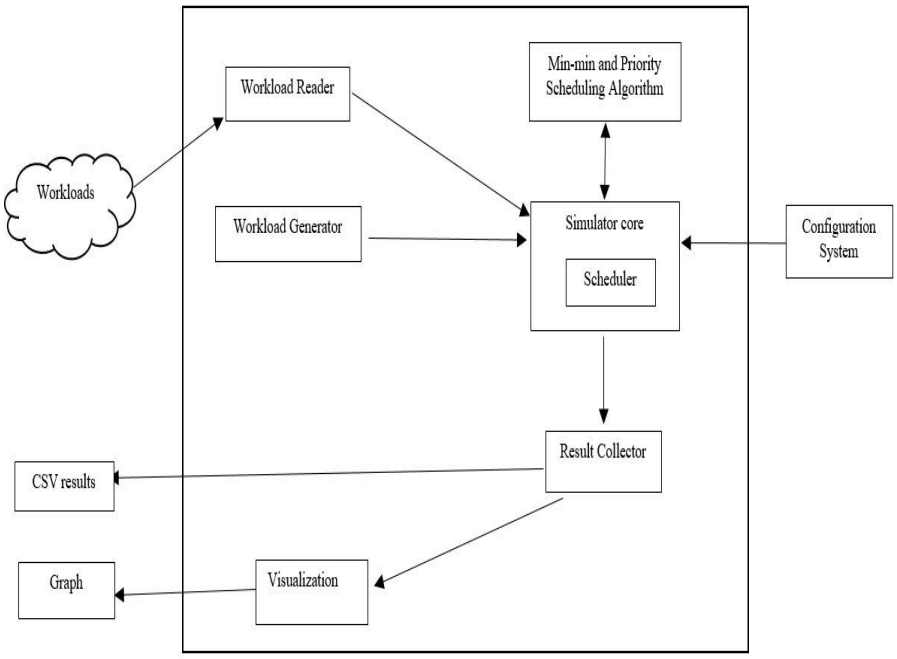


Fig. 2. Proposed System

One of the most used scheduling techniques in batch systems is priority scheduling. A priority is given to every process. The highest priority process should be carried out first, and then others. Priority processes are carried out in order of arrival to the processing queue. Any resource demand, including memory and time, can be used to determine priority [12]. If we have one short task, the Min-min could simultaneously perform several long tasks while performing one shorter. In this case the total makepan is calculated by quick task execution. In proposed system, the incoming tasks are classified their categories that may reduce the execution time of Min-min algorithm. In HPC system, machine clusters are assigned with appropriate task to process. The Min-min algorithm's main disadvantages are load imbalance, and job starvation with large service time. Since HPC system processing involves job scheduling and then investigating the variety of scheduling techniques that were implemented in the past became important.

The proposed system is based on Min-min algorithm and priority scheduling algorithm. Compute the job size and predicted execution time when the user submits a batch job (job size is equal to the number of cores the job requires). When computing the job size, select the task with the shortest anticipated execution time if the size of the job is less than or equal to the resources available. When a job's size is less than or equal to the threshold value, it is assigned to a small compute node; however, because the small compute node is busy during that period, it is assigned to a large compute node. If we have one short task, the Min-min could simultaneously perform several long tasks while performing one shorter. In this case the total makepan is calculated by quick task execution. In proposed system, the incoming tasks are classified their categories that may reduce the execution time of Min-min algorithm. In HPC system, machine clusters are assigned with appropriate task to process. The Min-min algorithm's main disadvantages are load imbalance, and job starvation with large service time. Since HPC system processing involves job scheduling and then investigating the variety of scheduling techniques that were implemented in the past became important. On the other hand, jobs given to large compute nodes have sizes that are neither smaller than nor equal to the threshold value. Wait till the resources are available if the size of the task does not exceed or equal the available resources. After allocating resources, update the job set and compute node list.

## 5. Conclusion

Some existing algorithm having to be considered the performance of the system is not satisfied. The issues are still and researchers are trying to solve them. Jobs can be categorized according to user requirements, and then the optimal running time can be determined based on the various objectives for every job. It will improve the quality of job scheduling indirectly in a cloud environment and HPC. Resource and Job Management for HPC is a continuously evolving domain. Algorithm is modified for scheduling using HPC. High performance can get than

other existing algorithm. Priority of small job is assigned to small node reduced waiting time than large node. On particular target HPC systems, job scheduling methods with acceptably thorough models for ability conflict, job interaction, and interference will be evaluated. Modelling and simulation is very important role for researcher, ongoing research are testing and simulation using simulator.

## Acknowledgments

I would like to express my appreciation to my dedicated supervisor for their continuous guidance, my family for their constant support and encouragements.

