Job Scheduling with Heuristic Techniques

Parvinder Kaur
Asstt. Professor,
Department of Computer Science,
Guru Nanak College,
Batala.

Amritpal Singh
Asstt. Professor,
Department of Computer Science,
Guru Nanak College,
Batala.

Amandeep Kaur
Asstt. Professor,
Department of Computer Science,
Guru Nanak College,
Batala.

Kiranpreet Kaur
Asstt. Professor,
Department of Computer Science,
Guru Nanak College,
Batala.

ABSTRACT

Job of parallel processing is to increase the execution time of a large job by frament them into small and manageable sub-jobs that run concurrently. Every sub-job has manages its own processor so output speed can be increased.By applying this methodology large calculations can be easily solved which have time contraints.Jobs can be parallel programmed using the concept of the scheduling. The execution time can be reduced by managing the jobs in this way.The proposed technique is capableschedule the execution on parallel system that proceedswith perfomance charactersitice and structure of the application.The scheduling problem is solved using heuristice and Number of appriximations. Comparisions have been also proposed between them. For optimization of the result well known NP-complete problem in job scheduling is used. GA (Genetic Alogrithm) part of a fast heuritic technique, has been proposed to schedule all the parallel jobs in homogeneous parallel environment.Comparision has been done between Shortest Job First (SJF), First Come First Serve (FCFS), and Round Robin (RR) scheduling techniques. The comaprison of all the algorithm returns the optimized result and selection of the best optimized selection is done.

Keywords

Scheduling, Parellel Processing,Task Scheduling, Directed Acyclic Graph, Heuristics Serach, Genetic Algorithm,Operating System.

1. BRIEF INTRODUCTION

On a set of homogenous processor when different tasks are run on parellel processor is known as job scheduling [1]. The determine the sequence of task on each and every processor [2]. The processing time of each task depends upon order and sequence [3]. The total finishing time also effected by processor perfomance [4].The major components under consideration for job scheduling are:

- The sequence of jobs in a specific order.
- The number of processors used.
- Check the performance homogeneous processors and evaluation them.
- Scheduling and Mapping and of the tasks onto the processors [5].

These four components play a vital role in searching and optimizing problem. To calculate optimized result these four components have to be taken together due to their dependency upon each other.Job scheduling have a combination of homogeneous processor in a parellel envoriment which is proposed with the help of Genetic Algorithm [6]. A Genetic Algorithm is also known as Heuritic Technique which able to generate indiviuals. In GA, a set of genes are created which further generates a string known as individuals. The strings are also known as chromosome [7]. The gives us an optimized output. From other set of indiviuals a set of other generation chromosome are generated. This whole process is done by the fitness function [8]. Based on these attributes such as fitness function, population of chromosome, probability of occurance of crossover, selection criteria and probability of occurance of mutation.These attributes are used for the optimization of job scheduling [9].

This paper is divided into five parts. In, Section I Brief introduction is discussed. Section II gives problem defination along with concise description of the methodology. In the section III proposed heuristic technique is explained. Section IV dicusses the experimental results and performance analysis of the study. At end in the section V conclusion are defined.

II. PROBLEM DEFINITION

A short review of classification of parallel algorithms, job scheduling and method based on the, the availability of the information, characteristics of the tasks to be scheduled and the parallel environment [10]. The main concept divide the major job into set of sub jobs can be run on homogenious parellel system [11].This will gernerate order of the task which can be represented in a graphical form [12]. The DAG(Directed Acyclic Graph) [13] model can be used to reprsent divided sub tasks. The major purpose is to joint out the determine scheduling method to array the jobs in parellel processing system [14]. A deterministic scheduling is that in which all information for jobs and relations with each other are known by scheduling algorithm in advance [15]. The setup of homogeneous processors defines that the processors may have equal processing speeds for finding the exact results in time [16]. A study has been done on the deterministic task scheduling model in the homogeneous parallel multiprocessor environment [17]. This will offeres us the correctness of results [18]. The main aim of using these type of model is to decrease the makespan. It will help in lessen the total finish time of each job in the parallel multtiprocessor system [19]. The waiting time and execution time in parallel environment is combination of total finish time [20]. By considering a set of m homogeneous multiprocessors in the parallel multiprocessor computing environment, an derived equation can be reprsented as written below:

