IDENTICAL PARALLEL MACHINES SCHEDULING USING GENETIC ALGORITHM

Aseel Jameel Haleel
Department of Production Engineering and Metallurgy/ University of Technology
en.aseel@yahoo.com

ABSTRACT

Minimizing the scheduling production time consider one of the most important factors for companies which their objectives is achieve the maximum profits. This paper studies the identical parallel machine scheduling problem which involves the assignment numbers of job (N) to set of identical parallel machine (M) in order to minimize the makespan (maximum completion time of all job). There are numerous troubles in solving the large size of "parallel machine scheduling" problem with an excessive jobs and machines, so the genetic algorithm was proposed in this paper which is consider an efficient algorithm that fits larger size of identical "parallel machine scheduling" for minimizing the makespan. Most studies in the scheduling field suppose setup time is insignificant or included in the processing time, in this paper both the sequence independent setup times and processing time were considered. The solutions of algorithms are coding in (MATLAB). A numerical example of (11) jobs are schedule on (3) machines to demonstrative the effectiveness of algorithm solution. The result show the algorithm can effectively solve large size of scheduling problem and given the best schedule with minimum makespan.

KEY WORDS : Parallel machine, Job schedule, Genetic algorithm, Setup time, Makespan .

جدولة المكائن المتوازية المتماثلة باستخدام الخوارزمية الجينية

اسيل جميل هليل / الجامعة التكنولوجية

الخلاصة

يعتبر التقليل من وقت جدولة الانتاج من اهم العوامل بالنسبة للمصانع التي تسعى لتحقيق اقصى قدر من الارباح. يدرس هذا البحث عملية جدولة الاعمال على المكائن المتوازية والمتماثلة لتقليل اقصى وقت انهاء لكل وظيفة . العديد من المشاكل تظهر اثناء جدول حجم كبير من الاعمال والمكائن لذلك تم اقتراح الخوارزمية الجينية في هذه البحث والتي تعتبر خوارزمية فعالة والتي تناسب مع الحجم الكبير من مشاكل "جدولة المكائن المتوازية المتماثلة" لتقليل وقت انهاء الاعمال. معظم الدراسات في مجال الجدولة افترض وقت الإعداد غير هام أو تم تضمينها ضمن وقت المعالجة، في هذه البحث تم اعتماد كلا النوعين من الاوقات.تم برمجة الخوارزمية الجينية باستخدام برنامج (MATLAB) ولإظهار فعالية الخوارزمية الجينية تم تطبيق مثال لجدولة (11) وظيفة على (3) مكائن . اظهرت النتائج فعالية الخوارزمية الجينية في حل الحجم كبير من مشاكل الجدولة واعطاء افضل جدولة باقل وقت.

الكلمات المفتاحية: جدولة الاعمال، المكائن المتوازية، الخوارزمية الجينية، اوقات الاعداد

