SERBIAN JOURNAL OF ELECTRICAL ENGINEERING
Vol. 6, No. 3, December 2009, 515 - 524

UDK: 004.738.2

Influence of Load Balancing on Quality of Real Time Data Transmission*

Nataša Maksić[1,a], Petar Knežević[2], Marija Antić[1], Aleksandra Smiljanić[1]

Abstract: The routing algorithm with load balancing presented in [1] represents the modification of OSPF protocol, which enables the optimization to achieve higher network throughput. In the case of routing with load balancing, packets belonging to the same stream use different paths in the network. This paper analyzes the influence of the difference in packet propagation times on the quality of real-time data transmission. The proposed algorithm was implemented and the simulation network was formed to measure the jitter.

Keywords: Load balancing, Loose source routing, OpenVZ, OSPF, Real-time data transmission.

1 Introduction

IP television and telephony, Internet radio, video and audio conferences, as well as other types of real-time data transmission are more and more present in IP networks. To make the transmission of these contents possible, it is necessary to provide guaranteed quality of service. The important performance metrics of real-time multimedia data transmission are bandwidth, reliability, delay and jitter.

The use of load balancing enables the more even distribution of link loads and thereby avoids bottlenecks in certain communication scenarios. The proposed routing is optimized to achieve higher guaranteed node traffic loads in comparison with OSPF. In addition to routing optimization, the application of the proposed protocol can substantially simplify and accelerate bandwidth reservation procedure. Namely, each router can keep the information about the maximum traffic it can generate, and also the information about capacity reserved for the existing sessions. When the new session request is processed,

[1]Department of Telecommunications, Faculty of Electrical Engineering, University of Belgrade, Bulevar Kralja Aleksandra 73, 11120 Belgrade, Serbia
[2]Iritel, Belgrade, Serbia
E-mails: [a]maksicn@etf.rs; [b]petar@iritel.com; [c]amarija@etf.rs; [d]aleksandra@etf.rs
*Award for the best paper presented in Section *Telecommunications*, at Conference ETRAN 2009, June 15-19, Vrnjačka Banja, Serbia.

only source and destination router need to check if they have enough available capacity. In this way, fast and decentralized bandwidth reservation process can be achieved, which is very important for multimedia applications.

In this algorithm, the traffic between two nodes is routed in two phases. In the first phase, the portions of the traffic generated by the source router are directed to balancing routers. In the second phase, the balancing routers route the traffic to the destination routers. In both of the phases standard OSPF is used as the underlying routing protocol. The portions of each flow are transmitted through the balancing routers according to balancing coefficients. The balancing coefficient is the characteristic of the node and it is independent of the packet's source and destination. The same algorithm is executed on each router and calculates the optimal values of balancing coefficients.

Balancing of the IP traffic results in different paths of the packets belonging to one stream and therefore in different packet propagation times through balancing area. The packet propagation time is essential for multimedia applications due to its influence on the stream jitter and delay. For this reason, it is important to have insight into performance metrics in the network with load balancing which provides the described advantages related to guaranteed bandwidth and capacity reservation.

This paper is organized as follows. In the second section, a brief description of our implementation is presented. This description is needed for the analysis of the performance metrics such as transmission reliability, jitter and delay. In the third section, we discuss the performance metrics of the data transmission in the network with load balancing and make the comparison with performance metrics in the network with OSPF. The fourth section describes the methods of virtualization in computer systems and presents the virtualization technique that was chosen for simulation network. The fifth and sixth sections address the influence of load balancing on the real-time stream jitter. In the fifth section, the simulation network for jitter measurement is described and in the sixth section the results of jitter measurements are presented and analyzed.

2 Short Description of the Implementation

The implementation of the routing with load balancing can be divided into two entities: module for calculating balancing coefficients and module for directing the packets to balancing routers.

The module for calculating balancing coefficients is functionally integrated in OSPF implementation and uses the shortest paths, which are provided by OSPF. The balancing coefficients determine which portions of the traffic are to be directed to the corresponding balancing routers. The balancing of the traffic is executed in OSPF area, which at the same time represents the balancing area.

For the calculation of balancing coefficients, the information about the ratio between the projected traffic inserted in the balancing area and the projected traffic terminated in the balancing area is needed. The OSPF Opaque LSAs are used for distribution of this information in accordance with RFC 2370 [3].

