WJCMS

Wasit Journal of Computer and Mathematics Science

Journal Homepage: https://wjcm.uowasit.edu.iq/index.php/WJCM

e-ISSN: 2788-5879 p-ISSN: 2788-5879

Development of an Anomaly Detection Algorithm in Distributed Systems based on Runtime and System Status

Razieh Asgarnezhad[1], Ali Naseer Kadhim Alwali[1,2], Mhmood Hamid Sahar Alsaedi[1] and Samer Alwan Zaboon Albwhusseinsarr[1,2]*

[1]Department of Computer Engineering Isfahan (Khorasan) Branch Islamic Azad University Is- fahan , Iran
[2]Ministry of Education Educational Directorate of Wasit province Wasit Iraq

*Corresponding Author: Samer Alwan Zaboon Albwhusseinsarr

DOI: https://doi.org/10.31185/wjcm.Vol1.Iss1.25
Received: December 2021; Accepted: February 2022; Available online: March 2022

ABSTRACT: Computational grades have emerged as a new approach to solving large-scale problems in the fields of science, engineering, and commerce. The computing grid is a hardware and software infrastructure that provides affordable, reliable, comprehensive, and affordable access to the computational abilities of others. A computational grid is associated with a set of resources on a large scale. Computational grades have emerged as a new approach to solving large-scale problems in the fields of science, engineering, and commerce. The computing grid is a hardware and software infrastructure that provides affordable, reliable, comprehensive, and affordable access to the computational abilities of others. A computational grid is associated with a set of resources on a large scale. The purpose of this thesis is to provide a method that optimizes resource management and scheduling in at least one direction. The main focus of the research is on the time criterion, the deadline for doing things and receiving the response from the grid are parameters that can be examined. From a more general perspective, the aim of the research is to get the answer as quickly as possible from the calculation grid. The proposed algorithm improves the scheduling and resource management of the grid in the direction of improvement, and the structure and form of this problem have not yet been resolved. The proposed solution has been considered hypothesis and removed some of the definitions of grid scheduling, such as cost, quality of service, architectures, and others, but ultimately heed it and timed the grid in some ways.

Keywords: Anomaly Detection Algorithm, Distributed Systems, Runtime, System Status

1. INTRODUCTION

Computational grades have emerged as a new approach to large-scale problem solving in the scientific, engineering, and commercial fields. the computational grid is a hardware and software infrastructure that provides reliable, stable, comprehensive, and inexpensive access to the computing capabilities of others [1].

One of the most important parts of a grid system is the timer. the task is to schedule resources and properly divide tasks among computing resources. error control is also one of the tasks of this component. better scheduling increases service quality. different methods can be used for scheduling. existing algorithms can be divided into two classes, online mode and categories [2]. in the online mode, a task is assigned to the appropriate resource or machine (in terms of timing and strategy) as soon as the grid scheduler is reached; but in the case of batch tasks, they are not scheduled immediately, but become a group (set of tasks) and with a specific time period, the scheduling operation is performed on them, in fact, the scheduling operation is done in groups and categories. it is done [3]. online algorithms are suitable in situations where the rate of entry of tasks into the environment is low, and vice versa regarding batch strategies, it is better to have a high rate

of entry and arrival of tasks. it is clear that in the case of a grid environment, its high dynamics make batch algorithms perform better, and we continue to use this approach frequently. also in the case of relational tasks, tasks can be dependent or independent of each other, it is assumed that each application can be divided into independent tasks and in fact the need to use the workflow diagram to specify it is not a task to prioritize and delay. to evaluate and compare different algorithms and strategies, we need several quantitative and standard criteria, as well as a suitable simulation platform [4].

A computational grid is associated with a set of resources on a large scale. computational grades have emerged as a new approach to large-scale problem solving in the scientific, engineering, and commercial fields. the aim of this dissertation is to present a method that optimizes resource management and grid scheduling in at least one direction. the main focus of the research is on the time criterion, the deadline for doing the work and receiving the answer from the grade parameters can be examined. from a more general point of view, the aim of the research is to receive the answers to the given tasks as quickly as possible from the computational grade. the proposed algorithm has improved the scheduling and resource management problem in the grid in a direction and the structure and form of this problem has not been solved yet. the proposed solution considers hypotheses and excludes items from the definition of grid scheduling, such as cost, quality of service, architectures, and so on; but in the end, it has improved it and moved the schedule in the grid in some directions [5].

The grid environment is an optional and dynamic environment, meaning that unlike clusters where the rate of change of available resources is usually low, in machine grid systems and in general all available resources of the grid over time is possible. is to change. the efficiency of any grid system depends to a large extent on the efficiency and effectiveness of resource management methods as well as the policies and scheduling mechanisms used to perform tasks in this environment. for example, we know how critical and vital the act of load sharing is in grid systems, because the effective use of load sharing will benefit both machine owners and resource customers. grid scheduling is actually the task of mapping tasks to processor machines, the degree of complexity of which depends on the number of resources and tasks of the system. during this report, in addition to these cases, we introduced most of the challenges and obstacles to grid scheduling. also, by introducing different models and different scheduling stages, we were able to identify the problem hypotheses and our scheduling algorithm within set the standard and defined grid [6].

By performing various simulations, we presented a comprehensive and complete comparison of the most famous heuristic algorithms and plotted the results in the form of tables and graphs. Then, by introducing the combined GA and PSO algorithms, we expressed our idea about the application of this new method for the grid scheduling problem. Comparison of this strategy with the most popular heuristic methods (ie Min-min) proved the superiority of the idea of this research and the results of the simulations and the obtained values showed the improvement and reduction of the total completion time. The study and application of a new algorithm and strategy has always been an open issue for the grid scheduling problem; Creating and determining appropriate fitting functions, defining objectives, criteria and optimization parameters and, above all, simulation and creating different grid environments and scheduling algorithms, are part of the work of researchers in this field.

