DOI: http://dx.doi.org/10.26483/ijarcs.v9i2.5835
Volume 9, No. 2, March-April 2018
ISSN No. 0976-5697
International Journal of Advanced Research in Computer Science
RESEARCH PAPER
Available Online at www.ijarcs.info

LIST SCHEDULING ALGORITHMS CLASSIFICATION: AN ANALYTICAL STUDY

Akanksha, Dr. Nirmal Kaur and Ravreet Kaur
Department of Computer Science and Engineering
UIET, Panjab University,
Chandigarh, India

Abstract: Multiprocessor scheduling is another name of task scheduling in form of algorithms that are mostly utilized in systematic as well as engineering appliance that is also known as the issue of NP-complete. The main aim of scheduling is the reduction of execution time. The illustration of task scheduling for multiprocessor scheduling is shown by DAG (Directed Acyclic graph). The categorization for this is into Static as well as dynamic scheduling. The list task scheduling is the example of static task scheduling algorithm. Varied task scheduling algorithms, like ISH, HLFET, MCP, ETF, CNPT and DLS are reviewed in this paper. The comparison of list task scheduling isdependent on metrics, termed as SLR, load balancing, efficiency and speed up.

Keywords: Parallel processing, list scheduling, DAG (Directed Acyclic Graph), heuristic based algorithm

I. INTRODUCTION

There is a lot of demand for high speed computing in different application areas. Because of the increasing power of computing, more and more power search has started increasing. A huge issue couldn't be explained by means of sequential machine in some span of time; therefore, a parallel machine with number of issues has been categorized and has been assign to associate problems on separate processors [1].

Parallel computing is most effective method for fulfilling the calculative constraint for different engineering and scientific applications. Two causes are there for the attractiveness and popularity of parallel computing:

- Lessing computer hardware cost
- Application performance that cannot be executed by conservative computers

Task allocation for the accessible processors for precedence limitation amid tasks for the processors is the aim of task scheduling. In this, the execution time should be less. The representation of application plan is characterized by DAG (Directed Acyclic Graph).Task scheduling is known as NP-completed for some of the restricted cases [2].Two variants are there, which are connected with task scheduling algorithm. Initial variant is either for judging the presence of communication time and the subsequent variant can be multiprocessor or homogenous system. It is consisted of three components:

- Performance of Homogenous processor
- Task mapping on processors
- Execution sequence for task on every processor.

II. DAG (DIRECTED ACYCLIC GRAPH)

DAG is also termed as TG (Task-Graph) and be considered as general model for parallel program composed of vertices and edges set. DAG is consisted of four tuples, G= {V,E,W,D} in which every vertex, that is V={T1,T2,…, Tn} is taken as graph's task. E as directed edges seteij depicting the two tasksdependency, Ti plusTj. Every graph edge is connected by weight D, known as communication time/ communication cost [3].