In the module for directing the packets to balancing routers, the load balancing is accomplished by using loose source routing, described in RFC 791 [4]. Source routing option specifies the list of routers through which packet should pass on its way to destination. This option is the part of IP header. As a part of the implementation, the application which adds the loose source routing option based on the given parameters is developed. This application sets up the contents of the loose source routing option and destination IP address in order to achieve the desired path of packet. Using netfilter/iptables software [5], all packets which enter the balancing area are directed to this application. Packets are processed by the balancing application only once, when they enter the balancing area. This application receives the balancing coefficients from the program which calculates these coefficients through a TCP connection. Packets are directed to balancing routers proportionally to balancing coefficients, i.e. one portion of the packet flow is directed to one balancing router, next portion to another router, etc. Packets are consecutively sent through one balancing router while the sum of the packets' lengths is less than the number of bytes calculated by multiplying the balancing coefficient for that router and configurable number.

In order to achieve the improved performance, the module for calculating the balancing coefficients and the module for directing the packets to the balancing routers are separated. The final goal is to move the module for directing the packets to balancing routers to the Linux kernel. That would result in the reduced execution time due to less task switching between kernel and application.

3 Real-Time Data Transmission in the Network with Load Balancing

The important performance metrics of the real-time multimedia transmission are bandwidth, reliability, delay and jitter. The guaranteed bandwidth results from the way of calculating balancing coefficients: the obtained routing scheme guarantees the traffic which a router can deliver to the balancing area or receive from the balancing area.

The reliability of the transmission is inherited from the reliability of the transmission in the existing IP networks. In the second section, it is described that the load balancing is accomplished by using the loose source routing. In that way, the existing mechanism of directing packets based on the OSPF protocol and source routing is used during the load balancing. The only new module on the packet path is the program which adds the loose source routing option in the packet header. The influence of this simple program on the transmission reliability is negligible.

The delay of the real-time signal reproduction in the network with load balancing is the largest delay on the paths used for load balancing. If the balancing was performed in the network with links of high latency, such as satellite intercontinental links, the problem of the delay would be remarkable. The presented implementation of routing with load balancing limits the balancing area to OSPF area and the latencies in the network with load balancing will also be limited.

In the case of load balancing, when switching from longer to shorter path, the change of packet order in the receiving stream can occur if the difference of the packet propagation times is bigger than the time between the generation of subsequent packets. However, this should not represent the problem. Namely, the change of packet order can also occur during the routing in the OSPF network, in the case of reconfiguration due to network topology change. Therefore, the applications should be able to handle the change of packet order.

The evaluation of the jitter can be performed based on the formulas from RFC 3550 [6]. $D(i, j)$ represents the time difference between the packets i and j at the reception side in relation to time difference at the sender side, i.e. the difference in relative generation time and time of reception for two subsequent packets.

$$D(i,j) = (R_j - R_i) - (S_j - S_i) \tag{1}$$

The relative generation times of packets S_i and S_j are noted in RTP packet headers as timestamps. The reception times of packets, R_i and R_j are measured in the application which is the packet destination. Formula (1) can be rewritten as (2), in which $D(i, j)$ is viewed as the difference in propagation times of two packets.

$$D(i,j) = (R_j - S_j) - (R_i - S_i) \tag{2}$$

Equation (3) is used to calculate the jitter. It takes into account the change of the current value of $D(i-1,i)$ in relation to the previously measured jitter value scaled by factor 1/16. Scaling with factor 1/16 softens the influence of rapid changes of values of $D(i-1,i)$ and produces a more stable measure of jitter.

$$J(i) = J(i-1) + \frac{\left(\left|D(i-1,i)\right| - J(i-1)\right)}{16}. \tag{3}$$

The different packet transmission times on the different paths result in the increase of the absolute value of $D(i-1,i)$, where $i-1$ and i represent the packets transmitted on different paths. The increase of the absolute value of $D(i-1,i)$ compared to the previous value of jitter results in the increased jitter.

In most of the cases, the value of $D(i-1,i)$ will be greater for the pair of packets transmitted on different paths than for the pair of packets transmitted on the same path, if the propagation times on those paths are different. In other words, the increase of jitter due to transmission of subsequent packets on different paths is caused by the difference in propagation times on those paths.

