IJCSE International Journal of Computer Sciences and Engineering Open Access
Review Paper Volume-6, Issue-3 E-ISSN: 2347-2693

Comprehensive study about different scheduling techniques for parallel applications in cloud computing

S. Rana[1*], A. Kumar[2]

[1*] Dept. Of CET, Guru Nanak Dev University, Amritsar, India
[2] Dept. Of CET, Guru Nanak Dev University, Amritsar, India

Corresponding Author: sadhna93rana@gmail.com, Tel.: 9463283202

Available online at: www.ijcseonline.org

Received: 14/Feb//2018, Revised: 22/Feb2018, Accepted: 16/Mar/2018, Published: 30/Mar/2018

Abstract— Parallel computing gives us an environment through which we can execute numerous assignments at the same time. It enables us to take care of enormous problem by separating it into multiple small problems. As energy utilization while satisfy deadline constraint by PCs has become a concern in recent years. This paper has exhibited a comprehensive review on the different swarm intelligence based energy efficient scheduling techniques. It has been observed that the scheduling in parallel condition is NP-hard in nature. The research on meta-heuristic based job scheduling methods have demonstrated that the utilization of Quick energy aware processor merging has low convergence rate overall world wide minimum even at high numbers of dimensions. Gravitational Search optimization algorithm has been generally acknowledged as a global optimization algorithm of current enthusiasm for disseminated advancement and control. Particle swarm optimization is constrained to beginning arrangement of particles, wrongly chosen particles tends to poor outcomes. Moreover, comparison among different job scheduling methods have displayed that no strategy is ideal for each case. At last, a few considerations about future challenges have been exhibited.

Keywords—Scheduling,Energy,DeadlineandPower

I. INTRODUCTION

Parallel computing provides us an environment through which we can execute many tasks simultaneously. It gives us opportunity to solve big problem by dividing that big problem into several smaller problems. As energy consumption while satisfying deadline constraint by computers has become a concern in recent years.

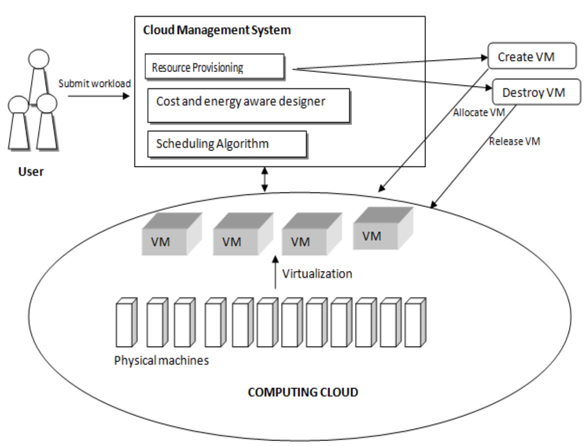

Figure1.Cloud computing environment for parallel applications

In computing, workflow is a general representation of different scientific applications in distributed system. Large workflow is representing by DAG (Directed Acyclic Graph) where nodes indicates task and edges presents data flow.

As in this paper, Section I contains the introduction, Section II contain the related work, Section III contain the methodology which defines taxonomy of various scheduling techniques, Section IV describes results and discussion with the help of tables, Section V concludes research work with future directions

II. RELATED WORK

Guoqi Xie, et al. (2017) [1] proposed two algorithms EPM and QEPM for vitality diminishment continuously parallel application in a heterogeneous distributed computing framework than different techniques like DEWSTS calculation. EPM centre around to pick best processor to kill for limiting vitality consume while satisfy deadline constraint and QEPM focus to reduce computation complexity.

Aeshah Alsughayyir, et al. (2016) [2] provided energy minimization for computation in multi-clouds uses DVFS when HPC tasks were under deadline constraint. They used algorithm Energy Aware Scheduling Algorithm and compare it with Cloud Min-Min Scheduling algorithm. Comparison shows that EAGS algorithm consumes low energy by average of 63.9%.

