Journal of Computer Science, Information Technology and Telecommunication Engineering (JCoSITTE)
Vol. 6, No. 1, March 2025, pp. 741~747
ISSN: 2721-3838, DOI: 10.30596/jcositte.v6i1.21731

Evaluation and Comparison of Load Balancing Algorithm Performance in the Implementation of Weighted Least Connections and Round Robin in Cloud Computing Environment

Mohammad Mika Fawwazi[1], Eka Putra[2], Nadya Andhika Putri[3]
[1,2,3] Faculty of Science and Techonology, Department of Computer System, University of Pembangunan Pancabudi, Indonesia

ABSTRACT

Weighted Least Connections (WLC) and Round Robin algorithms are two commonly used load balancing methods in cloud computing environments. Both have different approaches in distributing requests to servers, which impacts system performance. WLC takes into account the number of active connections and the capacity of each server, so that servers with larger capacities receive more requests, while Round Robin distributes requests sequentially regardless of server conditions. This study compares the performance of the two algorithms based on several parameters, including response time, throughput, and CPU utilization. The results show that WLC is superior in systems with heterogeneous servers, where WLC is able to adjust the load distribution based on the capacity and number of active connections, thereby improving system efficiency and performance. Faster response time and balanced CPU utilization are achieved by WLC, while Round Robin is more suitable for environments with servers with similar specifications. Although Round Robin works well in simple conditions, this algorithm often causes load imbalance on low-capacity servers in complex environments. Based on the results of the study, WLC is recommended for environments with server heterogeneity and dynamic loads, because this algorithm significantly improves resource efficiency and reduces server bottlenecks. Thus, WLC provides more optimal performance than Round Robin in scenarios that require more intelligent load distribution.

Keyword : load balancing, round robin, Weighted Least Connections, response time, throughput, CPU utilization

Corresponding Author:
Mohammad Mika Fawwazi,
Department of Computer System,
University of Pembangunan Pancabudi, Indonesia
Jl. Gatot Subroto, Kota Medan, Sumatera Utara 20122,
Email : mikafawwazii@gmail.com

Article history:
Received Nov 11, 2024
Revised Nov 20, 2024
Accepted Mar 10, 2025

1. INTRODUCTION

In today's digital era, cloud computing has become the foundation for many internet applications and services. Cloud computing infrastructure enables service providers to offer scalable and flexible resources at an efficient cost(Shafiq et al., 2022);(Ma & Chi, 2022). In a cloud computing environment, effective load management is key to ensuring that the system remains responsive and reliable despite fluctuating demand. Load balancing algorithms play a central role in distributing requests fairly among multiple servers or nodes, aiming to optimize resource usage and improve overall system performance(Bryhni et al., 2000);(Arfah et al., 2024). With the exponential growth in the use of cloud-based applications, ranging from web services to mobile applications and enterprise systems, the challenges in load management are becoming increasingly complex(Neghabi et al., 2018);(Nugroho, 2021). For example, applications serving millions of concurrent users require a load balancing system that can not only cope with high request volumes but also adapt to dynamic load variations. In this context, choosing the right load balancing algorithm becomes crucial to ensure optimal system performance. Load balancing algorithms serve to efficiently distribute requests among servers in a distributed system(Kashani & Mahdipour, 2022);(Shahid et al., 2023). Two widely used algorithms in practice are Weighted Least Connections and Round Robin. Although both algorithms aim to distribute the load evenly, they have very different approaches to achieving this goal. In the load balancing algorithm, there will be round robin and Weighted Least Connections algorithms(Kanellopoulos & Sharma, 2022). Weighted Least Connections is an algorithm that considers the number of active connections and the weight of each server when distributing requests. The server's weight determines how much the server contributes to the overall system capacity, and the number of active connections

indicates the current load borne by the server while round robin is a simpler method that distributes requests in turns to the servers in the list. Each server receives a turn to handle the request without considering the current load or capacity of each server(Semong et al., 2020).

