Gigabit Ethernet Backbones with Active Loops[1]

R.García, M.P. Malumbres and J. Pons
Department of Computer Engineering (DISCA)
Technical University of Valencia, Spain
{roman, mperez , jpons}@disca.upv.es

Abstract

The current standard Ethernet switches are based on the Spanning Tree (ST) protocol. Their most important restriction is that they can not work when the topology has active loops. In fact, the ST protocol selects a tree from the real topology by blocking the links that are not involved in the tree. This restriction produces a network traffic unbalancing behavior saturating those links near the root switch while rest of links will be idle or with a very low utilization.

This paper proposes a new transparent switch protocol for Gigabit Ethernet backbones that considerably improves the performance of current ones. The proposed protocol is named ALOR for Active Loops and Optimal Routing. ALOR protocol could be used in the last stage of a fat tree network in order to allow a final backbone with active loops. So, rings, mesh and other regular/irregular active loop topologies can be used to connect the Gigabit switches in order to obtain better performance results.

1. Introduction

Routers and switches (bridges) are the basic devices for LAN interconnection. They have different properties and each one has its own scope. For example, routers forward packets using hierarchical addressing (i.e. IP addressing) but they expect that every station be configured to work with them.

Switches can be used to build a diameter-limited network, usually named " Switched LAN" or "extended LAN". The current standard switch is a transparent switch. Transparent means that the stations do not need to use special software to work with switches. The most attractive feature of transparent switches is their easy installation procedure and null maintenance.

This paper does not try to go back over the old dilemma about routers vs. switches. This paper is focused in an old restriction associated with switches, their inability to work with loops [1].

A switch is a smart hub. It learns where the stations are, so it can forward frames to their destination using the appropriate paths. The learning process is simple:

(a) Station-n transmits a frame. The frame header has the fields “destination address”, “source address”, “data” and others.
(b) Switch-j receives the frame on port-i. The switch reads the “source address” field and learns that station-n must be reached using port-i.

The learning process is, obviously, a continuous process. When a switch do not know where a destination station is, it simply sends a copy of the frame on all its ports but the one on which frame was received. Finally, any switch in the LAN learns the way to reach any active station and route frames in consequence [5].

There is an important restriction to make the learning process feasible; loops are forbidden. Why?. Because a loop implies alternative paths, so a station can be detected by multiples ports in the same switch and this confuses the learning process. And, more important, broadcast frames would be caught in the loop infinitely with no solution. Therefore, the current standard switches use ST algorithm to transform any topology into a tree.

The figure-1 (a) shows a tree with three hierarchy levels. Root-switch is in the first (top) level of the hierarchy. At the bottom there are the leaf-switches. They connect with stations. If only one technology is used (i.e. 10 Mbps Ethernet) it is clear that an excess of traffic will saturate switches near to root. In order to alleviate this problem, engineers have designed powerful switches with multiple ports (24 ports per switch, and even more, are usual) and/or use “fat tree” topologies.

Figure-1 (b) shows a typical fat tree based in Ethernet technologies. A problem in a fat tree is that the best performing technology has to be kept for the backbone LAN, and cannot be used in the rest of the LANs in order to balance traffic and bandwidth. A variant of this problem occurs when it is necessary to update the end-stations technology. For example, currently it is usual to install Fast Ethernet (100 Mbps) at the stations. So, it is

[1] This work was funded by an UPV grant (ref. 2001.0019)

