SinkrOn
JURNAL & PENELITIAN TEKNIK INFORMATIKA
Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 4, October 2022
DOI : https://doi.org/10.33395/sinkron.v7i4.11651
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Analysis performance of content delivery network by used Rateless Code method

Sahat Parulian Sitorus[1)*], Elysa Rohayani Hasibuan[2)], Rohani[3)]
[1)2)3)]University of Labuhanbatu, North Sumatra, Indonesia
[1)]sahatparuliansitorus4@gmail.com, [2)]elysa.hasby@gmail.com, [3)]pasariburohani@gmail.com

Submitted : Aug 8, 2022 | **Accepted** : Sep 4, 2022 | **Published** : Oct 3, 2022

Abstract: Content Delivery Network (CDN) is a server system network to content contribution in an application/web to various accessors around the world so that data sent and received by users was faster. Server selection algorithm determines the extent to which the quality of server service to client requests. This study to analyze CDN performance, used Geographical algorithm Domain Name Server (GeoDNS), Round Robin (RR), Weigthed Round Robin (WRR), dan Least Connection (LC). In addition, Rateless Code (RLC) is also applied to reduce packet loss, there are two methods used, namely the first method is rateless code 50 and the second method is rateless code 100. Thus, the trial was carried out 20 times. Besides that, it is also applied Rateless Code (RLC) to less packet loss. From the research results, the use of CDN can improve server service performance. CDN managed to reduce the maximum to the average delay about 67,6%. From the studied CDN algorithm, Geo DNS experienced the lowest delay up to about 67.6 ms, however LC experienced the lowest packet loss, namely about 2,84%. The application of rateless code on CDN with LC algorithm succeeded in reducing packet loss to about 2.84%, but delay increased to about 75.3%.

Keywords: Analysis; CDN; LC; without RLc; Qos; NS-2; rateless Code

INTRODUCTION

On the internet network, scattered servers provide services such as website content and other data. Server can stand alone without connecting to other servers. Stand-alone servers have limitations in serving clients because they depend on the server's own resources. When the number of clients exceeds the server's ability to serve, the server is overloaded and the quality of service can decrease (Suryanto et al., 2018). One solution to improve server quality is the use of more than one server that can interact to share service tasks, and can be distributed in different geographies. The technology that connects these servers is one of them Content Delivery Network (CDN)(Mai et al., 2014).

CDN technology used more than one server (multi server) distributed in different geographies, serving many users using certain techniques and algorithms. One of the techniques is used to place a load balancer as a network entry point (Lewin, 2006). Clients that need services, the search engine google.com for example, will contact a certain server that functions as a load balancer. This load balancer server will calculate using a certain algorithm, which server is the most efficient in providing services to clients. The client taht gets an answer about which server will serve, will seamlessly continue the request to the destination server (Zheng & Boyce, 2001). Through the use of CDN technology, the distribution of server services will be more efficient and the client will be closer to the server.

The load balancer uses a certain algorithm to find which server is more efficient in serving the client. Some of the algorithms used in load balancers such as round robin (RR) (Mahajan et al., 2013), weighted round robin (WRR) (Aribowo, 2016), least connection (LC) (Yu et al., 2012) dan Geo DNS (Howley, 2009). A round robin decides which servers to serve based on a turn, while a weighted round robin allows one server to experience more than one turn. While the least connection decides the client that handles the client that has fewer connections or clients. Geo DNS decides which server to serve based on geographical proximity.

One type of data that flows on the internet network is video which has experienced significant development in video streaming applications. One of the video streaming data security techniques that can prevent packet loss effects is rateless code. Rateless code is a fountain code, which reproduces the packets to be sent, but only requires a portion of the packet to be decoded. So that the loss of some packets has no effect on the quality of the data sent. One application of the use of rateless code in video transmission is shown in Al-Akaidi (2012).

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 4, October 2022
DOI : https://doi.org/10.33395/sinkron.v7i4.11651
e-ISSN : 2541-2019
p-ISSN : 2541-044X

This study was examined the use of algorithms RR, WRR, LC, and Geo DNS at network CDN through a simulation method. The applications studied are the use of CDN for streaming video. To improve reception quality, this researcher integrates rateless code at streaming video application through network CDN (Triukose et al., 2011).

LITERATURE REVIEW

2.1. Content Delivery Network (CDN)

CDN is an interconnected system on the internet that provides internet application content to multiple clients by duplicating content on multiple servers and redirecting content to users (Madi, 2012). CDN is used by internet service providers (ISPs) to deliver static or dynamic web pages. This technology is also suitable for audio streaming applications, video, and internet television (IPTV). Picture 1. showed the configuration CDN showed the configuration CDN is (هدش ه ت اقم تياس ديدج عجرم تعم ت ن زا *Content Delivery Networks : State of the Art* , n.d.):

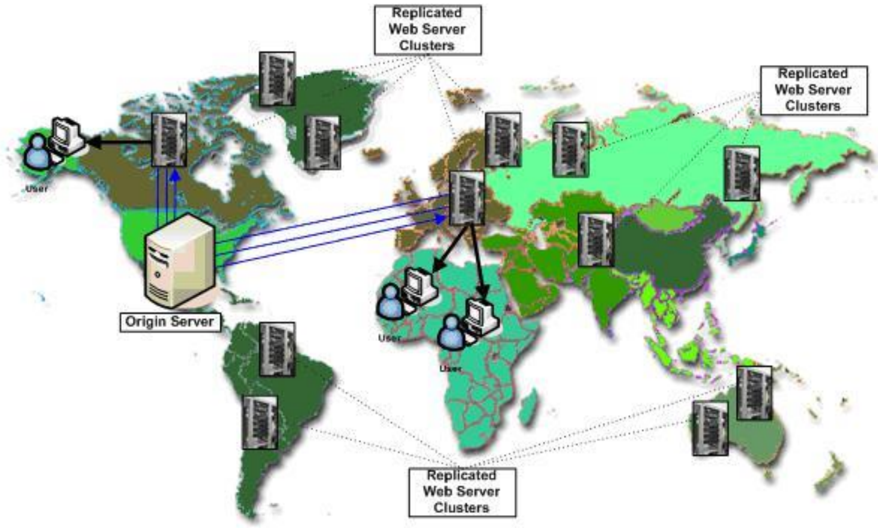

Figure 1. CDN

2.2. Using technique CDN

There are several realization techniques CDN, including the use of load balancers and IP forwarding.