The nature of the computational grid has created challenges for resource management and scheduling of this system. The point is that rarely can an algorithm or fitness function be found that can cover and optimize all of these challenges [7]. Most of the algorithms presented so far have improved the problem of scheduling and resource management in the grid in a direction and the structure and form of this problem has not been solved yet. Each of these proposed solutions considers hypotheses and excludes items from the definition of grid scheduling, such as cost, quality of service, architectures, and so on; But in the end, they improved it and moved the grid scheduling forward in some ways [8]. We also use this rule for our proposed algorithm and by considering the hypotheses and presenting a new definition of scheduling, we try to look at the issue of scheduling and resource management in the grid from a new perspective and in one of its critical criteria. (Such as Makespan time, Flowtime, Resource utilization, etc.) Make improvements. The scheduler examines the issue of grid scheduling, assuming that tasks are independent and that the scheduling operation is static. Tasks can be interdependent or independent, it is assumed that each application can be divided into independent tasks, and in fact there is no need to use a workflow diagram to determine the priority and lag of tasks (this The assumption is perfectly consistent with nature and parallel to the grid).

2. LEARNING FROM RARE CLASSES

To evaluate and compare different algorithms and strategies, we need a few quantitative and standard criteria, as well as a suitable simulation platform. The most important criteria for us are the time of completion and turnaround time, and we do not pay attention to other factors such as the cost of resources. Various models have been proposed to formulate distributed environments; Using a special model, the efficiency of algorithms can be measured in different distributed environments and with different characteristics. Since the mapping and scheduling of tasks is assumed to be static, it is possible to have a relative estimate of the time of performing different tasks on different sources. The model used is a matrix containing the expected times to perform tasks (ETC matrix). The computational basis in this research is the

proposed model [9], because this model agrees and compares all research and studies in the field of grid scheduling. Therefore, the results obtained from it can be compared and evaluated with the results of other researches.

3. RELATED WORKS OF ANOMALY DETECTION TECHNIQUES

Experience working with network computing systems has shown that efficient programming and system performance are not necessarily the same. In other words, a scheduler may not be able to optimize both the program and the performance of the system. One solution is to use a multilayer system. In addition, because the expected scale in grid systems is large, an RMS usually consists of several RMSs that work together. Figure 1 shows a diagram for a system with multiple connected RMSs that each RMS has several levels. Figure 2 shows an abstract model of the main functions supported by a grid management system. This model has multiple units and four interfaces: resource user interface, resource provider interface, resource manager support interface and resource manager peer interface [9].

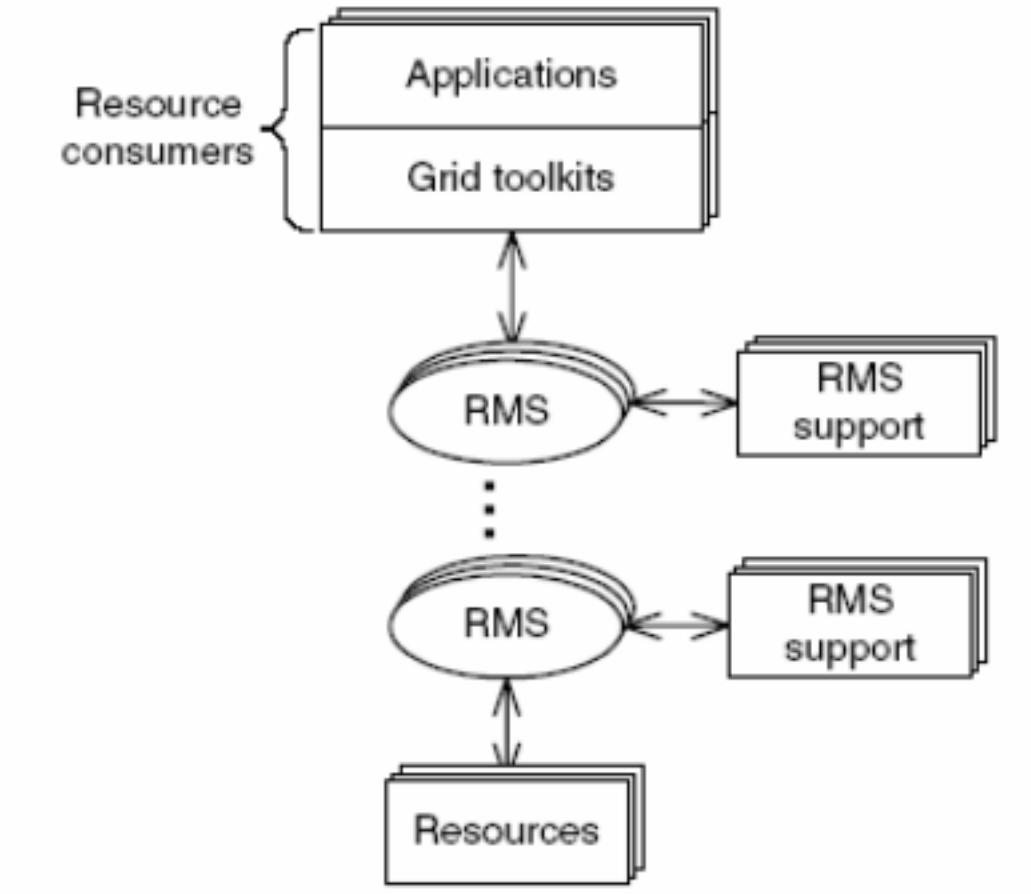

FIGURE 1. Background of RMS system [9]

Organizing machines in the grid affects the RMS communication pattern and therefore plays a decisive role in architectural scalability. Figure 3 shows the classification of machine organization. Organizing for machines involved in resource management, in fact, sheds light on how scheduling works, the structure of communication between machines, and the different roles that machines play in scheduling.

