*Chapter 1*

# STP-BASED NETWORKS: ANALYSIS AND DESIGN

***Medagliani P.,† Ferrari G.,† Germi G.,‡ and Cappelletti F.‡*** *
† University of Parma, Parma, Italy
‡ Selta spa, Roveleto di Cadeo (PC), Italy

**Keywords:** Spanning Tree Protocol, Open Shortest Path First, Hot Standby Router Protocol, Virtual Local Area Network, Opnet, simulation, performance, configuration, guidelines.

**Abstract**

In this work we analyze, through simulations, the performance of *Spanning Tree Protocol* (STP)-based Ethernet networks with ring and double ring topologies. In particular, we consider both the presence and the absence of *Virtual Local Area Networks* (VLANs), and we derive the optimized STP parameters which minimize the STP convergence time and maximize the network stability. Two possible techniques for STP internal timers management are evaluated. The presence of failures (either broken links or nodes) is also taken into account, in order to determine the proper STP parameters which guarantee connectivity recovery and convergence in all possible network scenarios. Some of the simulation results are also verified through an experimental testbed. Finally, the use of "transparent" switches is proposed as a solution to (i) accelerate the STP convergence, (ii) increase the reaction capability to failures, and (iii) overcome the limitations, imposed by the STP, on the maximum sustainable number of nodes. In particular, this approach allows to extend the number of nodes in the network, still guaranteeing the possibility of incorporating VLANs. In order to evaluate the impact of failures in a realistic network, the Open Shortest Path First (OSPF) protocol and the Hot Standby Router Protocol (HSRP) are introduced in an STP-based network. This analysis shows that the use of OSPF protocol and the HSRP does not affect the STP performance, even if a longer delay is required in order to start the transmission of ping messages and a reduced reaction capability to node/link failures must be accounted for.

*E-mail address: {paolo.medagliani, gianluigi.ferrari}@unipr.it. {g.germi, f.cappelletti}@selta.it

# 1. Introduction

Ethernet is a widely used connection technology for Local Area Networks (LANs) [1]. Its simplicity, low cost, and high data transfer capacity have favored its use in several application scenarios. For example, Ethernet is the technology of choice to interconnect racks of servers with low latency and high reliability, or to store data on remote hard disks (i.e., Storage Area Networks, SANs). In all these cases, the ability of the network to react against failures is of paramount importance.

In order to improve the network robustness against failures, a first solution consists in guaranteeing redundancy of paths between a source and its destination. For example, in most process automation plants a long connectivity loss cannot be tolerated. Therefore, exploiting path redundancy to prevent from data loss and react against possible failures is highly desirable. However, the use of redundant paths is not allowed because it leads to the creation of loops in the network, which may quickly saturate its transport capacity. A solution to this problem is given by the adoption of the *Spanning Tree Protocol* (STP) [2], which eliminates the presence of loops in the network and provides alternative paths when the active one fails.

In [3], the author presents a framework for the performance enhancement of Ethernet networks, considering the bandwidth limitations introduced by the STP. Due to the wide diffusion of Ethernet networks in many application fields, the security aspects of this technology must be also taken into account. In [4], the authors analyze the stability of STP and develop a spanning tree port cost-based approach to resist to possible external attacks. In [5], the authors propose the division of an STP-based network into two tiers in order to increase security and hide network infrastructure operations.

Even if STP-based Ethernet networks have a capillary diffusion, in the literature there are a few papers analyzing their performance. Moreover, the choice of the optimized STP parameters, which guarantee network convergence, i.e., absence of loops, is typically left to heuristic trials. In this work, we present a simulation-based performance analysis of STP-based Ethernet networks and, on the basis of the obtained results, we derive some guidelines for optimized configuration of STP parameters. In particular, network behavior is analyzed through the Opnet simulator [6]. Since no suitable models are provided by Opnet, a custom model, through which the convergence of the STP is evaluated and the optimized parameters that allow fastest network convergence are derived, is implemented. The optimized values of the STP parameters are obtained both in the presence and in the absence of VLANs connected to the switches. In addition, the robustness of the STP-based networks against node failures is evaluated and a set of optimized configuration rules for the STP parameters is derived. The use of "transparent" switches is proposed as a possible approach to overcome the limitation, on the maximum sustainable number of nodes, imposed by the STP. Finally, the impact of layer 3 protocols, namely the Open Shortest Path First (OSPF) protocol and the Hot Standby Router Protocol (HSRP), on the performance of STP-based networks is considered.

The structure of this work is the following. In Section 2., an overview of the considered protocols is provided. In Section 3., the performance of STP-based networks is first evaluated in the absence of failures, considering two different internal timer management strategies. In particular, the maximum network dimension (in terms of nodes) is derived

as a function of the main STP parameters. Then, this analysis is extended to account for possible failures. In order to overcome the limitations (in terms of network dimension and convergence speed) imposed by the STP, in Section 4. we propose the use of "transparent" switches, which are properly characterized. In Section 5., on the basis of the previous results we summarize simple design guidelines for configuring the STP parameters. Finally, Section 6. concludes the work.