$$T = \{t_{i:}\ i = 1, 2, 3 \ldots m\}$$

In figure 1, fully connected processors with via identical links or communication lines of three parallel system is shown with the help of a DAG. A Directed Acyclic Graph has the parallel application connected with links and can be represented by a set of nodes N and links L. Let by taking Gr = (V, L, W, C), where the vertices set V consist of m tasks and are represented as:

$$V = \{t_{j:}\ j = 1, 2, 3 \ldots n\}$$

A directed edge set D consist of u edges and all are denoted as:

$$D = \{e_{u,:}\ u = 1, 2, 3 \ldots r\}$$

Both V and D are deeply interconnected in the parallel multiprocessor system [21].

Figure 1: A fully connected having three parallel processor

Above figure 1 it is clear the precedence relationships among tasks in multiprocessor network enviroiment. Two task can be signify having link with the directed edges [22]. These edge generate a system that has no cycle. By taking the linkage between two differnt jobs occuring in a order as task J_1 and next job J_2 and these two jobs are from the subset task group D. Two jobs will be connected by a directed edge e_1 [23]. So here in the chances of these two jobs reprsent that job J_2 cannot be scheduled until J_1 has been completed. In othere words job J_2 is a predecessor of job J_1i.e. Job J_2 is a successor of job J_1. Similarly the job j_2, j_3......j_n connectivity with edges e_2,e_3,....e_n can be represented which produce a Directed Acyclic Graph having tasks and edges. From communication point of view every successor job j_m sends a message having a set of contents according to the requirement of next job i.e. to the next job t_n. This will produce execution set up between jobs in this graph [24]. Every edge has a weight or the data label between two jobs can be symbolize by the term X. It defines the execution duration the corresponding job and are differ from processor to processor in the multiprocessor parallel system [25]. Every elements set has the weight between nodes of vertices of the graph in the Directed Acyclic Graph and is symbolize as shown in the equation below as:

$$X = \{X_{i,j:}\ i = 1, 2, 3 \ldots m,\ j: 1, 2, 3, \ldots n\}$$

There is a data communication between the jobs. Data communication between the two jobs can be characterize by using this method i.e. if they are scheduled to different processors, but both jobs are scheduled to the identical processor, then the weight associated to the edge becomes null [26]. Figure 2 is an example of a complete Directed Acyclic Graph.The weight between a set of jobs be symbolized by the term communication cost PP and can be represented as:

$$PP = \{pp_{i:}\ i = 1, 2, 3 \ldots n\}$$

Figure 1, DAG naming D consist of a set of jobs J: $\{j_{j:}\ m = 1, 2 \ldots x\}$ and set of processors PR = $\{PR_i : i = 1, 2, 3,4,5\}$. The relationship between processor and jobs is defined by processor-task execution time matrix [27].This is shown in the Table 1. The execution time of each set of job vary due to different weight or communication cost in the system Processor display different processing speed in the homogeneous system. Let us consider that processor pr_1 is much faster than pr_2, pr_3 and so on. Processor pr_2 is faster than pr_3, pr_4 and so on. (i.e., the order of speed and processing capabilities can be expressed as pr_1>pr_2> pr_3> pr_4> pr_5). As shown in Table 1 job j_1 takes 4 time units to complete their execution on processor j_1 and takes 9 time units and 10 time units to complete their execution on processor pr_9 and pr_{10} respectively [28]. On the basis of the size of the jobs processed on same processors, the execution time has been calculated.

III. INTRODUCTION TO HEURISTIC (GENETIC) ALGORITHM

Every Genetic Algorithm passes through a cycle of phases and these phases given below:

- From the string population and creation of strings,
- Evaluation of Strings
- Strings selection
- To create a new population reproduction of strings.

The individuals are encoded in the population string known as chromosomes. It is likely to evaluate the of individuals in a population after cromosome has been coded. A good coding scheme [29] take advantage o foperators and make the object function easy to calculate. Every individual is given by a fitness value given by the objective function during the selection operation. Select the fittest individual of the current population to provide as parent of the next generation. Two prime opertor of the reproduction crossover and mutation.