In Flat organization, all machines can communicate directly with each other without the need for an interface. In hierarchical organization, machines on one level can communicate directly with machines that are directly above or below them, as well as with their counterparts on the same level. In a cell structure, machines in a cell can communicate with each other through flat organization. In each cell, machines are assigned that are responsible for communicating with machines outside the cell. The internal structure of one cell is hidden from the others. The cells themselves can be organized hierarchically or flat [10].

In grid systems, service quality is not limited to network bandwidth; It also includes Node processing and storage capabilities. Therefore, the main focus in a grade is more on providing service quality in End-to-End mode, than just

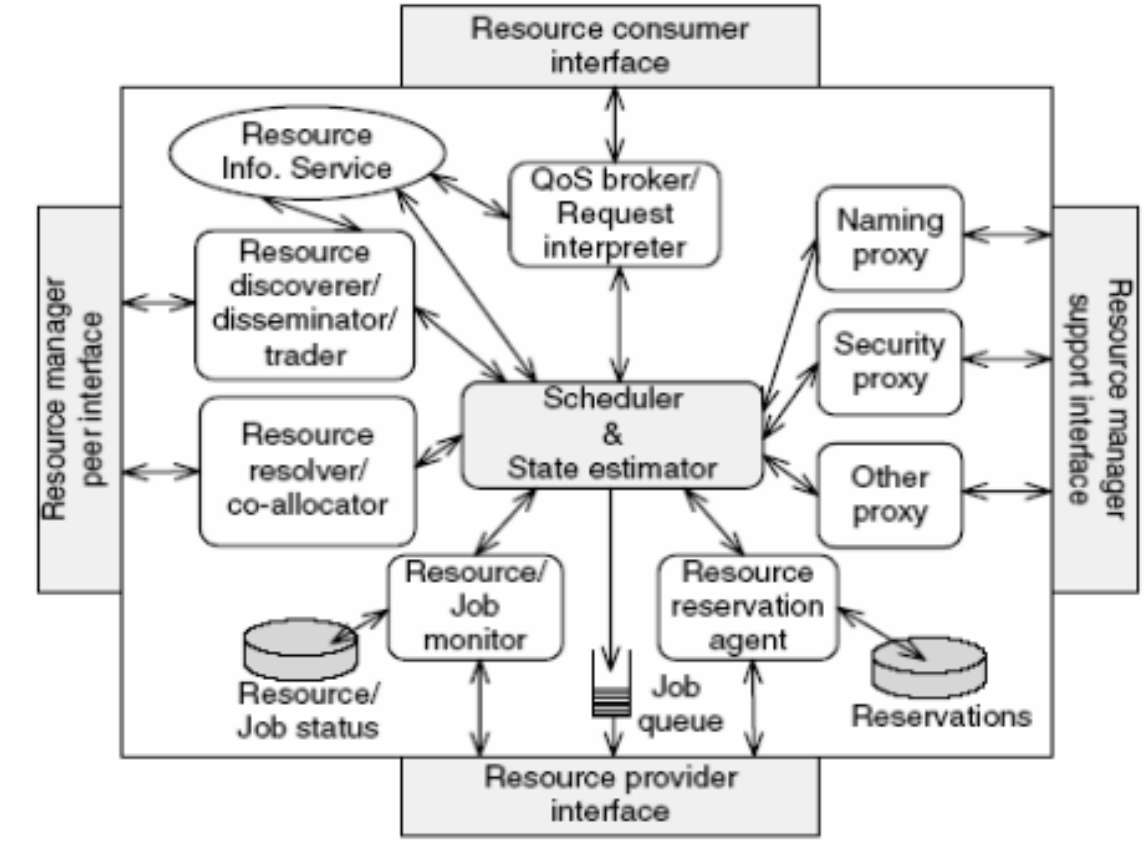

FIGURE 2. **Abstract structure of resource management system [9]**

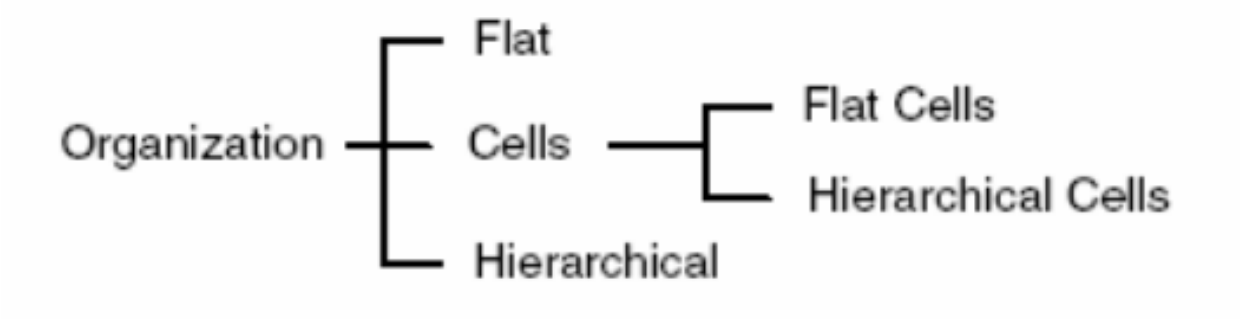

FIGURE 3. **Classification of machine organization in the grid**

network service quality. Figure 2-8 shows the service quality support classification. An RMS that can provide multiple service quality parameters specifically but cannot enforce them with relevant policy-making has, in fact, provided Soft support. Also, if all nodes in the grid can execute the items guaranteed by the RMS, the support provided will be of the Hard type [11].