2.2.1. Load balancer

Load balancer is a technique for distributing traffic loads on two or more internet connection lines so that traffic can run optimally, throughput maximum, reduce response time and avoid overload on any of the connection lines (Wirawan, 2011). Load balancer is also distribute workloads evenly across two or more computers, CPU network link, hard drive, or other resources, to get optimal use of resources (Mathew et al., 2012).

2.2.2. IP forwarding

Differential with load balancer, connection CDN with IP forwarding allow clients to contact any server on the network CDN. Then server then analyze the condition of the client and resend the request to the client more efficiently. This study does not use IP forwarding (Usep Taufiq Hidayat et al., 2013).

2.3. Server Selection Algorithm

There are Server Selection Algorithm applied in the task of maximizing CDN performance, namely using Geo DNS, RR, WRR and LC.

2.3.1. Geographic Domain Name Server (Geo DNS)

Geo DNS is a technique of dividing or mapping internet users by region. CDN servers are placed in each area that has been mapped according to the mapping and each client that performs a regulation will be served by the server closest to the client.

2.3.2. Round Robin (RR)

RR is a scheduling algorithm that divides the server load evenly according to the number of connections or response time.

2.3.3. Weighted Round Robin (WRR)

WRR is a scheduling algorithm that distributes all server loads with different processing capacities. Each server can be assigned an integer weight indicating the processing capacity, where the initial weight is one.

2.3.4. Least Connection (LC)

LC is a scheduling algorithm that will direct active server network connections with the least number of server load connections.

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 4, October 2022
DOI : https://doi.org/10.33395/sinkron.v7i4.11651
e-ISSN : 2541-2019
p-ISSN : 2541-044X

2.4. Rateless Code

Rateless code is a coding process that duplicates the original packet, becoming more. Because during the transmission process, packets may be lost, the duplicated packet can be used as a substitute. Loss of packets can be caused by damage, delays or non-compliance with the terms of service (QoS), resulting in packets being discarded (F et al., 2006). The basic purpose of rateless code is to maintain traffic quality even if the server condition deteriorates and repair any data packets lost during the transmission process. Rateless code consist of encoder dan decoder. Encoder duplicates the original packet, while the decoder replaces the lost packet with the duplicated packet. This research uses rateless code Luby Transform (LT).

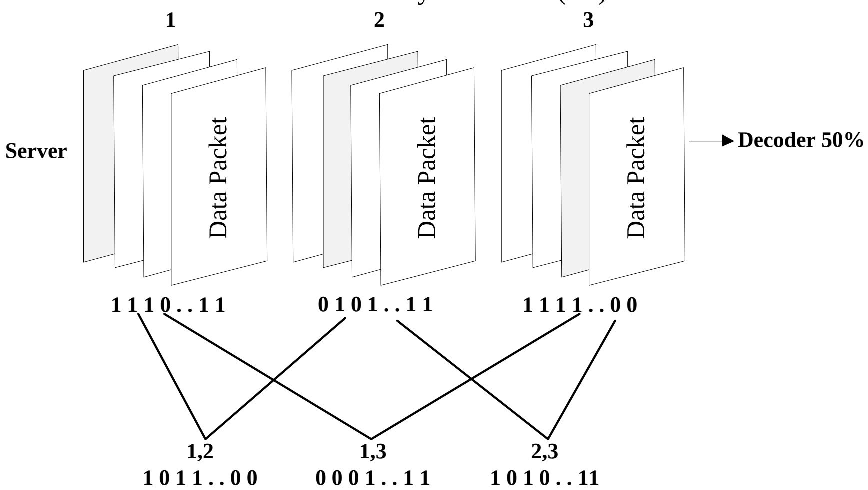

Figure 2. Encoder Rateless Code

Figure 2 encoder rateless code shows the process of adding three redundancy packets from the original three data packets: 1110..11, 0101..11 and 1111..00. Combination packet 1 and 2 to be 1011..00. Generating redundancy using the XOR operation as shown in Table 1.

Table 1. XOR

A	B	XOR
0	0	0
0	1	1
1	0	1
1	1	0

The six data packages: 1110..11, 0101..11, 1111..00, and redundant : 1011..00, 0001..11, 1010..11 sent to the recipient. The receiver needs a minimum of three packets out of six packets to receive data correctly. Because of the original data packet is 3 pieces and the data packet redundancy is 3, the rateless code process uses generation of 100%. Implementation rateless code at CDN. This research was conducted by implementing an encoder on the server and a decoder on the client. On the server side, the rateless code encoding is done with a doubling rate 50% and 100%.