Received: 2017 - 9 - 26
Accepted: 2018 - 1 - 23

INTRODUCTION

The concept of parallel machines was widely used in the manufacturing. There are many types of scheduling issues, the “parallel machines scheduling” (PMS) “is one of them in which scheduling jobs in parallel machines were considered” [Chaudhry, 2011]. In machine scheduling problem reducing the makespan is an efficient method to minimize the job lateness with job tardiness. Additionally, prompts of the minimization the work-in-process buffer, and reducing abnormalities shop flow congestion because of uncompleted jobs. Therefore, reducing make span is a standout amongst the most essential criteria for production and service industries [Tavakkoli-Moghaddam, 2009]. For several years the parallel machine issues were studied due to its importance to academicals and industrial sectors. Many researchers were addressed issues related to “parallel machines scheduling”. Tavakkoli-Moghaddam et al. proposed a genetic algorithm (GA) approach to realize the bi-objective “parallel machine scheduling” problem. The execution of the proposed model is demonstrating by a various numerical experiments. The results demonstrate the effectuality of the proposed model and genetic algorithm for small and large sized issues [Tavakkoli-Moghaddam, 2009]. Chiang was applied a memetic algorithm to solve the a total weighted tardiness objective function for an “identical parallel machine” batch a scheduling problem with conflicting job families whose jobs arrived dynamically [Chiang 2010]. Also the genetic algorithm was proposed to minimizing the completion time for machine scheduling and worker assignment problem in identical parallel machine models [Chaudhry, 2010]. Jouglet and Savourey were identified some properties related to minimizing total tardiness in identical “parallel machine scheduling” issues [Jougle, 2011]. Chang was integrated dominance properties with genetic algorithms for “parallel machine scheduling” issues with setup times [Pie- Chann Chang, 2011]. Shaoo et al. worked on the genetic algorithm based multi-objective reliability optimization in interval environment. They have explained the goal to solve the constrained multi objective reliability optimization problem of a system [Sahoo, 2012]. Alwarsamy used used the neural networks to studied Multi objective optimization of “parallel machine scheduling” and choose the best optimal schedule which minimizes the make span, total tardiness and total earliness [Muralidhar, 2013]. Selvi proposing the mathematical models to solve scheduling issues involving “identical parallel machine”, where the goals are to optimize the multi objective scheduling issues using Genetic algorithms [Selvi, 2014]. German et al. developed an algorithm to find the optimal scheduling solution for the parallel machine schedule problem, the integer linear programming model and longest processing time algorithm used to generate the initial solution, then the developed algorithm used to improve the solution to minimizing the make span [German, 2016].

PROBLEM FORMULATION

Parallel Machine Scheduling

The issues of classical identical “parallel machine scheduling” can be expressed as follows: a set of the (N) jobs to be processed on a number of available identical parallel machines (M). Only one job can be process on each machine at a specific time, and each job can be processed on one machine. Each job has differed processing time and ready at the beginning of the scheduling horizon. In a manufacturing environment, setup times contain all activities that are performed on material, in order to prepare them for the main process phase. The sequence independent setup times, may be distinction for each machine based on machines characteristics. Setup time is important between the processing of different jobs, so the study was considered the setup times in order to reduce makespan (the total completion time of all

the jobs) [Chaudhry, 2011]. Setup time can be categorized into two types: (dependent and independent) sequence setup times. Sequence independent setup time depends on the job to be processed while sequence dependent setup time depends on both the job to be processed and the one yet processed [Maciel Manoel Queiroz 2013][Djamel Nait Tahar 2006].
The inputs parameters were used in the proposed model are shown below:
(M) Machines total number.
(N) Jobs total number to be processed.
(P_{im}) Processing time of job (i) on machine (M); i=1, 2, . . . , N M = 1, 2, . . .,M.
(S_{ijm}) Setup time to change from job (i) to job (j) on machine M : j = 1,2, ...,N.

Proposed Genetic Algorithm

Genetic algorithm was utilized widely in combinatorial optimization issues such as scheduling issues. The flowchart of proposed genetic algorithm is given in **Figure(1)**

Four stages can be identified in performing GA for machine scheduling [Ak, 2012]:
- Genes definition and chromosomes encoding
- Initial population
- Fitness function calculation
- Selection and genetic operators

Genes Definition and Chromosomes Encoding

The main instruction to build Genetic Algorithms called genes. A “chromosome is a sequence of genes”. In this paper chromosome representation every job in the schedule as a gene in a chromosome. The operations of representation of individual genes are called Encoding. This operation can carry out using bits, numbers, trees, arrays, lists or any other objects. In this paper the value encoding will be used in which every chromosome representation as a string of values and the values can be anything connected to the problem (number set, character set, etc.). An asterisk in used in order to distinguish the machines on the chromosome. **Figure(2)** illustrates the Chromosome value encoding [Ak, 2012].