## Conflicts of Interest

The authors have no conflicts of interest to declare.

## References

[1] Rodrigo, G. P.; Elmroth, E.; Ramakrishnan, L. (2018): ScSF: A Scheduling Simulation Framework, Springer, Springer International Publishing AG, part of Springer Nature 2018.
[2] Souza, A.; Rezaei, M.; Laure, E.; Tordsson, J. (2019): Hybrid Resource Management for HPC and Data Intensive Workloads,19th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID), 2019.
[3] Matsui, Y.; Watashiba, Y.; Date, S.; Yoshikawa, T.; Shimojo, S. (2020): Job Scheduling Simulator for Assisting the Mapping Configuration Between Queue and Computing Nodes, Springer nature Switzerland AG 2020, pp 1024-1033, 2020. https://doi.org/10.1007/978-3-030-15032-7_86 .
[4] Jones, W. M.; Ligon, W. B.; Pang, L. W.; Stanzione, D. (2005): Characterization of Bandwidth-Aware Meta-Scheduler for Co-Allocating Jobs Across Multiple Clusters, The Journal of Supercomputing 34(2):135-163, November 2005.
[5] Chung, M. T.; Pham, K. T.; Thoai, N.; Kranzlmueller, D. (2019): A New Approach for Scheduling Job with The Heterogeneity-aware Resource in HPC Systems, IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems, 2019.
[6] Fan, Y. (2021): Job Scheduling in High Performance Computing, 20 September,2021.
[7] Raqqaq, S.; Khan, F.; Wahid, A.; Sha, M. A.: Scheduling Algorithms for High-Performance Computing: An Application Perspective of Fog Computing, Springer Nature Switzerland AG 2019,2021.
[8] Amalarethinam, D.I. G.; Kavitha, S. (2016): Enhanced Min-Min Algorithm for Meta-task Scheduling in Cloud Computing, IJCTA, 9(27), pp 85-91, 2016.
[9] Yoon, J. W.; Hong, T. Y.;Park, C. Y.; Noh, S. Y.; Yu, H. C. (2020): Log Analysis-Based Resource and execution Time Improvement in HPC: A Case Study, Applied Science, 10 April,2020.
[10] Legrand; A.; Marchal, L., Casanova, H. (2003): Scheduling distributed applications: the SimGrid simulation framework, In Procedings of the third International Symposium on Cluster Computing and the Grid, pp 138-145, IEEE,2003.
[11] Dutot, P. F.; Mercier, M.; Poquet, M.; Richard, O. (2016): Batsim: a Realistic Language-Independent Resouces and Jobs Management Systems Simulator, In 20th Workshop on Job Scheduling Strategies for Parallel Processing, Chicago,United States, May 2016.
[12] https://www.guru99.com/priority-scheduling-program.html
[13] Buyya, R.; Murshed, M.; Abramson, D.; Venugopal, S. (2005): Scheduling parameter sweep applications on global Grids: a deadline and budget constrained cost-time optimization algorithm, International Journal of Software: Practice and Experience (SPE), 35(5):491-512,2005.
[14] https://github.com/aleasimulator/alea
[15] Dalibor, K.; Hana, R. (2010): Alea 2: Job Scheduling Simulator, ICST, ISBN 78-963-9799-87-5,2010.
[16] Chen, H. K.; Wang, F.; Helian, N.;Akanmu, G. (2013): Min-Min Scheduling Algorithm for Efficient Resource Distribution Using Cloud and Fog in Smart Buildings, Advances on Broadband and Wireless Computing, Communication and Applications (BWCCA 2018), 19 October 2018.
[17] http://www.e2matrix.com/blog/2018/01/22/job-scheduling-algorithms-in-cloud-computing/
[18] https://jsspp.org/workload/
[19] https://www.cs.huji.ac.il/labs/parallel/workload/logs.html

## Authors Profile

**Lett Yi Kyaw** received the M.C.Sc. degree from the University of Computer Studies, Myeik, Myanmar in 2009. She is currently working as an Associate Professor in the University of Computer Studies, Taungoo, Myanmar. Her other research interests are Cloud Computing, High Performance Computing, Machine Learning and Deep Learning. She can be contacted at email: lettyikyaw@ucsy.edu.mm.

.

**Kyar Nyo Aye** received the Ph.D.(IT) degree from the University of Computer Studies, Yangon, Myanmar in 2013. She is currently working as an Associate Professor in the University of Computer Studies, Taunggyi, Myanmar. Her other research interests are Cloud Computing, High Performance Computing, Machine Learning and Deep Learning. She can be contacted at email: kyarnyoaye@ucsy.edu.mm.