METHOD

The method used in this research is as follows:

3.1. Network Simulator-2 (NS-2)

NS-2 is a network simulation software that is widely used in studying the dynamic structure of networks. NS-2 able to simulate wired networks and wireless networks and the protocols include routing algorithms, communication protocols and others (Wahyuni & Santoso, n.d.). NS-2 using two types of programming languages, C++ dan TCL. C++ was used as simulation process core, while TCL for network configuration. NS-2 is open source under the GPL (Gnu Public License) can be used on Windows operating systems and Linux operating systems. TclCL and OTcl is a component TCL with the function to bridge the network configuration with the simulation process. NS-2 executed via the command line execution command. The simulation results are in the form of notes or traces that can be used by Network Animator (NAM) or plot grafik Xgraph.

3.2. Evalvid (Evaluasi Video)

Simulator NS-2 display data presentation using Xgraph. However, Xgraph loses details of data submission events and only displays average data for the parameter under review. Therefore it showed the parameter evaluated, this study uses evalvid. Evalvid is framework and tool set for evaluating the quality of the video sent

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 4, October 2022
DOI : https://doi.org/10.33395/sinkron.v7i4.11651
e-ISSN : 2541-2019
p-ISSN : 2541-044X

over a real or simulated communication network. The main components of the structure of Evalvid are described as follows:

1) Source : The source of the video can be raw file YUV with resolution Quarter Common Intermediate Format (QCIF, 176 x 144) or in Common Intermediate Format (CIF, 352 x 288).
2) Video Encoder dan Decoder : Evalvid supports two codecs MPEG4, such as codec NCTU and ffmpeg.
3) VS (Video Sender) : Component VS reads the compressed video file from the encoder output, fragments each large video frame into smaller segments and then sends these segments via UDP packets on a real or simulated network. For every package delivery UDP, framework record the timestamp, packet id and packet size in sender trace file with the help of tcp dump or win dump, if network is real link. Yet , if the network is simulated, a sender trace file is provided by the sending entity. The VS component also generates a video trace file that contains information about each frame in the real video file. The video trace file and sender trace file are then used for subsequent video quality evaluations (Zheng & Boyce, 2001).
4) ET (Evaluate Trace) : Evaluation takes place on the sender side. Therefore, the time stamp information, packet id, and packet size received at the receiver must be sent back to the sender. Based on original encoded video files, trace video files, file sender trace and file received trace, ET component generates packet loss report, jitter as well as reconstructed video files to see the video results on the receiving side are damaged or not.
5) FV (Fix Video): digital video quality assessment is done frame by frame. Therefore, the total number of video frames on the receiving side, including the incorrect ones, must be the same as the original video on the sending side.
6) PSNR (Peak Signal Noise Ratio) : PSNR is one of the objects to assess application QoS on video transmission.
7) MOS (Mean Opinion Score) : a subjective measure of digital video quality in applications.

3.3. Network Configuration

Network Configuration CDN is simulated randomly selected and generates an NS2 model as shown at figure 7, where CDN modeled consists of 3 servers and 19 clients. Client perform service requests there are 10 clients while 9 other clients generate background traffic. Network configuration CDN with GeoDNS technique is showed at figure 3.

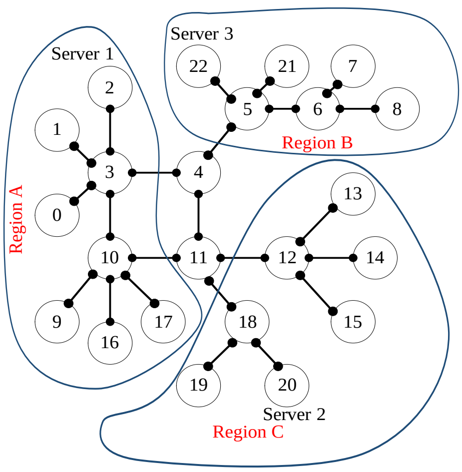

Figure 3. Configuring CDN with GeoDNS Technique

CDN evaluation using the RR, WRR and LC algorithms is done by adding an imaginary node to NS-2.

3.4. Evaluated traffic

The video traffic specifications used in the simulation are shown in Table 2, where the video speed has bit rate : 286435,406 bps.

Table 2. Video Traffic Specification

Parameter	Note
Name *video*	akiyo_cif.yuv
Frame rate	30fps
Frame type	IPP
Codec	MPEG4
bit rate (bps)	286435 bps

The video has a frame rate of 30 frames per second with the IPP framed mpeg4 codec.

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 4, October 2022
DOI : https://doi.org/10.33395/sinkron.v7i4.11651
e-ISSN : 2541-2019
p-ISSN : 2541-044X

3.5. Evaluated method

Evaluated method is GeoDNS, RR, WRR and LC and application of *rateless code* on the best method.

3.6. Parameters evaluated

Parameters evaluated is variable measuring the achievement of improvements in the SDN network using simulation NS-2.

The transmission network is in the form of a simulation line as a component of the cable network which is represented by parameters bandwith and delay. To find out how far the network performance. Then the parameter evaluation delay, packet loss, and jitter.

3.6.1. Delay is the time delay caused by the process of sending data packets to the intended recipient. The equation shows how to find the value of delay;

$$\text{Delay} = \frac{(\text{Receive package time} - \text{Package delivery time})}{\text{Total}}$$

3.6.2. Packet loss

Packet loss is the number of data packets lost during the transmission process. One of the causes of packet loss is the queue that exceeds the buffer capacity at each node. Some of the causes of packet loss are:

1. Congestion, caused by the occurrence of excessive queues in the network
2. Node that works beyond the buffer capacity
3. Memory which is limited to nodes
4. Policing or control over the network to ensure that the amount of traffic flowing is in accordance with the magnitude bandwidth. If the amount of traffic flowing in the network exceeds the existing bandwidth capacities then policing control will get rid of the excess traffic. The formula for finding packet loss is as follow;

$$\text{Packet Loss} = \frac{(\text{Number of packets sent} - \text{Number of packets received})}{\text{Number of packets sent}} x100\%$$

3.6.3. Jitter

Jitter is the variation of the delay affected by the background traffic minus the adjacent frame in the transmission of data packets on CDN. If the jitter is large but the delay is small then the network performance cannot be said to be bad because the amount of jitter can be compensated with a small delay value. Jitter will reduce network performance when the value is large and the delay value is also large.