Initial Population

A collection of chromosomes is called population. The two main portion of population used in genetic algorithm are the generation of initial population and the population size. An initial population generated mostly at random. The population size based on the complexity of the problem. Virtually, a population size of around 100 individuals is perfectly frequent, but it's can be changed according to the time, the memory and the problem type [Alcan, 2011]. In this paper the population size was (50 individuals and 100 iteration)

Fitness Function Calculation

The chromosome fitness in a genetic algorithm is the value of an objective function. Fitness function for each chromosome is considered to be the sum of different optimization criteria. For computing fitness function, firstly the chromosome decoded then the objective function evaluated. The objective function is minimizing the total completion time of all the jobs (make span) [Jiang, 2017]. To determine the fitness value for each chromosome a simple way is proposed to calculated the total time as shown in equation (1) and chooses the minimum summation as the feasible solution as:

$$T = \sum Pim + Sijm \quad (1)$$

where

T total completion time

P_{im} processing time

S_{ijm} setup time

Selection and Genetic Operators

Selection is the process of selecting two parents from the population for reproduction. Reproduction is an important element in GA, because it allows the exchange of information obtained by the chromosomes and its transmission to the next generation. The selection process executes by Selecting parents, crossing the parents to make new chromosomes (offspring or children) for the next generation, and exchange old chromosomes in the population with the new ones. All chromosomes of every generation include feasible and unfeasible chromosomes. Different selection methods were used to select parent, the important are roulette wheel selection and ranking selection [Savas Balin, 2011]. In this paper roulette wheel selection was used which select parents by simulating a roulette wheel. The algorithm uses a random number to choose one of the wheel sections with a probability of chosen a specific chromosome in respect to its fitness. After the parent selection the genetic operator crossover and mutation will be applied. These operations were explained below:

- **Crossover Operation**

Crossover is the main genetic operator. It generates offspring by combining both chromosomes feature. In this paper, two points crossover were used which is selected the two parent chromosomes randomly based on fitness, and randomly chosen the crossover points. **Figure(3)** illustrates the crossover operation when the crossover points are 3 and 6 [Alcan, 2011].

- **Mutation Operation**

The mutation operation makes new chromosomes by causing small perturbations in genes. During past years several mutation operators were proposed such as inversion, insertion, displacement, swapping mutation [Tavakkoli-Moghaddam,2007]. In this paper, swapping mutation was used, in which we select two random positions and then swap their genes as shown in **Figure(4)**

precursory experiments detected that there were three significant factors: population size, crossover and mutation probability. The factors values (Min and Max) can be represented as shown in **Table 1**.

COMPUTATIONAL EXPERIMENTS

Machine scheduling problem of (N) jobs on (M) identical Parallel machines solved using genetic algorithm described above sections. To solve the presented model, numerical example of (11) jobs were scheduled on (3) identical parallel machines. **Table 2** shows the genetic algorithm factors. **Table 3** shows the model's data (processing times and setup times) which are randomly generated.

RESULT

The results of experiments shown in **Table 4** which represent the best scheduling solution of (11 jobs) on (3 machines) with genetic algorithm. The optimal solution obtained with GA is (2.4102 Min) which is represent the minimum makespan. **Figure(5)** shows the completion time of the best make span solution and repetition average. The completion time decreases through the last iteration because the genetic algorithm tries to improve the machine scheduling until reach the value of best completion time.

CONCLUSIONS AND RECOMMENDATIONS

This paper solving scheduling problem of identical parallel machines using genetic algorithm optimization technique. This algorithm was proposed to minimize the total completion times considering a set of jobs that have independent setup time and processing time on a set of identical parallel machines. The proposed algorithm is greatly successful in finding optimal schedules on parallel machines additionally its effective in solving large size issues, the study displayed that the computational time (make span) required to solve the problem with (J = 11 and M = 3) was less than 3 min. For future work, the problem can be extending to include the tooling constraints (schedule set of jobs with processing time and tool requirements on identical parallel machines). Hybrid algorithm combining the discrete particle swarm optimization and genetic algorithm to solve the scheduling problem. Also, can using Genetic Algorithm (GA) in scheduling non-identical parallel machine with fuzzy processing time.