Individual's fitness in a population generates a good coding model [30] which will help operators. Then the object function is easy to calculate. During selection, every individual is given a fitness value by the objective function. Selecting the fittest individual of the current population to represent as parent of the next generation. Reproduction have two types of operators mutation and crossover. The crossover operator select randomly a pair of individuals among those selected previously and exchange some part of the information. The mutation operator select an individual randomly and works with task duplication heuristics. in order to make total execution time of the schedule should be minimum[31].

i). Creation of the initial population:

The first step in the genetic algorithm. Parameters like number of tasks,number of processors and population size are required to produce initial population. The preliminary population is initialized with randomly created individuals. The length of all individuals in an initial population is equal to the number of tasks in the Directed Acyclic Graph. Every job is randomly allocated a processor [32].

ii) Proposed Gentic Alogrithm(GA):

It utilize the fitness function which is based on the total completion time for the schedule. It is combination execution time and communication delay time. The fitness function divided the evaluation into two pieces: processor fitness and task fitnes. The task fitness make sures that all jobs are performed and scheduled in valid order. A valid order defines that a precedence relations are satisfied nad successor task cannot planned until predecessor has been completed. The processor fitness element attempts to decrease the processing time. Consider the following jobs K1 and K2 for single processor and multiprocessor parallel system jobs schedules with task size equal to 50 jobs The processor chosen for scheduler K1 is c_1 and the execution time for all task are given in Table 1.

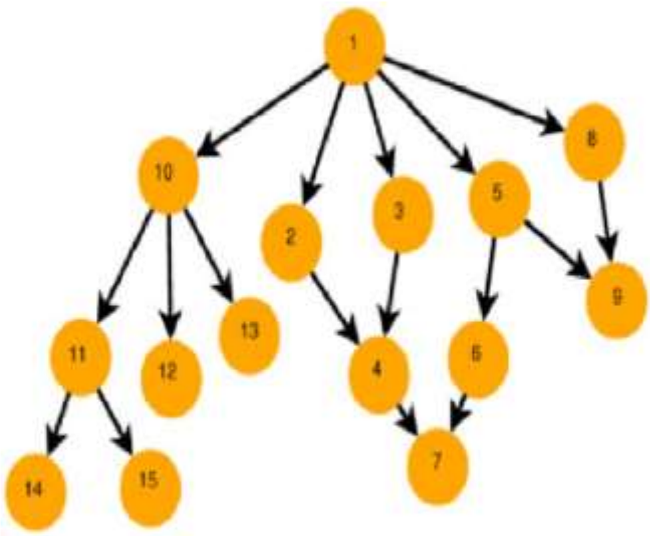

Figure 2: DAG of job size=15 with job precedence.

Kji's execution time of different jobs on different processors as shown in Table 1.

Table 1: Shows a job execution matrix on different processors with job size = 10.

JOBS / Processor	Job1	Job2	Job3	Job4	Job5	Job6	Job7	Job8	Job9	Job10
Processor 1	4	3	6	5	2	8	7	4	9	10
Processor 2	4	4	7	6	3	9	8	5	10	11
Processor 3	5	5	7	7	3	9	9	5	10	11
Processor 4	6	5	8	7	4	10	9	6	11	12
Processor 5	7	6	9	8	4	11	10	6	11	13

The total finish time of scheduler S1 and S2 is:

K1: $t_1 \rightarrow t_2 \rightarrow t_3$ ---------------------------$\rightarrow t_9 \rightarrow t_{10}$

Total Finish Time = Execution time + Comm. time. =4+3+6+5+2+8+7+4+9+10=58 time units

All the jobs are executed on same processor, thats why communication time is zero. The processors selected for schedule K2 are same as given in table 1. Here proposed Genetic algorithm use jobs duplication heuristics to lessen the data response time of its descendant jobsF, permitting them to start in advance.