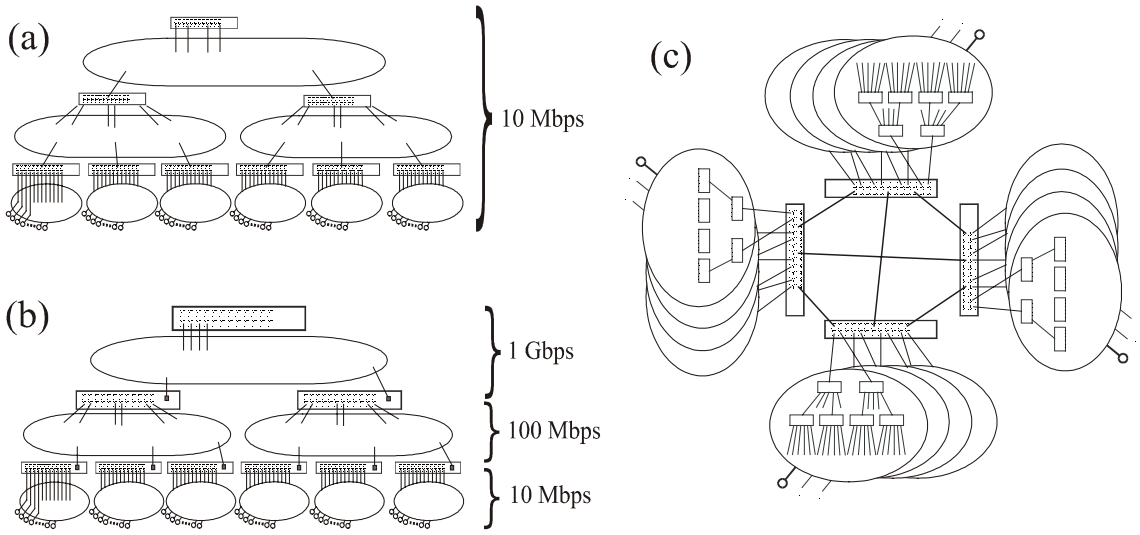

Figure 1. (a) A tree topology. All switches have the same technology (b) A fat tree topology with a collapsed backbone. (c) A fat tree with a distributed backbone running ALOR.

only possible to make a fat tree with two hierarchical levels (1Gbps on top and 100 Mbps down). Considering the maximum number of ports per switch (24 ports is the common available) it is a restriction in order to connect a great number of stations (like in a campus LAN).

Note that the problem is that the backbone with gigabit Ethernet switches is a collapsed backbone. The backbone is formed for only one gigabit switch. If we finally need to use three o more gigabit Ethernet switches as backbone (distributed backbone), we have again the problem of the tree [2].

This paper proposes a new protocol (ALOR) for gigabit switches that allow the use of active loop topologies. Therefore, strongly connected regular topologies, like meshes, as well as irregular topologies with active loops, can be used without wasting bandwidth. As loops imply alternative paths, the ALOR protocol could use optimal routes.

ALOR works on top of ST protocol. It uses information gathered by ST. It is a distributed protocol but, in this case, only for gigabit switches. Although ALOR is proposed for gigabit switches at network backbone , it can be used on all network switches [3][4].

In section 2 a detailed description of the protocol is given. In section 3 a preliminary performance evaluation was done comparing ALOR performance with respect to the ST standard. Finally, in section 4 some conclusions are drawn.

2. Active Loops and Optimal Routing protocol (ALOR)

The ALOR protocol is based on the idea that switches can learn "which stations are associated with each gigabit switch" or, in other words, where each station is. This is the main difference with respect to the ST based standard, in which the switches only learn "from which direction" the station has been listened to. Thus, the ST is not strictly required, and the most favorable route can be used considering all the existing links. The optimal route criterion is based on number of hops.

In order to completely describe the ALOR protocol, first we are going to explain how a giga-switch can identify the ports that belong to the defined giga-tree. Then we will describe how ALOR learns the location of each active station in the network and as consequence the best route to reach it. Also, an example is given to show the different stages of the learning process. And finally, some additional details are exposed about the ALOR implementation.

2.1. Determining the gigabit ports that belong to the giga-tree

A gigabit switch has all its ports working at 1 Gbps., but only some of them are involved in the giga-tree (the distributed backbone). It is necessary to define an automatic method that allows the gigabit switch to know which are those ports. ALOR protocol proposes that gigabit switches exchange ALOR configuration messages with the following format:

destination address = ALOR group (multicast address),
source address = MAC address of the source switch,
type = ALOR id. ,
data = transmission port status OR acknowledge.