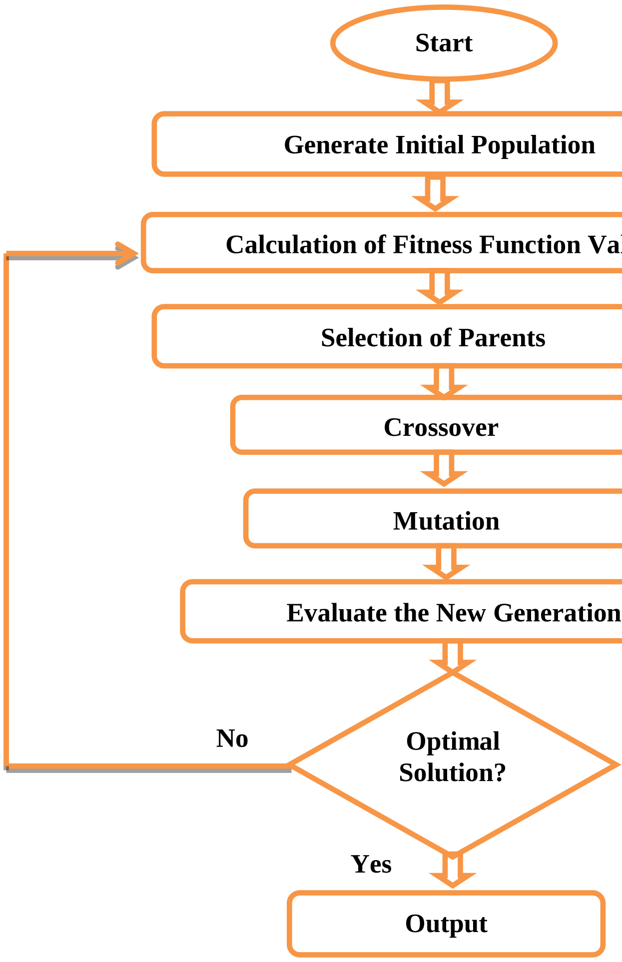

Fig.1 flowchart of proposed of GA

Sequence of jobs on each machine

Machine A: 1-2-3

Machine B: 4-5-6-7

Machine C: 8-9

1	2	3	*	4	5	6	7	*	8	9

Fig.2 chromosome value encoding

Parent1	A	B	C	D	E	F	G	H	I
Parent2	1	2	3	4	5	6	7	8	9
offspring	A	B	C	4	5	6	G	H	I

Fig.3 The crossover operation

(a)

Parent	1	2	3	*	4	5	6	7	*	8	9
Offspring	3	2	1	*	4	5	6	7	*	8	9

(b)