# 2. Protocol Overview

## 2.1. Spanning Tree Protocol

The segments of a Local Area Network (LAN) are connected through switches, which operate at the Layer 2 of the ISO/OSI stack [7]. These devices forward the packets received from an input port towards one or more output ports. In order to have correct network operation, logical (or layer 2) path loops between the nodes must be avoided, i.e., there must be a unique active path between any pair of switches. In the opposite case, packets would be endlessly forwarded by the nodes, with catastrophic effects on the network performance. These problems are not relevant when transmitted packets have a Medium Access Control (MAC) address field with information known by the switches. In this case, each switch is aware of the devices connected to its ports. On the other hand, a broadcast message or a message directed to a node with an unknown MAC address is forwarded by a switch to all the active ports, except for the input one. In this case, in the presence of a loop, the packets will be replicated by all switches in the network, thus quickly saturating the network.

A possible approach would be physically avoiding loops during the network creation phase. However, this choice can be unreliable in the case of a link failure, after which some areas of the network could become unreachable and isolated. A better approach would exploit the redundancy of paths from a source to a destination. After network start-up, only one of the redundant paths becomes active, whereas the remaining paths are left inactive. In the presence of a failure, the original path is replaced with one of the inactive paths, guaranteeing correct network operations.

The algorithm which manages the activation of the links is known as Spanning Tree Algorithm (STA) and is embedded into the STP, which is a part of the IEEE 802.1D standard [8]. Since the operations, needed to manage the STP, are performed by all the switches in the network, the STA is totally distributed. The goal of the STP is the creation of a tree which allows to route data packets to any segment of the network, avoiding loops and leaving only one active path between any pair of source-destination nodes.

The limitations of this protocol can be summarized as follows: (i) the convergence time increases when the number $N$ of switches in the network increases; (ii) the control traffic introduced by the STP degrades the network performance; (iii) the inactive paths do not increase the overall capacity of the network; and (iv) the maximum number of switches in the network is limited. Solutions to these problems are provided through possible enhancements of the basic STP, such as the Rapid STP [9] and the Multiple STP [10].

The main phases of the convergence process of the STP are: (i) election of a root node (i.e., a root bridge, RB, according to the STP reference names), (ii) determination of the least cost paths, (iii) deactivation of the remaining paths, and (iv) resolution of the paths

with equivalent costs. The last phase occurs only when there is more than one path with the same characteristics, whereas the other three steps occur at the network start-up.

Every switch has a unique identifier and an associated priority. In the case of different priorities, the switch with the lowest priority becomes the RB. On the other hand, if all priorities are equal, the node with the lowest identifier will become the RB [11]. Once the tree is created, every node has a minimum-cost path towards the RB. Note that the STP does not guarantee that the path between any source-destination pair is the one with minimum cost, because the minimization of the path cost is not the goal of the STP. The use of minimum-cost paths to the RB is guaranteed by the following rules:

- after the election of the RB, every switch computes the cost of every path from itself to the RB and chooses the one with least cost (the associated port is referred to as root port, RP);
- the switches in the same network segment cooperatively select the switch with least cost (the port which connects a switch to the network segment is referred to as designated port, DP).

Every switch needs to have a complete knowledge of the network (i.e., of the priorities and the identifiers of the other nodes). To this end, a periodical "special" packet, referred to as Bridge Protocol Data Unit (BPDU), is transmitted. A BPDU contains information about the transmitting node, i.e., the states of its ports, its priority, the cost of the path from the switch which originates the BPDU to the RB, and the identifier of the RB. The BPDUs are not forwarded by the receiving switch. According to the STP, there are three kinds of BPDU: (i) Configuration BPDU (CBPDU or simply BPDU), used by the switches to create and maintain the network tree; (ii) Topology Change Notification (TCN), used to notify topology changes in the network or the addition of a new switch; and (iii) Topology Change Acknowledgment (TCA), used to confirm a topology change in the network.

In order to prevent the formation of loops in the network, the STP defines five possible states for the ports of a switch: (i) disabled, (ii) listening, (iii) learning, (iv) forwarding, and (v) blocking. When the network is created, all ports connected to valid links are in listening state, while the others are in disabled state. In the former state, the switches receive the BPDUs from the other nodes. In the presence of a loop, a port of a switch, which becomes aware of the presence of the loop, is turned into the blocking state, in order to prevent the transit of BPDUs. After a time interval, referred to as forward delay ($FD$), equal to 15 s by default, the ports in listening state switch to learning state [12]. In this phase, the nodes become aware of the surrounding switches. After another $FD$ interval, the ports in learning state are switched into forwarding state and data packets can then circulate throughout the network.

