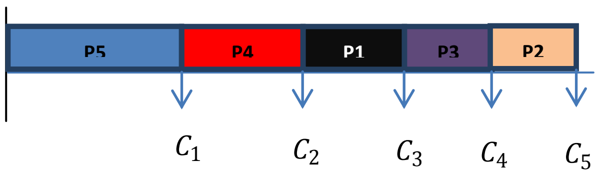

Fig 1: Schedule LPT of instances 1

$C_1 = p_{5,1} = p_5 \times (M + (1 - M)r^a) = 9 \times (0.25 + (1 - 0.25)1^{-0.1})$=9
$C_2 = C_1 + p_{4,2} = C_1 + p_4 \times (M + (1 - M)r^a) = 9 + 8 \times (0.25 + (1 - 0.25)2^{-0.1})$=9+7.5981=16.5981
$C_3 = C_2 + p_{1,3} = C_2 + p_1 \times (M + (1 - M)r^a) = 16.5981 + 6 \times (0.25 + (1 -$
0.253− *0.1*=16.5981+5.5318=22.1299
$C_4 = C_3 + p_{3,4} = C_3 + p_3 \times (M + (1 - M)r^a) = 22.1299 + 3 \times (0.25 + (1 -$
0.254− *0.1*=22.1299+2.7087=24.8386
$C_5 = C_4 + p_{2,5} = C_4 + p_2 \times (M + (1 - M)r^a) = 24.8386 + 2 \times (0.25 + (1 -$
0.255− *0.1*=24.8386+1.7770=26.6156
$C_{max} = C_5 = 26.6156$

However: $\sum_{j=1}^{j=5} p_j$ =28. So, we gain the difference time: 1.3844. This is due to learning effect.

1.5 Shortest processing timeheuristic: H2

The shortest job are handled first and completed. We denote by H_{SPT} this heuristic.

Example 2:

We consider the same instance of example 1.
We execute in this order:

$p_2 p_3 p_1 p_4 p_5$

The schedule for this instance is giving by the following figure:

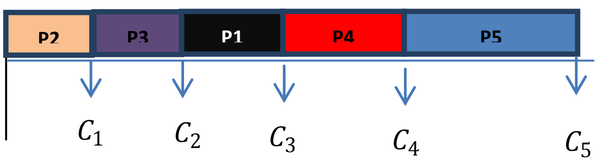

Fig 1: Schedule SPT of instances 1

$C_2 = C_1 + p_{3,2} = C_1 + p_3 \times (M + (1-M)r^a) = 2 + 3 \times (0.25 + (1 - 0.25)2^{-0.1})$=2+2.8493=4.8493
$C_3 = C_2 + p_{1,3} = C_2 + p_1 \times (M + (1-M)r^a) = 4.8493 + 6 \times (0.25 + (1 -$
$0.253 - 0.1$=4.8493+5.5318=10.3811
$C_4 = C_3 + p_{4,4} = C_3 + p_4 \times (M + (1-M)r^a) = 10.3811 + 8 \times (0.25 + (1 -$
$0.254 - 0.1$=10.3811+7.2233=17.6044
$C_5 = C_4 + p_{5,5} = C_4 + p_5 \times (M + (1-M)r^a) = 24.8386 + 9 \times (0.25 + (1 -$
$0.255 - 0.1$=17.6044+7.9965=25.6009
$C_{max} = C_5 = 25.6009$

However: $\sum_{j=1}^{j=5} p_j$ =28. So, we gain the difference time: 2.3991. This is due to learning effect.

1.6 Ascending Heuristic: H3

Now we arrange all processing time in according with ascending order.

We denoted by H_{Asc} this heuristic.

$$H_{Asc} = \sum_{j=1}^{n} p_j \,(M + (1-M)j^a)$$

For instance 1, the order of processing time is the following:

$$p_2 p_3 p_1 p_4 p_5$$

V. AN EXACT BRANCH-AND-BOUND ALGORITHM

Our algorithms are based on a similarapproach proposed for solving other parallelmachine scheduling problems. Our algorithms canefficiently solve problems of medium size.

We describe the components of a newbranch-and-bound algorithm that not only embeds the newly proposed lower and upper bounds.

This new solutionencoding is based on branch and bound algorithm for identical parallel machine.Obviously, whenever a solution having a maximum completion time satisfying $C_{max} < UB$ isfound, the incumbent value is updated and all the active nodes *N* having C_1 (*N*)≥*UB* are pruned with C_1 is the time cumulating in the node.

VI. EXPERIMENTAL RESULTS

The processing times were generated according to the following distributions:
Class 1: discrete uniform distribution on [1, 100];
Class 2: discrete uniform distribution on [20, 100];
Class 3: discrete uniform distribution on [50, 100];
Class 4: normal distribution with mean 100 and standard deviation 50;
Class 5: normal distribution with mean 100 and standard deviation 20;
For each n, m∈ {2, 3, 5}
For each n, m, class we generate 100 instances. Thus we have 1500 instances.

