Research Article

Implementation of Load Balancing Using the PCC (Per Connection Classifier) Method on Computer Networks to Improve Responsiveness

[1]Arfianita, [1*]Abdul Wahid, [1]Satria Gunawan Zain, [1]Jumadi M Parenreng

[1] Department of Computer Engineering, Universitas Negeri Makassar, Makassar, South Sulawesi, Indonesia

* Corresponding Author: wahid@unm.ac.id

Abstract: Dependence on a single ISP can lead to the risk of internet disconnection, necessitating a solution to enhance network performance and responsiveness. This research explores the implementation of load balancing using the Per Connection Classifier (PCC) method to improve the responsiveness of computer networks utilizing three ISPs (Internet Service Providers). The test results indicate that the PCC method across the three ISPs can increase throughput by 9.32 Mbps, reduce delay by 43.54 ms, and decrease jitter by 74.17 ms while packet loss remains stable at 0.0%. Additionally, the recursive gateway failover technique in PCC load balancing demonstrates a significant increase in throughput up to 83.3 Mbps, a reduction in delay by 0.27 ms, and a reduction in jitter by 1.15 ms. Packet loss remains stable at 0.0% under varying ISP conditions. Moreover, a comparison between the PCC and PBR methods with the least connection algorithm shows that both methods effectively enhance network responsiveness, with the PCC method exhibiting superior performance in throughput.

Keywords: Failover Recursive Gateway; ISP; Least Connection; Load Balancing; PCC

Citation: Arfianita,A., Wahid, A., Zain, S.G., Parenreng,J.M. (2024). *Implementation of Load Balancing Using the PCC (Per Connection Classifier) Method on Computer Networks to Improve Responsiveness*. Iota, 2024, ISSN 2774-4353, Vol.04, 04. https://doi.org/10.31763/iota.v4i4.810

Academic Editor : Adi, P.D.P

Received : August, 08 2024

Accepted : September, 04 2024

Published : November, 02 2024

1. Introduction

Communication is becoming increasingly important as advances in data communication technology continue to evolve. Computer networks are currently among the most widely used information technologies. Their benefits and roles are crucial for various users, including individuals, institutions, companies, and organizations, both private and governmental. Computer networks facilitate many valuable functions, supporting various activities and specific needs. With computer networks, users can easily share data, send information, and engage in other activities related to information exchange[1].

An internet connection is a global network that enables devices to connect and share information instantly. The rapid growth of internet-based telecommunications technology has transformed how users communicate, work, and access information. A fundamental expectation of internet services is their constant availability. In a world increasingly dependent on Internet connectivity, it is essential to ensure that Internet services remain consistently available[2].

The smooth running of various daily activities depends on a stable internet connection. However, network issues can disrupt critical processes such as business operations, communication, and information access. Unstable connections and internet outages are often caused by signal interference and insufficient output due to inadequate bandwidth and excessive network load. This issue usually results from relying on a single Internet Service Provider (ISP). If the ISP line fails or a problem arises in the local network connected to the Internet, no recovery measures are available to address the disruption[3].

Table 1. Literature of Study

Previous Study	Strengths	Weakness
Load balancing optimization of two ISPs for Mikrotik-based bandwidth management [4].	The load balancing method optimally balances the bandwidth usage of two or more ISPs by ensuring efficient internet connection utilization.	If one of the ISPs experiences problems, the internet connection will be disrupted, affecting internet network performance.
Integration of ISP Broadband_UNM and 3 LTE Using PCC Method to Improve Internet Connection Stability [5].	This research shows that if one ISP is disabled, the system automatically switches its connection to the active ISP.	The drawback of this research is that the failover mechanism can only monitor the ISP link. So, when there is a problem with the internet connection, failover cannot monitor it on the router side.
Installation and Configuration of Haproxy as a Load Balancing Web Server at PT. Pupuk Sriwidjaja Palembang [6].	This research shows that load balancing can distribute traffic sequentially according to server capacity so that all servers receive a balanced load.	This research has yet to comprehensively evaluate load balancing performance, such as availability, throughput, delay, jitter, and packet loss.

Table 1 summarizes several studies on load-balancing optimization with various ISPs. The first study shows that the load balancing method on Mikrotik effectively balances bandwidth usage but is vulnerable to ISP interference without failover. The second study implemented the Per Connection Classifier (PCC) method to switch to the active ISP if one fails automatically, but failover only monitors the ISP link, not the router. The third study, using HAProxy, emphasizes even server load distribution but still needs to evaluate performance thoroughly.

Therefore, to avoid connection failures when relying on a single Internet Service Provider (ISP), this research implements load balancing using three ISPs with the Per Connection Classifier (PCC) method to improve responsiveness in computer networks. By utilizing three ISPs, the total bandwidth capacity available for network traffic increases. Responsiveness can be enhanced by measuring and optimizing the Quality of Service (QoS) in load balancing. Quality of Service (QoS) assesses network performance and ensures the implemented load balancing system meets established quality standards.

This research's novelty lies in applying load balancing techniques that utilize three ISPs with the Per Connection Classifier (PCC) method to improve computer network responsiveness. By integrating three ISPs, this research aims to increase the bandwidth capacity available for network traffic and optimize the quality of service (QoS) to meet predefined quality standards. This innovative approach expands network coverage and capacity and overcomes the weakness of previous methods limited to monitoring ISP links without considering issues on the router side. By applying the failover recursive gateway load balancing technique, this research can monitor and detect disruptions in each ISP's internet connection, automatically switching to an ISP with a stable connection if one of the ISPs experiences problems. In addition, it also compares the PCC method with Policy Routing (PBR), which uses the least connection algorithm to distribute the connection load and improve network responsiveness evenly.

2. Method

2.1 *Load Balancing*

Load balancing is a method used to distribute traffic across multiple connection lines evenly. The goal is to improve performance, maximize throughput, reduce response time, and prevent overload on any single connection line [7][8]. Load balancing does not duplicate connections; instead, it distributes existing connections across multiple servers [9]. Workloads can be distributed across various resources such as computers, networks, CPUs, and hard drives to maximize resource utilization, thereby maintaining performance and service availability [10][11].