According to the STP, the RB transmits a BPDU with a period denoted as "hello time" (equal to 2 s by default). At the network start-up, each switch elects itself as the RB and transmits a BPDU. If the priority information conveyed by a BPDU from a given switch is higher than the one stored in the receiving switch, the latter updates its status, electing as RB the transmitting switch, and stops transmitting its own BPDUs. In fact, from this moment on it will retransmit only the BPDUs received from the elected RB.

The conditions to be satisfied to guarantee convergence of the STP are the following: (i) all switches elect the same RB; (ii) one of the switches in the loop has a port in blocking

state; (iii) the remaining ports of that switch and of all the other switches are in forwarding state; and (iv) the previous three conditions are stable during the time. Once all previous conditions are met, only the RB will broadcast the BPDUs every 2 s and the other switches, upon the reception of a BPDU, will retransmit it and refresh their internal information.

Another important timer of the STP is the max age (denoted as $MA$) timer, which defines the time interval after which a reset of the switch is required if no refreshing BPDU is received. When a BPDU is retransmitted by a switch, the latter modifies only the cost of the path to the RB and a timer, referred to as message age (denoted as $m_{\text{age}}$), used to measure the "distance" of a node from the RB. In particular, this value is generally increased by 1 s or 2 s, depending on the state of internal timers.[1] This value is used in combination with the $MA$ in order to guarantee the reliability of the information conveyed in a BPDU. More precisely, the RB generates a BPDU with $m_{\text{age}} = 0$ s. Then, the switches which receive this BPDU assume that the information transported by the BPDU is valid for a time interval equal to $MA$. When the BPDU is relayed, $m_{\text{age}}$ is incremented and the receiving switch assumes that the information conveyed in the BPDU is valid for $MA - m_{\text{age}}$ (dimension: [s]). When the message age of a switch becomes larger than the max age, the switch sends a TCN BPDU to the other switches in order to notify them of the occurred problem. In this case, the STP does not converge.

## 2.2. Per-VLAN Spanning Tree

VLANs are instrumental to logically segment the network into areas. This solution is less expensive and more flexible than the traditional approach based on the use of dedicated switches. This technology, included into the IEEE 802.1Q standard [13], allows to interconnect the switches which share the same VLAN, even if they belong to geographically separated networks. This operation, referred to as trunking, is based on the use of tags which identify which packets belong to which VLAN. A port which conveys tagged traffic is named trunk port, whereas a port, through which untagged packets enter inside the VLAN, is referred to as access port. In the case of mixed traffic, instead, the port is referred to as hybrid.

There are two possible ways of assigning a device to one or more VLANs: (i) dynamic and (ii) static. In the former case, database-based software packages are used to associate a switch with a specific VLAN. In the latter case, instead, assignments are based on a strict association between a port and the corresponding VLANs. This approach corresponds to the use of port-based VLANs. With this mechanism, all users connected to a port are automatically associated to the assigned VLAN. The STP can be applied to every created VLAN. In this case, the mechanism of the extension of the STP, referred to as per-VLAN Spanning Tree (PVST), remains the same, except for the fact that the tree is computed also for each VLAN (i.e., for the switches belonging to the same VLAN).

[1] Generally, a 1 s increment is associated with STP convergence, whereas a 2 s increment denotes no STP convergence in the network.

## 2.3. Open Shortest Path First Protocol

The OSPF protocol is one of the most used Internet routing protocols nowadays and it operates at the layer 3 of the ISO/OSI stack [14]. The Internet is composed by several routing domains, called Autonomous Systems (ASs) and the data traffic is routed along different paths according to the weights associated to each link by network operators. The OSPF protocol gathers link state information from available routers and constructs a topology map of the network. Therefore, each router has a complete knowledge of the network topology and, using the information associated with the weights at each link, can determine the shortest path to a specific destination. The main advantages of the OSPF protocol with respect to a distance vector-based routing protocol can then be summarized as follows: (i) it converges faster; (ii) the routing update packets are small, as it doesn't send the entire routing table; (iii) it is not prone to routing loops; (iv) it scales very well for large networks; and (v) it recognizes the bandwidth of a link and takes it into account in link selection.

The routers that belong to the same area, i.e., a set of networks and hosts within an AS that have been administratively grouped together, periodically send Hello messages in order to notify surrounding routers of their activity. If a router does not receive Hello messages for a period of time larger than RouterDeadInterval (typically 40 s), it assumes the connectivity with its neighbor is lost and starts generating new Router Link-State Advertisement (LSA) messages to notify to the other routers the topology change. The transmission of the LSA messages forces, in surrounding routers, the recomputation of the shortest path and the following update of the routing tables stored in each node. The duration of the recovery phase is given by the sum of three contributions: (1) the failure detection time, (2) the LSA flooding time, and (3) the time to complete the calculation of the new paths and update the routing tables. Given an Hello interval time equal to 10 s, a failure detection takes between 30 s and 40 s. The LSA flooding time is given by the propagation delay and the delays related to rate limitations on the transmission of a group of LSA messages (denoted as Link-State Update, LSU). Finally, since the computation of the shortest path makes use of the Dijkstra's algorithm [15], which requires significant time in order to complete data processing, an additional time interval, referred to as spfDelay, equal to 5 s, is introduced, in order to collect a large number of LSA message and reduce the number of reprocessing run required by the arrival of a new LSA message.