In a cloud computing environment, the main challenges in load balancing involve handling dynamic loads, scalability, and maintaining consistent performance(Rahmatika et al., 2024);(Al Khowarizmi, Rahmad Syah, Mahyuddin K. M. Nasution, 2021). Cloud computing environments often involve thousands of servers spread across multiple geographic locations, with demands that can change rapidly. Therefore, load balancing algorithms must be able to adapt to load fluctuations in real-time, ensuring that all servers function at optimal capacity without experiencing overload(Malik et al., 2021). The problems in this study are related to load dynamics, scalability, and availability. In the context of a cloud computing environment, the load is not only influenced by the volume of requests but also by the characteristics of the application, such as data size, processing complexity, and interactions between components that are currently still not well managed, then the number of servers and resources is difficult to adjust according to needs and server failures often occur due to the lack of good server management availability. So to solve these problems, this study will involve experiments with simulations and performance testing of round robin with Weighted Least Connections in a cloud computing environment(Vecliuc et al., 2022). The parameters to be measured include average response time, request success rate, server resource usage, and impact on system availability. Data collected during the experiment will be analyzed to determine which algorithm provides the best results under various load conditions(Shafiq et al., 2022). The implementation of effective load balancing algorithms is crucial to ensure optimal performance in cloud computing environments. By comparing Weighted Least Connections and Round Robin, this study aims to provide in-depth insights into how each algorithm operates in the context of varying loads and dynamic system conditions(Belgaum et al., 2020). This study is expected to make a significant contribution to the understanding of how load balancing algorithms affect the performance of cloud computing systems. With comprehensive evaluation and comparison results, this study can help service providers and application developers in selecting the most suitable load balancing algorithms for their needs, as well as provide guidance for system optimization.

2. RESEARCH METHODS

This section will explain the methodology that will be used to evaluate and compare the performance of two load balancing algorithms, namely Weighted Least Connections (WLC) and Round Robin (RR), in a cloud computing environment. This study uses an experimental approach by creating a simulation of a cloud computing environment that supports load balancing. The simulation is carried out to replicate real-world scenarios where applications run on multiple virtual servers. Two load balancing algorithms, WLC and RR, will be applied to distribute requests from users to these servers. Then, the Weighted Least Connections (WLC) algorithm distributes requests based on the number of active connections and the weight given to each server, while the round robin algorithm will distribute requests alternately without taking into account the load on the server side. The parameters that will be processed include throughput, response time, utilization, scalability and fault tolerance. The following are the stages of the research methodology

2.1 Dataset

At this stage we will use a dataset of 1000 data. At this stage, data processing will be carried out to eliminate data that has no value. This data will contain throughput, response time and CPU utilization. The following data is used in table 1

Tabel 1. Dataset

Reques t Volume	WLC_Throughpu t (requests/sec)	RR_Throu ghpu t (requests /sec)	WLC_Respons e Time (ms)	RR_Respons e Time (ms)	WLC_CP U Utilizatio n (%)	RR_CPU Utilizatio n (%)
1000	800	750	120	140	55	60
2000	1600	1500	130	150	60	65
5000	4000	3700	140	160	65	70
10000	8500	7900	150	175	70	75
20000	17000	15500	160	190	75	80

11909	962,1622	897,6476	120,4004	140,5005	55,2002	60,2002
1209	978,3784	912,4124	120,4404	140,5506	55,22022	60,22022
1228	994,5946	927,1772	120,4805	140,6006	55,24024	60,24024
1247	1010,811	941,9419	120,5205	140,6507	55,26026	60,26026

3.2 Research Architecture

In this section there will be stages in implementing Weighted Least Connections (WLC) and Round Robin (RR), in a cloud computing environment. At this stage, the process of data collection, load balancing process with Weighted Least Connections (WLC) and Round Robin (RR) and conducting performance evaluation will be explained. The research architecture can be seen in Figure 1 below:

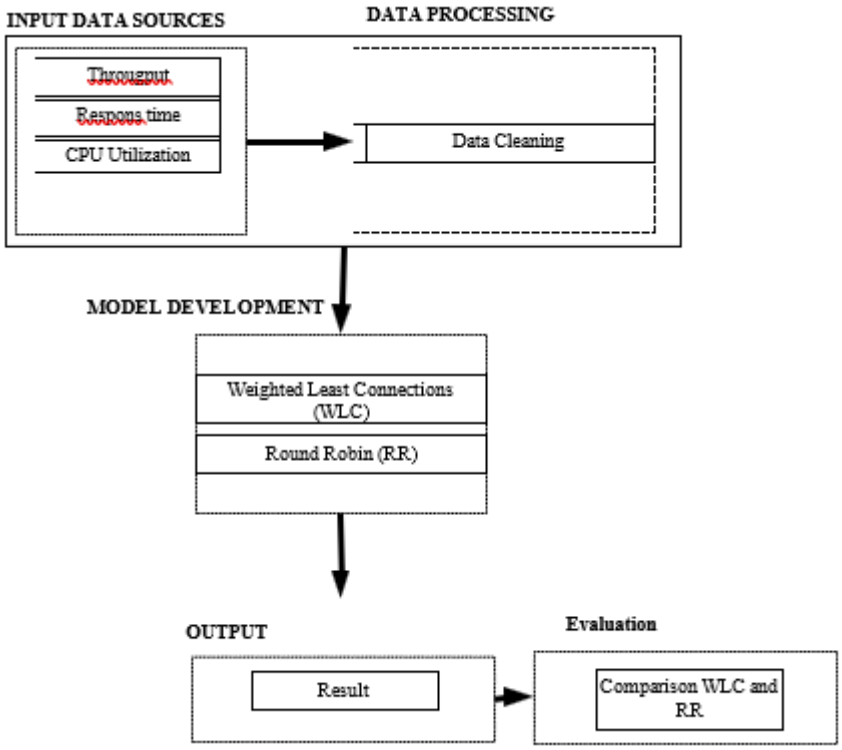

Fig 1. Research Architecture

In Figure 1 there will be a research architecture in conducting a comparison of Weighted Least Connections (WLC) and Round Robin (RR), in a cloud computing environment. At this stage, 1000 data will be collected which will be preprocessed to clean the data from empty values which will then be applied to Weighted Least Connections (WLC) and Round Robin (RR), in a cloud computing environment which will produce a performance comparison that will see which is more effective in a cloud computing environment.

3. RESULT AND DISCUSSION

In this section, we will discuss the evaluation and comparison of performance between load balancing algorithms such as the implementation of Weighted Least Connections and Round Robin in a cloud computing environment. Each algorithm has advantages and disadvantages for the workload distribution process and resource efficiency. The Weighted Least Connections algorithm is more responsive to servers with different connections, while the Round Robin algorithm tends to divide the load evenly without taking into account the state of the server. In the context of cloud computing, load distribution needs to be done optimally so that it produces a fast response time on the application and does not burden the server. Then the Weighted Least Connections and round robin algorithms are tested to ensure that the evaluation of the throughput, latency an cpu usage content can be seen in their performance. This discussion will explain how each algorithm works in a cloud computing environment and situations where one algorithm may be superior to the other.

3.1 Weighted Least Connections

In this context, the Weighted Least Connections (WLC) algorithm will distribute the load in a cloud computing environment that focuses on efficiency with the number of active parameters on the server. This algorithm will consider the weight in the calculation that will be used to measure the capacity of each server that has the benefit of greater connection reception on a large server capacity. The test results carried out in the application of the Weighted Least Connections (WLC) algorithm produce effective performance in distributing the load to server environments that have different specifications. In this case, servers that have better weights will receive more connections while servers with low specifications will receive a small load. It can be concluded that the Weighted Least Connections (WLC) algorithm can distribute the load dynamically based on the capacity of each server. The following are the results of the application of Weighted Least Connections (WLC) on response time, throughput and CPU Utilization

