Volume 8, No. 4, May 2017 (Special Issue)
ISSN No. 0976-5697
International Journal of Advanced Research in Computer Science
REVIEW ARTICLE
Available Online at www.ijarcs.info

Various Approaches in Enhancing Scheduling Mechanism

Mini Singh Ahuja
Assitant Professor
GNDU RC Gurdaspur
Punjab, India
miniahuja06@gmail.com

Mehmapreet Kaur
Student M.Tech (CSE)
GNDU RC Gurdaspur
Punjab, India
sodhimehma@gmail.com

***Abstract*:** Complex computational environment exists where jobs cannot be processed by single processor. Multi cluster environment is one such environment in which computational intensive task exists. Task secluding multi cluster environment is critical factor. Users from various locations submit their tasks to multi clusters. This provides the challenge to schedule the resources among the tasks. An efficient task scheduling increases the performance of considered environment. The proposed work deals with the study of various techniques used for job scheduling in multi cluster environment. The comparative survey of techniques is presented so that optimal technique can be selected for future

***Keywords*:** Computational Environment, Multi Cluster, Task Scheduling, Resource

1. INTRODUCTION

Scheduling in multi cluster environment is critical area of research. Legion of scheduling algorithms are playing a part to schedule resources in multi cluster environment. Task can arrive from geographical large area and this present a challenge that which task must be given a resource at first place. In multi cluster environment Computer, data, and other resources are shared[1]. The service provider and consumers must agree upon what to be shared within given environment. The global nature of resource sharing is unique facility provided within multi cluster environment. The resource may be present within different administrative domain and demanded by node belonging to some other domain[2].
Scheduling is the process of assigning resources to jobs based on objective functions defined. Type of scheduling depends upon the objective function associated with the resource. Scheduling resources has following phases associated with it.

- Resource Discovery
- Resource Filtering
- Resource Selection
- Resource Scheduling Policy

Before allocation within multi cluster environment resources must be discovered. Resource may or may not be available[1]. Hence this phase becomes critical for monitoring of resource within the system[3]. The available resources must be checked to determine whether they satisfy the requirements or not. Hence filtering is compulsory. Resource selection out of available resources is next phase. This phase is critical since out of available resources of same type resource with optimal condition is selected for allocation. This is required so that job can be completed well within time. Healthier resource selection is the target of this phase. Scheduling policy decides whether the resource allocation is primitive or not. Resource allocation is said to be primitive if once allocation resource can be prompted from task even if it is not yet fully completed.
It is possible that multiple objective functions are associated with the resource. Multi heuristic job scheduling algorithms are followed in such situations. Proposed work deals with the study of various algorithms used for scheduling resources among jobs. Rest of the paper is organized as follows: Section II provides details of various algorithms used for job allocation. Section III provides comparison of various techniques used for scheduling, section IV provides conclusion and future scope whereas section V provides references used within the proposed work.

II. RESOURCE SCHEDULING ALGORITHMS

The critical part of parallel system is the mechanism of distributing jobs within the parallel systems. Legions of algorithms are utilized for this purpose. The algorithms which take part in the parallel system are described in this section.

- *BAT ALGORITHM*

It is a meta heuristic algorithm used to optimally allocate resources to tasks. It is based on echolocation behavior of BATS in terms of varying pulse rate of emission and loudness. BAT algorithm has the advantage of simplicity and flexibility. Because of simplicity and flexibility BAT algorithm can be used to solve wide range of problems[4].

- *CUCKOO ALGORITHM*

This algorithm is also meta heuristic in nature. Cuckoo algorithm starts with the initial population. The cuckoo lays its eggs and then competition to survive begins. Cuckoo lays its eggs in nest of other birds. Hence cuckoo has to preserve eggs from the birds whose nest it uses. The process continues and at the end cuckoo with same profit values survive and algorithm converges[5].