## 2.4. Hot Standby Router Protocol

The HSRP is a Layer-3 Cisco proprietary standard for establishing a fault-tolerant gateway [16]. This protocol is designed for use over multi-access, multicast or broadcast capable LANs (e.g., Ethernet), and is based on the use of a single *virtual* IP address for a set of routers which can act as gateways. In order to prevent from network misconfigurations, only one router is active, whereas the remaining nodes are in the standby state. If the active router fails, a priority-based scheme is used in order to determine the router that must switch into the active state and act as network gateway.

In order to notify the neighboring routers of its address and its priority, a router periodically sends an Hello message to a multicast address, i.e., to all routers in the network with HSRP capabilities, in order to notify them of its activity and communicate to them its priority. As soon as a router has become active, the other routers switch into the standby

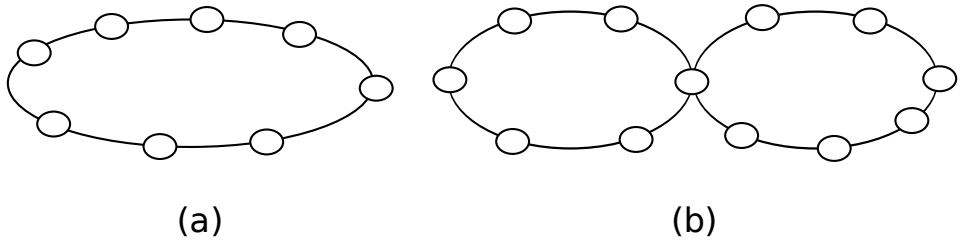


Figure 1. Topologies considered for simulation-based analysis: (a) ring topology and (b) double ring topology.

state and only the active router sends the periodic Hello message. If the active router fails, the other routers can no longer receive the Hello messages. After a time interval, referred to as Hold time and equal to 10 s by default, the second router in the priority list, which was in standby, switches in the active state and replaces the failed router, acquiring the virtual IP address of the active router.

# 3. Ring and Double Ring Networks

## 3.1. Scenarios Without Failures

In order to analyze the STP and characterize its performance, a proper Opnet simulator model has been developed. This model, which allows to periodically extract (or log) the states of the ports of each switch, has been derived from the model of a Cisco CS2948 switch and presents 4 layer-2 ports. The links which connect any pair of switches are Ethernet 100 Mbps connections. We first consider (a) ring and (b) double ring topologies, as shown in Fig. 1. In particular, in double ring topologies, the switches are configured so that the RB is the node in the center of the double ring. Since the goal of this work is to provide some guidelines for the configuration of the parameters of the switches in STP-based networks, we derive, for given values of the max age and forward delay, the network convergence time. We point out that a ring topology has been considered as this represents the worst-case scenario for an STP-based network. The double ring topology, instead, is a simple, yet representative, example of a possible extension of the ring topology.

We also validate some of the simulation results through an experimental testbed formed by 9 Cisco 2811 switches[2] equipped with a 4-port HWIC-4ESW interface which operates at 100 Mbps, configured with a VLAN, referred to as VLAN 1, on all the ports.

The first indicator considered for STP convergence analysis, according to the definition provided in Subsection 2.1., is the number of exchanged packets in the network. This number, as a function of time, is shown in Fig. 2. The network parameters are $N = 30$ switches, $MA = 20$ s and $FD = 15$ s. A ring network topology is considered. From the results in Fig. 2, one might be tempted to conclude that the network has converged after approximately 50 s. However, observing the simulator log files and the states of the ports

[2]The Cisco switches, i.e., CS2948, used in the simulator have the same functionalities of the switches in the experimental testbed.

