A Comparative analysis among Task scheduling for grouped and ungrouped Grid Application

Alankrita Aggarwal[1], Shivani Gaba[2], Shally Nagpal[3]and Anoopa Arya[4]

[1,2,3,4] *Panipat Institute f Engineering and Technology, Samalkha-132101, Haryana India*

Abstract

Grid Computing is new, and inescapable innovation in the fields of logic, and designing, and just as in business, and modern undertakings. This innovation assists with associating all assets e.g. supercomputer, stockpiling frameworks. This paper centers mostly around the booking methodologies for gathered and ungrouped Grid applications. The planning system assumes a significant part in the lattice asset. A decent errand booking technique is expected to decrease the all-out time taken for work execution in the network.

Keywords

Grid Computing, Gridsim toolkit, Scheduling strategy

1. Introduction

Use Grid empowers the sharing, choice, and total of a wide assortment of assets including supercomputers, stockpiling frameworks, information sources, and concentrated gadgets that are topographically circulated and possessed by various associations for settling enormous scope computational and information escalated issues in science, designing, and business. The idea of Grid processing [1] [2] began as a venture to connect topographically scattered supercomputers, yet presently it has become a long way past its unique aim.

The remainder of this paper is coordinated as follows. Segment 2 arrangements with the extent of the Gridsim toolbox in this paper. Segment 3 arrangements with the plan of the engineering of the planning technique. Area 4 arrangements with the execution of gathered and ungrouped gridles with booking procedure three scheduler Section 5 depicts the execution of the assembled and ungrouped gridles. Area 6 portrays reenacted geography and recreation results are displayed in segment 6.

2. Related Work

2.1 Scope

This paper focuses on the scheduling strategy of Gridsim Toolkit. The Gridsim simulation sends, and receives the Gridlets to/from the grid resources in sequential manner; the user or broker entity gets a Gridlets from Gridlet list, submit the Gridlet to the grid resource, and receives the processed Gridlet from the resource.

Then, the broker [5][6] gets the second Gridlet from Gridlets list, and performs the subsequent steps. Sending, and receiving one Gridlet at one time increase the Gridlet transmitting time, and cost over the Grid network as well as overhead processing time, and cost of each Gridlet. Another point is that, if an individual Gridlets with small length is sent to a grid resource with large MIPS, the Gridlet

International Conference on Emerging Technologies: AI, IoT, and CPS for Science & Technology Applications, September 06–07, 2021, NITTTR Chandigarh, India
EMAIL: alankrita.agg@gmail.com(A. 1); sgsgknl@gmail.com (A. 2); shally.ngpl@gmail.com(A. 3); anoopa.arya71@gmail.com(A. 4)
ORCID: 0002-0931-1118 (A. 1); 0000-0002-7919-0702(A. 2); 0000-0002-1632-577X (A. 3)

processing effort at the grid resource would not utilize the maximum capability of that resource; this will result in wastage of network bandwidth in terms of grid transmitting cost and time.

2.2 Design

The proposed framework acknowledges the complete number of Girdles, normal million guidelines (MI) of those Gridles (Gridlet length), permitted deviation level of the MI, granularity season of reenactment, overhead handling time per Gridles (Gridlet overhead preparing time), and the matrix assets from the client. Then, at that point, it will introduce the Gridsim substances and the boundaries.

The framework then, at that point makes the network asset [4] as indicated by the client for preparing the Gridles. Subtleties of the framework assets are acquired from the document containing a rundown of assets with their attributes. Absolute PEs (handling components), and MIPS of every PE are made dependent on the subtleties found in the record and accordingly, every asset is constructed uniquely, in contrast, to document preparing MIPS. The Gridles are made on client-determined boundaries (absolute number of Gridles, normal MI of each Gridles, and the level of the deviation of the MI). Then, at that point, the framework begins the reenactment.

It first assembles the trait of the accessible matrix assets made in the asset creation area in the framework. Then, at that point, it duplicates the MIPS of every asset with granularity time. Then, at that point, bringing about esteem shows the absolute MI that a specific asset can measure in the given granularity time [6-7]. The framework then, at that point assembles the prepared Gridles from the mimicked network through its I/O line [8-11]. At last, the framework will show the prepared Gridles gatherings and a portion of their qualities to the client.

3. Implementation

Figure 1 portrays the design of the Gridlet scheduler that will be utilized in the proposed framework. The scheduler gets the attributes of Grid assets from the asset document determined by the client.

The asset MIPS, granularity time, and Gridlet length are utilized for the Gridlet gathering task. The gathering task brings about a rundown of Gridlet gatherings, and a rundown of assets [12- 16].Figure 2 shows Gridlet grouping task. In this scenario, 100 Gridlets with small lengths (MI) are divided into six subgroups according to available resources, and the granularity time.