The primary goal of load balancing is to evenly distribute traffic across nodes in a cluster to enhance network quality. This includes improving service reliability and user satisfaction, optimizing resource utilization, reducing job execution and wait times, boosting service performance, maintaining server cluster stability, and creating a fault-tolerant, scalable, and easily modifiable system [8].

2.2 *Per Connection Classifier (PCC) Method*

The Per Connection Classifier (PCC) method is used to classify connection traffic passing through a router based on the source address (src-address), destination address (dst-address), source port (src-port), and destination port (dst-port). The router will maintain a memory of the gateway path used by the first packet of a connection. Thus, subsequent packets associated with the connection will be forwarded through the same gateway[12]. PCC is a feature that monitors incoming and outgoing connections on the router using specific criteria. This PCC function balances load by combining two or more Internet Service Providers (ISPs) [9].

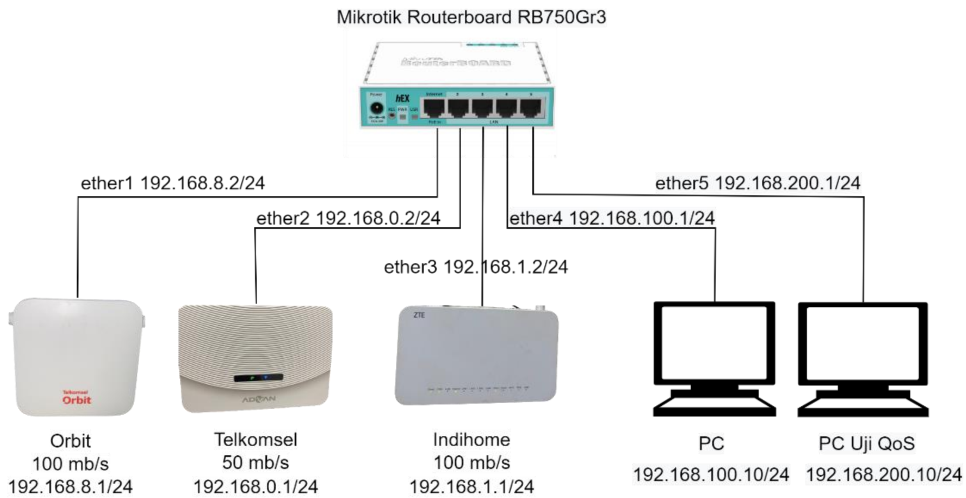


Figure 1. Design System

Figure 1 shows the design system involving several vital components using the PCC (Per Connection Classifier) load balancing method on the MikroTik RB750GR3 router. Mikrotik is a data flow regulator in the local area network[13]. There are three internet connections from different providers, namely Orbit with 100 Mb/s bandwidth (IP 192.168.8.1/24), Telkomsel with 50 Mb/s bandwidth (IP 192.168.0.1/24), and Indihome with 100 Mb/s bandwidth (IP 192.168.1.1/24). An Internet Service Provider (ISP) is a company that provides Internet access to users. ISPs build internet network infrastructure, such as fiber optic cables and communication satellites. To connect to the Internet, users must subscribe to an ISP. ISPs provide certain conditions to users, such as subscription fees, data transfer speeds, and internet access time limits [14]. The MikroTik router connects each internet connection to a different port: Ether1 (IP 192.168.8.2/24) for Orbit, Ether2 (IP 192.168.0.2/24) for Telkomsel, and Ether3 (IP 192.168.1.2/24) for

Indihome. On the local network (LAN) side, there are two devices: a PC with IP 192.168.100.10/24 connected to Ether4 (IP 192.168.100.1/24) and a QoS Test PC with IP 192.168.200.10/24 connected to Ether5 (IP 192.168.200.1/24).

QoS testing using Wireshark. This test ensures that implementing load balancing using the PCC method can increase responsiveness on computer networks. Wireshark is an open-source packet data capture application that can scan and capture traffic on the Internet. It is commonly used as a troubleshooting tool on problematic networks, and since it can read any packet traffic content, it is also widely used for testing software [15].

2.3 Failover Recursive Gateway

Failover is a technique that prevents failures due to increased traffic caused by high workloads when facing significant demand from ISP users. When one ISP device fails, another ISP device acts as a backup or replacement to maintain service continuity [16]. Additionally, gateway selection needs to be flexible enough to adjust to the traffic conditions of the moment because traffic is dynamic and subject to frequent and unpredictable variations[17]. A failover recursive gateway is a technique in computer networks used to improve service availability by automatically switching to another gateway in the event of a failure of the primary gateway. The term "recursive" in this context refers to the ability of the system to iteratively search for an alternative gateway if the primary one is no longer usable.