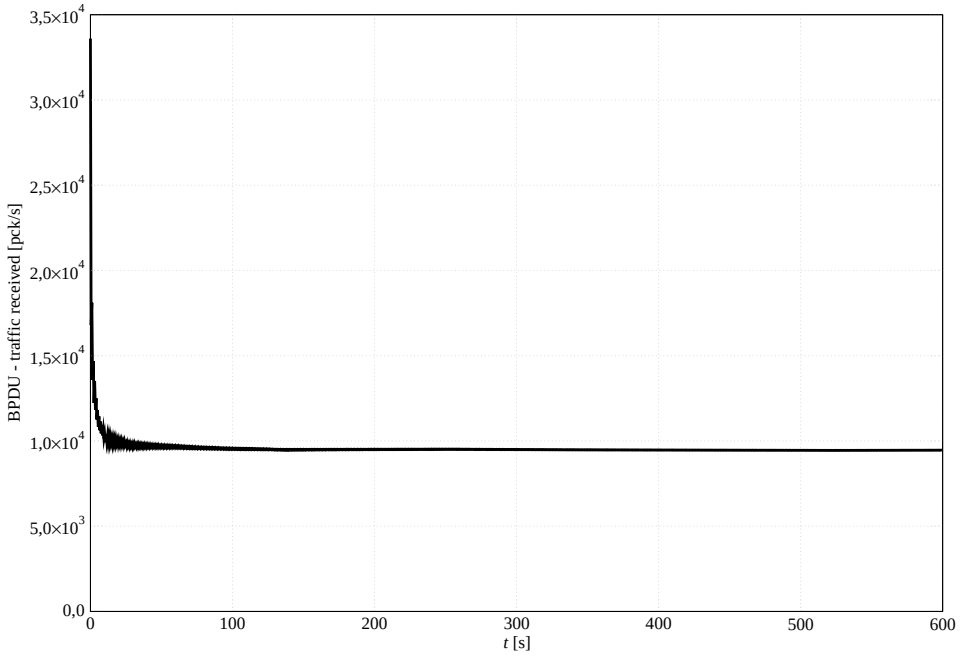


Figure 2. Number of exchanged packets in a scenario with $N = 30$ switches, $MA = 20$ s and $FD = 15$ s.

of each switch—not shown here for lack of space—it can be concluded that the network does not meet the convergence conditions described in Subsection 2.1., even if BPDUs are regularly transmitted. In fact, this analysis technique does not take into account the different types of transmitted BPDUs. In the results presented in Fig. 2, the traffic is due to the presence of TCN BPDUs and not to CBPDUs transmitted after network convergence. In particular, when the value of $MA$ is not correctly selected for the considered network, the most distant nodes receive BPDUs with $m_{\text{age}}$ equal to $MA$. According to the STP, as soon as a switch experiences this situation, it starts sending a TCN message to the other switches, which will eventually reply with TCA messages, upon acknowledgement of the topology change. This means that the chosen values of $MA$ and $FD$ are too small for the considered scenario and the STP does not converge. In this case, the network will be divided into two areas and the RB in the network will not be unique.

The convergence of the network can be verified by careful examination of a properly generated (through our simulator) log file at each node. In particular, we have considered scenarios with different numbers of switches, varying the max age and forward delay, in order to obtain the minimum values of $MA$ and $FD$ which guarantee network convergence. In Fig. 3, the minimum values of $MA$ required for convergence is shown as a function of the number of switches in the network. The $FD$ is not shown, since it can be extracted according to the relation

$$2(FD-1) \geq MA$$

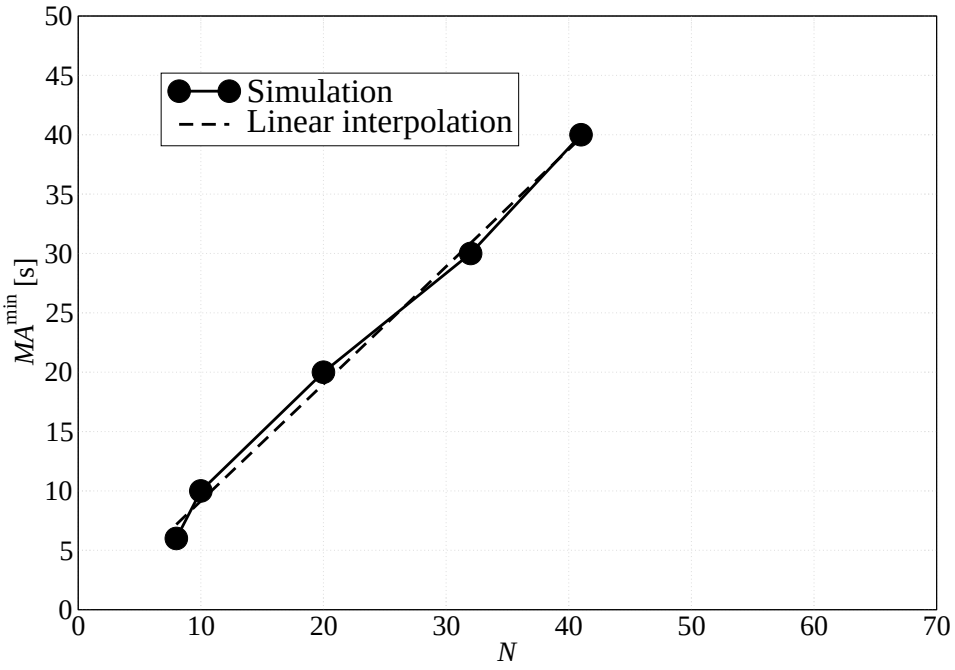


Figure 3. Minimum *MA* considering ring topology with Cisco switches.

required by the IEEE 802. 1D standard. Therefore,

$$FD^{\min} = \frac{MA^{\min}}{2} + 1.$$

By linearly interpolating (as a function of $N$) the *MA* values in Fig. 3, one finds that

$$MA^{\min} = 0.9876 \cdot N - 0.7242 \simeq N - 0.7. \quad (1)$$

Setting *MA* to the minimum value given by equation (1) guarantees fastest convergence. However, equation (1) is no longer valid when there is a link or node failure since the network topology changes (this case will be explained in Subsection 3.2.). The first point of the curve shown in Fig. 3 has been also verified experimentally (without VLAN 1). Experimental results show that there is convergence with $N = 8$, $MA = 6$ s, and $FD = 4$ s, whereas with $N = 9$ and the same STP parameters the network does not converge.