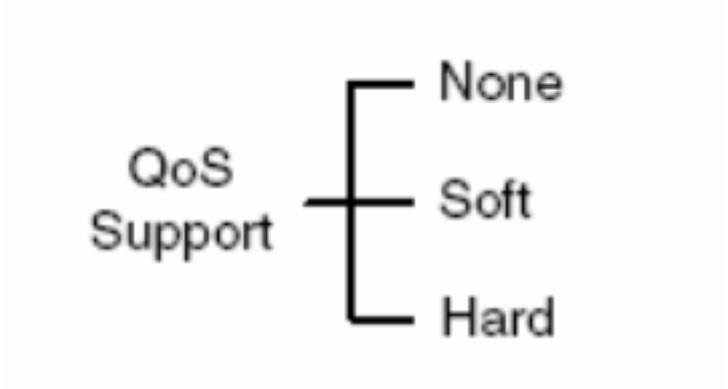

FIGURE 4. Service quality support classification

At the time of the introduction of distributed systems, the available communication bandwidth was very limited; But today we are witnessing a rapid growth in communication capacity between different machines, and this has made the operation and establishment of grid networks more practical. However, the problem of low communication bandwidth and limited communication capacity is still a limiting factor in the implementation of distributed applications, so it can be said that Data communication capacity is one of the most important resources in the grid. [12].

There are connections in the grid at two levels: the relationships between the internal components of the grid with each other (within the grid) and the relationships between the components of the grid with the elements outside it (external grid). Intra-grade communications are important in terms of sending and receiving tasks and related data to various intra-grade machines. Outdoor communication is the communication between the internal components of the grid and the Internet, and for example, it becomes valuable when building applications such as search engines. In-grid machines may, in addition to having connections to internal grid components, also have separate connections with different bandwidths for Internet connection (external grid) [13]

Security in distributed systems is at different levels. A distributed system may cover several sites that are geographically dispersed. In this environment, all machines located on different sites may be covered by a single resource management and scheduling system. It is true that the security of machines located on a site is provided locally at the same site, but the task of ensuring the security of resources at the level of the distributed system is the responsibility of the resource management system and scheduling. Therefore, any ideal resource management and scheduling system should provide the necessary security of access to the covered resources at the level of the distributed system [14, 15].

Sometimes the two concepts of scheduling and resource reservation are misused. Scheduling means automatically finding the most suitable machine (source) to perform a given task. Grid schedulers, therefore, act on the basis of the resources currently available to the grid to find the best source. But there is a function called resource reservation in the grade to increase the quality of service [16, 17].

In principle, having a quality assurance service is preferred to not having one, so schedules that do a quality assurance service are preferable to schedules that fail to do so [18]. There are various methods, methods and ideas for scheduling and mapping tasks. The three major scheduling models in the grid environment are the Centralized model, the Distributed model, and the Hierarchical model. In the following, we will have an overview of each of these models and their advantages and disadvantages [19, 20].

Like the centralized scheduling model, hierarchical scheduling has scalability problems and low fault tolerance. But compared to centralized scheduling, one advantage of the hierarchical scheduling model is that central scheduling and local scheduling can have different policies in scheduling tasks [21, 22].

Scheduling optimization is the process of finding an optimal or near-optimal schedule in the form of a set of tasks and processor machines. The problem of mapping tasks to processors is a matter of exponential complexity. Heuristic and metaheuristic algorithms such as GA: Genetic Algorithms or SA: Simulated Annealing algorithms can be used to optimize the scheduling problem in the grid [23].

4. PROPOSED METHOD:

As it was examined, the previous algorithms used execution time and total completion time to schedule tasks. In our proposed algorithm, which is a combination of GA and PSO intelligent algorithms, we have tried to make comparisons based on total completion time.

In the previous section, we simulated and compared heuristic algorithms. To compare with exploratory methods, we select the best strategy, the Min-min method, from the previous section. We have seen that Min-min is more successful than the other methods in almost all cases and comparisons, so comparing the intelligent method of combining GA and PSO with the Min-min strategy is in fact a complete and comprehensive comparison. Shows the superiority of each method. Here, after each section, the regulatory parameters of the algorithm and the corresponding evolution strategy will be explained. The pseudo-code related to the combination of these two algorithms is given below. In fact, by combining these two algorithms, we want to use their features together.

```
  Initialize GA and PSO parameters.
  Create population and initialize them randomly.
  Calculate cost of each population.
  Repeat below operations for N (the number of iteration)
time or until termination condition is met:
    Do it for PSO iteration time:
      Do it for all population:
        Update velocity
        Update position and reflect it if necessary.
        Evaluation or update costs.
        Update personal and global best.
      Do it for GA iteration time:
        CROSSOVER(Arithmetic)
        MUTATION: For Size of Mutation Population time,
each time a member of population randomly is selected and
after mutation insert into mutation population.
        SELECTION
        Evaluation or update costs.
        Update local and global best.
  At the end we have the best solution and each iteration
best solutions.
```

As shown in the pseudo-code above, we first optimize an initial population generated by the PSO algorithm, then give this population as the initial population to the genetic algorithm, and finally get the best answer.

5. EXPERIMENTAL RESULTS

Considering the previous section and the nature of distributed environments such as grids that have a high number of tasks with variable lengths, it can be concluded that the calculation and execution time versus the total completion time can be ignored. And a good mapping and scheduling ultimately returns the necessary benefits to the user. These results and interpretations are valid with a slight difference for all twelve types of ETC matrices. However, for a general comparison, Figure 9 shows the results; Interpretation of this diagram The overall performance of an algorithm is distributed in all

