Load balancing algorithm for a local video network

S Suherman, Mahdi Aziz, Erna B. Nababan

Universitas Sumatera Utara, Medan 20155, Indonesia
suherman@usu.ac.id

Abstract. Server grows as client increases. If single server hardware up scaling has reached the limit, then multiple servers become solution. For local server requirement, the aforementioned solution has been sufficient. Load balancer distributes tasks to participating servers through designated algorithm. This paper assesses some load balancer algorithms for local video on demand network through experimental network that is developed in the lab for further research. As socket programming implemented, the WW is outperformed other algorithm in term of delay, jitter, losses as well as group of picture (GoP) delay.

1. Introduction

The development on applications, networks and the growth of users require servers within internet to be able to handle requests and bigger workloads. It is predicted that by the end of 2018 internet users reach more than 4087 Million users with multimedia as the biggest internet traffics [1].

When an internet server has high number visitors more than the maximum capacity it can cope, overload causes disturbances and service may stop [2]. Server cluster is one of solution that locates a group of servers to work at the same location to accomplish some tasks together. The common device that is employed by clustered servers is a load-balancer.

A load balancer distributes loads among the clustered servers by using a particular algorithm [3, 4]. The round robin algorithm uses a sequential time-sharing where client requests are directed to servers sequentially. Round robin allocates task similarly to servers. The drawback is that a server that has completed its task should wait its turn to have another task. This makes the servers do not work optimally. In order to optimize server capabilities, weighted round robin doubles the load for particular server. Each server is given weighted. The better the server performances, the higher the server weight values. First in first out (FIFO) lets the server retrieves client requests to load balancer dynamically. This allows server to work on its maximum performance, however low request results inefficient system. EDF (Earlier Deadline First) is an optimal pre-empted algorithm that guarantees the works completed on time. This queue discipline executes tasks based on their deadlines.

Several works have been performed to maximize the basic aforementioned algorithms. Menon [5] describes load balancing utilizing machine learning to capture application characteristics. Huang [6] combined FIFO and LIFO queues by using switched delay lines. It uses a two stages cache. Azqueta et al [7] proposes load balancer for large data by splitting tables into an appropriate number of regions and distributing them across servers. This paper examines some selected load balancer algorithms when they are employed to serve client requests for local video on demand application such as video network of an audio visual laboratory [8] or even an aeroplane cabin video network [9].
2. Methodology
In order to assess the load balancer performances; clients, load balancer and server applications are developed by using java socket programming. There are two computers employed as servers, and three computers for clients. Each client generates 5 to 10 multiple requests so that the actual clients are 15 to 30 clients. A computer is dedicated as a load balancer. Table 1 shows the specification of computers used for the experiments.

| Function         | Hardware                                      | Operating system |
|------------------|-----------------------------------------------|-------------------|
| Server 1         | Intel I3, 2.26 GHz, RAM 1 GB                   | Win 7 32 Bit      |
| Server 2         | Intel Core 2 Duo, 4M Cache, 2.40 GHz, RAM 2 GB | Win 7 32 Bit      |
| Load Balancer    | Intel I5, 6M Cache, up to 3.80 GHz, RAM 4 GB  | Win 7 32 Bit      |
| Client 1         | Intel Cel. 4 Core, 2M Cache, 2.00 GHz, RAM 2 GB| Win 8 32 Bit      |
| Client 2         | Intel I5, 6M Cache, up to 3.80 GHz, RAM 4 GB  | Win 7 32 Bit      |
| Client 3         | Intel Core 2 Duo, 4M Cache, 2.40 GHz, RAM 2 GB | Win 7 32 Bit      |

All computers are connected through a gigabit Ethernet switch by using UTP Cat 5E cables with class C IP addresses. The arrangement is shown in Figure 2.

The traffic is set to be a short video stream and the assessed load balancing algorithms are round robin (RR), weighted round robin (WRR), random and source IP Hash algorithms. Applications on clients request video files to load balancer, load balancer forwards request to servers, servers send video packets to clients. Delay and packet loss are recorded in clients. Experiments are conducted 100 times for each algorithm.

Delay, jitter and packet loss of the transmitted packets are determined as the performance parameters. Delay in the time difference between the packets received time in the clients and the packet send time in the server. However, since the timing is not synchronized, a simple time adjustment was conducted before the experiment. Jitter is calculated as neighbouring delay variation. Packet loss is measured as the percentage of loss packets out of the transmitted ones. The parameters are taken as the average value.