In Fig. 4, the minimum *MA* required for convergence, in a scenario with the double ring topology in Fig. 1 (b), is shown as a function of $N$. The considerations carried out for the scenario with ring topology still hold in this scenario. The minimum value of *MA* is given, as a function of $N$, by the following expression:

$$MA^{\min} = 0.5292 \cdot N - 0.9643 \simeq \frac{N}{2} - 1. \quad (2)$$

Note that expression (2) holds in networks where the RB is the central node of the double ring topology. In other scenarios, this expression is not valid and the convergence is no longer guaranteed.

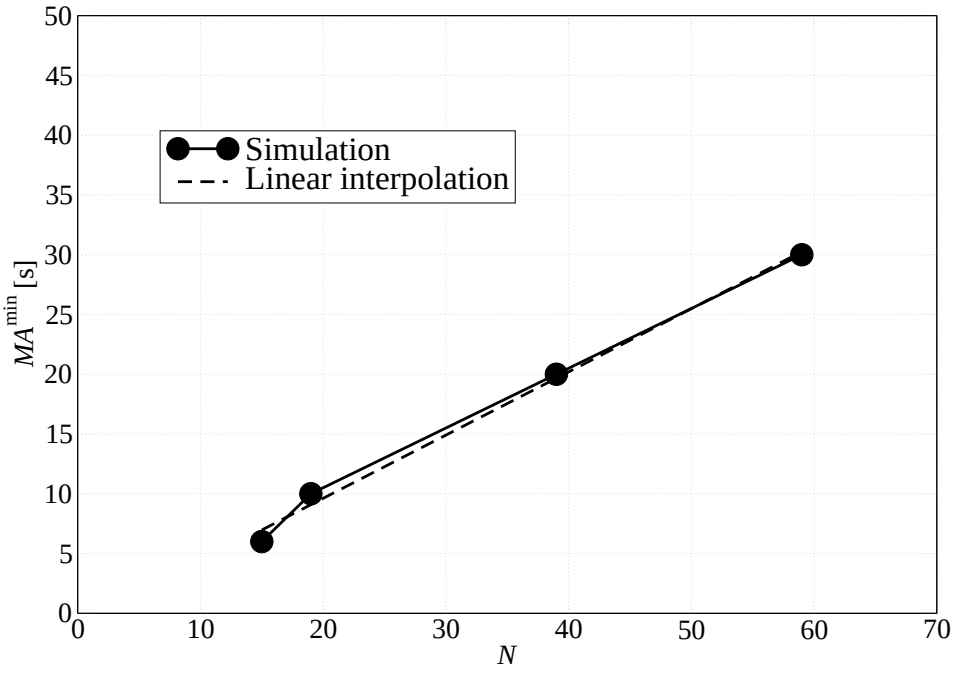


Figure 4. Minimum *MA* considering double ring topology with Cisco switches.

As we have seen, the main limitation to the formation of large STP-based ring and double-ring networks is given by the absence of convergence, caused by the fact that the $m_{\text{age}}$ becomes larger than the *MA* at the nodes which are distant from the RB. A possible solution would be the modification of the increment which the $m_{\text{age}}$ undergoes when the BPDU crosses a node. In particular, this increment can be expressed as follows:

$$\begin{aligned} m_{\text{age}} &= \lfloor BPDU_{\text{cross}} + m_{\text{a-io}} + D_{\text{ma}} \rfloor \\ &= \lfloor BPDU_{\text{cross}} + 1 + 0.5 \rfloor \end{aligned} \tag{3}$$

where $BPDU_{\text{cross}}$ is the BPDU crossing time at switch, $m_{\text{a-io}}$ is the message age increment overestimate, and $D_{\text{ma}}$ is the medium access delay. The first term is given by the difference between the transmission instant and the reception instant of the BPDU, i.e., the physical time interval required by the BPDU to cross a switch. The second and third terms, instead, are derived from statistical considerations and are equal to 1 s and 0.5 s, respectively, as indicated by Cisco [12]. In particular, $m_{\text{a-io}}$ is the minimum increment necessary to avoid an underestimation of the BPDU age. $D_{\text{ma}}$, instead, is the time necessary for a device to gain the access to the medium for initial transmission. In other words, $D_{\text{ma}}$ corresponds to the time between the instant at which the switch decides to retransmit the BPDU and the instant at which the BPDU effectively begins to leave the switch.