Grid Resource Creation

The resource record contains an overview of open resources on the Grid. Each resource has its characteristics showed in the archive, specifically, resource name, plan, working structure, and different machines, number of PEs, MIPS of each PE, time area, planning cost, correspondence speed, sporadic seed, and resource load during the top hour, off-top hour, and event.

Gridlet scheduling algorithm

1. The scheduler receives the Gridlet_list created in the system.
2. The scheduler receives the resources_list provided by Grid Information Service.
3. Set the Total_length of Gridlet to zero.
4. Keep a record of the ID of the resource(whose MIPS is being used in the current comparison)
5. Set resource ID and job execution cost of each Grouped_Gridlets.
6. Store the collected Grouped_Gridlets into a new gridlet_list_3
7. Get the total simulation time.

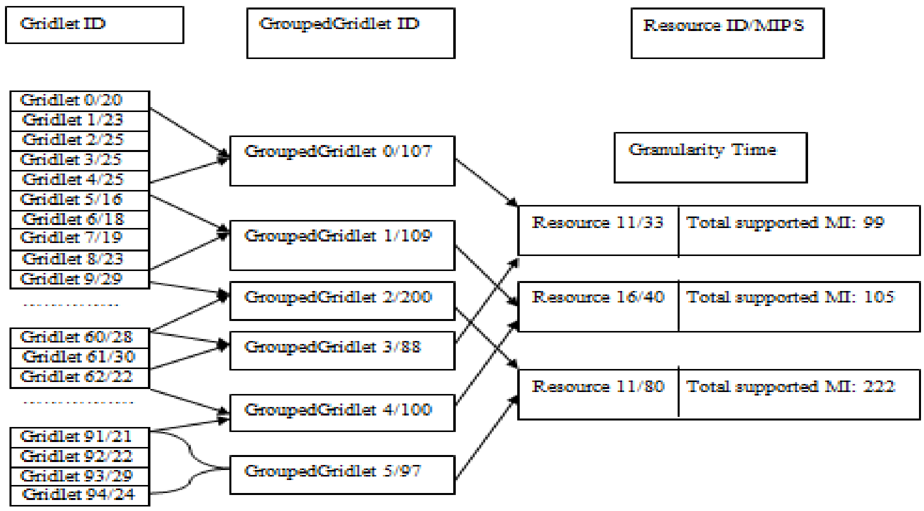

Figure 1: Gridlet Grouping Task

4. Results

Following are the tables and respective evaluation of the grouped and ungrouped data. The table is representing the values given for ungrouped and grouped data and simulations are done. Figure 2 and Figure 4 shows the simulation of the gridlet after the input data is given. Figure 3 represents the simulations when ungrouped and grouped data is given at the intervals of 10seconds.

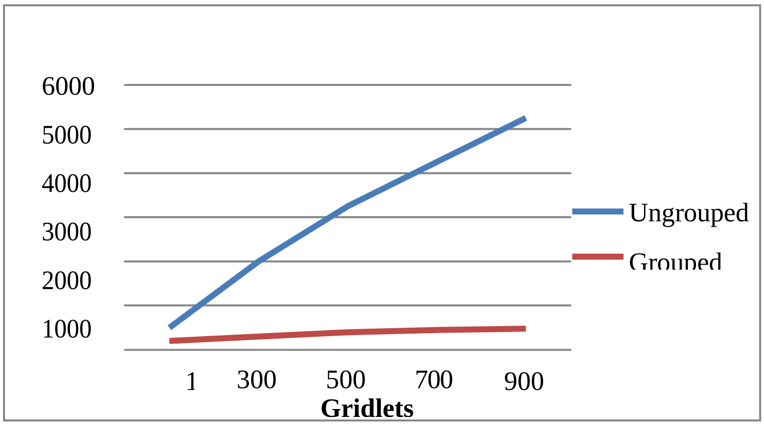

Figure 2: Simulation with and without Grouping for Granularity Time of 10 Seconds

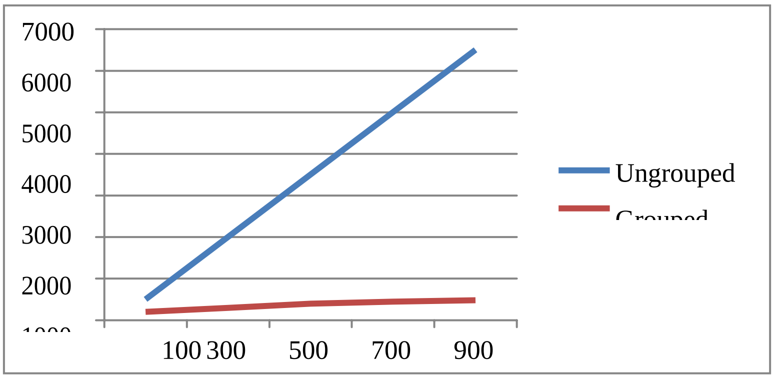

Figure 3: Simulation with and Without Grouping for Granularity Time of 15 Seconds

The absolute handling cost relies upon the MI, and all-out CPU time (each second) utilized for preparing the Gridlet.