Parent	1	2	3	*	4	5	6	7	*	8	9
Offspring	1	2	3	*	8	5	6	7	*	4	9

Fig.4 swap two jobs within (a) on machine (b) different machine

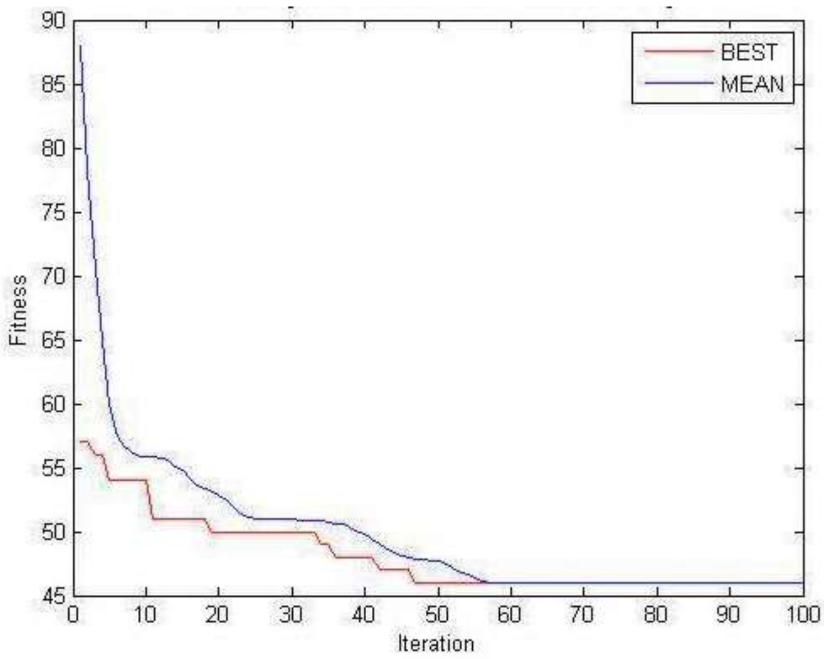

Fig.5 completion time of the best makespan solution

Table (1) genetic algorithm factors

Factor	(Min, Max)
Population size	(10,150)
Crossover probability	(0.5, 0.9)
Mutation probability	(0.0001, 0.01)

Table (2) genetic algorithm factors

Population size	50
Crossover probability	0.5
Mutation probability	0.01

Table (3) processing time and setup time

number of machine (3) number of jobs (11)											
Jobs	1	2	3	4	5	6	7	8	9	10	11
Processing time Pi_1	11	12	6	12	10	5	7	9	12	12	6
Processing time Pi_2	12	12	8	11	6	8	12	11	12	10	5
Processing time Pi_3	11	12	10	11	10	8	10	6	10	5	7
Setup time on machine 1 From/to	1	2	3	4	5	6	7	8	9	10	11
1	5	6	7	8	6	7	7	7	10	6	8
2	5	7	10	5	8	8	5	5	5	6	7
3	9	7	7	5	7	5	9	6	9	5	5
4	9	8	8	6	7	5	6	10	9	5	6
5	6	9	6	10	9	8	8	5	10	10	5
6	10	9	9	6	8	9	5	9	5	8	6
7	5	6	6	9	8	10	8	8	7	8	6
8	7	9	8	6	10	5	6	10	6	5	7
9	7	8	9	10	6	8	8	5	9	10	5
10	9	5	10	6	9	5	9	5	10	7	10
11	9	5	10	6	9	5	9	5	10	7	10
Setup time on machine 2 From/to	1	2	3	4	5	6	7	8	9	10	11
1	7	5	8	8	9	8	6	6	9	6	5
2	7	10	9	9	9	6	10	8	5	7	6

3	7	10	8	5	8	6	7	9	10	8	7
4	10	8	7	10	7	7	6	6	9	9	9
5	7	5	8	9	9	6	10	5	7	7	5
6	5	6	6	7	8	10	10	6	8	7	9
7	9	7	9	7	7	6	7	6	6	10	5
8	7	9	6	7	10	6	5	7	7	5	8
9	6	5	9	6	10	6	6	8	10	10	7
10	7	5	6	8	8	6	7	5	8	10	9
11	5	6	7	8	8	7	8	6	8	9	9

Setup time on machine 3 From/to	1	2	3	4	5	6	7	8	9	10	11
1	10	8	9	9	5	6	10	8	9	7	10
2	10	10	8	5	6	7	5	8	6	5	9
3	7	9	7	8	7	5	9	9	7	8	7
4	9	8	5	8	9	10	6	8	7	6	8
5	6	6	9	10	9	6	7	6	10	7	5
6	5	6	5	8	5	5	8	5	5	8	10
7	9	10	5	9	7	7	10	10	10	6	10
8	8	5	8	7	8	6	7	6	8	6	9
9	7	7	5	7	7	7	10	5	7	8	6
10	10	6	9	9	8	7	6	8	6	6	8
11	8	10	9	5	8	10	9	10	7	9	5

Table (4) best scheduling solution

Machines	Scheduled Jobs
M1	6 7 1 3
M2	2 4 5
M3	11 9 8 10

REFERENCES

Alcan P., Başlı H., " An Application with Non-identical Parallel Machines using Genetic Algorithm with the Help of Fuzzy Logic" , Proceedings of the World Congress on Engineering, Vol II, London, U.K, 2011.