Expression (3) for the message age applies to Cisco switches (i.e., nodes with switching capabilities running the Cisco kernel). A Linux kernel is also publicly available [17]. The Linux kernel neglects the contribution of the message age increment overestimate. There-

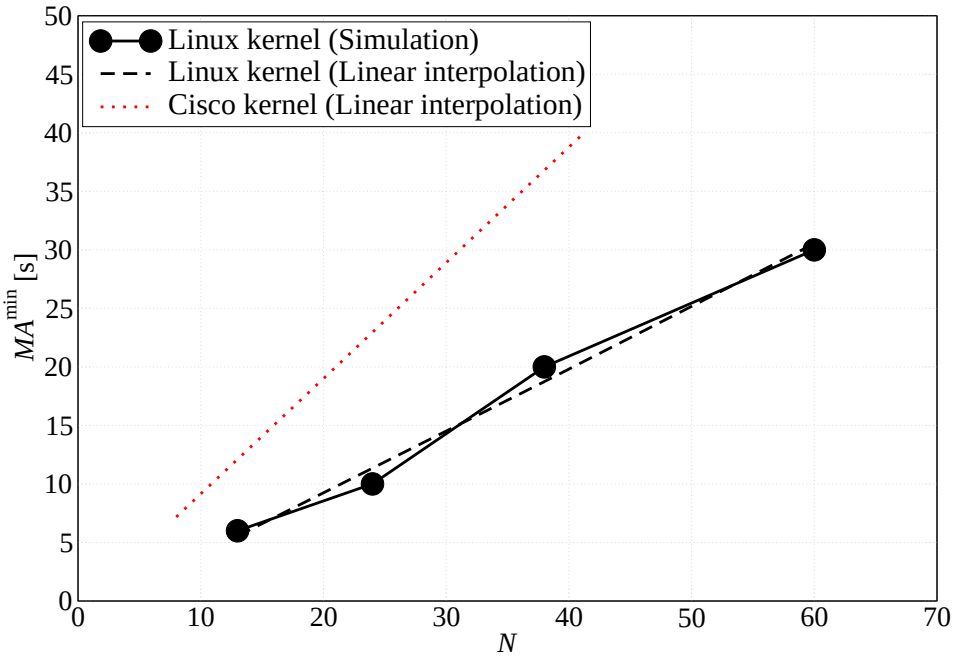


Figure 5. Minimum *MA* considering ring topology with switches with Linux kernel.

fore, the message age can be given the following expression:

$$m_{\text{age}}^{\text{Linux}} = \lfloor BPDU_{\text{cross}} + D_{\text{ma}} \rfloor = \lfloor BPDU_{\text{cross}} + 0.5 \rfloor.$$

In Fig. 5, the minimum *MA* of a network is shown, as a function of $N$, in scenarios with ring topology and switches running the Linux kernel. Since the crossing time is lower in nodes with the Linux kernel, the maximum number of switch, for which convergence is still guaranteed, is larger than in the case with Cisco kernel. In particular, given a value of *MA*, the number of switches which can be supported using the Linux kernel is twice that supported using the Cisco kernel. More precisely, the minimum message age depends on $N$ as follows:

$$MA^{\text{min}} = 0.5277 \cdot N - 1.309 \simeq \frac{N}{2} - 1.3. \tag{4}$$

In Fig. 5, for comparison purposes, the performance with the Cisco kernel (dotted line) is also shown—this curve is the linear interpolation curve in Fig. 3.

The use of the Linux kernel allows to reduce the convergence time of the STP. In fact, since the convergence occurs after a time interval equal to $2FD$, the possibility of using a lower value of *MA* and, consequently, a lower value of $FD$ reduces the convergence time. On the other hand, as already mentioned, a switch can detect a link or node failure if it does not receive any refreshing BPDU for a period of time longer than $MA - m_{\text{age}}$. In scenarios with the Linux kernel, since the value of $m_{\text{age}}$ increases more slowly (when crossing switches) than in scenarios with the Cisco kernel, the information stored in a node is valid for a longer time interval. Therefore, the failure reaction capability is reduced.