Mateusz Zotkiewicz, et al. (2016) [3] presented a DAG for online energy aware and communication scheduling strategy in SaaS. It contains two phases, first phase set virtual deadline of one task to central scheduler and these deadlines are determined by strategy that chooses that task which is less dependent on other. Second stage, jobs are allotted to servers according to load. It was basically implemented on Green Cloud Simulator.

Adam Gregory, et al. (2016) [4] provided resource manager for batch workload includes Map Reduce jobs. It empowers specialist organization to accomplish exchange off between vitality utilization, missed due date and executing time. The optimization problem in this paper is to manage the resources to solve he uses Constraint Programming.

Hao Li, et al. (2016) [5] used deadline requirement Vitality mindful calculation for planning the work process in heuristic approach for vitality utilization. For execution utilizes two sub calculations: undertaking mapping calculation for outline to their utility VMs and job merging algorithm for merging tasks in workflow which minimize time consume in data center .

Vahid Arabnejad, et al. (2015) [6] proposed PDC algorithm for E-science Scheduling for commercial clouds. PDC is used to minimize costs and deadline constraint. It had four phases : to separate greatest characteristic parallelism of work process; corresponding due date conveyance to parcel the client level characterize in first stage ; task determination as per need and occurrence choice from best as per execution. CloudSim simulation is used to implement PDC along with GAIN and IC-PCP.

Zhongjin Li, et al. (2015) [7] provided an algorithm for minimizing energy and cost .CEAS (cost and energy aware scheduling) uses CloudSim and four workflow applications. CEAS calculation utilizes five sub-calculations: VM choice calculation for mapping assignments to ideal VM for use cost; two undertakings are converged to limit cost and vitality utilization; VM reuse arrangement to use the VMs that are perfect; and undertaking slacking calculation to spare vitality by DVFS.

Kenli Li, et al. (2014) [8] proposed on scheduling BoT(Bags of Tasks) which was made up of independent stochastic tasks with execution time on heterogeneous platform with energy and deadline budget constraints on single processor. Here Clark's equation was used for find the variance of schedule length and expected value. ESTS calculation was utilized to for high planning execution BoT application with low time complexity.

Hamid Mohammadi Fard, et al. (2012) [10 proposed a general structure and heuristic calculation for multi-target static scheduling of work processes in heterogeneous registering situations. The calculation utilizes constraints indicated by the client for every target and approximates the ideal arrangement by applying a twofold system: augmenting the separation to the imperative vector for prevailing arrangements and limiting it generally. He presents four-objective makespan, economic cost, energy consumption, and reliability. The algorithm Played out a related bi-criteria scheduling heuristic and a bi-criteria genetic algorithm.

Qingjia Huang, et al.(2012) [11] provided an upgraded vitality proficient planning (EES) algorithm to diminish vitality utilization with SLA in Data centre running parallel applications towards the objective of guaranteeing the activity complete before the due date settled on before. Fundamental objective was to think about the slack space for the non-basic employments and endeavor to plan the tasks adjacent running on a uniform recurrence for worldwide optimality.

III. METHODOLOGY

Scheduling

Scheduling is the strategy by which work showed by a couple of means is allotted to scattered resources that aggregate the work. Scheduling makes it possible to have PC multitasking with a singular central getting ready unit (CPU). A scheduler may go for one of various targets, for example, increasing throughput (the total aggregate of work completed per time unit), restricting response (time from work getting the opportunity to be enabled until the point that the moment that the essential point it begins execution on resources), or constraining torpidity (the time between work getting the opportunity to be engaged and its subsequent fulfillment), boosting fairness (measure up to CPU time to every methodology, or all the more generally reasonable conditions as showed by the need and workload of every technique).