- *FIFO Job Scheduling*

In this job scheduling algorithm job is distributed to the processor within the system on the first come first serve basis. This algorithm may or may not yield optimal solutions.[6] The allocated processor releases the job when the job burst time finishes. This algorithm is strictly non primitive in nature. It is hence rarely utilized in the parallel environment.

- *TIME SHARING*

In this scheduling time is shared among multiple jobs. The time sharing system utilizes time quantum. The processor is switched among the processors based on time quantum. The process

CONFERENCE PAPERS
National Conference on Emerging Trends on Engineering & Technology (ETET-2017)
On 21st April 2017
University Inst. of Engg. & Tech. & University Inst. of Computer, SBBS University, Punjab (India)

continues until all the jobs finish execution. The time sharing system involves states such as waiting, active and ready. The time scheduling on parallel system can be implemented using local scheduling. The processing node has processors associated with them. Threads ready to be executed are placed within first come first serve buffer.[7] When the processor is available thread is fetched from the queue and executed. The time sharing environment generally adopt pipeline concept for executing instructions concurrently within uniprocessor systems.

- *ANT COLONY OPTIMIZATION*

This algorithm is considered one of the best algorithms in order to schedule the resources and allocate jobs to the processors. The characteristics of ants are followed in this case. The ants communicate with each other and let the information spread to detect optimal path. The base of parallel ant colony is implemented using parallel construction phase. In parallel ant colony algorithm multiple colonies are built simultaneously. The output of all the colonies is compared with each other. The output generated by colonies is checked for optimality. [7]
The only problem with the ACO is the convergence of ant colony algorithm is slow. The distance covered using this algorithm is less. In order to resolve the problem honey bee algorithm can be utilized.

- *HONEY BEE ALGORITHM*

This algorithm utilizes better features of Ant Colonies algorithm along with high distance capabilities. The Honey bee algorithm utilizes optimal path finding along with distance coverage. It utilizes the foraging system associated with honey bees to find the path out of available alternatives. This algorithm converges much faster as compared to existing ant colony algorithm.[8]

- *GANG SCHEDULING*

This scheduling produces optimal result as compared to all other parallel scheduling algorithms. This algorithm is primitive in nature. Hence deadlock never occurs within the system. The tasks that form a job are grouped together and then scheduled in this approach. The job priority is also considered in this case. The job with the highest priority is executed at first place in this case. In the early system this approach produces optimal results. But nowadays more advanced algorithms exists which produces better result as compared to this algorithm. [9]

- *GENETIC ALGORITHM FOR PARALLEL SCHEDULING*

The genetic algorithm has phases associated with it. The algorithm continues until optimal result is obtained. The phases associated with genetic algorithms are initialization, mutation crossover etc. The genetic algorithm goes through the generations and in case of complex problems consumes large amount of time. So for smaller problems genetic algorithm is not preferred. The genetic algorithm terminates when solution reaches satisfactory level. The prescribed tolerance hence plays critical part in terminating genetic algorithm. [10]–[12]
Ant colony and honey bee algorithm can be used in order to enhance job scheduling performance. The ant colony algorithm can be used to select path which is optimal in nature. The ant colony algorithm however is distance dependent. In order to resolve the problem honey bee algorithm can be used. Honey bee may not always give optimal path but speed is enhanced considerably. The honey bee algorithm is not distance dependent. Parallel ant colony algorithm is need of the hour in which multiple process can be executed at the same time [7], [8], [13]. Combination of BAT and CUCKOO algorithm hybridization can be used in future end ever since it is not yet demonstrated and can become hot topic of research in future

III. Comparison of Various Job Scheduling Algorithms

The following table provides the detailed comparison of various job scheduling algorithms along with their time complexities.