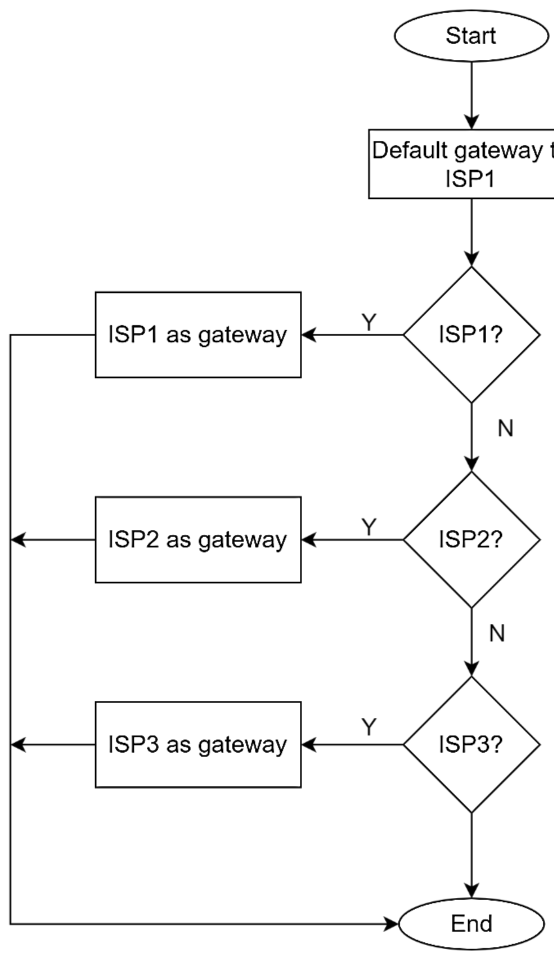


Figure 2. Flowchart Failover Recursive Gateway

Figure 2 illustrates switching the default gateway between ISPs to ensure internet connection availability. The process starts with setting the default gateway to ISP1. The system then checks the connection to ISP1. If it is available, the process is completed. Otherwise, the system checks the connection to ISP2. The gateway is switched to ISP2, and the process is completed if available. Otherwise, the system checks the connection to ISP3. The gateway is redirected to ISP3, and the process is completed if available. If no connection is available to ISP1, ISP2, or ISP3, the process ends without a connection.

2.4 Least Connection Algorithm

The least connection algorithm uses Policy-Based Routing (PBR), a network management method that establishes rules for allocating network traffic to achieve balanced load distribution. The network administrator creates these rules to ensure the efficient use of network resources[18]. The least connection algorithm is a load-balancing method that prioritizes servers with the fewest connections to receive additional load. In other words, the server with the lightest load will be given more connections to handle [18]. Least Connections distributes traffic by selecting the server with the fewest active connections[11].

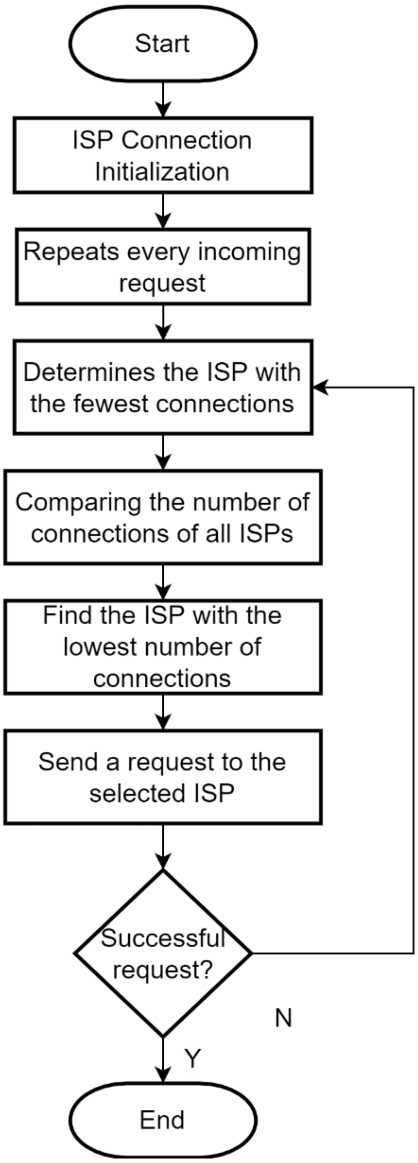


Figure 3. Flowchart Least Connection Algorithm

Figure 3 shows the flowchart of the least connection algorithm used for load balancing on a network with 3 ISPs. This algorithm starts with initializing the number of active connections for each ISP. Each ISP has a counter that indicates the number of active connections. When a request comes in, the algorithm compares the number of active connections of each ISP to determine the ISP with the lowest number of connections. The request is then forwarded to that ISP. This process is repeated for each request, balancing the traffic load among all ISPs. This algorithm optimizes network performance and ensures efficient use of resources.

2.5 QoS Measurement on Responsiveness Improvement

Responsiveness in computer networks is the system's ability to respond quickly to user or data requests. In implementing load balancing using the PCC method, responsiveness can be measured by how efficiently and balanced the traffic load distribution among several paths or connections is. Quality of service (QoS) measures how effectively a network provides services to users [19]. It helps ensure that various services or applications in the network get the appropriate level of service as needed. Quality of service (QoS) parameters include throughput, delay, jitter, and packet loss.

2.5.1 Throughput

Throughput is the data transfer rate calculated as the total number of packets that successfully arrive at the destination during a specific time interval, which is then divided by the time interval. Throughput is the overall rate at which a system processes and completes tasks[19][20]. The throughput value can be calculated using the following equation 1.

$$Throughput = (number\ of\ data\ sent)/(data\ transmission\ time) \quad (1)$$

Table 1. Throughput Standardization

Throughput Categories	Throughput	Index
Excellent	>2,1 Mbps	4
Good	700-1200 kbps	3
Medium	338-700 kbps	2
Poor	0-338 kbps	1

Table 2 shows the standardization of throughput according to THIPON (Telecommunication Internet Protocol Harmonization Over Network). More than 2.1 Mbps throughput falls into the excellent category (index 4). A throughput of 700-1200 Kbps falls into the good category (index 3). A throughput of 338-700 Kbps falls into the medium category (index 2). Throughput 0-338 Kbps falls into the poor category (index 1).

2.5.2 Delay

Delay is the time to transfer and process a request on the server[20]. The delay value can be calculated using the following equation 2.

$$Delay = (delay\ total\ )/(total\ numbers\ of\ delays) \quad (2)$$

Table 2. Delay Standardization

Delay Categories	Delay	Index
Excellent	<150 ms	4
Good	150 s/d 300 ms	3
Medium	300 s/d 450 ms	2
Poor	>450 ms	1

Table 3 shows the delay standardization according to THIPON. A delay of less than 150 ms is categorized as excellent (index 4). A delay of 150-300 ms falls into the good category (index 3). A delay of 300-450 ms falls into the medium category (index 2). A delay of more than 450 ms falls into the poor category (index 1).