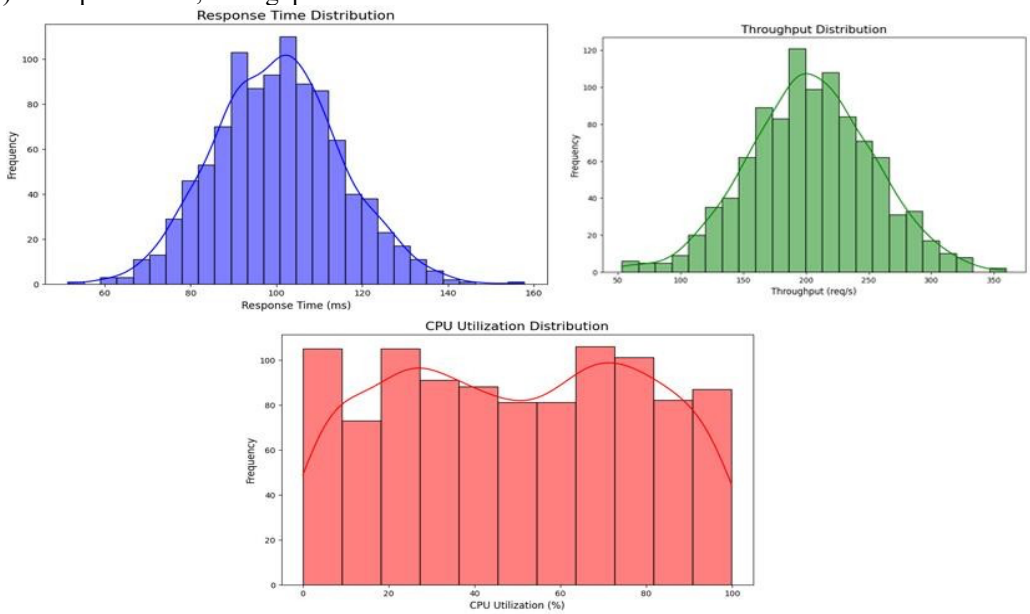

Fig 2. Result Weighted Least Connections (WLC)

In Figure 2 there will be a graph of the implementation of the Weighted Least Connections algorithm where each parameter such as response time, throughput and CPU utilization produces a value from the server side that shows the performance and efficiency of the system. This graph shows that response time decreases along with optimal load distribution, throughput increases with better resource allocation, and CPU utilization is balanced according to the existing server capacity, ensuring that no server experiences overload or suboptimal utilization.

3.2 Round Robin

In this section, there will be an application of the round robin algorithm which is one of the load balancing methods. This algorithm will distribute the load evenly to each server according to the requests of each available server. Each server will carry out the process sequentially according to the rotation of the algorithm. The results of the application of the round robin algorithm show that the distribution of requests to the server is divided evenly without considering the load and active connections on each server so that the response time, CPU utilization and throughput produced are not optimal in the cloud computing environment. In testing with servers that have non-homogeneous capacities, Round Robin results in an increase in response time on servers with lower specifications. This happens because slower servers or with smaller CPU capacities still receive the same number of connections as stronger servers, thus affecting the overall response time. As a result, some servers become slower in processing requests, especially when there is a spike in load. The following is a graph produced in a cloud computing environment with throughput, response time and CPU unitilization parameters using the round robin algorithm shown in Figure 3