RESULT

4.1. Design of Research

Design of research is a framework, shape, description, arrangement / structured arrangement used as a diagram to be able to implement CDN design using NS-2. The purpose of the design of research is to find out in detail the flow of the research methodology and it is designed like a diagram structure that can be seen in figure below:

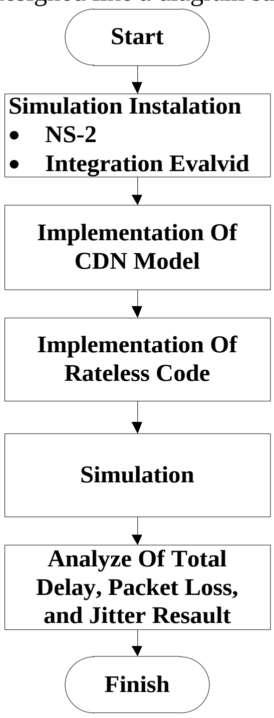

Figure 4. Design of Research

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 4, October 2022
DOI : https://doi.org/10.33395/sinkron.v7i4.11651
e-ISSN : 2541-2019
p-ISSN : 2541-044X

This research is started from NS-2 it will be integration with Evalution Video (EvalVid) to generate video traffic. Then the implementation of the model on the CDN will be carried out. Implementation rateless code to node-node NS-2 simulation. At the same time, contributions in the form of changes/modifications are made. Then the simulation is carried out with video transmission data and the implementation of the calculation of delay, packet loss, and jitter along with the results of the analysis below.

4.2. CDN Design Using NS-2 Simulation

CDN design consists of node 0 to node 22. Therefore, we see the picture below:

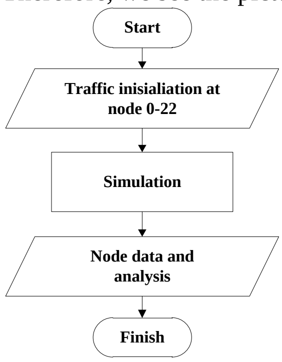

Figure 5. CDN Design Using NS-2 Simulation

After carrying out the configuration process, the next step is to perform a simulation. Simulation sequence is showed at figure 6.

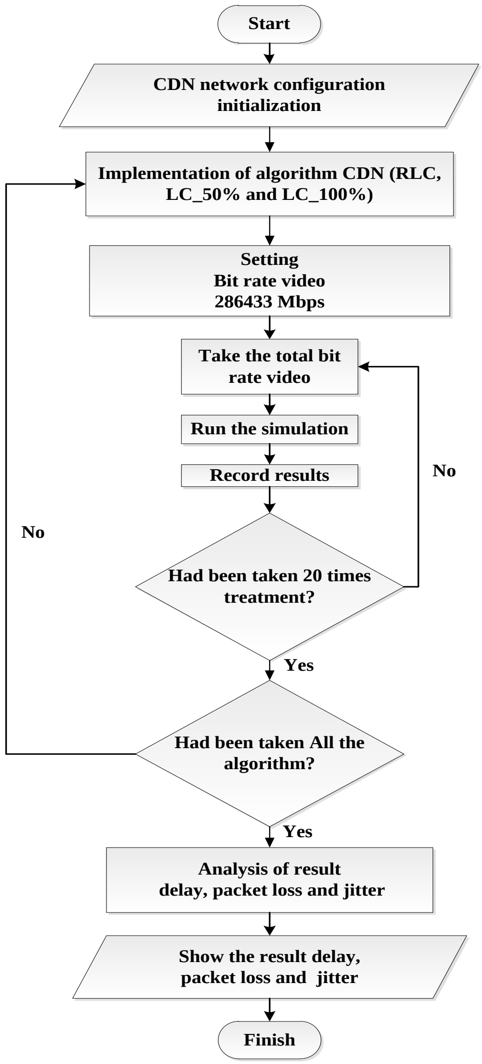

Figure 6. Flowchart of simulation implementation

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 4, October 2022
DOI : https://doi.org/10.33395/sinkron.v7i4.11651
e-ISSN : 2541-2019
p-ISSN : 2541-044X

At figure 6 it is showed the flow diagram of the simulation implementation, where the first step is to initialize the CDN configuration, after the CDN is configured, the network is implemented to the CDN so it can be simulated.

Before simulating a CDN, first make settings on the video bit rate, after that it will carry out a simulation process with the application of the algorithm. Then the simulation is repeated 20 times, after the simulation is complete, an output will be generated in the simulation process in the format .txt recorded all events during the simulation. In this simulation, CDN applies the algorithm was Rateless Code (RLC). The algorithm was carried out up to 20 trials.

DISCUSSIONS

Content Delivery Network (CDN) evaluated is the algorithm Geo DNS, LC, RR and WRR. To determine the extent of the influence of CDN on service performance, the data is compared with the network without load balancer (non CDN). At non CDN is the client chooses 1 of 3 servers randomly.

4.1. Delay

The Result of Network Simulator (NS-2) with 20 repetitions shows that the delay value does not very much in each experiment. Just the order of the 1st and 12th experiments for non CDN and the order of the 1st and 11th experiments with slightly increased RR. It is showed at table 4.1. Non CDN, video services experience higher delays than CDN. On average, non-CDN experienced delays ranging from 75,79%, 73,01%, 52,22% and 68,7% higher than DNS network with algorithm Geo DNS, LC, RR and WRR.