2.5.3 Jitter

A jitter is a fluctuation or change in the period of data packet arrival or variation in delay time. Jitter refers to the variation in the delay of data packet reception[20][21]. The jitter value can be calculated using the following equation 3.

$$Jitter = (total\ variation\ of\ delay)/(total\ packets\ of\ data\ received) \qquad (3)$$

Table 3. Jitter Standardization

Jitter Categories	Jitter	Index
Excellent	0 ms	4
Good	1-75 ms	3
Medium	76-125 ms	2
Poor	126-225 ms	1

Table 4 shows the jitter standardization according to THIPON. Jitter 0 ms falls into the excellent category (index 4). Jitter 1-75 ms falls into the good category (index 3). Jitter 76-125 ms is medium (index 2). Jitter 126-225 ms falls into the bad category (index 1).

2.5.4 Packet Loss

Packet loss occurs when one or more packets fail to reach their destination. It is typically caused by network congestion. The packet loss rate is the percentage of packets lost compared to the total packets sent[20][22]. The packet loss value can be calculated using the following equation 4.

$$Packet\ loss = \frac{(captured\ packets - packets\ sent}{total\ captured\ packets} x\ 100\% \qquad (4)$$

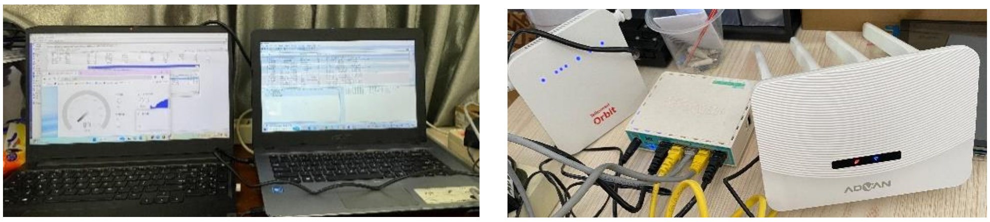
Table 4. Packet Loss Standardization

Packet loss Categories	Packet loss	Index
Excellent	0%	4
Good	3%	3
Medium	15%	2
Poor	-25%	1

Table 5 shows the standardization of packet loss according to THIPON. A packet loss of 0% is categorized as excellent (index 4). A packet loss of 3% is categorized as good (index 3). A packet loss of 15% is categorized as medium (index 2). Packet loss of -25% is categorized as poor (index 1).

3. Result and Discussion

In this section, we will discuss in detail the results of the research built. Figure 4 illustrates the load balancing setup. In Figure 4(a), PC 1 is shown as the system configuring load balancing, while PC 2 tests the system's responsiveness. Figure 4(b) depicts the connection between devices such as the modem router, MikroTik router, and PCs, enabling the load balancing functionality.



(a) (b)

Figure 4. Load Balancing Setup

3.1 Responsiveness of Load Balancing PCC Method 3 ISPs

Quality of Service (QoS) testing was carried out before and after implementing load balancing on 3 ISPs using 3 ISPs to determine the increase in responsiveness on a computer network.

Furthermore, Figure 5(a) shows the throughput test results at ISP 1 increased with a value of 0.811 Mbps, ISP 2 increased with a value of 8.5 Mbps, and ISP 3 increased with a value of 0.362 Mbps. Figure 5(b) shows the delay at ISP 1 decreased with a value of 8.37 ms, ISP 2 decreased with a value of 0.69 ms, and ISP 3 decreased with a value of 14.6 ms. Figure 5(c) jitter at ISP 1 decreased with a value of 15.5 ms. ISP 2 decreased with a value of 1.23 ms, and ISP 3 decreased by 26.6 ms. Figure 5(d) shows that packet loss at ISP 1, ISP 2, and ISP 3 has not changed, remaining at a value of 0.0%.

Based on TIPHON throughput test results, ISP 1 is in the good category with index 3, ISP 2 is in the very good category with index 4, and ISP 3 is in the medium category with index 2. The delay of the three ISPs is very good with index 4. Packet loss on the three ISPs is in the very good category with an index of 4. And the jitter of the three ISPs is in the good category with index 3.

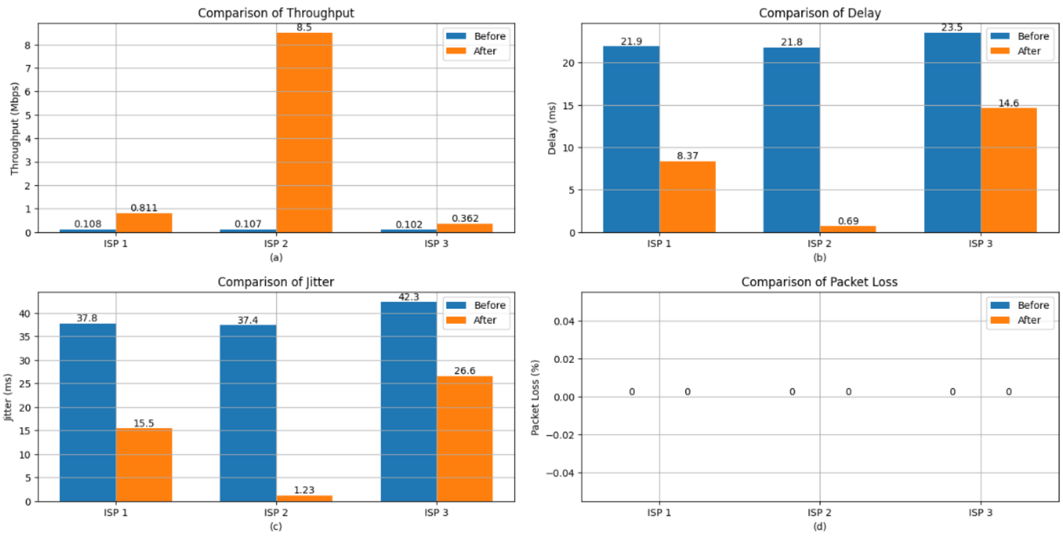


Figure 5. QoS Load Balancing Graph of PCC Method 3 ISPs

Table 5. Responsiveness of PCC Method Load Balancing

QoS Parameters	Responsiveness
Throughput (Mbps)	Increase 9.32 Mbps
Delay (ms)	Reduction 43,54 ms
Jitter (ms)	Reduction 74,17 ms
Packet Loss (%)	Remains 0,0 %

Table 6 shows the responsiveness of PCC load balancing. The increase in throughput by 9.32 Mbps indicates that the network can transmit more data per second and improve access speed. The delay was reduced by 43.54 ms, indicating that the time taken for data to travel from source to destination was reduced. Jitter was reduced by 74.17 ms, which indicates better data delivery stability. Packet loss did not change or remained stable at 0.0%.