In the OSPF areas, which are limited by the number of routers and the distance between the routers, the difference in transmission time will not significantly affect the jitter.

4 Virtualization in Computer Systems

Virtualization in computer systems enables the execution of programs compiled for one computing environment on another computing environment. One of the important applications of virtualization is the execution of multiple operating systems on one computer. This application is gaining on importance, because it enables the better utilization of the modern computers that have strong computing capabilities.

One method of executing multiple operating systems on one computer relies on providing virtual hardware platform identical to the hardware platform for which the operating system was intended. This type of virtualization is called full virtualization. Operating system executed in this environment has impression that it is executed alone on its hardware platform and it is oblivious of the other operating systems executed on that computer. As a consequence, operating systems can be executed on such virtual machines without any modifications. The disadvantage of this method is that it requires additional processing in order to provide fully virtualized environment, which can affect certain functionalities, including networking functionality.

The improved performances are offered by paravirtualization method, in which the executed operating systems are slightly modified in a way that they are aware of the virtual environment. Such modifications can provide the optimized execution of certain functionalities. For example, networking subsy-stems can be modified so that the packets are exchanged between the executed operating systems in a more direct and efficient way, which can provide better networking performances, including reduced delay and smaller jitter.

The most efficient method for executing multiple operating systems on one computer is the operating system-level (OS-level) virtualization. In OS-level virtualization, operating systems are executed as containers formed by base operating system. This means that only one instance of kernel is executed and every additional executed operating system has its data structures and execution of the applications separate from other executed operating systems. In OS-level virtualization, the introduced overhead in execution of operating systems and communication between systems is reduced in comparison with full virtualization and paravirtualization. The disadvantage of OS-level virtualization is that all executed operating systems need to be of the same type, because only one instance of kernel is executed.

5 Simulation Network

The development of virtualization platforms enables the execution of significant number of operating systems on limited number of computers. This provides an economic and efficient way for creation of large networks which can be used for testing and evaluation of various network protocols. Additional advantage of using virtualization for simulation of router network is the possibility to execute and test the software implementation without modifications which would be necessary in the case of using one of the simulation platforms like ns-2 [7].

Because of improved performances of the OS-level virtualization, we used OpenVZ virtualization software [8]. The network simulation using Xen paravirtualization software is presented in [2]. OpenVZ is the open-source OS-level virtualization software for Linux operating system. Simulation network was formed from CentOS 5 Linux operating system [9] installed in OpenVZ containers. On each system one IP router is executed, and the resulting network is the network of IP routers which can be used for testing and evaluation of behavior of network protocols. Our simulation network consists of 22 Linux routers which are executed on 7 computers. The number of routers which can be executed on the same computer concurrently depends on the processor speed and the amount of memory installed in that computer. The computers are connected through a switch, while VLANs are used to separate traffic of the different connections between virtual containers.

The topology of the simulation network is shown in the Fig. 1. Used topology contains routers with different numbers of neighbors and different distances between them. In the Fig. 1 the dashed contours group the routers which are executed on the same computer.

In order to measure the influence of the load balancing on the jitter and delay, Linux routers were executing the implementation of routing with load balancing, described in the second section.