Table 3. Delay Characteristics

Experiment procedure	Delay (ms)				
	Non CDN	Geo DNS	LC	RR	WRR
1	0,2643734	0,06663	0,0746518	0,1308019	0,0867261
2	0,2803317	0,067416	0,0753887	0,1332597	0,0873026
3	0,280363	0,067557	0,0754475	0,1329682	0,0873266
4	0,2801863	0,067609	0,0754226	0,1324716	0,087509
5	0,2803377	0,067526	0,0754603	0,1312684	0,087462
6	0,279745	0,067829	0,0755161	0,1343835	0,0873571
7	0,2796914	0,067589	0,0754689	0,1344546	0,0875764
8	0,2803081	0,067721	0,0753665	0,1341323	0,0872529
9	0,2803715	0,067806	0,0753349	0,1322128	0,0875321
10	0,280204	0,067829	0,0753718	0,1333128	0,0878038
11	0,2803726	0,06775	0,0753445	0,1313691	0,0875302
12	0,2804693	0,06741	0,0753225	0,1258215	0,087681
13	0,2792903	0,067534	0,0749631	0,1352767	0,0873321
14	0,2803713	0,067584	0,0753554	0,1358861	0,0871902
15	0,2802813	0,06768	0,0753685	0,133849	0,0872989
16	0,2799121	0,067609	0,0753075	0,1332214	0,0874576
17	0,2802995	0,067695	0,0754791	0,1324805	0,0873271
18	0,2798484	0,06777	0,0753961	0,1326009	0,0875185
19	0,2796439	0,067558	0,0754741	0,1346084	0,0876176
20	0,279529	0,068076	0,0760313	0,147345	0,0882532
Average	0,2792965	0,067609	0,0753736	0,1335862	0,0874527

Of the four algorithms CDN, Geo DNS consistently managed to achieve the lowest delay, an average of 67.6 ms, followed by LC, WRR and RR.

4.2. Packet Loss

The simulation results show that the packet loss rate only varies at certain points in each experimental sequence. It is shown in Table 4.2. Non CDN, lvideo services experience very high packet loss on average 44.19%.

Table 4. Packet Loss Characteristics

Experiment procedure	Packet Loss				
	Non CDN	Geo DNS	LC	RR	WRR
1	42,72%	3,31%	2,25%	12,54%	14,68%
2	43,95%	3,99%	2,48%	13,53%	13,72%

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 4, October 2022
DOI : https://doi.org/10.33395/sinkron.v7i4.11651
e-ISSN : 2541-2019
p-ISSN : 2541-044X

3	43,89%	3,97%	2,52%	13,26%	13,50%
4	43,88%	3,62%	2,42%	13,26%	13,62%
5	43,79%	3,88%	2,40%	13,40%	13,70%
6	43,92%	3,47%	2,58%	13,51%	13,78%
7	43,95%	3,72%	2,30%	13,58%	14,07%
8	43,76%	3,83%	2,50%	13,52%	13,85%
9	45,28%	3,82%	2,56%	13,58%	13,81%
10	43,95%	3,73%	2,48%	13,26%	13,79%
11	43,82%	3,89%	2,54%	12,98%	13,68%
12	43,93%	3,79%	2,49%	12,51%	13,53%
13	43,95%	3,78%	2,51%	13,71%	13,72%
14	43,85%	3,56%	2,40%	13,55%	13,91%
15	43,92%	3,74%	2,31%	13,31%	13,84%
16	43,84%	3,97%	2,45%	13,55%	13,52%
17	43,79%	3,72%	2,49%	13,42%	13,76%
18	44,01%	3,54%	2,32%	13,28%	13,49%
19	44,10%	3,66%	2,56%	13,49%	13,70%
20	49,56%	11,42%	10,25%	20,16%	20,14%
Average	44,19%	4,12%	2,84%	13,67%	14,09%

LC experienced the lowest packet loss of 2.84% followed by Geo DNS 4,12%, RR 13,67% and WRR 14,09%.

4.3. Jitter

Comparison of simulation results from non CDN and CDN for jitter parameters can be seen in Table 4. From Table 5, it showed that added configuration CDN cause of Jitter to be up. On average, RR has the highest jitter reaching about 9,72 ms.

Table 5. Jitter Characteristics

Experiment procedure	**Jitter**				
	Non CDN	**Geo DNS**	**LC**	**RR**	**WRR**
1	0,007957	0,008	0,0085	0,00971	0,008942
2	0,007382	0,008	0,00857	0,00985	0,009074
3	0,007581	0,008	0,00854	0,00979	0,008986
4	0,00763	0,008	0,00854	0,00962	0,008933
5	0,007506	0,008	0,00856	0,00971	0,009008
6	0,007458	0,008	0,00853	0,00979	0,008966
7	0,007544	0,008	0,00853	0,00974	0,009072
8	0,007523	0,008	0,00852	0,009709	0,008975
9	0,007367	0,008	0,00857	0,009703	0,00897
10	0,007627	0,008	0,00853	0,009698	0,009157
11	0,007623	0,008	0,00852	0,009764	0,009019
12	0,007786	0,008	0,00852	0,009754	0,008986
13	0,007545	0,008	0,00857	0,009721	0,009008
14	0,007189	0,008	0,00855	0,009723	0,008983
15	0,00759	0,008	0,00856	0,009637	0,009013
16	0,007426	0,008	0,00856	0,009653	0,008993
17	0,007505	0,008	0,00854	0,00972	0,00896
18	0,008076	0,008	0,00855	0,009756	0,00904
19	0,007509	0,008	0,00853	0,009735	0,009024
20	0,009684	0,006	0,00857	0,0098	0,009119
Average	0,008175	0,081	0,00854	0,009729	0,009001

To see the movement of jitter changes in each experimental sequence can be seen in Figure 7 is showed that the changes to the non CDN and the three load balancer algorithms are stable. However, the graph shows that the

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 4, October 2022
DOI : https://doi.org/10.33395/sinkron.v7i4.11651
e-ISSN : 2541-2019
p-ISSN : 2541-044X

magnitude of the jitter value of non CDN is much smaller than that of CDN (algorithm Geo DNS, LC, RR and WRR).