K2: Total finish time = Execution time + Comm. Time = 42time units

The scheduler K1 demonstrate a total finish time of 58 time units, whereas scheduler K2 illustrate a total finish time of just 42 time units. Therefore, fitness function helps in decrease the total finish time. Fitness values (task and processor) have been assessed for all chromosomes and the probability of higher fitness is to be selected for reproduction from current generation to the next generation.

iii) Selection operation:

This is the basic design of fitness function, so how to design the fitness function will directly affect the performance of genetic algorithm. To select the superior and eliminate the inferior, GA uses the selection operator. According to their fitness value individual are selected. Once fitness values have been evaluated for all chromosomes, we can select good chromosomes through rotating roulette wheel strategy. This operator generate next generationby selecting best chromosomes from parents and offspring[33].

iv) Crossover operator:

This operator randomly choose two parent chromosomes (chromosomes with higher values have more chance to be selected) and randomly select their crossover points and make them to generate two child (offspring) chromosomes. We have checked one and two point crossover operators. In first crossover, the segments to the right of the crossover points are exchanged to form two offspring as shown in Figure 3(a) and in second crossover [34], the middle portions of the crossover points are exchanged to form two offspring as shown in Figure 3(b).

v) Reduction of Idle time:

By decreasing the idle time of a processor, waiting for the data from other processors based on that a mutation operation is designed. It works by randomly selecting two jobs and swapping them with each other. In first step, it randomly selects a processor, and then randomly selects a job on that processor. This job is the first job of the pair to be swapped. Secondly, At random selects a second processor and randomly selects a job. If the two selected jobs are the same jobs the search continues on. If the two jobs are dissimilar then they are swapped over.

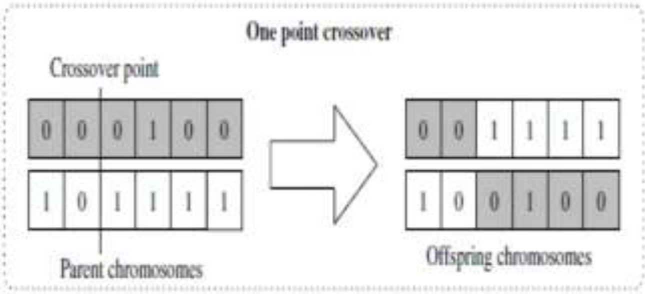

Figure 3(a): One Point Crossover

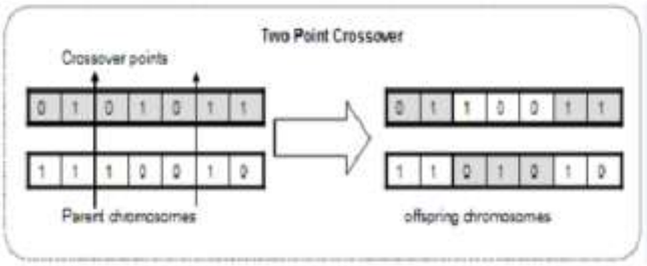

Figure 3(b): Two point crossover

The steps of the Genetic Algorithm (GA) are as follows:

Step 1. Study the DAG.

Step 2. Position the parameters.

Step 3. Initialize list of chromosome having its p-size after choosing the chromosome randomly.

Step 4. Calculate Fitness function on the basis of number of processor and number of jobs.

Step 5. Crossover swapping on the decided population.

Step 6. Mutation process/ swap mutation process on chromosome selected.

Step 7. Hheuristic operation applied at the end.

Step 8. Testing procedure and mapping is done at last.

III. RESULTS AND DISCUSSION

The performance comparisions with Shortest job First (SJF), First Come First Serve (FCFS) and Round Robin (RR) scheduling method on parallel systems and execution of the schedules are shown in Figure4 and 5 for job size equal to 10.

Performance analysis:

Speed up (SP_{sp}): Speed up is defined as the completion time on a uniprocessor divided by completion time on a multiprocessor. In case of homogeneous system, it is denoted as: SP_{sp} = p(1)/p(m). But in case of heterogeneous system, it is denoted as SP_{sp}= (min (p(1)) / p(m) i.e., the best uniprocessor completion time divided by the completion time on a heterogeneous multiprocessor system. The speedup is measured with the execution of tasks on single processor which shows 58 time units for job size equal 10 jobs divided by execution time units on PGA, SJF, FCFS and Round Robin (RR) scheduler as shown in Figure 5.

Efficiency (¢):

(T_{sp} / m), where m is the number of processors.