3.2 Failover Recursive Gateway Load Balancing PCC Method in Responsiveness Improvement

Quality of Service (QoS) testing was carried out before and after implementing a failover recursive gateway to determine the increase in responsiveness on a computer network.

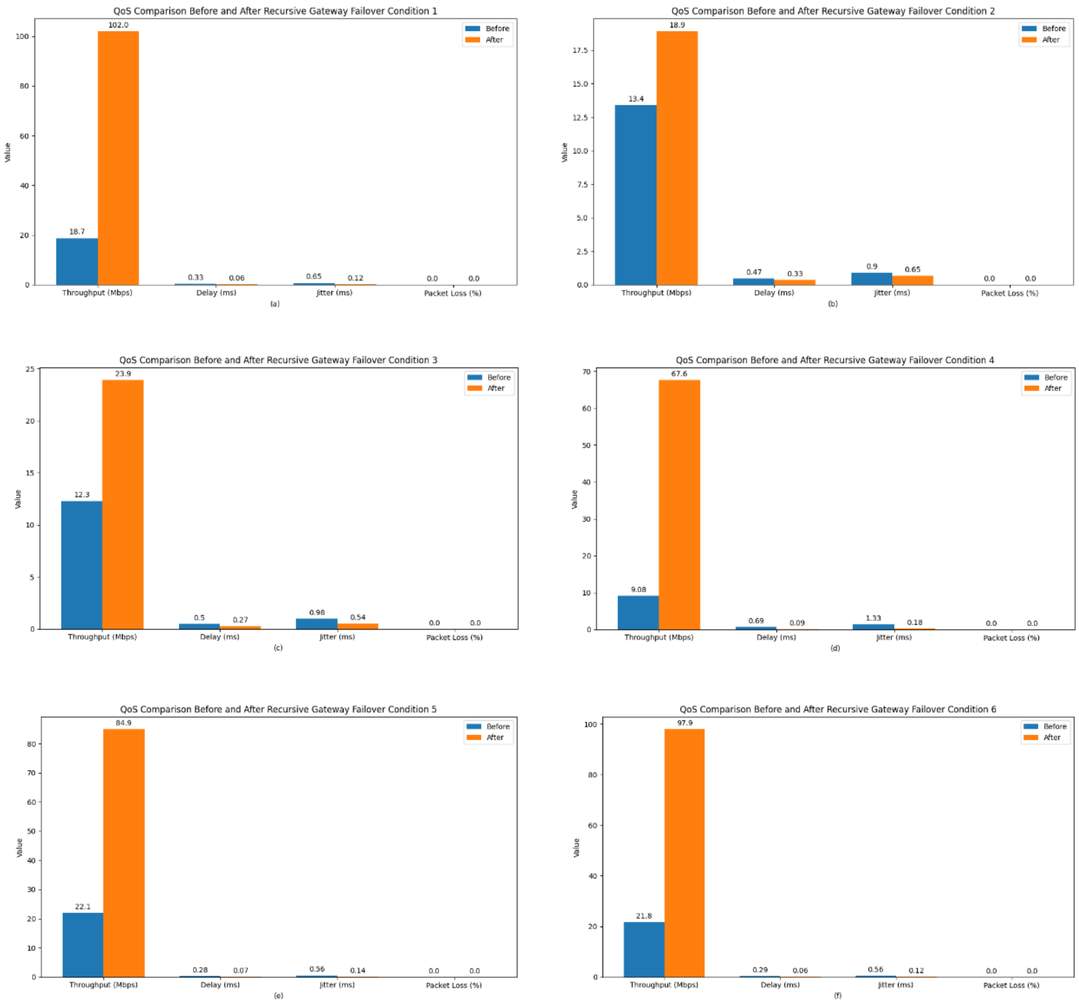


Figure 6. QoS Failover Recursive Gateway Graph

Figure 5 shows Quality of Service (QoS) testing before and after applying the PCC method with a failover recursive gateway technique for various ISP conditions. Figure 6(a) is a graph of the QoS testing conditions of ISP 1 on, ISP 2 on, and ISP 3 off. Throughput increased from 18.7 Mbps to 102 mbps. The delay was reduced from 0.33 ms to 0.06 ms. Jitter reduced from 0.65 ms to 0.12 ms. Packet loss does not change and remains at 0.0%. Figure 6(b) is a graph of QoS testing conditions 2 ISP 1 on, ISP 2 off, and ISP 3 on. Throughput increased from 13.4 mbps to 18.9 mbps. The delay is reduced from 0.47 ms to 0.33 ms. Jitter reduced from 0.9 ms to 0.65 ms. Packet loss does not change and remains at 0.0%. Figure 6(c) is a graph of QoS testing conditions 3 ISP 1 off, ISP 2 on, and ISP 3 on. Throughput increased from 12.3 mbps to 23.9 mbps. The delay was reduced from 0.5 ms to 0.27 ms. Jitter reduced from 0.98 ms to 0.54 ms. Packet loss did not change and remained at 0.0%.