The gigabit switches must accept whatever kind of ALOR control messages, even if they are received from a port in blocking status. When ST detects a protocol change, the ALOR protocol uses configuration messages as follow:

(a) Every gigabit switch must transmit the message to all its ports only one time.
(b) When a message arrives at a gigabit switch port, it must acknowledge it. Acknowledgement messages becomes more robust the ALOR protocol.
(c) Root-switch transmits first
(d) Rest of gigabit switches transmit configuration messages when the first ALOR configuration message is received from a neighbor.

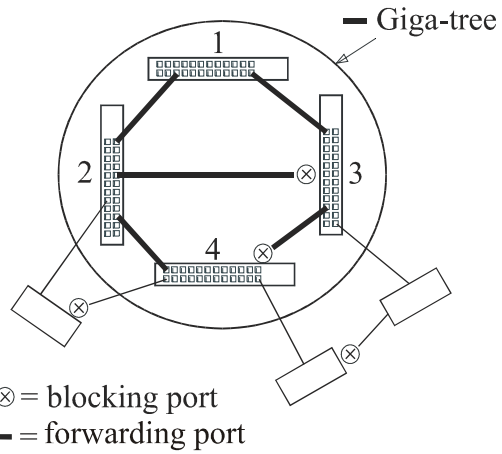

Figure 2. Giga-tree, a tree at the top level of a campus backbone.

Note: An ALOR configuration message cannot travel from a gigabit switch to another gigabit switch through a normal switch (Figure 2). There are two reasons:

(a) It would mean that there is a loop in the topology because any gigabit switch must reach another gigabit switch through root. The ST protocol guarantees there is no active loops.
(b) A normal-switch does not accept ALOR messages from a blocking port. It will only accept ST protocol messages.

2.2. ALOR Learning algorithm

The ALOR protocol learning process is based on the tree generated by the ST protocol, and evolves from the ST leaves towards the root. This is therefore a bottom-up process based on the following:

A gigabit switch is proprietary of all the stations that it listens from all its ports other than the ports in the giga-tree. Thus, it can associate a "cost to reach" equal to zero to the MAC address of the source station (hop count is the simplest metric, but other metrics are also possible).

But the main goal of the learning process is to share the information among all the switches of the switched-LAN. Switch knowledge is transmitted to the neighboring switches through ALOR location messages. Basically these messages contain a list of the new stations (MAC-addresses) and the cost (hops) to reach them.

Thus it is necessary to plan a spreading strategy to obtain a full propagation. This process consists of the following steps:

Bottom-Up process: This process is initiated by the leaf-switches. They transmit their knowledge to the switches with ALOR location messages using the root-port (port used to reach root switch) to reach the higher hierarchy levels and the blocking ports to reach the other branches of the tree (lateral propagation).

Any switch other that the leaf-switches waits to receive an ALOR location message for all the designated ports before repeat the process. Finally the bottom-Up process stops at the root switch. At this moment, the root switch has a full knowledge of all the active stations in the switched-LAN and their location (cost). Note that the routes known by root are optimal since the ST is an optimal tree.

It is necessary a second propagation, top-down phase, in order to allow the root switch to spread its knowledge to the rest of the switches. Then all the switches will know where the new stations are and how much cost to reach them.

Top-Down process: This process is initiated by the root-switch. A switch transmits their knowledge to the switches using the designated ports to reach the lower hierarchy levels and the blocking ports to reach the others branches of the tree (lateral propagation).

A switch repeats the top-down process when it receives an ALOR location message by its root port. Obviously, the process stops at the leaf-switches.

2.3. ALOR Learning example

The example is based on a campus LAN with a fat tree topology (see figure 3.a) that uses gigabit switches in the backbone. We suppose that the ST is already formed and the gigabit switches know which ports are involved in the giga-tree. The cache memory is empty. Four stations “a”, “b”, “c” and “d” transmit (i.e. a broadcast frame). The frames reach the corresponding gigabit switches and ALOR learns, in each switch, that a new station can be reached by port-I with cost 0. ALOR store in cache <station><cost><by port>. For example, in Switch-1 store that “host-a can be reached at cost 0 by port-i” (a0i).