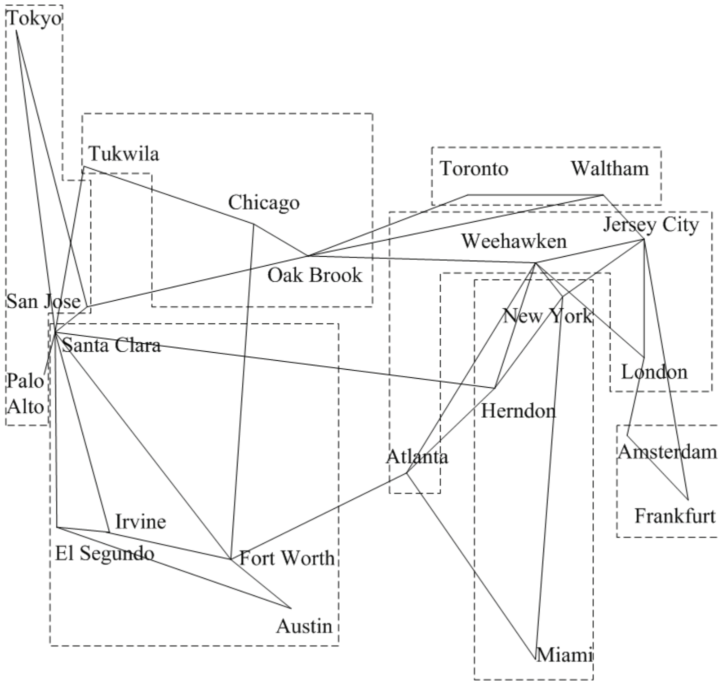

Fig. 1 – *Simulation network.*

6 Simulation Results

In described simulation network, the measurements of jitter were conducted when the RTP audio stream was transmitted between the two nodes.

Fig. 2 shows set of measurement results of RTP stream jitter. The measured jitter in the network without balancing is depicted in Fig. 2a. It can be seen that values of jitter are below 0.5 ms, which is very low. Such low results are the consequence of low overhead introduced by OpenVZ virtualization.

The measured jitter in the network with load balancing with two balancing routers is shown in Fig. 2b. The difference in propagation times for two balancing paths is smaller than the spacing between the subsequent RTP packets at the sender side. The increases of jitter in the moments of changing balancing paths are clearly visible. According to equations (1), (2) and (3) and the discussion of those equations in the third section, the value of jitter converges to difference of propagation times of subsequent packets. Since in the event of path change only one pair of packets is transmitted on different paths, the value of jitter increases only for that pair of packets and then decreases towards the value of jitter measured on the current path.