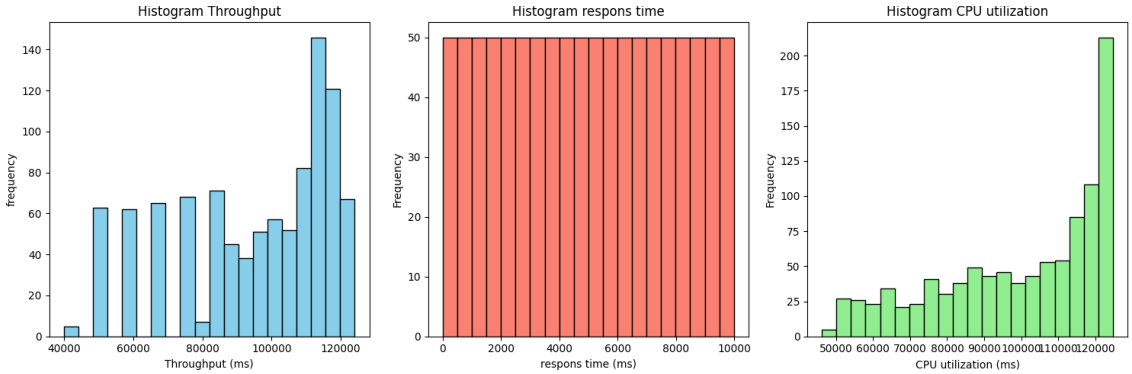

Fig 3. Result Algoritma Round robin

Then in Figure 3 there will be throughput results that produce stable values but there are fluctuations when processed on servers that have quite large capacity differences. The round robin algorithm is not able to distribute the load to servers with better specification levels, so that servers that have large capacities do not work optimally. Likewise with the results of the CPU utilization parameters that produce differences in performance between servers such as servers that have lower capacities will often be overloaded in receiving loads, which has an impact on the stability of the system as a whole.

3.3 Comparison

In this section, a comparison will be made between the Weighted Least Connections (WLC) and Round Robin algorithms which are methods of load balancing. Each algorithm has advantages and disadvantages in distributing the load, server performance and system stability. In the Weighted Least Connections (WLC) algorithm, in distributing the load, it will consider the number of active connections and the capacity of the server. Servers with better weight will receive a lot of load in the connection network and vice versa so that the load distribution is more balanced. On the other hand, the round robin algorithm in distributing the load does not look at the capacity of the server so that there is a significant difference in the server's ability to receive the load. For the response time parameter, the Weighted Least Connections (WLC) algorithm can reduce the response time by dynamically allocating the load according to the server's capabilities, while for the round robin algorithm, the response time tends to increase because servers with low specifications will receive the same number of requests as servers with large specifications. For the parameters of the throughput side, the Weighted Least Connections (WLC) algorithm produces stable and higher throughput values on the server side with good specifications because this algorithm maximizes the use of better servers and request processes in a unit of time, while the round robin algorithm will produce constant or non-maximum throughput due to differences in server capacity. While for the CPU utilization parameter, the Weighted Least Connections (WLC) algorithm produces a more even distribution of the load received because servers with better capacity get more load compared to servers with low capacity, while the round robin algorithm causes an imbalance in CPU utilization. Servers with low capacity tend to be overloaded faster, while servers with high capacity may not be fully utilized. This leads to inefficiency in resource utilization. The following is a comparison graph between the round robin algorithm and the Weighted Least Connections (WLC) algorithm in Figure 4.