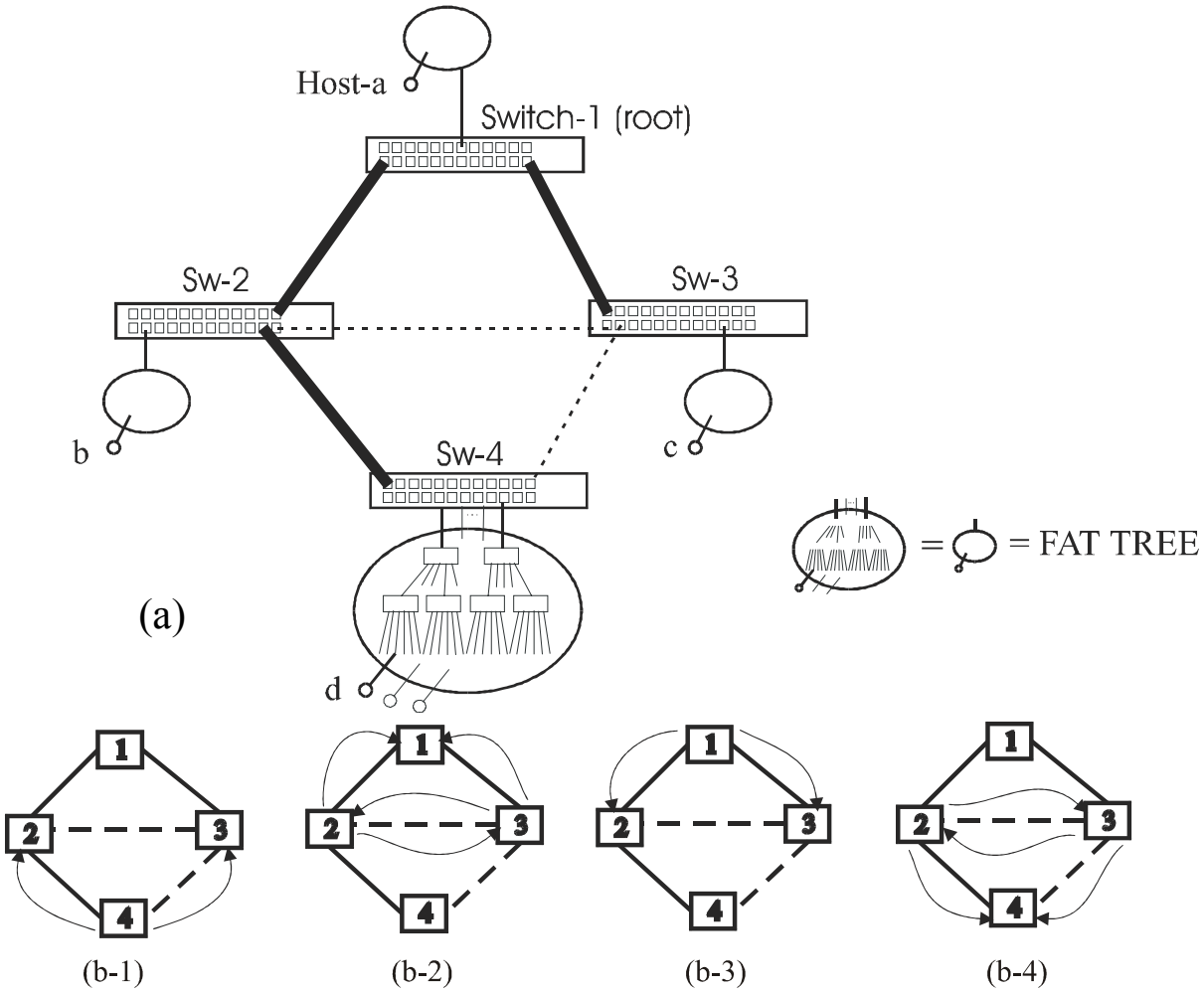

Figure 3. (a) A fat tree campus topology LAN used in the example. The fat lines between gigabit switches show the ports enabled by the ST and the dotted lines show the ports blocking. (b) ALOR protocol evolution.