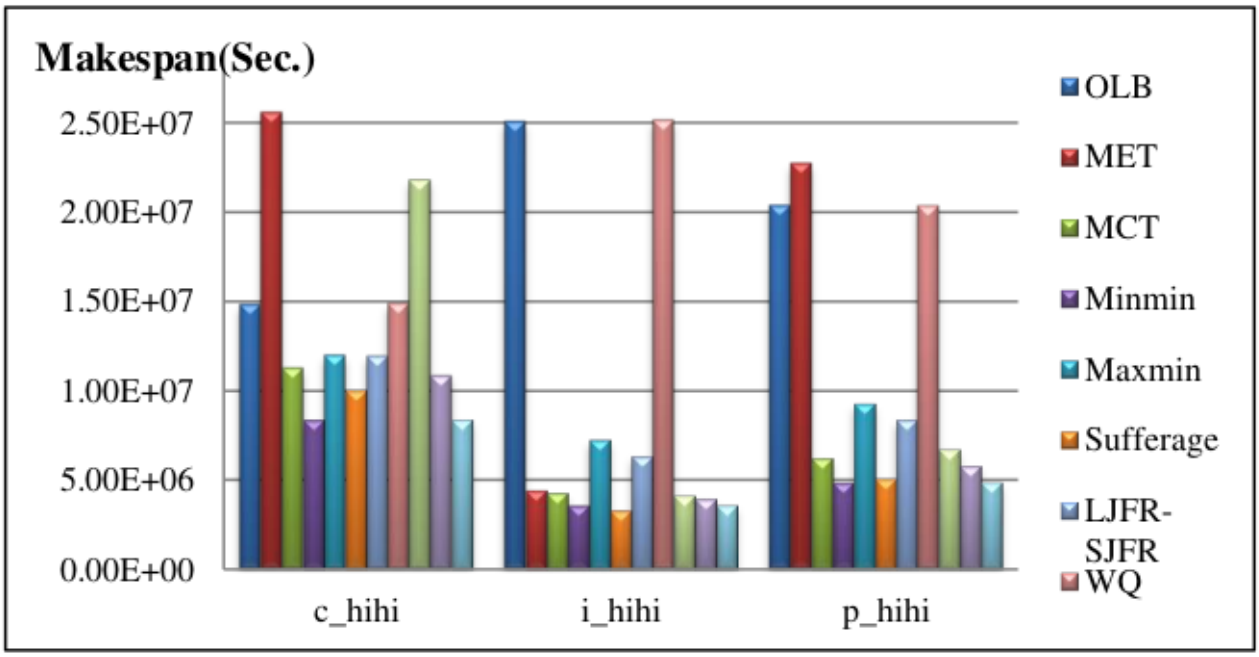

FIGURE 5. **Makespan values for the environment with heterogeneous tasks and machines**

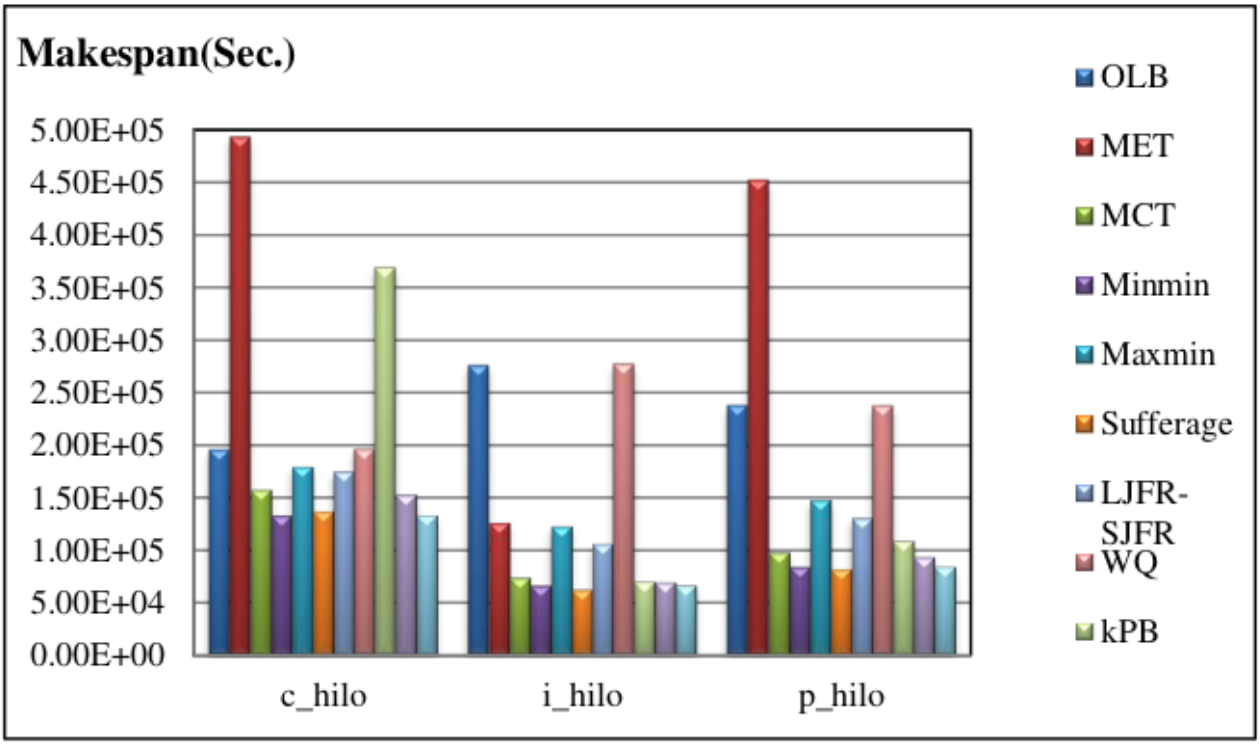

FIGURE 6. **Makespan values for the environment with heterogeneous tasks and homogeneous machines**