Figure 6(d) is a graph of QoS testing conditions 4 ISP 1 on, ISP 2 off, and ISP 3 off. Throughput increased from 9.08 mbps to 67.6 mbps. The delay was reduced from 0.69 ms to 0.09 ms. Jitter reduced from 1.33 ms to 0.18 ms. Packet loss does not change and remains at 0.0%. Figure 6(e) is a graph of QoS testing conditions 5 ISP 1 off, ISP 2 on, and ISP 3 off. Throughput increased from 22.1 Mbps to 84.9 Mbps. The delay was reduced from 0.28 ms to 0.07 ms. Jitter reduced from 0.56 ms to 0.14 ms. Packet loss does not change and remains at 0.0%. Figure 6(f) is a graph of QoS testing conditions 6 ISP 1 off, ISP 2 off, and ISP 3 on. Throughput increased from 21.8 Mbps to 97.9 Mbps. The delay was reduced from 0.29 ms to 0.06 ms. Jitter reduced from 0.56 ms to 0.12 ms. Packet loss did not change and remained at 0.0%.

Based on TIPHON, the QoS test results from various ISP conditions on the throughput value are in a very good category with an index of 4. The delay value is in a very good category with an index of 4. The jitter value is in a very good category. Although the value of QoS parameters was in a good category before applying the failover recursive gateway technique, it was higher after applying it.

Table 6. Failover Recursive Gateway Responsiveness

QoS Parameters	Responsiveness
Throughput (Mbps)	Increase up to 83,3 Mbps
Delay (ms)	Reduction up to 0,27 ms
Jitter (ms)	Reduction up to 1,15 ms
Packet Loss (%)	Remains 0,0 %

Table 7 shows the responsiveness of the recursive failover gateway from various ISP conditions. The increase in throughput up to 83.3 Mbps shows that the network can send more data per second and increase access speed. The reduction in delay up to 0.27 ms shows that the time required for data to travel from source to destination is reduced. The reduction in jitter by 1.15 ms shows better data delivery stability. Packet loss does not change or remains stable at 0.0%.

3.3 Comparison of PCC Load Balancing Method with PBR Load Balancing Least Connection Algorithm.

Table 7. Distribution of Connection Load Using PCC Load Balancing

No	Src. Address	Src. Port	Dst. Address	Dst. Port	Hash Value	Mark Connection		
						ISP 1	ISP 2	ISP 3
1	192.168.100.10	5970	20.198.119.84	443	2	-	-	√
2	192.168.100.10	6082	74.125.200.188	5228	1	-	√	-
3	192.168.100.10	58324	172.64.155.141	443	0	√	-	-
4	192.168.100.10	6058	31.13.95.61	443	0	√	-	-
5	192.168.100.10	52933	104.18.32.115	443	1	-	√	-
6	192.168.100.10	6113	31.13.95.61	443	2	-	-	√

Table 8 shows how connections from a single source address (Src. Address) are distributed to different ISPs based on the hash value and connection mark. All connections originate from the same IP address, 192.168.100.10. This is the device's IP address on the local network, making the outgoing connection. The source port is different for each connection, indicating the different sessions or connections created by the device. Connections are routed to different destination IP addresses and destination ports. This includes connections to different servers on the internet on different ports (e.g., port 443 for HTTPS, port 5228 for Google Services, etc.).

The hash value is the result of the hashing algorithm used by the PCC method to determine where the connection will be routed. This hash value is then used to determine which ISP to use for the connection. A connection with a hash value of 0 will be forwarded to ISP 1, a hash value of 1 will be forwarded to ISP 2, and a hash value of 2 will be forwarded to ISP 3.

Table 8. Distribution of Connection Load Using Load Balancing Least Connection Algorithm

Testing	Number of Connections			Distance		
	ISP 1	ISP 2	ISP 3	ISP 1	ISP 2	ISP 3
1	8	10	12	1	2	3
2	7	13	9	1	3	2
3	8	10	10	1	2	2
4	16	8	8	2	1	1
5	11	10	12	2	1	3
6	16	7	11	3	1	2
7	13	12	13	2	1	2
8	27	58	27	1	2	1
9	11	17	10	2	3	1
10	36	17	13	3	2	1
11	13	13	9	2	2	1
12	5	5	8	1	1	2
13	11	11	11	1	1	1

Table 9 tests load balancing connection distribution using the PBR method with the least connection algorithm. The PBR method with the least connection algorithm directs network traffic to the ISP with few active connections. The test table shows the number of connections and distance from each ISP. Distance in routing determines the priority of the route. Routes with lower distances will be selected first. Distance is set based on the number of active connections. The ISP with the fewest connections is given the lowest distance, so that route will be selected first.

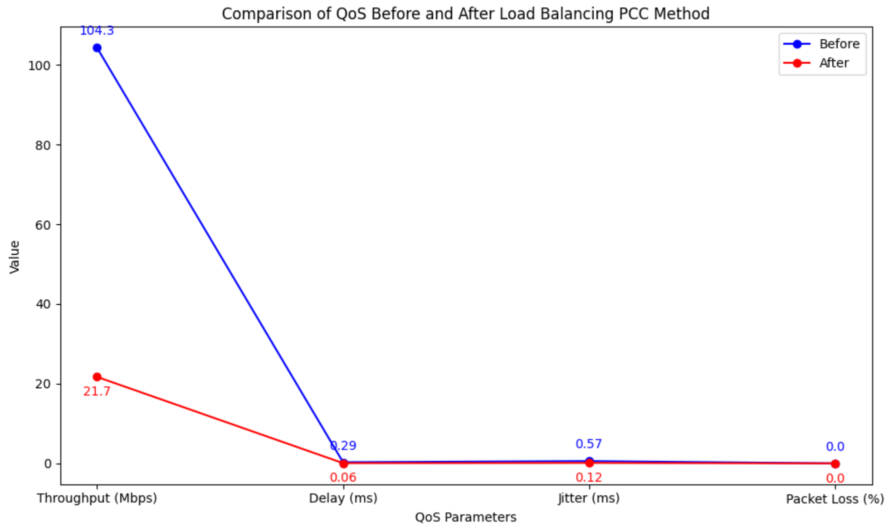


Figure 7. QoS Graph of PCC Method Load Balancing

Moreover, Figure 7 is a graph of QoS testing before and after implementing the PCC load balancing method. Throughput increased from 21.7 Mbps to 104.3 Mbps. The delay is reduced from 0.29 ms to 0.06 ms. Jitter reduced from 0.57 ms to 0.12 ms. Packet loss did not change and remained at 0.0%.