	Switch-1				Switch-2					Switch-3					Switch-4			
	2	3	i	M+	1	3	4	i	M+	1	2	4	i	M+	2	3	i	M+
tx-1			a	a0i				b	b0i				c	c0i			d	d0i
rx-1 tx-2							d0		d14			d0		d14				
rx-2 tx-3	b0 d1	C0 D1		b12 c13 d22 d23		c0 d1			c13		b0 d1			b12	b0 d1	c0 d1		b12 c13
rx-3 tx-4					a0 b1 c1 d2				a11	a0 b1 c1 d2				a11				
rx-4						a1 b1 c0 d1					a1 b0 c1 d1				a1 b0 c1 d1	a1 b1 c0 d1		a22 a23

Table 1. Table showing the ALOR protocol evolution on LAN from figure 3.a.

Figure 3.b shows the ALOR protocol evolution. (b-1) First ALOR configuration messages; transmission and reception (tx-1 and rx-1). (b-2) tx-2 and rx-2. (b-3) tx-3 and rx-3. (b-4) tx-4

Table 1 shows the detailed evolution of the ALOR learning process. For each switch, the table shows a column with the number of the port from which it receives the ALOR message. To make the example simpler, port-N means the port that connects with the switch-N. The column labeled "i" groups the rest of the ports that are not involved with the input/output of ALOR messages, but are the ports that connect with the rest of the fat tree, so with the final stations. The column labeled "M+" is the cache memory where the results of the ALOR learning process are summarized.

The first ALOR location messages are transmitted by switch-4 (a leaf-switch) (Row labeled "tx-1" in Table 1). It transmits a message to switch-2 and another copy to switch-3. Message data field contains all new data from its cache memory "M+". In the example, switch-4 sends "d0". The next row of table 1 (labeled "rx-1, tx-2") shows two steps:

(a) In the first one, switches 2 and 3 receive the message by their port-4 containing data "d0". Them both switches learn that its neighbor switch-4 can reach station-d with cost 0, so if they can reach switch-4 with cost 1 (one hop), then they can reach station-d with cost 1. Both switches store in cache "d14" ("I can reach station-d with cost=1 by my port-4").

(b) The second step describes the transmission come out by switches 2 and 3 (figure 3.b2). In this point switch-2 sends an ALOR location message to its gigabit switch neighbors 1 (bottom-up propagation) and 3 (lateral propagation). Switch-2 sends data ("b0", "d1"). Approximately at the same time switch-3 does the corresponding. Switch-3 sends data ("c0", "d1").

The rest of the process (next rows of the table) is a repetition of those steps. It is important to highlight that "M+" can store more than one route to a single station. For example, switch-1 in row labeled "rx-2, tx-3"store "d22" and "d23", so it knows it has two optimal routes to reach station-d in with cost 2, one by port-2 (switch-2) and another one by port-2 (switch-3). The same happens in the last row of the table in switch-4 with station-a.

Finally, at the end of this ALOR learning cycle, all gigabit switches know the optimal routes to the stations in the example. The process is repeated continuously. At the end of the top-down process a new bottom-up is started by the leaf-switches.

2.4. Learning fidelity criterion

ALOR switches acquire their knowledge from the transmission of location messages. We should consider the implications of lost location messages due to transmission errors. In case of transmission errors, the cross information that each switch has from the neighbors will not be coherent with the information recorded by these neighbors. But this is not, in fact, a big problem. When a gigabit switch does not know an optimal route (ALOR) for a station it will always use the normal Spanning Tree information.