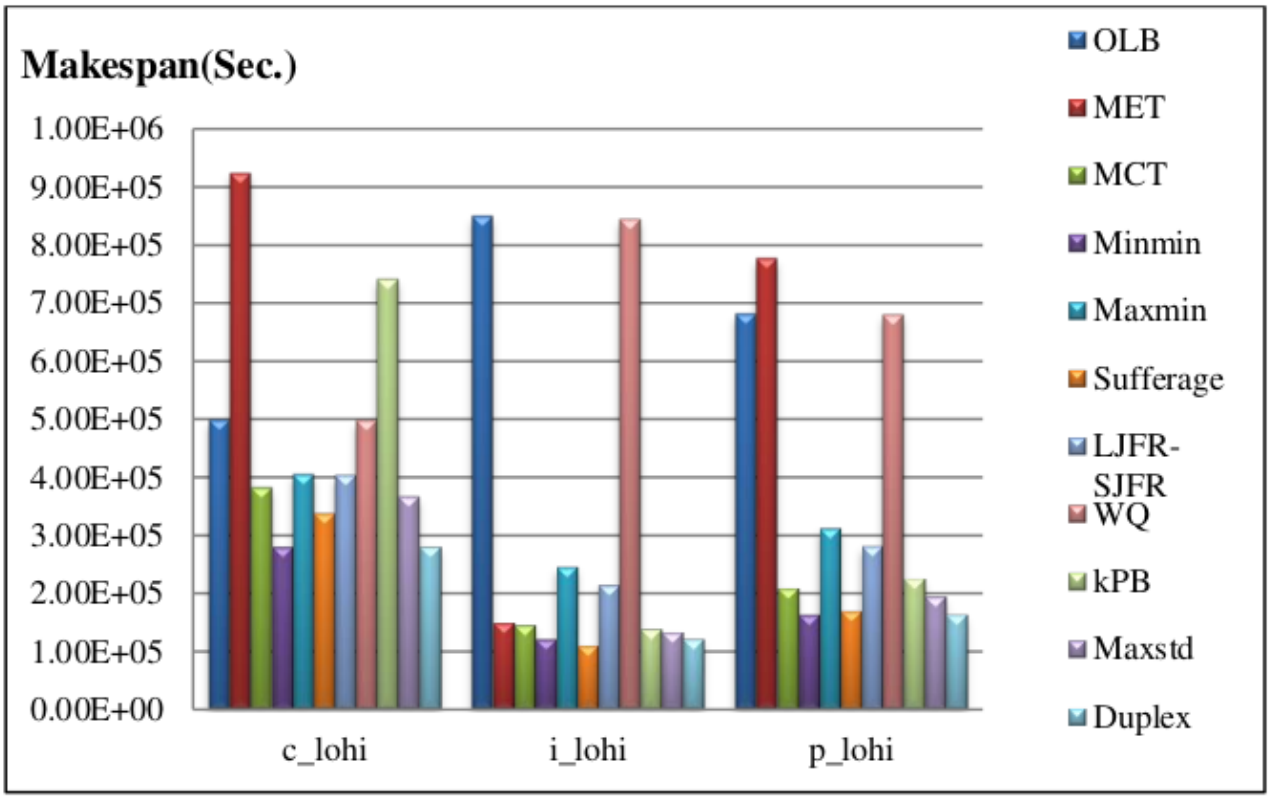

FIGURE 7. **Makespan values for the environment with homogeneous tasks and heterogeneous machines**

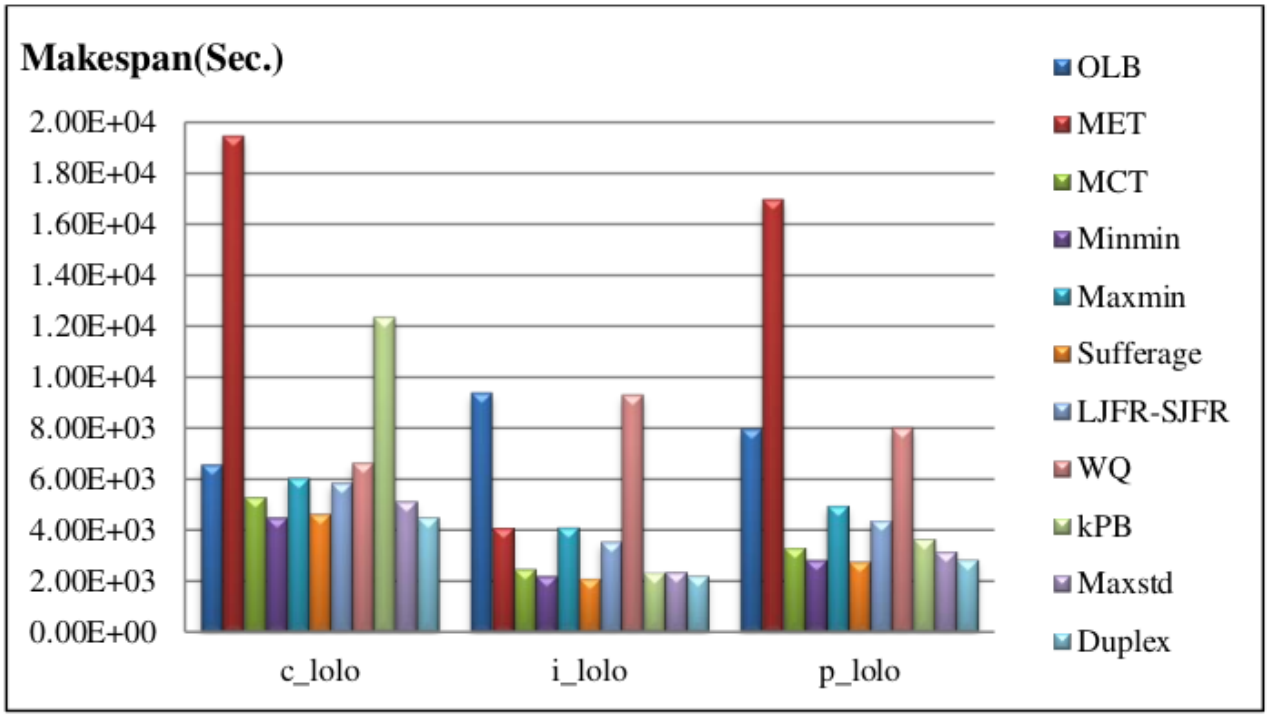

FIGURE 8. **Makespan values for the environment with homogeneous tasks and machines**

conditions and environments. Methods such as MET and kPB perform very poorly and unacceptably in compatible environments because these strategies seek the fastest or closest machine (kPB) for each task and because in the environment Only one machine has this property (ie, a specific machine is the fastest machine for all tasks), a severe load imbalance occurs in the system; All diagrams also confirm this.