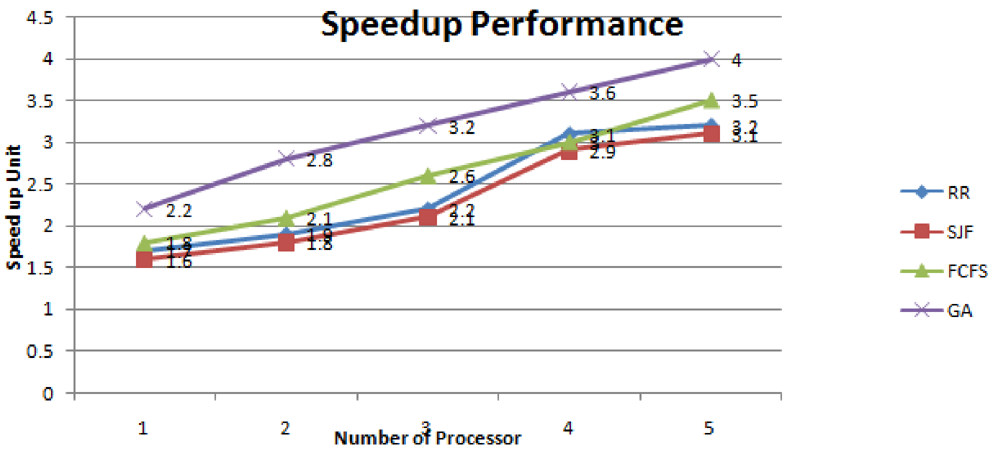

Figure 4: Speedup v/s number of parallel multiprocessor system for job size = 10 jobs

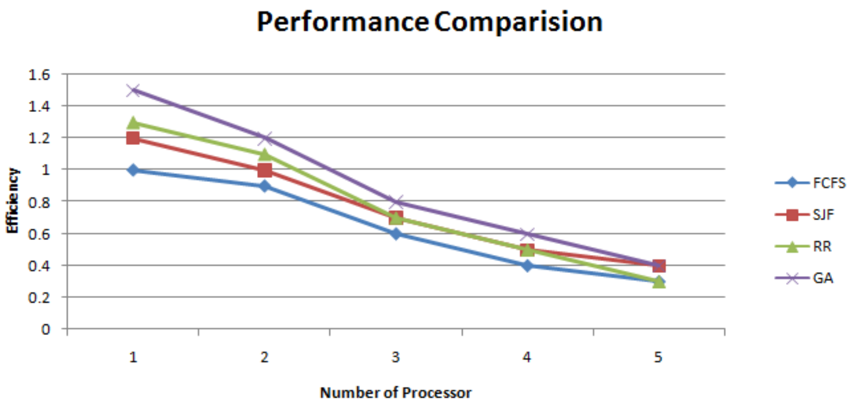

Figure 5: Performance comparisons of the GA, SJF, FCFS, and RR for job size = 10 tasks

III. CONCLUSION

The Heuristic Genetic Algorithm (GA) has been proposed . This proposed technique helps in scheduling homogeneous parallel multiprocessor system to lessen thewaiting or idle time and execution time. It also help to increase the throughput of the system. The proposed system is a better method for allocating the jobs to the homogeneous parallel multiprocessor system. The propsed technique is compared with SJF, FCFS and Round Robin (RR) scheduling techniques.

REFERENCES

[1] Sara Baase, Allen Van Gelder,"Computer Algorithms", Published by Addison Wesley, 2000.

[2] SartajSahni, "Algorithms Analysis and Design", Published by Galgotia Publications Pvt. Ltd., New Delhi, 1996.

[3] Anup Kumar, Sub Ramakrishnan, ChinarDeshpande, Larry Dunning, " IEEE Conference on Parallel processing", 1994, Page no.83-87.

[4] AnanthGrama, GeorageKarypis, Anshul Gupta, VipinKumar,"Introduction to parallel computing", Published by Pearson Education, 2009.

[5] KalyanmoyDeb,"Optimization for Engineering Design", Published by PHI, 2003.

[6] DezsoSima, Terence Fountain, Peter Kacsuk, " Advanced Computer Architectures", Published by Pearson Education, 2009.

[7] David E. Goldberg,"Genetic Algorithms in Search, Optimization and Machine Learning", Published by Pearson Education, 2004, Page No.60-83.

[8] Mitchell, Melanie," An Introduction to Genetic Algorithm, Published Bu MIT Press 1996

[9] Michael J Qumn, "Parallel Computing Theory and Practices, 2nd Edition", Published by Tata McGraw Hill Education Private Ltd, Page No. 346-364.