The following situation could happen (see figure 3.a): Suppose that Switch-4 wants to transmit a frame to station-c and does not know that an optimal route exists through its blocking port to switch-3. Then switch-4 will send the frame through the spanning tree towards switch-2. Now, it is possible that switch-2 knows an optimal route to station-c through its blocking port towards switch-3. So, it is not a big problem if an ALOR location message is lost.

Like in the ST protocol, the information learnt by switches has an expiry time. ALOR uses the same expiry criterion as ST. There is a long cache time (5 to 15 min.) for the normal operation and a short cache time (3 to 15 sec.) when a "topology change" is produced.

The expiry time is controlled by the owner switch. When the station-related information is no longer valid, the proprietary switch will set an infinite cost associated with that station in the next location message, (infinite = 255).

2.5. ALOR Routing Protocol

ALOR routing is performed in a distributed manner. When a station frame reaches a gigabit switch, it is checked in ALOR cache for the destination station. If there is a route, and it is different to the route through the ST, then the frame is sent through the corresponding port, otherwise ALOR leaves this job to ST.

An interesting case arises when two of the switches know additional routes which are optimal. In that case the traffic can be distributed in a proportional way. For example, if there are two optimal routes, traffic can be split fifty-fifty but this policy can produce a bad collateral effect if it is done improperly. If frames can reach a destination through different routes, it is possible a second frame reaching destination before the first one. Indeed, this is a situation that can happens in any distributed protocol that continuously learns new routes. Note that the ST protocol does not guarantee this situation.

Finally, for broadcast frames, the best way to spread them is through the Spanning Tree.

3. ALOR Performance Evaluation

In order to evaluate the performance of the proposed ALOR protocol we have used SMPL[6]. We have defined two extended LAN models. The first one will use the ST protocol and the other will implement the ALOR protocol. In the simulations, we assume that giga-switches are able to process every frame in each port at full rate (non blocking switching). Contention will appear when two or more frames want to cross the switch using the same output link. Also, simulations assume that every giga-switch has a sub-LAN associated with it. Stations in the sub-LAN generate an exponential pattern traffic –being □ the inter arrival frame rate. All traffic considered in simulations crosses al least one giga-switch (from stations in a sub-LAN to another sub-LAN) and it is uniformly distributed. Frame size is fixed to 1Kbytes. Finally, the traffic generated by ST and ALOR protocols was also considered, however no significative differences were found). As performance metrics we have selected the average backbone transit delay (measured in microseconds) and the □ factor, related to the exponential traffic pattern, as the network traffic load indicator.

In the first simulation we compare a campus LAN with three giga-switches as backbone using ST and ALOR. Figures 4 .a and 4.b shows both backbone topologies. Note that ST protocol does not use the link between switches 2 and 3.

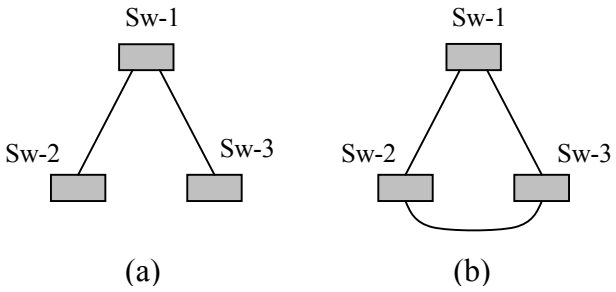

Figure 4. Three giga-switches forming a (a) tree topology (Spanning Tree protocol). (b) ring topology (ALOR protocol).

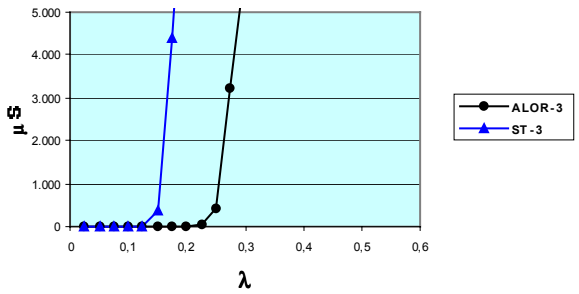

Figure 5. Average frame transit delay in the backbone with ST-3 (Figure 4.a) and ALOR-3 (Figure 4.b).