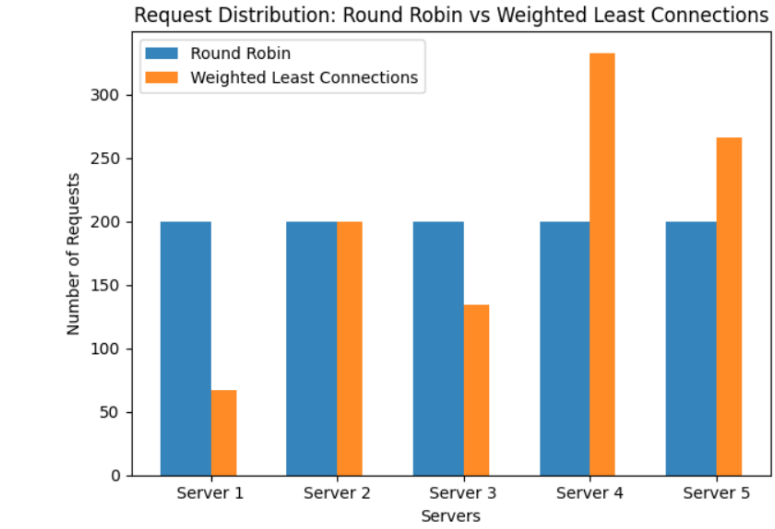

Fig 4. Result Comparison

4. CONCLUSION

The Weighted Least Connections (WLC) algorithm offers better performance in terms of response time, throughput, and CPU utilization in more complex environments with servers of different capacities. By considering the number of active connections and the capabilities of each server, WLC is able to optimize resource allocation, prevent bottlenecks, and improve system stability. Meanwhile, Round Robin is a simpler choice and is more suitable for systems with homogeneous servers, where each server has the same specifications, and in situations with relatively constant loads. In such conditions, the Round Robin algorithm performs quite well even though it does not optimize server capacity efficiently. However, in situations where efficient resource use is critical and more intelligent load distribution is required, WLC is clearly superior to Round Robin because it is able to adjust the load according to the actual capacity of each server, resulting in more optimal performance.

REFERENCES

Al Khowarizmi, Rahmad Syah, Mahyuddin K. M. Nasution, M. E. (2021). Sensitivity of MAPE using detection rate for big data forecasting crude palm oil on k-nearest neighbor. International Journal of Electrical and Computer Engineering (IJECE), 11(3), 2697–2704. https://doi.org/10.11591/ijece.v11i3.pp2697-2704

Arfah, M., Fachrizal, F., & Nugroho, O. (2024). DEVELOPING A MODEL OF ASSOCIATION RULES WITH MACHINE LEARNING IN DETERMINING USER HABITS ON SOCIAL MEDIA. Eastern-European Journal of Enterprise Technologies, 2.

Belgaum, M. R., Musa, S., Alam, M. M., & Su'ud, M. M. (2020). A systematic review of load balancing techniques in software-defined networking. IEEE Access, 8, 98612–98636.

Bryhni, H., Klovning, E., & Kure, O. (2000). A comparison of load balancing techniques for scalable web servers. IEEE Network, 14(4), 58–64.

Kanellopoulos, D., & Sharma, V. K. (2022). Dynamic load balancing techniques in the IoT: A review. Symmetry, 14(12), 2554.

Kashani, M. H., & Mahdipour, E. (2022). Load balancing algorithms in fog computing. IEEE Transactions on Services Computing, 16(2), 1505–1521.

Ma, C., & Chi, Y. (2022). Evaluation test and improvement of load balancing algorithms of nginx. Ieee Access, 10, 14311–14324.

Malik, N., Sardaraz, M., Tahir, M., Shah, B., Ali, G., & Moreira, F. (2021). Energy-efficient load balancing algorithm for workflow scheduling in cloud data centers using queuing and thresholds. Applied Sciences, 11(13), 5849.

Neghabi, A. A., Navimipour, N. J., Hosseinzadeh, M., & Rezaee, A. (2018). Load balancing mechanisms in the software defined networks: a systematic and comprehensive review of the literature. IEEE Access, 6, 14159–14178.

Nugroho, O. (2021). Identifikasi Asal Daerah Berdasarkan Dialek Menggunakan Metode Evolving Multilayer Perceptron. Universitas Sumatera Utara.

Rahmatika, A., Nugroho, O., & AnuR, T. A. (2024). USING RELATIONAL LEARNING IN EXPLORING THE EFFECTIVENESS OF USING HASHTAGS IN FUTURE TOPICS AND USER RELATIONS IN X. Eastern-European Journal of Enterprise Technologies, 2.

Semong, T., Maupong, T., Anokye, S., Kehulakae, K., Dimakatso, S., Boipelo, G., & Sarefo, S. (2020). Intelligent load balancing techniques in software defined networks: A survey. Electronics, 9(7), 1091.

Shafiq, D. A., Jhanjhi, N. Z., & Abdullah, A. (2022). Load balancing techniques in cloud computing environment: A review. Journal of King Saud University-Computer and Information Sciences, 34(7), 3910–3933.

Shahid, M. A., Alam, M. M., & Su'ud, M. M. (2023). Performance evaluation of load-balancing algorithms with different service broker policies for cloud computing. Applied Sciences, 13(3), 1586.

Vecliuc, D.-D., Leon, F., & Logofătu, D. (2022). A comparison between task distribution strategies for load balancing using a multiagent system. Computation, 10(12), 223.