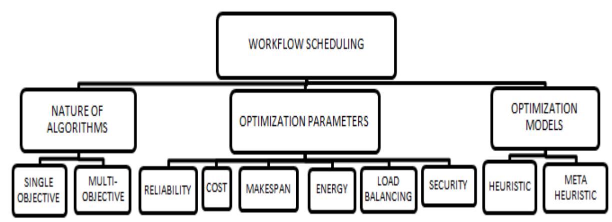

Figure2. Taxonomy of workflow scheduling

In Heterogeneous cloud computing system, DAG is used to represent real parallel applications like FFT, Diamond graph and Gaussian elimination. For suppliers, limiting the aggregate vitality utilization of an application is a standout amongst the most essential concerns. For client, the due date imperative of an application is a standout amongst the most vital nature of administration (QoS) necessity. To decrease vitality utilization in HCS, different methods including dynamic voltage-recurrence scaling (DVFS) and memory enhancement have been created. [1]Turning off processors is an accessible path in a few stages. DEWSTS calculation centre around both static and dynamic vitality utilization by turning off processor with modest number of tasks.EPM and QEPM calculations limit both static and dynamic vitality utilization by turn off best processor.

Energy consumption

Energy consumption is defined as the energy consumed by equipment when assignments are executed. DVFS algorithm calculation is for the most part used to lessen vitality .The power utilization for an application execution is made out of static and dynamic vitality utilization .Most work focuses on the dynamic energy consumption. Research into improving the energy consumption often focuses on the hardware, or programming techniques to improve energy efficiency. Real-time measurements are referred to as Dynamic Analysis, whereas Static Analysis means analyzing the code without necessarily having to run it.

Figure3. Types of Energy Consumption

***[13]*Types of Energy consumption**

Dynamic Energy :
Dynamic energy consumption refers to measure the energy draw of its hardware when operations are performed. This method poses several problems: the equipment and time required to perform the measurement are expensive, it takes a significant amount of time to run a sufficient number of tests, and the tests themselves have to be accurate and cover at least regular usage, preferably more. Nevertheless, a proper analysis meeting these requirements can be used to identify bottlenecks in both the code and actual hardware, or satisfy specific energy or time constraints. This option is attractive for businesses, those working with potentially dangerous equipment, and other users in charge of maintaining hardware on a large scale; case studies have been performed on varied hardware such as networking devices, mobile consumer electronics and CPU hardware.

Static Energy:
The investment required to perform dynamic analysis, and its susceptibility for errors, opens up demand for a faster analysis lacking many of the requirements posed by actual power measurement. This form of analysis, which focuses on analyzing code or systems without running it or activating them, is referred to as static analysis. While it does not share the disadvantages of a dynamic analysis, it also does not provide the same advantages: without real measurement, the result of a static analysis will always be approximate. Accuracy instead depends on how accurate the models and rules for reasoning about the energy consumption are. The main draw of this form of analysis is the ease of application. Rather than having to acquire measuring equipment and perform all the required steps for a dynamic analysis, it may be executed on any system, independent of the actual hardware it is running on. However, one problem not present with dynamic analysis is the fact that, in general, it is very difficult, if not impossible without intervention, to completely predict a program's behavior. Concessions have to be made to accuracy in order to be able to perform the analysis as best as possible. Several projects present that focus on static analysis of code, some of which are described in greater detail below. In particular, this thesis focuses on extending an existing analysis, and applying these extensions.

Processor merging:
Processor merging is defined as completely satisfy the requirements of the jobs submitted by user. Two types of methodology exist in parallel environment that is either split job into sub jobs so called tasks and processor merging. Since sometimes, it is not possible to divide job into tasks in such a case ,if required number of resources by given resources is more than available number of resources then to implement given job processor merging will be used. Example we assume that job demand 66 processors, and let be assume server A has 32 resources, B has 16 resources and C has 32 resources, then job is not divisible. So job must be divisible.