Figure 5 shows the average transit delay that a frame requires to cross the backbone. The ALOR protocol gets near two times more network traffic than ST.

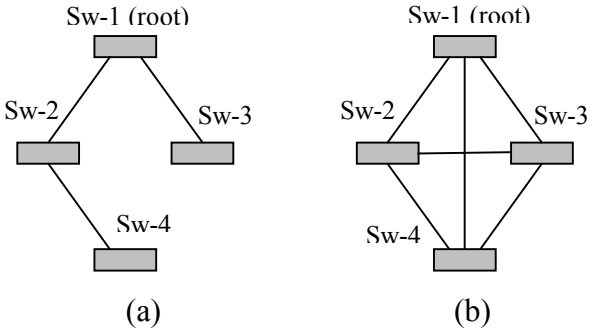

Figure 6. (a) Shows the giga-switches connected in a tree topology (Spanning Tree protocol). (b) Shows the same four giga-switches but with full connected topology (ALOR protocol).

In the second simulation we compare a backbone with four giga-switches using ST and ALOR. Figure 6 shows both topologies.

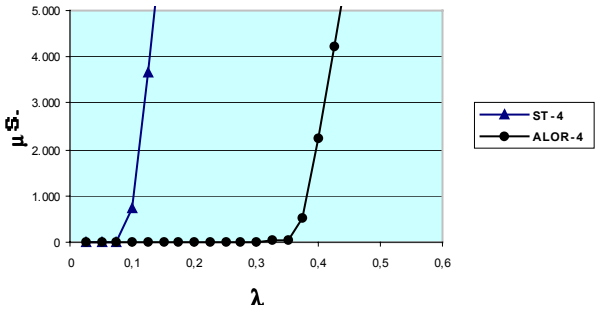

Figure 7. Average transit delay in the backbone with ST-4 (Figure 6.a) and ALOR-4 (Figure 6.b).

Figure 7 shows the average transit delay to cross the backbone. In this simulation the differences are greater. This is normal because we are now comparing a net with diameter=4 against a net with diameter=1. Simulation shows a rapid saturation in the link between switches 1 and 2.

4. Conclusions

A new ALOR protocol is proposed in this paper. ALOR is very simple and easy to implement. It works on top of the Spanning Tree (ST) protocol and allows gigabit Ethernet switches to work with active loop topologies. It is an important enhancement over the ST protocol that allows topologies with loops but blocks ports in switches in order to remove topology loops. The ALOR protocol can work efficiently on whatever kind of topology, being

able to always use the best routes to reach LAN destinations.

Although we have proposed to use ALOR only on the gigabit backbones of extended LANs, it can be applied to every switch in the LAN.

Finally, we have compared the current ST protocol with ALOR protocol by simulation. Simulation results show that ALOR considerably improves network performance on the network backbone even with the shortest topology, showing that as backbone size and switch connectivity increase the improvements also increase. So, ALOR is an alternative to be considered when designing Gigabit Ethernet Backbones.

5. References

[1] R. Perlman, *Interconnection Networks*, (2nd Ed.), Addison Wesley, 1999.

[2] R. Seifert, *Gigabit Ethernet*, Addison Wesley, 1998.

[3] R. García, J. Duato, JJ.Serrano, "A New Transparent Bridge Protocol for LAN Internetworking using Topologies With Active Loops", in the *Proceeding of the 1998 International Conference on Parallel Processing (ICPP98)*, 1998.

[4] R. García, J. Duato, "Suboptimal-Optimal Routing for LAN Internetworking using Transparent Bridges", *International Journal of Foundations of Computer Science,* Vol. 9, No. 2, 1998, pp. 139-156.

[5] IEEE, *Mac Bridges*, ANSI/IEEE Std. 802.1D, ISO/IEC 10038.

[6] M.H. MacDouall, *Simulating Computer Systems. Techniques and Tools*, MIT Press. 1987.