Round robin (RR) is implemented by allocating the tasks of transmitting video alternately. The WRR is implemented by allocating weight equals 1 for server 1 and 2 for server 2. Random algorithm uses pseudorandom to determine which server to serve. Source IP table is fixing the same IP to the same server.
3. Results and discussion

The average of video packet delay varies in each experiment. In some cases, delay reaches as low as 27 ms but sparks to more than 45 ms in other occasion. Since real time data requires at most 150 ms in order to achieving best performance; video packet requires strict requirement. As video frames are generated in form of group of pictures (GoP), delay for decoding are accumulated from some frames. Moreover a frame is often sent in some packets. Figure 2 shows the delay variations resulted by different algorithms.

![Delay variations](image)

The random decision generates the maximum delay up to 45.75 ms while WRR results the lowest maximum delay. In average, WRR produces the lowest delay about 38.85 ms followed by IP hash, RR and random. Figure 3 shows the average delay, jitter and loss comparisons of the four implemented algorithms. The WRR is consistently producing the best performance parameters: 38.85% delay, 4.58% losses and 1.85 ms jitter. This can be explained as follows: small network requires fast client response as well as fast server services. WRR is fast in selecting the serving server, as fast as RR, however the best server sends packets.
For video sequence IPPP with 30 fps and average frame I size 16KB, average frame P 3KB, packet size 1024 bytes, the video transmission delays for various GoP are shown in Table 2. Delay increases as GoP increases. In average, WRR performs best (Figure 4).

This delay illustrates how long video transmission occurs before decoding process is conducted in video player. For instance, delay 4.85 s for GoP 35 is significant and annoying. Therefore, it is important to select the appropriate load balancing to keep video transmission delay as low as possible.
Table 2. GoP transmission time (in second)

| GoP | RR | WRR | RANDOM | IP HASH |
|-----|----|-----|--------|---------|
| 5   | 1.20 | 1.17 | 1.21  | 1.19    |
| 10  | 1.80 | 1.75 | 1.82  | 1.78    |
| 15  | 2.39 | 2.33 | 2.42  | 2.37    |
| 20  | 2.99 | 2.91 | 3.03  | 2.96    |
| 25  | 3.59 | 3.50 | 3.64  | 3.56    |
| 30  | 4.19 | 4.08 | 4.24  | 4.15    |
| 35  | 4.79 | 4.66 | 4.85  | 4.74    |

Figure 4. Video transmission parameters

4. Conclusion
The paper has examined some load balancing algorithm used by a local video on demand network such as audio visual laboratory or in-flight entertainment. The assessment was in real network with java socket programming as the network protocol tools. There were two servers serving tens of requests generated by two computers. The assessment shows that WRR performs best over the other three compared algorithms in local multi-server network. It generated the lowest delay, jitter and losses. It constitutes to the lowest GoP video transmissions. WRR is fast in allocating server and choosing the best server to serve clients.

References
[1] Statistics on "Internet usage worldwide", Statistica, 2018.
[2] Ganesan, V., & MSK, M. (2017). A Secured Load Mitigation and Distribution Scheme for Securing SIP Server. Security and Communication Networks, 2017.
[3] D C Devi, V R Uthariaraj. (2016). Load balancing in cloud computing environment using improved WRR algorithm for nonpreemptive dependent tasks. The scientific world journal, 2016.
[4] Suherman, S., & Al-Akaidi, M. (2013), “Adjusting WiMAX for a Dedicated Surveillance Network”, International Journal of Electrical and Computer Engineering, 3(4), 492.
[5] Menon, H., Chandrasekar, K., & Kale, L. V. (2017, January). POSTER: Automated Load Balancer Selection Based on Application Characteristics. In Proceedings of the 22nd ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (pp. 447-448). ACM..
[6] P K Huang, C S Chang, Jay Cheng, D S Lee, "Recursive constructions of parallel FIFO and LIFO queues with switched delay lines", IEEE Transactions on Information Theory, 2007.
[7] Azqueta-Alzúaz, A., Brondino, I., Patiño-Martínez, M., & Jiménez-Peris, R. (2017). Load balancing for Key Value Data Stores. In EDBT (pp. 506-509).
[8] Al Mamun, M. (2014). Effectiveness of audio-visual aids in language teaching in tertiary level (Doctoral dissertation, BRAC University).
[9] S Galipeau, R Briski, G Adams, J Mills, and J Martin. 2008. Cabin management and entertainment system. U.S. Patent Application 11/974,086.