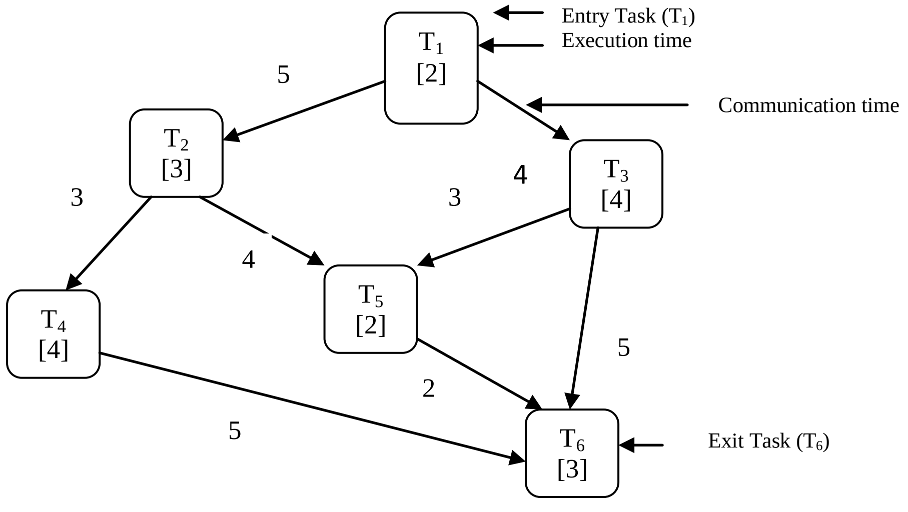

Figure 1. DAG model having six tasks

The representation of comm. of unication time is D (Ti, Tj). The mapping of task set on set P= (P1, P2,.. Pp) for p processors and every taskcan beimplemented on processor termed as computation time W and is termed as W (Ti) [4].

Communication time among two tasks is considered as zero on similar processor. Now, the precedence constraint constantly holds Tj that cannot be implemented till Ti that finishes when i<j. If the link among Ti and Tj is direct than Tj is considered as the Tisuccessor which is the Tjpredecessor. DAG Illustration has six nodes as illustrated in above figure 1 [5].

An entry task is the task in which there are no predecessors and the exit task is the task in which there are no successors. T1 is known as the entry task and T6 is the exit task.

Communication cost C(Ti, Tj)and computation time for specified DAG is defined below [6]:

Table I. Computation and communication time

	W (T1)=2	**W (T1)=2**	**W (T3)=4**
	W (T4)=4W (T5)=2	**W(T6)=3**	
C(T1, T2)=5	**C(T1, T3)=4**	**C(T2, T4)=3**	**C(T2, T5)=4**
C(T3, T5)=3	**C(T3, T6)=5**	**C(T4, T6)=5**	**C(T5, T6)=2**

III. LIST TASK SCHEDULING ALGORITHMS CLASSIFICATION

The classification of task scheduling is in two categories:

i. Deterministic task scheduling

ii. Non-deterministic task scheduling

Deterministic task scheduling is termed as "Static scheduling/compile time scheduling algorithm". The tasksinformation, like, task computation time and communication time with the tasks precedence constraintsare considered in this type[8].

Non-deterministic tasks are also termed as Run-time scheduling/ dynamic scheduling. For this, the task information is depicted only at the execution time [9].

Below figure is illustrating the classification of list task scheduling algorithms which came under heuristic dependent algorithms.