Table 1: Comparison of various Job Scheduling Algorithms

S. NO	***Technique***	***Parameters of scheduling***	***Facts Discovered***	***Methodology***	***Advantages***	***Disadvantages***
1	FIRST COME FIRST SERVE[2]	Submission Time, Processing Time, Finish Time	.Utilization is low for Resources .Complexity O(n)	MATLAB, NS2,CTC SP2	.Easy To Implement .Good speed	.Throughput can be low .Less Turnaround time,
2	BACKFILLING ALGORITHM[14]	Job Burst Time	.Shortest Job Move Ahead Of Other Jobs .Complexity is O(log n)	Simulation Is Event Based	.High throughput .low Turnaround time	.Complex in nature
3	CONSERVATIE BACKFILLING ALGORITHM[15]	Arrival Time, Processing Time, Finish Time	.Resource Utilization is Improved .Complexity is O(log n)	Simulation Is Event Based	.Resource Consumption is Enhanced.	.Low throughput
4	GANG SCHEDULING[9], [16]	Average Time Intervalare Considered	.Average Waiting Time .Complexity O(log n)	Simulation Is Discrete In Nature	.Reduced waiting time .Reduced turnaround time	.Resource utilization is poor
5	GENETIC ALGORITHM[10], [17]–[19]	Chromosomes	.Best For Large Complexity .Complexity O(n)	MATLAB, PYTHON	.Resources are utilized better	.High complexity .Time consumption is high
6	ANT COLONY[7], [20]	Burst Time, Number of jobs	.Best For Optimal Path Search, .Low Distance Handling .Complexity O(n)	MATLAB	.Resource discover with high accuracy	.Low distance levels can be handled
7	HONEY BEE[21], [22]	Burst Time, Number Of Jobs, Distance	.Best For High Distances .Complexity O(n)	MATLAB	.High distance levels can be handled	.Convergence rate is low
8	AMBF[23]	Front And	.Required Keeping	Trace Driven	.Low compl	.Throughput is reduced

		Back End Virtual Machines	Track of Backfilling Jobs .Complexity O(n2)	Simulation	exity .High convergence rate is observed	
9	PAIRED GANG SCHEDULING[24]	Burst Time, Number OF Jobs	.Improve Utilization of Resources .Complexity O(Log n)	Implemented In Cluster Based Environment	.Resource utilization is improved	.Time complexity is high
10	BUDDY ALLOCATION MECHANISM[23]	Jobs Are Partitioned	.Improve Utilization Of Resources .Complexity O(log n)	Cloudsim	.Complexity of allocation is reduced	.Waiting time and turnaround time is enhanced

IV. CONCLUSION AND FUTURE SCOPE

Job allocation and scheduling in multi cluster environment is critical. Allocation and scheduling must be performed carefully so that Makespan and Flowtime can be reduced. Waiting time associated with the task must also be reduced. This is possible only if correct algorithm can be selected for the job. Hybrid approach by combining various optimal algorithms can be used to optimize scheduling. In the future work we will design new hybrid algorithm which will combine the best features of BAT and Cuckoo algorithm which willgenerate optimal solution .

REFERENCES

[1] T. Kokilavani and D. I. G. Amalarethinam, "Reduced Makespan Task Scheduling Algorithm for Grid Computing," vol. 9, no. 27, pp. 71–76, 2016.

[2] V. Hamscher, U. Schwiegelshohn, A. Streit, R. Yahyapour, R. Buyya, and M. Baker, "Evaluation of Job-Scheduling Strategies for Grid Computing," *Grid Comput.*, vol. 1971, pp. 191–202, 2000.

[3] C. Wu, J. Li, D. Xu, P.-C. Yew, J. Li, and Z. Wang, "FPS: A Fair-Progress Process Scheduling Policy on Shared-Memory Multiprocessors," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 2, pp. 444–454, Feb. 2015.

[4] A. Hossein and G. X. Yang, "Bat algorithm for constrained optimization tasks," pp. 1239–1255, 2013.

[5] R. Rajabioun, "Cuckoo Optimization Algorithm," *Appl. Soft Comput.*, vol. 11, no. 8, pp. 5508–5518, 2011.

[6] I. J. I. Systems, M. S. Garshasbi, M. Effatparvar, and A. Branch, "High Performance Scheduling in Parallel Heterogeneous Multiprocessor Systems Using Evolutionary Algorithms," no. October, pp. 89–95, 2013.