5. Conclusion

In this paper, the positions are dealt with at the Grid resources. Each work is sent separately (one by one) to the Grid resource (s), dealt with only, and thereafter sent back to the customer. Sending, taking care of, and getting the positions one by one, forms the total amount of time expected to execute all of the situations from a customer. The supreme execution time wraps the best chance for sending each work to the Grid resource and the overhead dealing with time for each work at the Grid resource.

6. References

[1] Wang, Xin, Tapani Ahonen, and Jari Nurmi. "Applying CDMA technique to network-on-chip." IEEE transactions on very large scale integration (VLSI) systems 15.10 (2007): 1091-1100.

[2] P.K.Suri and Manpreet Singh, Analysis of Services, Challenges and Performance of a Grid, JCSNS International Journal of Computer Science and Network Security, VOL.7 No.7, July 2007.

[3] RajkumarBuyya and srikumarvenugopal A Gentle introduction of grid computing and technologies, published in 2005 in computer society of India.

[4] Anthony Sulistio and RajkumarBuyya, the GridSim Toolkit, Poster for the 3th International Conference on e-Science and Grid Computing (e-Science'07), Dec. 10-13, 2007, Bangalore, India.

[5] Agustin Caminero, Anthony Sulistio, Blanca Caminero, Carmen Carrion and RajkumarBuyya, Extending GridSim with an Architecture for Failure Detection, Proc. of the 13th International Conference on Parallel and Distributed Systems (ICPADS 2007), Dec. 5-7, 2007, Hsinchu,Taiwan.

[6] Anthony Sulistio and RajkumarBuyya, The GridSims: Real Tools for Simulated Parallel and Distributed Computing, International Science Grid This Week (iSGTW), Feature on 17 October2007.

[7] EwaDeelman, Gurmeet Singh, MironLivny, Bruce Berriman, and John Good, The Cost of Doing Science on the Cloud: The Montage Example, Proc. of the 2008 ACM/IEEE conference on Supercomputing (SC'08), Nov. 15-21, Austin, Texas,USA.

[8] Krzysztof Kurowski, JarekNabrzyski, Ariel Oleksiak and Jan Weglarz, Grid Scheduling Simulations with GSSIM, Workshop on Scheduling and Resource Management for Parallel and Distributed Systems (SRMPDS), Proc. of the 13th International Conference on Parallel and Distributed Systems (ICPADS 2007), Dec. 5-7, 2007, Hsinchu,Taiwan.

[9] Aggarwal, A., Dhindsa, K. S., & Suri, P. K. (2021). Performance-Aware Approach for Software Risk Management Using Random Forest Algorithm. International Journal of Software Innovation (IJSI), 9(1),12-19.

[10] Aggarwal, Neeraj MS Alankrita. "Load Balance Based on Scheduling and Virtual Machine." International Journal 3, no. 7(2013).

[11] Ruben, Mario, NithinSuryadevara, AnsgarPausch, Eufemio Pineda, JochenBürck, AnanyaBaksi, Tim Hochdörffer et al. "Chiral Resolution of Spin-Crossover Active Iron (II)[2x2]Grid Complexes." *Chemistry–A European Journal*(2021).

[12] Aggarwal A., Gaba S., Mittal M. (2021) A Comparative Investigation of Consensus Algorithms in Collaboration with IoT and Blockchain. In: Agrawal R., Gupta N. (eds) Transforming Cybersecurity Solutions using Blockchain. Blockchain Technologies.Springer, Singapore. https://doi.org/10.1007/978-981-33-6858-3_7.

[13] Li, Jian, De-Fu Yang, Yan-Chao Gao, and Xin Huang. "An adaptive sliding-mode resilient control strategy in smart grid under mixed attacks." *IET Control Theory & Applications*(2021).
[14] DK ShivaniGaba, ShifaliSingla.A Genetic Improved Quantum Cryptography Model to Optimize Network Communication," Special Issue, vol. 8, no. 9S, pp. 256–259, Aug.2019.
[15] Papadimitrakis, M., N. Giamarelos, M. Stogiannos, E. N. Zois, NA-I. Livanos, and A. Alexandridis. "Metaheuristic search in smart grid: A review with emphasis on planning, scheduling and power flow optimization applications." *Renewable and Sustainable Energy Reviews* 145 (2021):111072.
[16] Butt, Osama Majeed, Muhammad Zulqarnain, and TallalMajeed Butt. "Recent advancement in smart grid technology: Future prospects in the electrical power network." *Ain Shams Engineering Journal* 12, no. 1 (2021):687-695.
[17] Sarker, Eity, PobitraHalder, Mehdi Seyedmahmoudian, Elmira Jamei, Ben Horan, SaadMekhilef, and Alex Stojcevski. "Progress on the demand side management in smart grid and optimization approaches." *International Journal of Energy Research* 45, no. 1 (2021):36-64.