[10] Michael J Qumn, "Parallel Programming", Published by Tata McGraw Hill Education Private Ltd, Page No. 63-89.

[11] John L Hennessy, David A Pattern, "Computer Architecture, 3rd Edition", Published by Morgan Kaufmann & Elsevier India, Page No. 528-590.

[12] David E Culler, "Parallel Computer Architecture", Published by Morgan Kaufmann & Elsevier India.

[13] J P Hayes, "Computer Architecture and Organization", Published by McGraw Hill International Edition.

[14] J D Carpinalli, "Computer System Organization & Architecture", Published by Pearson Education.

[15] Sung-Ho Woo, Sung-Bong Yang, Shin-Dug Kim, and tack-Don Han ," IEEE Trans on parallel System, 1997, Page 301-305.

[16] M.SalmaniJelodar, S.N.Fakhraie, S.M.fakharie, M.N. Ahmadabadi," IEEE Proceeding", 2006, Page No 340-347.

[17] Man Lin and Laurence tianruoYang,"IEEE Proceeding", 1999, Page No 382-387

[18] Yajun Li, Yuhang yang, Maode Ma, Rongbo Zhu, " IEEE Proceeding", 2008.

[19] YI-Wen Zhong, Jian-Gang Yang, Heng-Nlan QI," IEEE Proceeding", August 2004, Page No. 2463-2468.

[20] Imtiaz Ahmad, Muhammad K. Dhodhi and ArifGhafoor," IEEE Proceeding", 1995, Page No. 49-53.

[21] CeydaOguz and M.FikretErcan," IEEE Proceeding", 2004, Page No. 168-170

[22] Kai Hwang & Faye A. Briggs,"Computer Architecture and Parallel Processing", Published by McGraw Hill ,1985.Page No. 445-47, 612.

[23] Andrei R. &Arjan J.C. van Gemund, "Fast and Effective Task Scheduling in Heterogeneous Systems", IEEE Proceeding, 2000.

[24] Michael Bohler, Frank Moore, Yi Pan, "Improved Multiprocessor Task Scheduling Using Genetic Algorithms", Proceedings of the Twelfth International FLAIRS Conference, 1999.

[25] Andrew J. page," Adaptive Scheduling in Heterogeneous Distributed Computing System", IEEE Proceeding, 2004.

[26] Jameela Al-Jaroodi, Nader Mohamed, Hong Jiang and David Swanson,"Modeling Parallel Applications Performance on Heterogeneous Systems", Proceedings of the International Parallel and Distributed Processing Symposium (IPDPS'03)

[27] Yu-Kwong and Ishraq Ahmad, "Static Scheduling Algorithms for Allocating Directed Task Graphsto Multiprocessors", ACM Computing Surveys, Vol. 31, No. 4, December 1999

[28] Wai-Yip Chan and Chi-Kwong Li, "Scheduling Tasks in DAG to Heterogeneous Processor System", Proceeding IEEE 1998.

[29] Sharma, M., Singh, G. and Singh, R., 2019. Design of GA and ontology based NLP frameworks for online opinion mining. Recent Patents on Engineering, 13(2), pp.159-165.

[30] C.P Ravikumar, A.K Gupta," IEEE proc. Comput. Digit. Tech.", Vol 142, No. 2, March 1995, Page No. 81-86.

[31] Jasbir Singh, Gurvinder Singh, "Task Scheduling using Performance Effective GeneticAlgorithm for Parallel Heterogeneous System", International Journal of Computer Science and Telecommunications [Volume 3, Issue 3, March 2012].

[32] Lee, Jaesung, and Dae-Won Kim. "An effective initialization method for genetic algorithm-based robot path planning using a directed acyclic graph." *Information Sciences* 332 (2016): 1-18.

[33] Feng, Lin, and Yang Qi-Wen. "Improved genetic operator for genetic algorithm." *Journal of Zhejiang University-SCIENCE A* 3 (2002): 431-434.

[34] Alhijawi, Bushra, and Arafat Awajan. "Genetic algorithms: Theory, genetic operators, solutions, and applications." *Evolutionary Intelligence* 17, no. 3 (2024): 1245-1256.