From the four algorithms CDN, choosen the best algorithms to implemented using a rateless code. The algorithm chosen is LC, with the reason that the packet loss is the lowest, while the delay, as is the case with Geo DNS, is still below the maximum allowable limit.

4.2. CDN Evaluation with LC Algorithm and Using Rateless Code

Rateless code used to suppress the number packet loss. Reduction packet loss needed, because a good video is by number packet loss minimum. The application of rateless code to send packets is applied with rate redundancy 50% and 100%.

4.2.1. Delay

The implementation of rateless code reproduces the packets sent by the server to the client. This causes an increase in the queue at each node and travel time on each transmission link. The increase in delay is a consequence of adding a rateless code. It shows at table 6 below:

Table 6. Result of Delay Characteristics

Delay			
Experiment procedure	**LC_NonRLC**	**LC_50%**	**LC_100%**
1	0,074425002	0,0799221	0,119520985
2	0,075661613	0,0646623	0,119396055
3	0,0755778	0,0814328	0,119115446
4	0,075573061	0,1082269	0,118895906
5	0,075551256	0,1647667	0,118039493
6	0,075624694	0,0935092	0,119591963
7	0,075523102	0,0822633	0,11814583
8	0,075563876	0,0834391	0,119350452
9	0,075611863	0,1152908	0,119457963
10	0,075826697	0,0675192	0,117212001
11	0,07553849	0,0725486	0,120712676
12	0,075608083	0,0916005	0,117365361
13	0,075639934	0,1657443	0,120599194
14	0,075740814	0,0581884	0,119352356
15	0,075711411	0,0878937	0,117865889
16	0,075659659	0,0642421	0,120236424
17	0,075638154	0,1160024	0,118242723
18	0,075597914	0,0653993	0,120044549
19	0,075702509	0,082056	0,119932802
20	0,076186902	0,0664511	0,116902724
Average	**0,075598142**	**0,09055794**	**0,11899904**

Table 6. shows the fluctuation of delay in each experiment. Without rateless code, experience video pack delay average about 75,6 ms, rise to about 90,6 ms when added rateless code with redundancy 50%. Addition ratelss code with redundancy 100% made up delay to be about 119 ms. Figure 4.8 shows clearly the change in delay due to the addition of rateless code.

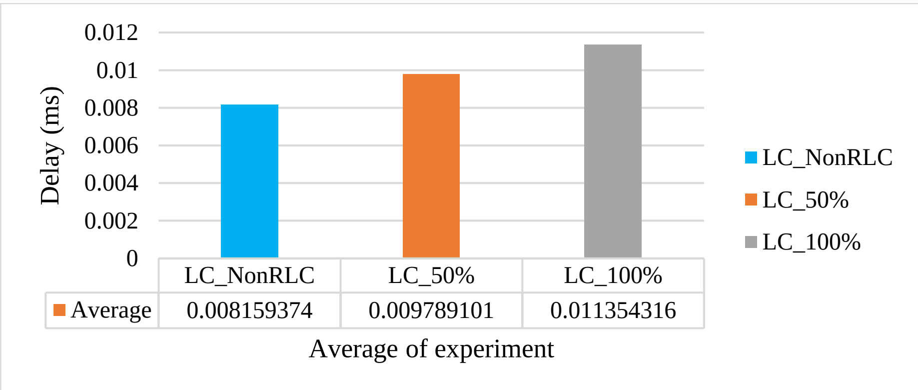

Figure 7. Comparison of Delay Characteristics

*name of corresponding author

SinkrOn
JURNAL & PENELITIAN TEKNIK INFORMATIKA
Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 4, October 2022
DOI : https://doi.org/10.33395/sinkron.v7i4.11651
e-ISSN : 2541-2019
p-ISSN : 2541-044X

4.2.2. Packet Loss

Despite the increased delay, rateless code managed to reduce packet loss. It is shown at picture 8. Data fluctuations occur on several fields of the experimental sequence. But in general, the increase in the amount of redundancy causes the delay to increase.

Table 7. Result of Packet Loss Characteristics

Packet Loss			
Experiment procedure	**LC_NonRLC**	**LC_50%**	**LC_100%**
1	6,70%	3,70%	4%
2	7,00%	3,90%	5,50%
3	7,20%	3,90%	5,00%
4	7,10%	3,90%	5,50%
5	7,10%	4,20%	4,90%
6	7,00%	3,90%	5,50%
7	7,10%	3,90%	5,40%
8	7,20%	3,70%	5,60%
9	7,10%	3,90%	5,50%
10	7,00%	3,90%	5,70%
11	7,30%	4,00%	5,50%
12	6,90%	3,90%	5,00%
13	7,10%	3,90%	5,10%
14	6,80%	4,00%	5,10%
15	6,90%	3,80%	5,60%
16	7,10%	3,70%	5,60%
17	7,20%	4,00%	5,30%
18	7,10%	4,20%	5,30%
19	6,90%	4,20%	5,10%
20	8,70%	3,90%	4,80%
Average	7,10%	3,90%	5,30%

Picture 8 shows the effect of applying rateless code on average on CDN networks. Redundancy 50% managed to reduce packet loss algorithm RLC at CDN of about 7,1% to be 3,9%. Meanwhile, the increase in redundancy that is too high results in insignificant packet loss improvements. In this case redundancy 100% causes packet loss reduction only up to 5,3%.