IV. RESULTS AND DISCUSSION

Table 1 Summary of distinguished related work ordered utilizing scientific categorization

Technique	Year	Objective	Energy	Deadline	Cost	Metrics
MOLS and Time Constraint Partioning [10]	2012	Multi - objective	✓	×	✓	Makespan , reliability
Energy-aware stochastic tasks scheduling algorithm (ESTS)[18]	2012	Bi-objective	✓	✓	×	Energy used
Enhanced energy-efficient scheduling (EES)	2012	Bi-objective	✓	✓	×	Schedule length and energy

Slack-time-aware two-phase scheduling framework ,ILP-based scheduler in offline[17]	2013	Bi - objective	✓	✓	×	Energy conservation
GMaP framework[9]	2013	Bi-objective	✓	✓	×	Energy cost and time
Energy-aware multi-job scheduling optimization model, bi-level genetic algorithm[12]	2014	Single	✓	×	×	Energy consumption
Energy-aware stochastic task scheduling algorithm ESTS[8]	2014	Bi-objective	✓	✓	×	Execution time and budget
Cat-Swarm-Optimization[14]	2014	Bi-objective	✓	×	✓	Cost and energy
Cost and energy aware scheduling (CEAS) [7]	2015	Multi-objective	✓	✓	✓	Makespan , cost
Proportional Deadline Constrained (PDC) [6]	2015	Bi-objective	×	✓	✓	Computation Cost and time
TaPRA and TaPRA-fast that solve the SJS problem. In online scheduling, OnTaPRA.[20]	2016	Single job scheduling and Online scheduling.	✓	×	×	Completion time
Constraint Programming, IBM CPLEX CP Optimizer[4]	2016	Bi-objective	✓	✓	×	Energy consumption and time complexity
Workflow Partitioning for Energy Minimization (WPEM) algorithm and Cat Swarm Optimization[24]	2016	Single objective	✓	×	×	Energy used
Energy aware scheduling (EAS)algorithm. The EAS comprises of two sub algorithms: job mapping algorithm and job blend algorithm[5]	2016	Bi-objective	✓	✓	×	Execution time and energy consume
Minimum Dependencies Energy-efficient DAG (MinD+ED) [3]	2016	Bi-objective	✓	✓	×	Execution time
Energy-aware scheduling algorithm EAGS[2]	2016	Bi-objective	✓	✓	×	The number of rejected applications
Fixed Frequency Island-Aware Largest	2017	Bi-objective	✓	✓	×	Deadline and energy

Task First (FFI-LTF) heterogeneous Island-Aware Largest Task First (HI-LTF) Fixed Frequency Island- and Task-Aware Largest Task First (FIT-LTF) [16]						
EPM and QEPM[1]	2017	Bi - objective	✓	✓	×	Computation time.

Table 2 Summarization of tools and applications

Technique	Software	Hardware	Application	Platform
MOLS and Time Constraint Partioning [10]	GroudSim	___	WIEN2k and MeteoAG	Grid and Cloud
Energy-aware stochastic tasks scheduling algorithm (ESTS)[18]	C++	___	BoT applications or parameter sweep applications,	Heterogeneous computing systems (HCS)
Enhanced energy-efficient scheduling (EES)	___	___	Gaussian Elimination algorithm and Random DAG Generator	Cloud
Slack-time-aware two-phase scheduling framework ,ILP-based scheduler in offline[17]	Online and offline	Four general purpose (GP) cores and four dedicated hardware accelerators (DHAs) and an L2 cache.	___	Multiprocessor systems-on-chips
GMaP framework[9]	Monte Carlo	___	___	Cloud
Energy-aware multi-job scheduling optimization model, bi-level genetic algorithm[12]	___	2.53 GHz Intel Xeon processor framework with 3.48 GB Smash running Windows XP.	MapReduce and Hadoop	Cloud
WSEPT (weighted shortest expected processing time) and Energy-aware stochastic task scheduling algorithm ESTS[8]	C++	___	Bag-of-tasks (BoT)	HCS
Cat-Swarm-Optimization[14]	Monte Carlo	___	___	Cloud
Cost and energy aware scheduling (CEAS) [7]	Cloudsim	___	Monotage, CyberShake, LIGO	Cloud
Proportional Deadline Constrained (PDC) [6]	CloudSim	___	EScience	Commercial clouds
TaPRA and TaPRA-fast that solve the SJS problem. In online scheduling, OnTaPRA.[20]	Offline and online simulations	___	SJS-Relax-LP and online scheduling	Cloud
Constraint Programming, IBM CPLEX CP Optimizer[4]	Java	3.2GHz Intel Center i7 CPU and 8GB of Slam window 7.2.6 GHz Intel CPU and 4 GB memory.	Map reduce	Cloud
Workflow Partitioning for Energy Minimization (WPEM) algorithm and Cat	CloudSim	___	____	Cloud