Ak B., Koc E., " A Guide for Genetic Algorithm Based on Parallel Machine Scheduling and Flexible Job-Shop Scheduling" Procedia - Social and Behavioral Sciences, Vol 62, PP 817 – 823, Elsevier, 2012.

Chaudhry I.A., Mahmood S.," Identical Parallel-Machine Scheduling and Worker Assignment Problem Using Genetic Algorithms to Minimize Makespan" Electrical Engineering and Applied Computing, Springer Science+Business Media B.V, 2011.

Chiang T-C, Cheng H-C, and Fu L-C, "A Memetic Algorithm for Minimizing Total Weighted Tardiness on Parallel Batch Machines with Incompatible Job Families and Dynamic Job Arrival", Computer Operation Research, 37:2257-2269, 2010.

Chaudhry I.A., Mahmood S., and Ahmad R.," Minimizing Makespan for Machine Scheduling and Worker Assignment Problem in Identical Parallel Models using GA" Proceedings of the world congress on engineering , Vol III pp 2464–2469, WCE, London, UK, 2010.

Djamel Nait Tahar_, Farouk Yalaoui, Chengbin Chu, and Lionel Amodeo," A Linear Programming Approach for Identical Parallel Machine Scheduling with Job Splitting and Sequence-Dependent Setup Times", Int. J. Production Economics 99: 63–73, 2006.

German Y., Badi I. , Bakir A., Shetwan A., " Scheduling to Minimize Makespan on Identical Parallel Machines" International Journal of Scientific & Engineering Research, Vol. 7, Issue 3, 2016.

Jouglet A., Savourey D., "Dominance Rules for the Parallel Machine Total Weighted Tardiness Scheduling Problem with Release Dates", Computer Operation Research, 38:1259-1266, 2011.

Jiang L., Pei J., Liu X., Pardalos P. M., Yang Y., Qian X., "Uniform Parallel Batch Machines Scheduling Considering Transportation using a Hybrid DPSO-GA Algorithm" International Journal Advance Manufacturing Technology, 89:1887–1900, 2017.
Muralidhar A., "Multi-objective Optimization of "parallel machine scheduling" Using Neural Networks", Vol .2, Issue 2, 2013.

Maciel Manoel Queiroz, André Bergsten Mendes, "Strategies for scheduling jobs in an identical parallel machine production environment", African Journal of Business Management, Vol. 7(16), pp. 1553-1559, 2013.

Pie- Chann Chang, "Integrating dominance properties with genetic algorithms for "parallel machine scheduling" issues with setup times", Applied Soft Computing 11:1263- 1274, 2011.

Sahoo, "Genetic algorithm based multi-objective reliability optimization in interval environment", Computers & Industrial Engineering 62 :152- 160, Elsevier, 2012.

Selvi V. ,"Multi Objective Optimization Issues On Identical Parallel Machine Scheduling Using Genetic Algorithms", International Journal on Recent Researches in Science, Engineering and Technology, Vol.2,Issue 7, 2014.

Savas Balin , "Non-identical parallel machine scheduling using genetic algorithm", Expert Systems with Applications 38 :6814–6821, 2011.

Tavakkoli-Moghaddam R., Taheri F., Bazzazi M., Izadi M.,and Sassani F., "Design of a genetic algorithm for bi-objective unrelated parallel machines scheduling with sequence-dependent setup times and precedence constraints", Computers & Operations Research 36 :3224 – 3230, Elsevier Ltd, 2009.

Tavakkoli-Moghaddam R., Mehdizadeh E., "A new ILP model foe identical parallel machine scheduling with family setup times minimizing the total weighted flow time by a genetic algorithm", IJE Transactions A: Basics Vol. 20, No. 2, 2007.