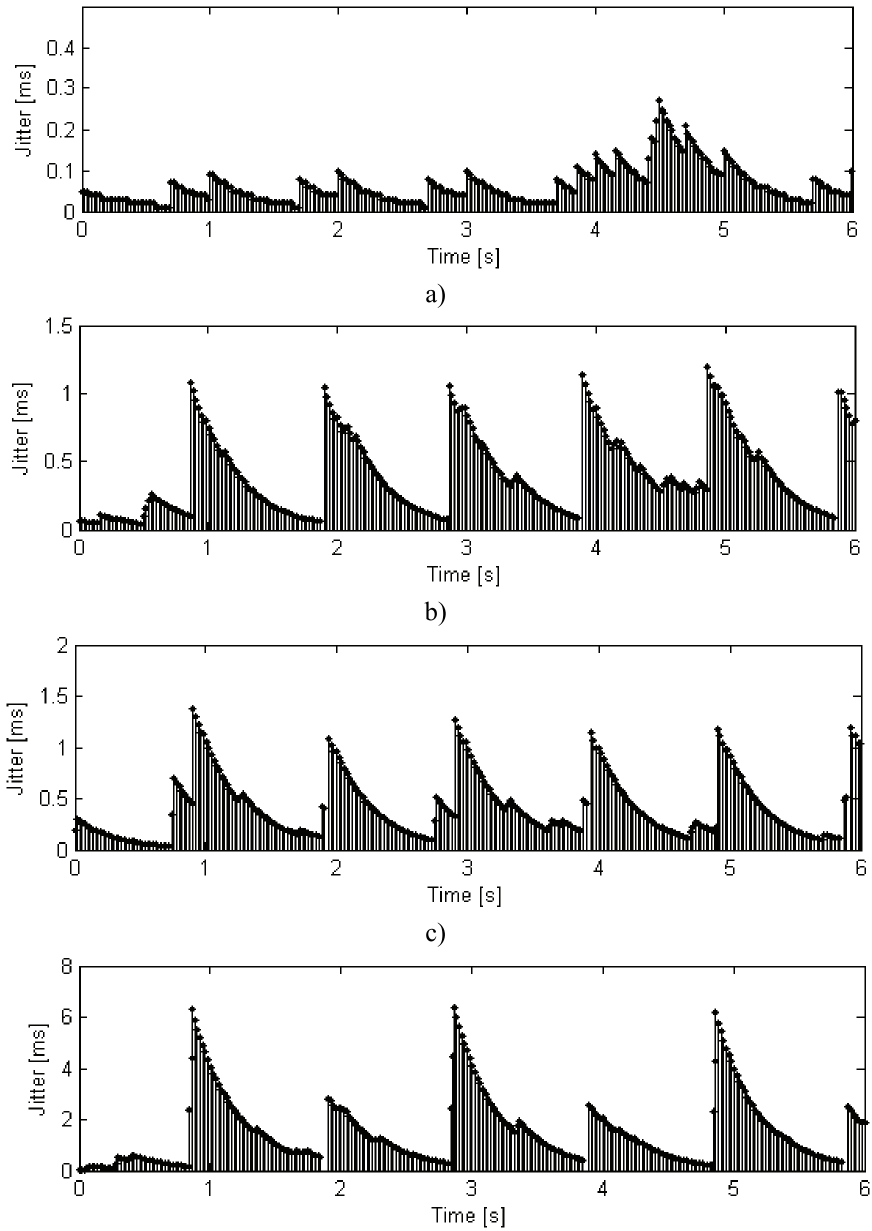

Fig. 2 – *Measured jitter:* a) *without load balancing*; b) *with two balancing routers and* 11 ms *propagation time difference*; c) *four balancing routers*; d) *two balancing routers and* 30 ms *propagation time difference*.

Fig. 2c shows the measured jitter in the network with different traffic vector, resulting in load balancing with four balancing routers. Although there are more path changes in this network compared to the network with two balancing routers, the measured jitter values in these two cases are similar. This similarity is the result of the values of balancing coefficients, link delays in the network and order of packet transmission on balancing paths.

When difference in transmission times of balancing paths is greater than spacing between RTP packets in the stream, the path change from the path with bigger delay to the path with smaller delay can lead to the change of packet order at the reception. The change of packet order results from the fact, that more than one pair of consequently received packets traveled over different paths. For each of those pairs, the value of jitter increases because it converges to the difference in propagation times on balancing paths, resulting in increased jitter values. Fig. 2d shows the measured jitter in the network with two balancing routers and propagation time difference of balancing paths of 30ms. Although the difference in propagation times is the same when changing from slower balancing path to faster and vice versa, increase of jitter in the case of switching from slower to faster path is significantly bigger than the increase of jitter in the case of switching from faster to slower path.

7 Conclusion

The routing algorithm with load balancing [1] enables achievement of higher guaranteed node traffic loads compared to standard OSPF and fast and automatic bandwidth reservation process desirable for multimedia application.

Characteristic of routing with load balancing is that packets belonging to one stream are transmitted over different paths between the two nodes. Different paths can have different packet propagation times and jitter values. In this paper we analyzed the real-time data transmission in the network with load balancing. The conducted measurement of jitter in the simulation network shows that jitter due to changing paths depends on the difference in propagation times over the different balancing paths. The higher values of the jitter occur when the difference in propagation times over different paths is higher than the time difference between subsequent packets at the source side. Since the balancing is performed inside the OSPF area, it can be expected that the difference in propagation times over different paths isn't too big. Also, the other performance metrics don't significantly differ compared to regular shortest path routing. These results show that, taking into consideration the increased throughput and possibility of simpler bandwidth reservation, the algorithm presented in [1] represents the suitable routing solution.

8 References

[1] M. Antic, A. Smiljanic: Oblivious Routing Scheme using Load Balancing over Shortest Paths, ICC 2008, Peking, May 2008, pp. 5783 – 5787.

[2] N. Maksic, P. Knezevic, M. Antic, A. Smiljanic: On the Performance of the Load Balanced Shortest Path Routing, PacRim09, Victoria, Canada, 2009.

[3] R. Coltin: The OSPF Opaque LSA Option, RFC 2370, July 1998, Online: http://www.rfc-editor.org/rfc/rfc2370.txt

[4] J. Postel: Internet Protocol, RFC 791, Sept. 1981, Online: http://www.ietf.org/rfc/rfc791.txt

[5] http://netfilter.org/

[6] H. Schulzrinne, S. Casner, R. Frederic, V. Jacobson: RTP: A Transport Protocol for Real-Time Applications, RFC 3550, July 2003,Online: http://www.ietf.org/rfc/rfc3550.txt

[7] http://www.isi.edu/nsnam/ns/

[8] http://wiki.openvz.org/

[9] http://www.centos.org/