Swarm Optimization[24]				
Energy aware scheduling (EAS)algorithm. The EAS consists of two sub algorithms: task mapping algorithm and task merge algorithm[5]	CloudSim	___	Bio informatics and earth science,	Cloud
Minimum Dependencies Energy-efficient DAG (MinD+ED) [3]	GreenCloud	___	Online scheduling, SaaS applications	Cloud
Energy-aware scheduling algorithm EAGS[2]	SimJava	___	Iaas applications	Cloud
Fixed Frequency Island-Aware Largest Task First (FFI-LTF) heterogeneous Island-Aware Largest Task First (HI-LTF) Fixed Frequency Island- and Task-Aware Largest Task First (FIT-LTF)[16]	Gem5 and McPAT for the Alpha coresOdroid-XU3 platform for the A7 and A15 cores	Exynos 5 Octa (5422) processor with Cortex-A7 and Cortex-A15 islands, and also using gem5 and McPAT for out-of-order (OOO) Alpha 21264 and in-order simple Alpha 21264 islands	Blackscholes, bodytrack, ferret, swaptions, and x264, PARSEC	Cluster
EPM and QEPM[1]	Java	2.6 GHz Intel CPU and 4 GB memory	FFT, Diamond graph and Gaussian elimination	Cloud

V. CONCLUSION AND FUTURE SCOPE

This paper has exhibited a far reaching survey on the diverse swarm intelligent based energy effective scheduling strategies. It has been watched that the planning in parallel condition is NP-hard in nature. The examination on meta-heuristic based employment planning strategies have demonstrated that the utilization of Quick energy aware processor merging has low convergence rate over worldwide minimum least even at high quantities of measurements. Gravitational search enhancement algorithm has been generally acknowledged as a worldwide advancement algorithm of current enthusiasm for dispersed optimization and control. Particle swarm advancement is constrained to beginning arrangement of particles, wrongly chose particles tends to poor outcomes. In addition, comparison among various job scheduling techniques have presented. The comparative analysis obviously demonstrates that the no procedure is ideal for each case.

In this manner, in near future we will propose a new hybrid Gravitational search algorithm and Quick energy aware processor merging algorithm for cloud computing environment. We mainly focus on the objectives like Speedup, Execution time, Power consumption and Energy cost.

REFERENCES

[1] Guoqi Xie,“Energy-aware Processor Merging Algorithms for Deadline Constraint Parallel Application in Heterogeneous Cloud Computing.”IEEE Transactions on Sustainable Computing,Vol.2 Issue2,pp.62-75, April 2017.

[2] Aeshah Alsughayyir, “Energy Aware Scheduling of HPC Tasks in Decentralised Cloud Systems”IEEE24th Euromicro International Conference on Parallel,Distributed, and Network-Based Processing,pp.617-621,2016.

[3] M. Zotkiewicz,”Minimum dependencies energy-efficient scheduling in data centers," IEEE Transactions on Parallel and Distributed Systems, vol. 27, no. 12, pp. 3561-3574, 2016.

[4] Adam Gregory,“A Configurable Energy Aware Resource Management Technique for Optimization of Performance and Energy Consumption on Clouds”IEEE 8th International Conference on Cloud Computing Technology and Science,pp.184-192,2016.

[5] Hao Li,“Energy-aware Scheduling of Workflow in Cloud Center with Deadline Constraint”IEEE 12th International Conference on Computational Intelligence and Security,pp.415-418,2016.

[6] Vahid Arabnejad,“Cost Effective and Deadline Constrained Scientific Workflow Scheduling for Commercial Clouds”IEEE 14th International Symposium on Network Computing and Applications,pp.106-113, 2015.