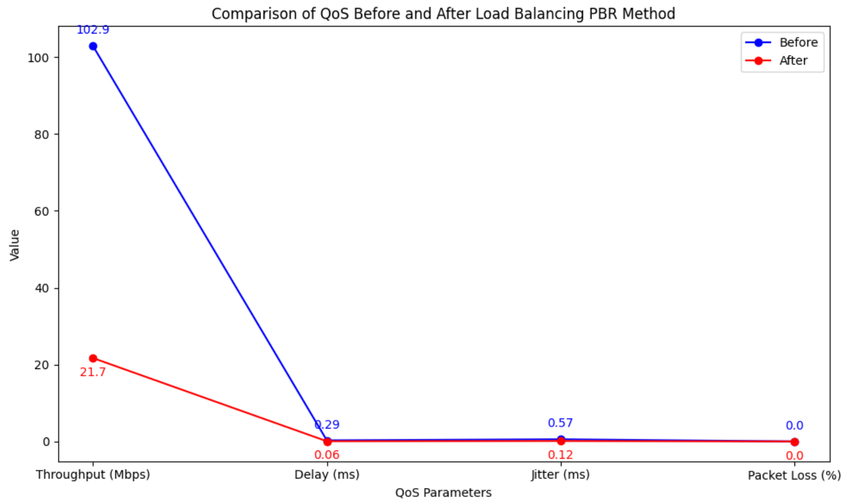


Figure 8. QoS Graph of Least Connection Algorithm PBR Method

Figure 8 is a graph of QoS testing before and after implementing the load-balancing PBR method with the least connection algorithm. Throughput increased from 21.7 Mbps to 102.9 Mbps. The delay is reduced from 0.29 ms to 0.06 ms. Jitter is reduced from 0.57 ms to 0.12 ms. Packet loss did not change and remained at 0.0%.

Table 9. Responsiveness of PCC and PBR Load Balancing Least Connection Algorithm

QoS Parameters	PCC	PBR Least Connection
Throughput (Mbps)	Increase 82,6 Mbps	Increase 81,2 Mbps
Delay (ms)	Reduction 0,23 ms	Reduction 0,23 ms
Jitter (ms)	Reduction 0,45 ms	Reduction 0,45 ms
Packet Loss (%)	Remains 0,0%	Remains 0,0%

Table 10 shows the increase in responsiveness after the implementation of load balancing. In the PCC method, there is a significant increase in throughput of 82.6 Mbps, while the PBR Least Connection Delay reduction increases throughput to 81.2 Mbps. Packet loss has not changed or remains stable at 0.0%. There is a reduction in delay of 0.23 ms and a reduction in jitter of 0.45 ms, and packet loss does not change, remaining at a value of 0.0% in both methods.

An increase in throughput means that the network can handle more data in the same period and improve the overall performance. Reduced delay indicates that the time taken for data to travel from source to destination is reduced. Reduced jitter indicates the time variation in data delivery is reduced. Packet loss that remains stable indicates that no packets are lost during the data delivery process.

4. Conclusion and Future Research

Based on the research and testing results, researchers can conclude that the Augmented Reality (AR) based car showroom application developed for the Alya Motor Car Showroom has successfully met the expected objectives. Black box testing shows that Implementing the load balancing Per Connection Classifier (PCC) method with 3 ISPs proved to improve the responsiveness of computer networks. Tests show an increase in throughput of 9.32 Mbps, a reduction in delay of 43.54 ms, and a reduction in jitter of 74.17 ms, and packet loss has not changed or remains stable at a value of 0.0%. Applying the failover recursive gateway technique in the PCC load balancing method also shows a significant increase in responsiveness in various ISP conditions. This technique effectively ensures the network remains responsive and stable despite changes in ISP conditions. Comparison between the PCC load balancing method and the PBR method of least connection algorithm shows that both can improve responsiveness. The PCC method provides higher throughput improvement than the PBR least connection method. Both methods are equally effective in reducing delay, jitter, and packet loss, which remains stable.

Future research on load balancing implementation should examine how the PCC (Per Connection Classifier) and PBR (Policy Based Routing) methods affect network security. This research can include an in-depth analysis of the effectiveness of the PCC method in maintaining the integrity and security of data transmitted over the network.

Acknowledgments: We extend our gratitude to all parties, particularly at Universitas Negeri Makassar, for their help in completing this research, refining the data, and processing it to finalize this article. We appreciate everyone's contributions.

Author contributions: Conceptualizations, methodology, validation, formal analysis, investigation, data curation, writing-original draft preparation, writing-review and editing, visualization, supervision, project administration, funding acquisition; Arfianita [A], Abdul Wahid [AW], Satria Gunawan Zain [SGZ], Jumadi M Parenreng [JMP], have read and agreed to the published version of the manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

[1] N. Nurmiati, L. Surimi, and S. Subardin, "Analisis Kinerja Load Balancing Terhadap Jaringan Internet Menggunakan Metode Equal Cost Multi Path (ECMP)," *Digit. Transform. Technol.*, vol. 2, no. 2, pp. 52–62, 2022, doi: 10.47709/digitech.v2i2.1779.

[2] N. Y. Arifin, Y. Suryani, and H. Bashori, "Analisis Dan Perancangan Jaringan Mikrotik Load Balancingpada Unit Laka Lantas Polresta Barelang," *Eng. Technol. Int. J.*, vol. 3, no. 2, p. 2, 2021, doi: 10.55642/eatij.v3i02.

[3] D. Gusrion, R. L. Gema, S. Safira, A. F. Hadi, and S. Andini, "Load Balance Pcc and Failover With Recursive Gateway," *J. Ipteks Terap.*, vol. 16, no. 4, pp. 715–724, 2022.