This problem also exists with lower percentages and concentrations in partial concordance environments. WQ and OLB have produced the same results in almost all environments, which are consistent with their definition. The subtle point and difference between these two methods is the lack of sequence (random mapping) of tasks in the WQ algorithm, which, of course, confirms the approximate ineffectiveness of this case (maintaining sequence mapping). The Duplex algorithm, which always chooses the best option between Max-min and Min-min, has a good performance and total completion time, because in almost all cases and conditions, the Min-min method will be better, and the Duplex result will be the same efficiency and response. Min-min. The top three methods, Maxstd, Min-min, and Sufferage, all use the same idea, the shortest completion time (ie, choosing the best machine for each task) and the best mapping. In fact, these algorithms, using their parameters and overview of all tasks (batch scheduling, not a specific task at any time) at any given time, select a task for mapping that has a delay in scheduling. Leads to more losses. So the secret of success and proper timing of these methods lies in choosing the best machine and the best mapping order. Finally, it can be seen from the tables and diagrams that inconsistent environments have a shorter average time because the resources and facilities distributed are equal and balanced (a particular machine is not superior and faster). It is noteworthy that real distributed environments are like grids, and it is appropriate to use these methods to work with them.

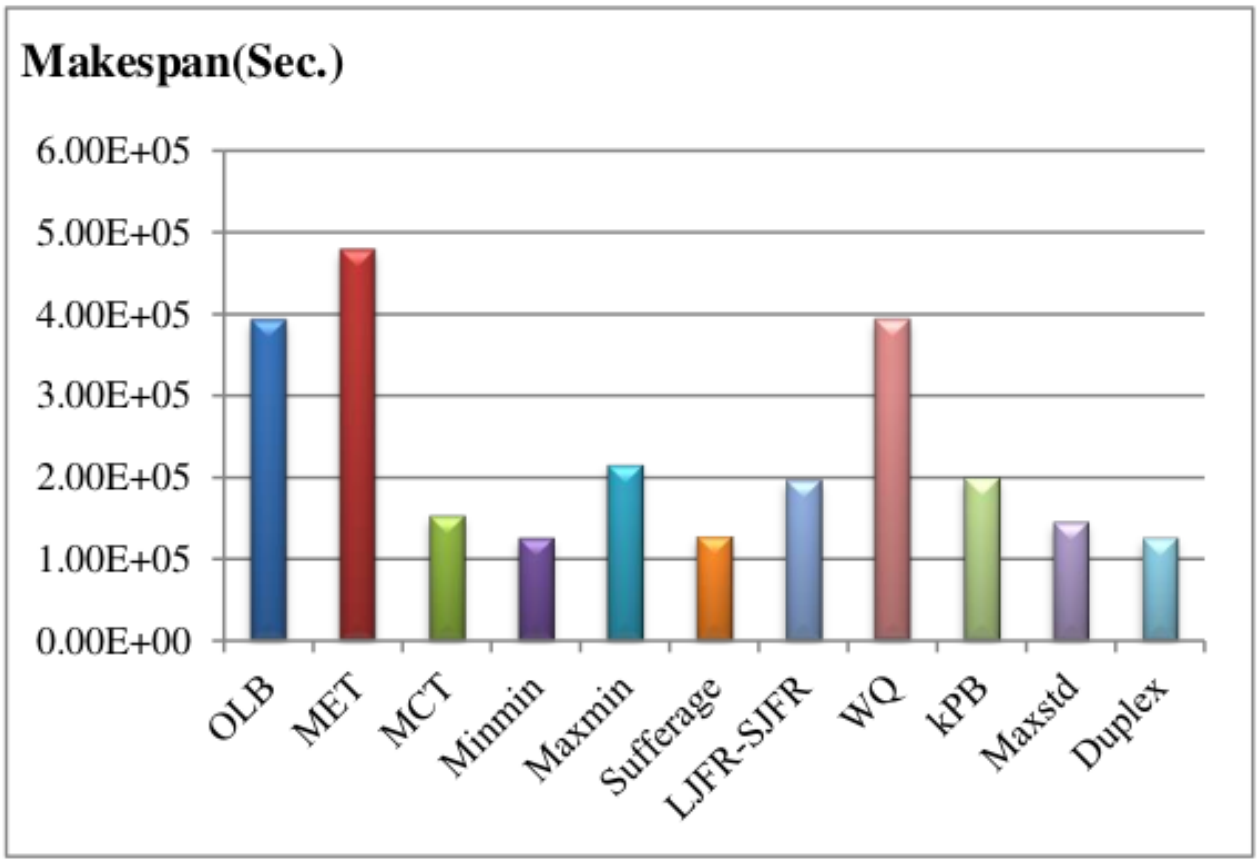

FIGURE 9. Geometric mean ofMakespan values for different distributed environments

6. CONCLUSION & FUTURE WORKS

The grid environment is an optional and dynamic environment, meaning that unlike clusters where the rate of change of available resources is usually low, in machine grid systems and in general all available resources of the grid over time is possible. Is to change. The efficiency of any grid system depends to a large extent on the efficiency and effectiveness of resource management methods as well as the policies and scheduling mechanisms used to perform tasks in this

environment. For example, we know how critical and vital the act of load sharing is in grid systems, because the effective use of load sharing will benefit both machine owners and resource customers.

Grid scheduling is actually the task of mapping tasks to processor machines, the degree of complexity of which depends on the number of resources and tasks of the system. During this report, in addition to these cases, we introduced most of the challenges and obstacles to grid scheduling. Also, by introducing different models and different scheduling stages, we were able to identify the problem hypotheses and our scheduling algorithm within Set the standard and defined grade. By performing various simulations, we presented a comprehensive and complete comparison of the most famous heuristic algorithms and plotted the results in the form of tables and graphs. Then, by introducing the combined GA and PSO algorithms, we expressed our idea about the application of this new method for the grid scheduling problem.