[7] Zhongjin Li, “Cost and Energy Aware Scheduling Algorithm for Scientific Workflows with Deadline Constraint in Clouds” IEEE Transactions on services computing,vol.pp,Issue.99,pp.1-15,2015.

[8] K. Li, “Energy-e_cient stochastic task scheduling on heterogeneous computing systems," IEEE Transactions on Parallel and Distributed Systems,vol. 25, no. 11, pp. 2867-2876, 2014.

[9] Yue Gao,“An Energy and Deadline Aware Resource Provisioning, Scheduling and Optimization Framework for Cloud Systems” IEEE, 2013 .

[10] Hamid Mohammadi Fard,"A Multi-Objective Approach for Workflow Scheduling in Heterogeneous Environments" 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, pp.300-309,2012.

[11] Qingjia Huang,"Enhanced Energy-efficient Scheduling for Parallel Applications in Cloud" 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing,pp.781-786, 2012.

[12] X. Wang,"An energy-aware bi-level optimization model formulti-job scheduling problems under cloud computing," Springer Soft Computing, vol. 20,no. 1, pp. 303-317, 2016.

[13] Stein Keijzers, "Energy Consumption Analysis of Practical Programming Languages" Master thesis August ,2014.

[14] Saurabh Bilgaiyan,"Workflow Scheduling in Cloud Computing Environment Using Cat Swarm Optimization" IEEE International Advance Computing Conference (IACC), pp.680-685,2014.

[15] Xinghui Zhao,"Energy-aware resource allocation for multicores with per-core frequency scaling" Springer,Journal of Internet Services and Applications, pp.1-15,2014

[16] Santiago Pagani,"Energy Efficiency for Clustered Heterogeneous Multicores" IEEE Transactions on parallel and distributed systems, vol. 28, no. 5, May, 2017

[17] Xin Li,"Slack-time-aware Energy Efficient Scheduling for Multiprocessor SoCs" IEEE International Conference on High Performance Computing and Communications,pp.278-285,2013.

[18] Kenli Li,"Energy-Aware Scheduling Algorithm for Task Execution Cycles with Normal Distribution on Heterogeneous Computing Systems" IEEE 41st International Conference on Parallel Processing, pp.40-47,2012.

[19] Hamid Reza Naji,"A High-Speed, Performance-Optimization Algorithm Basedon a Gravitational Approach" IEEE,pp.56-62, 2012.

[20] Li Shi, "Energy-aware Scheduling of Embarrassingly Parallel Jobs and Resource Allocation in Cloud" IEEE Transactions on Parallel and Distributed Systems,vol.28,Issue 6,pp.1-14,2016.

[21] Anita Choudhary, "Workflow Scheduling algorithms in cloud environment:a Review, Taxonomy, and Challenges"IEEE International Conference on Parallel, Distributied and Grid Computing,pp.617-624,2016

[22] Dhananjay Thriuvady, "Parallel ant colony optimization for resource constrained job scheduling",Springer,Volume 242, Issue 2, pp 355–372 ,2014.

[23] Piotr Swistalski,"Scheduling parallel batch jobs in grids with evolutionary Metaheuristics". Springer , Volume 18, Issue 4, pp 345–357,2014.

[24] Khadija Bousselmi "Energy efficient partitioning and scheduling approach for Scientific Workflows in the Cloud" IEEE International Conference on Services Computing,pp.146-154,2016.

[25] A.Jain,"Cloud Scheduling using Meta Heuristic Algorithms"IJCSE,Vol.5,Issue10,pp.132-139,2017.

Authors Profile

Ms S Rana pursed Bachelor of Computer Science from CTIEMT, PTU University Jalandhar in 2015. She is currently pursuing Master of Computer Science in Department of CET, Guru Nanak Dev University, Amritsarsince 2018.Her main research work focuses on Cloud and parallel computing.

Mr A Kumar is currently pursuing Ph.D. and currently working as Assistant Professor in Department of CET, Guru Nanak Dev University of Amritsar since 2018. He has published more than 20 research papers in reputed international journals. His main research work focuses on Parallel computing. He has 15 years of teaching experience and 6years of Research Experience.