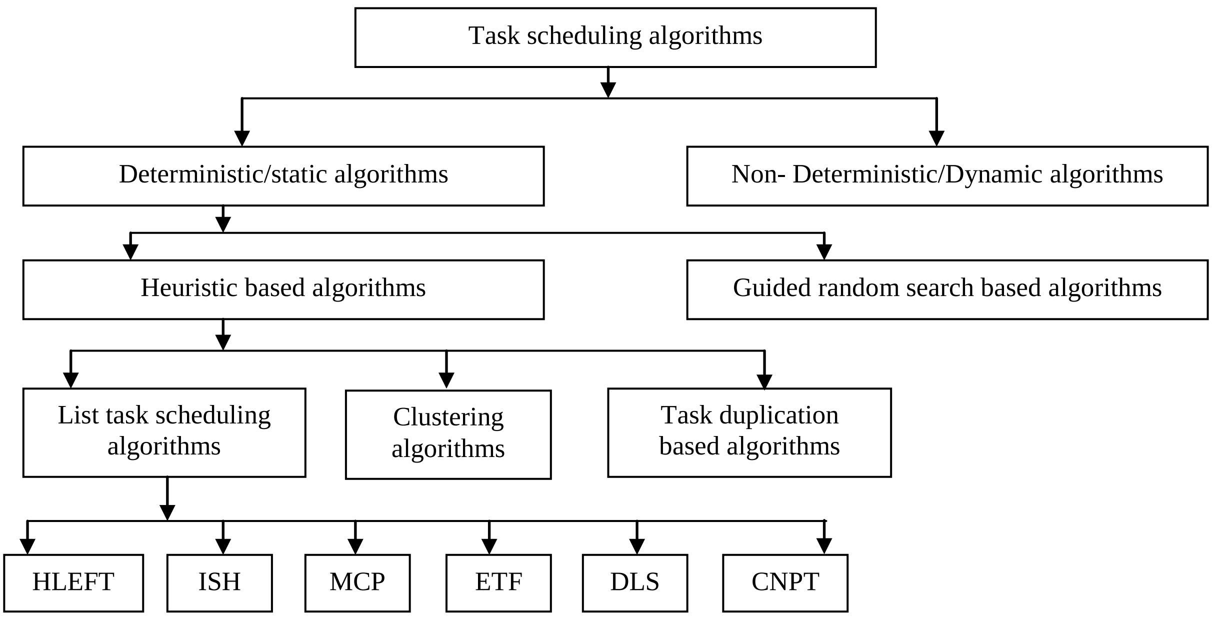

Figure 2. List scheduling algorithms classification

The task scheduling algorithms are categorized in deterministic and non-deterministic algorithms. The deterministic algorithm is divided in to heuristic with guided random search dependent algorithms. The heuristic dependent algorithm provides enhanced solution and satisfactoryperformance with the polynomial time complexity as compared to 'exponential time complexity'. It is further divided as: List task scheduling, clustering plus task duplication based algorithms [10].

'List task scheduling algorithms' is known as scheduling algorithms in which the assigning of tasks in done by DAG provided to processors. It is easy and provides less complexity as contrast to another algorithm. The clustering algorithm lessens the communication time among DAG tasks. It is an effective algorithm. It executes in two phases:

i. Task clustering
ii. Post clustering

Task duplication based algorithm provides better effectives and less scheduling length because of the reduction of communication time with the tasks. It assists for lessening the initialization time for the tasks that are waiting. These algorithms have an aim for processors usage in accurate time [11].

Table II. Task scheduling algorithms

Task scheduling algorithms	***Description***
HLFET (Highest level first with estimate time)	It is simple and known as the first algorithm of 'list task scheduling algorithm'. Task priority is chosen by the attributes of static levels. Consideration of communication is not taken place.
ISH (Insertion scheduling heuristic) algorithm	It is effective as it uses the appropriate time being developed by incomplete schedule on processors. The priority of task has been done with static b-level attribute. Provision of enhanced results in contract of HLFET algorithm is considered in this.
MCP (Modified Critical path) algorithm	It finds task priority by utilizing ALAP (As late as possible) attribute. It gives more priority to tasks that takes less start time. The main con is that it doesn't have communication time for task priority.
ETF (Earliest time first) algorithm	It finds the earliest start time for each task and later chooses the task having less initial time. Main limitation is that it reduces the scheduling length on each level
DLS (Dynamic level scheduling) algorithm	It finds the task priority on the tasks priority on dynamic basis. It is same as ETF algorithm but DLS utilizes DL attribute while ETF utilizes static level attribute. It doesn't sustain scheduling list on scheduling procedure.
CNPT (Critical node parent tree) algorithm	It achieves more accuracy and reduces complexity. The prioritization of task is determined with CN (Critical node) attribute. It has two stages; Listing plus Processor assigning phase. It has better performance as contrast to DLS, MCP plus ETF algorithms.

Below tables shows the time complexity of task-scheduling algorithms and priority attributes of task scheduling algorithms [12]. The algorithms considered are HLFET, ISH, MCP, ETF, DLS plus CNPT.

Table III.Task scheduling algorithms time complexity

S. No.	***Algorithms***	***Complexity***
1	HLFET	O (v2)
2	ISH	O (v2)
3	MCP	O (v2(logv)+p)
4	ETF	O (pv3)
5	DLS	O (pv3)
6	CNPT	O (v2)

Table IV. Priority Attribute of task scheduling algorithms

S. No.	***Algorithms***	***Priority attribute***
1	HLFET	Static-level
2	ISH	Static-level
3	MCP	ALAP
4	ETF	Static-level
5	DLS	Dynamic-level
6	CNPT	Critical-node

IV. COMPARATIVE METRICS

The analysis of performance could be executed in list task scheduling algorithms on the basis of comparison metrics, SLR (Scheduling length ratio), Speed up, Efficiency and Load balancing [13-14].