Comparison of this strategy with the most popular heuristic methods (i.e. Min-min) proved the superiority of our idea and the results of the simulations and the obtained values showed the improvement and reduction of the total completion time. The study and application of a new algorithm and strategy has always been an open issue for the grid scheduling problem; Creating and determining appropriate fitting functions, defining objectives, criteria and optimization parameters and, above all, simulation and creating different grid environments and scheduling algorithms, are part of the work of researchers in this field. In general, what was presented in this report was theoretical discussions, introduction of principles, and review of proposed scheduling methods. Our main purpose of this research was to provide a complete reference of comparisons in the first place and to present a better method than the existing methods in the second place. However, other ideas related to the problem of grid scheduling, such as fuzzy theory, neural networks, etc., can still be considered and applied.

FUNDING

None

ACKNOWLEDGEMENT

None

CONFLICTS OF INTEREST

The author declares no conflict of interest.

REFERENCES

[1] B. Zhang, H. Zhang, P. Moscato, and A. Zhang, "Anomaly detection via mining numerical workflow relations from logs," *2020 International Symposium on Reliable Distributed Systems (SRDS)*, pp. 195–204, 2020.
[2] L. Pan, Z. Gu, Y. Ren, C. Liu, and Z. Wang, "An anomaly detection method for system logs using Venn-Abers predictors," *2020 IEEE Fifth International Conference on Data Science in Cyberspace (DSC)*, pp. 362–368, 2020.
[3] S. O. Al-Mamory and H. Zhang, "New data mining technique to enhance IDS alarms quality," *Journal in Computer Virology*, vol. 6, no. 1, pp. 43–55, 2010.
[4] E. Alpaydin, *Introduction to Machine Learning*. Cambridge, Massachusetts, London, England: The MIT Press, 2020.
[5] J. P. Anderson, *Computer security threat monitoring and surveillance*. Fort Washington, Pennsylvania: James P. Anderson Company, 1980.
[6] R. Bace and P. Mell, "NIST special publication on intrusion detection systems, DTIC Document," 2021.
[7] E. Bloedorn, A. D. Christiansen, W. Hill, C. Skorupka, L. M. Talbot, and J. Tivel, "Data mining for network intrusion detection: How to get started," 2001.
[8] S. T. Brugger, "Data mining methods for network intrusion detection," 2004. http://neuro.bstu.by/ai/To-dom/My_research/failed%201%20subitem/For-research/D-mining/Anomaly-D/Intrusion-detection/brugger-dmnid.pdf.
[9] J. Balthrop, S. Forrest, and M. R. Glickman, "Revisiting LISYS: parameters and normal behavior," 2022.
[10] Y. Zuo, Y. Wu, G. Min, C. Huang, and K. Pei, "An intelligent anomaly detection scheme for micro-services architectures with temporal and spatial data analysis," *IEEE Transactions on Cognitive Communications and Networking*, vol. 6, no. 2, pp. 548–561, 2020.
[11] H. Cheng-Yuan, L. Yuan-Cheng, I. W. Chen, W. Fu-Yu, and T. Wei-Hsuan, "Statistical analysis of false positives and false negatives from real traffic with intrusion detection/prevention systems," *Communications Magazine*, vol. 50, no. 3, pp. 146–154, 2012.
[12] V. H. Le and H. Zhang, "Log-based anomaly detection without log parsing," *36th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pp. 492–504, 2021.
[13] Darpa, "MIT Lincoln Laboratory: Communications & Information Technology," 1998. http://www.ll.mit.edu/mission/communications/ist/index.html.
[14] D. E. Denning, "An Intrusion-Detection Model," *Software Engineering, IEEE Transactions on, SE-13*, pp. 222–232, 1987.
[15] S. S. Dongre and K. K. Wankhade, "Intrusion Detection System Using New Ensemble Boosting Approach," *International Journal of Modeling and Optimization*, vol. 2, 2022.
[16] Emerald, "Event Monitoring Enabling Responses to Anomalous Live Disturbances (EMERALD," 1996. http://www.sdl.sri.com/projects/emerald/.
[17] M. V. Mahoney and P. K. Chan, "An Analysis of the 1999 DARPA/Lincoln Laboratory Evaluation Data for Network Anomaly Detection," in *Recent Advances in Intrusion Detection 2820* (G. Vigna and C. K. E. Jonsson, eds.), pp. 220–237, Springer, 2013.
[18] J. Mchugh, "Testing Intrusion detection systems: a critique of the 1998 and 1999 DARPA intrusion detection system evaluations as performed by Lincoln Laboratory," *ACM Transactions on Information and System Security*, vol. 3, no. 4, pp. 262–294, 2020.

[19] R. G. Mohammed and A. M. Awadelkarim, "Design and Implementation of a Data Mining-Based Network Intrusion Detection Scheme," *Asian Journal of Information Technology*, vol. 10, no. 4, pp. 136–141, 2011.
[20] C. Modi, D. Patel, B. Borisaniya, H. Patel, A. Patel, and M. Rajarajan, "A survey of intrusion detection techniques in Cloud," *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 42–57, 2013.
[21] Nmap, "Nmap - Free Security Scanner for Network Exploration & Security Audits." http://nmap.org/.
[22] V. H. Le and H. Zhang, "Log-based anomaly detection without log parsing," *36th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pp. 492–504, 2021.
[23] R. Xu, Y. Cheng, Z. Liu, Y. Xie, and Y. Yang, "Improved Long Short-Term Memory based anomaly detection with concept drift adaptive method for supporting IoT services," *Future Generation Computer Systems*, vol. 112, pp. 228–242, 2020.