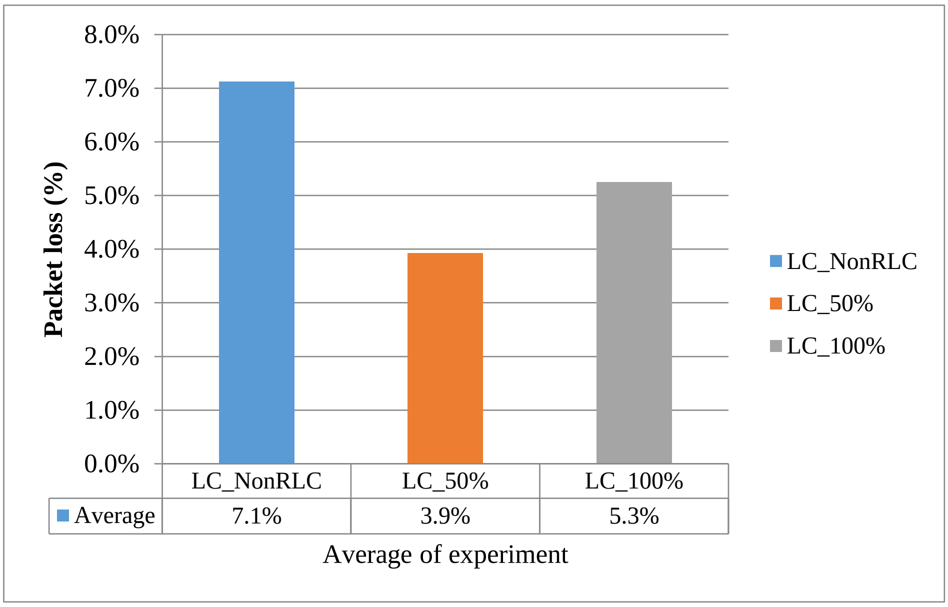

Figure 8. Comparison of Packet Loss Characteristics

4.2.3. Jitter

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 4, October 2022
DOI : https://doi.org/10.33395/sinkron.v7i4.11651
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Experiment test results applying LC 50% and LC 100% at CDN, to get the jitter value by taking the download value and bit rate of 279,72 kbps by done 20 times experiment. After the jitter test as shown in table 8, the jitter test, rateless code on the CDN.

Table 8. Jitter Test Results.

Jitter			
Experiment procedure	**LC_NonRLC**	**LC_50%**	**LC_100%**
1	0,0080796	0,0097932	0,011222
2	0,0081376	0,0097069	0,0112399
3	0,0081478	0,0098153	0,0114692
4	0,0081589	0,0096935	0,0112619
5	0,0082043	0,009832	0,0114878
6	0,0081531	0,0098607	0,0113681
7	0,0081614	0,0097925	0,0113035
8	0,0081709	0,0098398	0,0114831
9	0,008143	0,0097199	0,0112593
10	0,008138	0,0097902	0,0114507
11	0,0081399	0,0098382	0,0113171
12	0,0081748	0,0097001	0,0112596
13	0,0081672	0,0098079	0,0114457
14	0,0081405	0,0098077	0,0112332
15	0,0081472	0,009823	0,011474
16	0,0081498	0,0097165	0,0113922
17	0,0081368	0,0097459	0,0112495
18	0,008175	0,0097888	0,011445
19	0,0081405	0,009875	0,0111568
20	0,0082517	0,0098066	0,011446
Average	0,0082	0,0098	0,0113

At table 8 showed that when rateless code applied at CDN, packages have increased in value jitter. For example, for the application of LC without RLC in the order of the 1st experiment, namely about 0,0082 ms, LC 50% in the order of the 2nd experiment, namely about 0,0098 ms and then LC 100% increased value jitter is about 0,0113 ms.

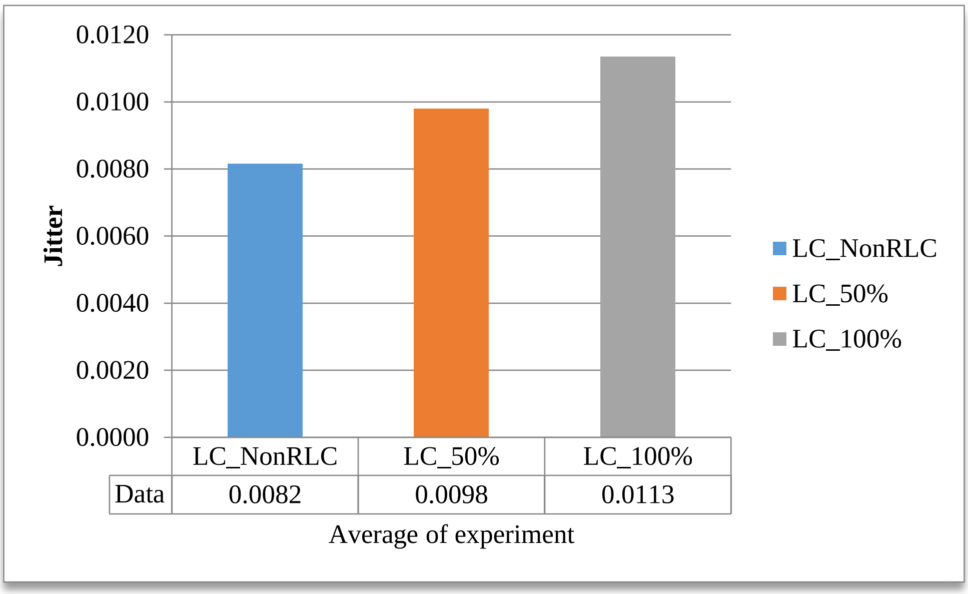

Figure 9. Comparative Characteristics Jitter

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 4, October 2022 e-ISSN : 2541-2019
DOI : https://doi.org/10.33395/sinkron.v7i4.11651 p-ISSN : 2541-044X