Table V. Task scheduling comparative metrics

Metrics	*Description*
SLR (Scheduling length ratio)	It is the time considered for executing on critical path as SL lower bound. For normalizing the SL for the lower bound, it can be described as: $\mathbf{SLR} = \frac{\mathbf{Makespan}}{\mathbf{Criticalpath}}$
Speed up	It is the proportion among 'sequential execution time' with 'parallel execution time' in which the sequential time execution time as the amount of total computation time for every task with'parallel time execution time' which is the SL on less amount of processors. $\mathbf{Speedup} = \frac{\sum_{i=1}^{n} \mathbf{Ti}}{\mathbf{Tp}}$ As shown, $\sum_{i=1}^{n} \mathbf{Ti}$ is the amount of computational task time in chronological order as 1,2,3,…n and $\mathbf{Tp}$ is schedule length and total parallel execution time of DAG
Efficiency	It is the measurement of processor utilization. Mathematically, it can be described as: $\mathbf{efficiency} = \frac{\mathbf{S_p}}{\mathbf{N_p}}$ As shown, $\mathbf{S_p}$ is the speed up and $\mathbf{N_p}$ is the number of processors
Load balancing	It can be described with the proportion of scheduling length to average execution time on each processor.

V. RELATED WORK

This section describes the work till date for different scheduling techniques for fulfilling varied QoS (Quality of service) metrics with energy saving methods.

Table VI. A glance of existing techniques

Author	*Proposed work*	*Research Gap*
Abdul Razaque et al.[1]	Task Scheduling in Cloud Computing on the basis of carbon footprint	It has smaller amount flexibility and less reliability approachfor more execution time.
Hao Wu et al. [2]	Optimization of Deadline Constrained DAG Applications by using VM (Virtual machine) concept.	The accessible task model is relied on task's execution times be hard for efficiently calculating in a cloud environment. Otherissue is the cloud environment virtualization overhead for proposed algorithm that could be improved on the virtualization overhead basis.
Amandeep Verma et al.[3]	Cost and Time aware Scheduling strategy for Workflow application being executed in Cloud	Because of less optimal schedule plan for real cloud environment, the computational cost is additional and there is a possibility of enhancement in the enhanced schedule planning.
Hamid Arabnejad et al.[4]	Scheduling Algorithm on the basis of budget constraint provided by user for Workflow Applications	Because of requirement of dynamic concurrent DAG scheduling issue, that executeparallel workflows which cannot execute together but might distribute the resources for total cost for the user that could be lessen
Jia Yu and R. Buyya[5]	DAG dependent scheduling for budget constraint satisfaction by metaheuristic genetic algorithm on efficacy grids	Usage of optimization method has not considered for solving QoSconstraints for security and reliability.
Wei Zheng et al.[6]	'Budget-Deadline Constrained Workflow Planning used for Admission Control for Bi-criteria DAG scheduling'	The computational complexity of presented work is higher so the success rate is reduced and in presented work, the middle DAG scheduling heuristic technique is required.
Zhuo Tang et al.[7]	DVFS enable energy effective workflow task scheduling	Power consumption lessens by 46.5% but slacked makespanenhances.
Weihong Chen et al.[8]	Effective Task Scheduling used for Budget Constrained Parallel appliance on Heterogeneous Cloud Computing scheme	The issue of proposed work is only appropriate for the homogeneous cloud environment.
JasrajMeena et al. [9]	Cost Effective GA in favor of Workflow Scheduling in Cloud in Deadline Constraint	There is a big issue of shutdown time of VMs and due to the general execution workflow cost is affected. Due to the absence of optimal schedule plan for a real cloud environment, the computational cost is more and there is a chance of improvisation in the optimal schedule planning.
Anton Beloglazov et al.[10]	Heuristics of energy awareness in resource allocation for effective data center management	Difficult to run on large-scale and at large-scale energy consumption is more. There is no any concept of the generic resource manager.