Table 2.GAP heuristic: a=-0.1_M=0

n	m	class	GAP H3	GAP H1	GAP H2
10	2	1	87,1%	6,4%	8,2%
		2	87,1%	6,5%	8,7%
		3	88,8%	2,4%	3,0%
		4	87,4%	6,4%	8,5%
		5	88,6%	2,6%	3,5%
	3	1	166,3%	5,9%	15,5%
		2	165,1%	5,4%	14,9%
		3	170,2%	11,7%	16,2%
		4	167,3%	5,8%	16,0%
		5	169,3%	10,5%	15,5%
	5	1	291,1%	1,9%	21,5%
		2	290,2%	1,7%	22,7%
		3	324,5%	1,0%	7,6%
		4	296,6%	2,9%	23,6%
		5	324,8%	1,2%	9,1%

Table 3.GAP heuristic: a=-0.322_M=0

n	m	class	GAP H3	GAP H1	GAP H2
10	2	1	63,6%	20,6%	8,4%
		2	63,4%	20,5%	9,0%
		3	67,2%	7,0%	3,0%
		4	64,3%	18,0%	7,9%
		5	67,0%	6,9%	3,5%
	3	1	114,0%	15,7%	9,8%
		2	112,9%	16,1%	9,8%
		3	122,9%	12,4%	11,6%
		4	115,1%	13,0%	10,7%
		5	123,0%	11,7%	11,1%
	5	1	187,2%	6,5%	18,3%
		2	185,8%	7,1%	19,8%
		3	227,8%	3,3%	7,5%
		4	192,3%	6,5%	19,7%
		5	228,6%	3,8%	9,1%

VII. CONCLUSION

The parallel machine scheduling problem has received important attention in the engineering and industry fields for its popularity information systems and manufacturing. The impact of the learning effect on production procedure is cost saving.

A limited number of studies on parallel machine scheduling problem with learning effect theory are available. However, the learning effect has not been considered in these studies. In this paper, we study theparallel machine scheduling problem with learning effects. The objective is to minimize the makespan. We proposed three heuristics and exact algorithms to solve the scheduling problem.Computational experiments were conducted to evaluate the performance and execution time of the heuristics under severaldifferent instances, and the impact of the learning effects is also discussed.

Extensions of the paper may consider different learning curves. In addition, we can give some more heuristics and lower bound to ameliorate the BB algorithm.

ACKNOWLEDGMENTS

This project was supported by the deanship of scientific research at Majmaah University under the research project. The author is highly grateful to Majmaah University, Deanship of Scientific Research and Research Center for based science, Majmaah University for support and encouragement. The author is also, obliged to College of Science in AlZulfi City, Majmaah University, KSA) for sparing to accomplish this work. The author would like to thank Deanship of Scientific Research, Majmaah University,Kingdom of Saudi Arabia for funding this work under project.

REFERENCES

[1]. Conway, R. W., Maxwell, W. L., & Miller, L. W. (1967). Theory of scheduling. Reading: Addison-Wesley.

[2]. Gawiejnowicz, S. (2008). Time-dependent scheduling. Berlin, New York: Springer.

[3]. Drozdowski, M. (2009). Scheduling for parallel processing. Berlin, London: Springer.

[4]. Biskup, D. (2008). A state-of-the-art review on scheduling with learning effects. European Journal of Operational Research, 188, 315–329.

[5]. Okołowski D, Gawiejnowicz S (2010). Exact and heuristic algorithms for parallel-machine scheduling with DeJong'slearning effect. Computers & Industrial Engineering, 59, 272–279

[6]. A. Bachman, A. Janiak, Scheduling jobs with position-dependent processing times, Journal of the Operational Research Society 55 (2004) 257–264.

[7]. D. Biskup, A state-of-the-art review on scheduling with learning effect, European Journal of Operational Research 188 (2008) 315–329.

[8]. A. Janiak, R. Rudek, A note on a makespan minimization problem with a multi-abilities learning effect, OMEGA – The International Journal of Management Science 38 (3–4) (2010) 213–217.

[9]. Badiru, A. B. (1992). Computational survey of univariate and multivariate learning curve models. IEEE Transactions on Engineering Management, 39, 176–188.

[10]. Raccoon, L. B. (1995). A learning curve primer for software engineers. Software Engineering Notes, 21, 77–86.

[11]. Graham, R. L., Lawler, E. L., Lenstra, J. K., &RinnooyKan, A. H. G. (1979). Optimization and approximation in deterministic sequencing and scheduling: A survey. Annals of Discrete Mathematics, 5, 287–326.

[12]. Toksarı, M. D., &Güner, E. (2008). Minimizing the earliness/tardiness costs on parallel machine with learning effects and deteriorating jobs: A mixed nonlinear integer programming approach. International Journal of Advanced Manufacturing Technology, 38, 801–808.

[13]. T. Eren, A bicriteria parallel machine scheduling with a learning effect of setup and removal times, Applied Mathematical Modelling 33 (2009) 1141–1150.

[14]. G. Mosheiov, Scheduling problems with a learning effect, European Journal of Operational Research 132 (2001) 687–693.

[15]. D.L. Yang, T.C.E. Cheng, S.J. Yang, C.J. Hsu, Unrelated parallel-machine scheduling with aging effects and multi-maintenance activities, Computers & Operations Research 39 (2012) 1458–1464.