[4] F. Utami, H. Lindawati, and SUzanzefi, "Optimalisasi Load Balancing Dua ISP untuk Manajemen Bandwith Berbasis Mikrotik," *Pros. Semin. Nas. Multi Disiplin Ilmu Call Pap.*, pp. 75–82, 2017.

[5] M. F. I. P. Putra, J. M. Parenreng, and M. Yahya, "Integration of Telkom ISP and 3 LTE Using PCC Method to Improve Internet Connection Stability," *Internet Things Artif. Intell. J.*, 2024.

[6] H. Setiawan and A. Paluthi, "Instalasi dan kongifurasi Haproxy sebagai Load Balancing Web Server di PT. Pupuk Sriwidjaja Palembang," vol. 1, no. 2, pp. 37–48, 2022.

[7] Nugroho & Rezka, "Simulasi Management Bandwidth Dan Load Balancing Server," *Simulasi Manag. Bandwidth Dan Load Balanc. Serv. Menggunakan Clear Os Pada Virtual Box*, vol. 14, pp. 80–93, 2014.

[8] N. Xuan Phi, L. Ngoc Hieu, and T. Cong Hung, "Load Balancing Algorithm on Cloud Computing for Optimize Respone Time," *Int. J. Cloud Comput. Serv. Archit.*, vol. 10, no. 3, pp. 15–29, 2020, doi: 10.5121/ijccsa.2020.10302.

[9] B. Gofindo Malau, "Implementasi Load Balancing Mikrotik Jaringan Internet Di Pardamean Sibisa, Ajibata, Toba Samosir, Sumatra Utara," *J. Comput. Sci. Technol.*, vol. 2, no. 1, pp. 20–29, 2022, doi: 10.54840/jcstech.v2i1.23.

[10] A. Rahmatulloh and F. MSN, "Implementasi Load Balancing Web Server menggunakan Haproxy dan Sinkronisasi File pada Sistem Informasi Akademik Universitas Siliwangi," *J. Nas. Teknol. dan Sist. Inf.*, vol. 3, no. 2, pp. 241–248, 2017, doi: 10.25077/teknosi.v3i2.2017.241-248.

[11] J. Zhou *et al.*, "Comparative analysis of metaheuristic load balancing algorithms for efficient load balancing in cloud computing," *J. Cloud Comput.*, vol. 12, no. 1, 2023, doi: 10.1186/s13677-023-00453-3.

[12] E. R. Amalia, Nurheki, R. Saputra, C. Ramadhana, and E. H. Yossy, "Computer network design and implementation using load balancing technique with per connection classifier (PCC) method based on MikroTik router," *Procedia Comput. Sci.*, vol. 216, pp. 103–111, 2022, doi: 10.1016/j.procs.2022.12.116.

[13] M. S. Nur Wahid, J. M. Parenreng, A. Wahid, W. Hidayat M, and M. Mujahidin, "Pengembangan dan Implementasi Jaringan LAN Berbasis Mikrotik untuk Peningkatan Akses Internet dan Konektivitas di MTS An Nahdlah Ujung Pandang," *J. Kemitraan Responsif untuk Aksi Inov. dan Pengabdi. Masy.*, vol. 1, no. 1, pp. 34–46, 2023, doi: 10.61220/kreativa.v1i1.20235.

[14] H. Santoso, "Strategi Memilih Internet Service Provider Terbaik untuk Perguruan Tinggi (Studi Kasus: STMIK ATMA LUHUR)," *Semin. Nas. Apl. Teknol. Inf.*, vol. 2012, no. Snati, p. 1, 2012.

[15] A. Wahid, M. E. Firdaus, and J. M. Parenreng, "Implementation of Wireshark and IPtables Firewall Collaboration to Improve Traffic Security on Network Systems," *Internet Things Artif. Intell. J.*, vol. 1, no. 4, pp. 249–264, 2021, doi: 10.31763/iota.v1i4.509.

[16] W. Adhiwibowo and A. R. Irawan, "Implementasi Redundant Link Untuk Mengatasi Downtime Dengan Metode Failover," *J. Pengemb. Rekayasa dan Teknol.*, vol. 15, no. 1, p. 48, 2019, doi: 10.26623/jprt.v15i1.1490.

[17] J. J. Galvez, P. M. Ruiz, and A. F. G. Skarmeta, "Responsive on-line gateway load-balancing for wireless mesh networks," *Ad Hoc Networks*, vol. 10, no. 1, pp. 46–61, 2012, doi: 10.1016/j.adhoc.2011.06.002.

[18] S. H. Z. M. I. N. A. Subandri, "Manajemen Internet Service Provider Dengan Menggunakan Metode Policy Based Route Studi Kasus: Gedung Serba Guna Istanaku," *Sist. Inform. Jar. Komputer, dan Apl.*, vol. 11, no. 2, pp. 24–28, 2021.

[19] S. Shahryari, S. A. Hosseini-Seno, and F. Tashtarian, "An SDN based framework for maximizing throughput and balanced load distribution in a Cloudlet network," *Futur. Gener. Comput. Syst.*, vol. 110, no. 2020, pp. 18–32, 2020, doi: 10.1016/j.future.2020.04.009.

[20] M. Rostami and S. Goli-Bidgoli, *An overview of QoS-aware load balancing techniques in SDN-based IoT networks*, vol. 13, no. 1. Springer Berlin Heidelberg, 2024. doi: 10.1186/s13677-024-00651-7.

[21] G.-C. Deng and K. Wang, "An Application-aware QoS Routing Algorithm for SDN-based IoT Networking," *IEEE Symp. Comput. Commun.*, 2018, doi: https://doi.org/10.1109/ISCC.2018.8538551.

[22] A. Akbar Neghabi, N. Jafari Navimipour, M. Hosseinzadeh, and A. Rezaee, "Nature-inspired meta-heuristic algorithms for solving the load balancing problem in the software-defined network," *Int. J. Commun. Syst.*, vol. 32, no. 4, pp. 1–26, 2019, doi: 10.1002/dac.3875.