VI. CONCLUSION

The list task scheduling algorithms classification HLFET, ISH, MCP, DLS, ETF, and CNPT algorithms which are of homogenous environment has been studied and analyzed in this paper. The pros and cons of time complexity are considered. The algorithms are dependent on few priority attributes. As per priority attributes, the assigning of priority is taken place. Some computation metrics, namely, SLR, load balancing, efficiency and speed up have been studied that provides the assistance for differentiating the algorithms. The list scheduling algorithms provides more effectiveness and less SL than another scheduling algorithm.

VII. REFERENCES

[1] Abdul Razaque, Nikhileshwara Reddy Vennapusa, “Task Scheduling in Cloud Computing”, Conference on Long Island Systems, Applications and Technology (LISAT), IEEE, pp. 1-5, 2016.

[2] H. Wu, X. Hua, Z. Li and S. Ren, “Resource and Instance Hour Minimization for Deadline Constrained DAG Applications Using Computer Clouds”, *IEEE Transactions on Parallel and Distributed Systems*, Vol. 27, Number 3, pp. 885-899, 2016.

[3] Amandeep Verma and Sakshi Kaushal, “A hybrid multi-objective Particle Swarm Optimization for scientific workflow scheduling”, Journal of Parallel Computing, Elsevier, Vol. 62, pp. 1-19, 2017.

[4] Jorge G. Barbosa and Hamid Arabnejad, “A Budget Constrained Scheduling Algorithm for Workflow Applications”, Journal of Grid Computing, Springer, Vol.12, Issue 4, pp.665-679, 2014.

[5] Jia Yu and Rajkumar Buyya, “A budget-constrained scheduling of workflow applications on utility grids using genetic algorithms”, Workshop on Workflows in Support of Large-Scale Science, IEEE, pp. 1-10, 2006.

[6] Wei Zheng and Rizos Sakellariou, “Budget-deadline constrained workflow planning for admission control”, Journal of Grid Computing, Springer, Vol. 11, Issue 4, pp. 633-651,2013

[7] Zhuo Tang, Ling Qi, Zhenzhen Cheng, Kenli Li, Samee U. Khan, Keqin Li, “ An Energy-Efficient Task Scheduling Algorithm in DVFS-enabled Cloud Environment”, Journal of Grid Computing, Springer, Vol. 14, Issue 1, pp. 55-74, 2016.

[8] Weihong Chen, Guo Xie, Renfa Li, Yang Bai, chunnian Fan and Keqin Li, “Efficient task scheduling for budget constrained parallel applications on heterogeneous cloud computing systems”, Future Generation Computer System, Elsevier, Vol. 74, pp. 1-11,2017.

[9] Jasraj Meena, Malay Kumar, And Manu Vardhan, “Cost Effective Genetic Algorithm for Workflow Scheduling in Cloud Under Deadline Constraint”, IEEE Access, Vol.4, pp. 5065-5082, 2016.

[10] Beloglazov A, Abawajy J, R. Buyya, “Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing”, Future Generation Computer Systems, Elsevier, Vol. 31, Vol. 28, Issue 5, pp. 755-68, 2012.

[11] P. Mahalakshmi and K. Ganesan, “Mamdani Fuzzy Rule Based Model to Classify Sites for Aquaculture Development”, Indian Journal of Fisheries, Vol. 62, Issue 1, pp. 110-115, 2015.

[12] John Wiley & Sons, New York. Novak, V., “A Review of Fuzzy Sets and Their Applications”, pp 248,1989. Published[Online]. Available: http://www.polytech.univ-smb.fr/fileadmin/polytech_autres_sites/sites/listic/busefal/Papers/44.zip/44_35.pdf (Last Accessed on 7 July 2017)

[13] Altaf Ur Rahman, Fiaz Gul Khan, and Waqas Jadoon, “Energy Efficiency techniques in cloud computing”, International Journal of Computer Science and Information Security (IJCSIS), Vol. 14, Issue 6, June 2016.

[14] Singh, S., & Kaur, N, “A Heterogeneous Static Hierarchical Expected Completion Time Based Scheduling Algorithm in Multiprocessor System.,”International Research Journal of Engineering and Technology (IRJET), 2016, Vol, 3, pp 951-956.