CONCLUSION

From this research, the following conclusions can be drawn :

1) Using CDN with load balancer with various algorithms that have the potential to improve server service performance. From this simulation, CDN managed to reduce the maximum to the average delay about 67,6%.
2) From algorithm CDN studied, GeoDNS experience the lowest delay until about 67,6 ms, however LC experienced the lowest packet loss, namely about 2,84%
3) application rateless code at CDN by algorithm LC managed to reduce packet loss to about 2,84%, but the delay increases to about 75,3%.
4) Redundancy rateless code can reduce performance. It is shown that redundancy 100% only give lift loss about 5,10%, higher than redundancy 50% who managed to suppress the loss up to about 3,49%.
5) In general, CDN managed to improve network performance by lowering delay and packet loss. While the rateless code managed to reduce the loss on the CDN even further.

REFERENCES

Aribowo, D. (2016). Cluster Server IPTV dengan Penjadwalan Algoritma Round Robin. *Setrum : Sistem Kendali-Tenaga-Elektronika-Telekomunikasi-Komputer*, *1*(2), 74. https://doi.org/10.36055/setrum.v1i2.481

F, E. P., Codes, R., & Fountain, F. L. (2006). *STREAMING OF SCALABLE VIDEO FROM MULTIPLE SERVERS USING RATELESS CODES Jean-Paul Wagner , Jacob Chakareski and Pascal Frossard erale de Lausanne (EPFL) Signal Processing Institute - LTS4*. *July*, 1501–1504.

Lewin, E. (2006). *(12) United States Patent*. *1*(12).

Madi, M. (2012). Replica Creation Algorithm for Data Grids. *Vasa*. http://medcontent.metapress.com/index/A65RM03P4874243N.pdf%5Cnhttp://etd.uum.edu.my/3352/

Mahajan, K., Makroo, A., & Dahiya, D. (2013). Round robin with server affinity: A VM load balancing algorithm for cloud based infrastructure. *Journal of Information Processing Systems*, *9*(3), 379–394. https://doi.org/10.3745/JIPS.2013.9.3.379

Mai, T., Anh, T., Huu, H. P., Nguyen, T., Shin, S., Kim, T. H., & Kim, C. S. (2014). *m Onl ad in e b e V y er th si is on fil O e i n s I ly . Bo ok m Onl ad in e b e V y er th si is on fil O e i n*. *9*(3), 317–330.

Mathew, V., Sitaraman, R. K., & Shenoy, P. (2012). Energy-aware load balancing in content delivery networks. *Proceedings - IEEE INFOCOM*, 954–962. https://doi.org/10.1109/INFCOM.2012.6195846

Suryanto, Prasetyo, T., & Hikmah, N. (2018). Implementasi Load Balancing Menggunakan Metode Per Connection Classifier (PCC) Dengan Failover Berbasis Mikrotik Router. *Seminar Nasional Inovasi Dan Tren (SNIT)*, *1*(1), A230–A238.

Triukose, S., Wen, Z., & Rabinovich, M. (2011). Measuring a commercial content delivery network. *Proceedings of the 20th International Conference on World Wide Web, WWW 2011*, 467–476. https://doi.org/10.1145/1963405.1963472

Usep Taufiq Hidayat, Rendy Munadi, Nachwan Mufti Ardiansyah, & Endang Chumaidiyah. (2013). 69-Article Text-119-1-10-20180521. *Analisa Perbandingan Perancangan Dan Kelayakan Implementasi Jaringanlte Dan Wimax Di Surabaya Pada Area Urban, Suburban, Dan Rural Denganpendekatan Techno-Economy*, *6*(1), 1–19.

Wahyuni, R. T., & Santoso, I. (n.d.). *Erbandingan Kinerja Am (Acknowledged Mode) Dan Um (Unacknowledged Mode) Pada Jaringan Umts Menggunakan Ns-2*.

Wirawan, I. M. W. (2011). Implementasi Load Balance Pada Jaringan Multihoming Menggunakan Router Dengan Metode Round Robin. *Jurnal Ilmu Komputer UNUD*, *4*(1), 15–22. https://ojs.unud.ac.id/index.php/jik/article/view/6490

Yu, M., Jiang, W., Li, H., & Stoica, I. (2012). Tradeoffs in CDN designs for throughput oriented traffic. *CoNEXT 2012 - Proceedings of the 2012 ACM Conference on Emerging Networking Experiments and Technologies*, 145–156. https://doi.org/10.1145/2413176.2413194

Zheng, H., & Boyce, J. (2001). An improved UDP protocol for video transmission over Internet-to-wireless networks. *IEEE Transactions on Multimedia*, *3*(3), 356–365. https://doi.org/10.1109/6046.944478

ا ز ن ت تعم عجرم ديدج تياس اقم ت ه هدش Content Delivery Networks : State of the Art ,. (n.d.).

*name of corresponding author