[7] P. Delisle and P. Delisle, "Parallel Ant Colony Optimization : Algorithmic Parallel Ant Optimization : Algorithmic Models Models and Colony Hardware Implementations and Hardware Implementations."

[8] B. Yuce, M. S. Packianather, E. Mastrocinque, D. T. Pham, A. Lambiase, and T. Parade, "Honey Bees Inspired Optimization Method: The Bees Algorithm," pp. 646–662, 2013.

[9] F. A. B. Silva, E. P. Lopes, E. P. L. Aude, F. Mendes, T. C. Júlio, H. Serdeira, M. Martins, and W. Cirne, "Response Time Analysis of Gang Scheduling for Real Time Systems."

[10] E. Gabaldon, J. L. Lerida, F. Guirado, and J. Planes, "Multi-criteria genetic algorithm applied to scheduling in multi-cluster environments," *J. Simul.*, vol. 9, no. 4, pp. 287–295, 2015.

[11] W. Abdulal, A. Jabas, S. Ramachandram, and O. Al Jadaan, "Task Scheduling in Grid Environment Using Simulated Annealing and Genetic Algorithm," 2012.

[12] S. G. Ahmad, C. S. Liew, E. U. Munir, T. F. Ang, and S. U. Khan, "A hybrid genetic algorithm for optimization of scheduling workflow applications in heterogeneous computing systems," *J. Parallel Distrib. Comput.*, vol. 87, pp. 80–90, 2016.

[13] R. Singh, "Task Scheduling in Parallel Systems using Genetic Algorithm," vol. 108, no. 16, pp. 34–40, 2014.

[14] F. Xhafa and A. Abraham, "Meta-heuristics for grid scheduling problems," … *Sched. Distrib. Comput.* …, pp. 1–37, 2008.

[15] A. Mishra, S. Mishra, and D. S. Kushwaha, "An Improved Backfilling Algorithm : SJF-BF," vol. 05, no. 01, 2011.

[16] L. R. Dror G. Feitelson, D. G. Feitelson, and L. Rudolph, "Parallel Job Scheduling: Issues and Approaches," *Jsspp*, vol. 949, pp. 1–18, 1995.

[17] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.

[18] R. Nedunchezhian and P.Vivekanandan, "a Fast Genetic Algorithm for Mining Classification Rules in," *Soft Comput.*, vol. 1, no. 1, pp. 10–20, 2010.

[19] J. Carretero, F. Xhafa, and A. Abraham, "Genetic Algorithm Based Schedulers for Grid Computing Systems," *Int. J. Innov. Comput. Inf. Control*, vol. 3, no. 6, pp. 1–19, 2007.

[20] D. Maruthanayagam and R. UmaRani, "Enhanced Ant Colony Algorithm for Grid Scheduling," *Int. J. Comput. Technol. Appl.*, vol. 1, no. 1, pp. 43–53, 2010.

[21] M. Wang and W. Zeng, "A comparison of four popular heuristics for task scheduling problem in computational grid," *2010 6th Int. Conf. Wirel. Commun. Netw. Mob. Comput. WiCOM 2010*, pp. 3–6, 2010.

[22] S. Sharma, A. Chhabra, and S. Sharma, "Comparative Analysis of Scheduling Algorithms for Grid Computing," pp. 349–354, 2015.

[23] A. Kousalya and R. Radhakrishnan, "A Comparative Study of Parallel Job Scheduling Algorithms in Cloud Computing," vol. 6, no. 3, pp. 2687–2690, 2015.

[24] H. D. Karatza, "SCHEDULING GANGS IN A DISTRIBUTED SYSTEM," vol. 7, no. 1, pp. 15–22,

CONFERENCE PAPERS
National Conference on Emerging Trends on Engineering & Technology (ETET-2017)
On 21st April 2017
University Inst. of Engg. & Tech. & University Inst. of Computer, SBBS University, Punjab (India)