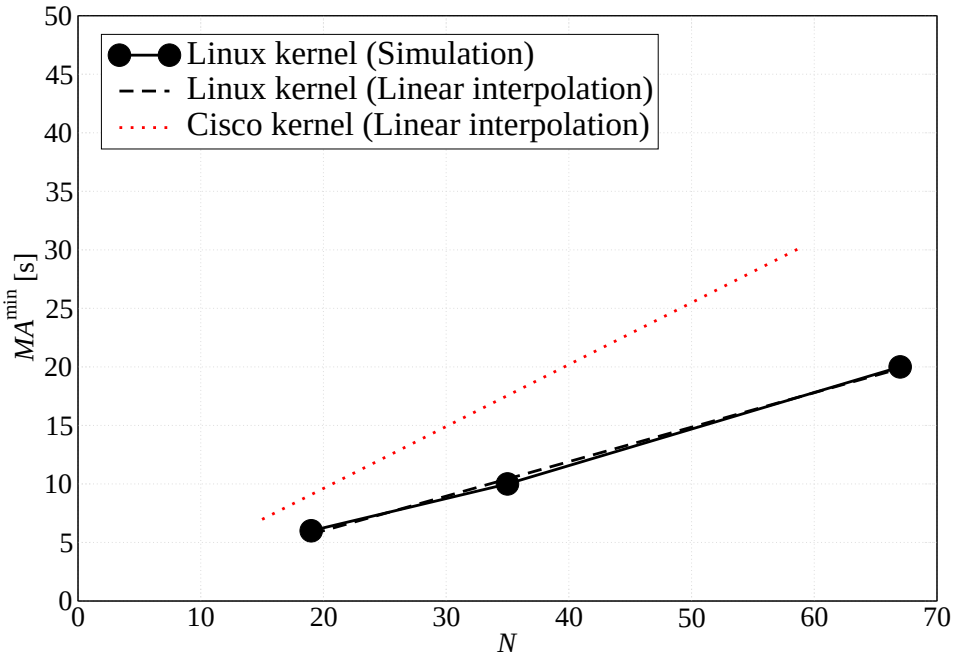


Figure 6. Minimum *MA* considering double ring topology with switches with Linux kernel.

Similar considerations can be made in a scenario with double ring topology and Linux kernel. The minimum required value of message age, as a function of $N$, is shown in Fig. 6. As assumed in scenarios with Cisco kernel at the switches, the RB is forced to be the switch in the center of the double ring, and the obtained results are valid only for this case. The relation between the number of switches and the minimum max age can be approximated as follows:

$$MA^{\min} = 0.2948 \cdot N - 0.1161 \simeq 0.3N - 0.1. \tag{5}$$

Similarly to the results presented in Fig. 5, the minimum max age allowed with the Linux kernel is almost twice that allowed with the Cisco kernel.

So far, the convergence has been analyzed in the absence of VLANs connected to the switches. We have then extended our analysis to account for the presence of VLANs, in the case of both single ring and double ring topologies. In particular, we have connected two VLANs (VLAN 1 and VLAN 2) to each node. From the analysis of the log files generated by the switches, it can be concluded that the convergence performance remains the same. In particular, the PVST allows to create both a common tree for all the switches and particular trees for each VLAN. Since in our simulations the VLANs have been connected to each switch using the previously described parameters, the common tree, the tree for VLAN 1, and the tree for VLAN 2 coincide. The convergence instant is still equal to $2FD$.

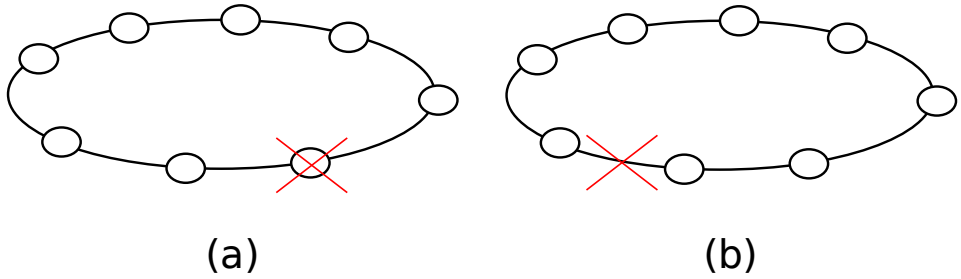


Figure 7. Example of network topology in the presence of (a) a node failure or (b) a link failure.

### 3.2. Scenarios With Failures

The results presented in the previous subsection have been obtained considering a network without failures. However, in order to test the validity of the STP, one must take into account also the reaction capability in the presence of a failure in the network. In Fig. 7, the illustrative failure configurations considered in this work are shown. In particular, Fig. 7 (a) refers to a *node* faiulre, whereas Fig. 7 (b) refers to a *link* failure. One can observe that, considering a network with $N$ nodes, after a single node failure, the number of nodes reduces to $N-1$, whereas in the case of a link failure the number of active nodes remains equal to $N$.

According to the STP, when a switch does not receive a refreshing BPDU for a period of time longer than $MA - m_{\text{age}}$, it sends a TCN BPDU in order to notify the neighboring switches of the lack of convergence in the network. After the transmission of the TCN, the switch which has notified the change of topology, starts sending BPDUs assuming to be the RB of the network. If the neighboring nodes have information about a better RB, they start replying with BPDUs in order to notify the node of their information. However, since the node which originates the TCN receives an information which is still "too old," it starts sending another TCN, thus originating the message exchange just described. This exchange of messages is a symptom of the fact that the STP parameters are too small for the considered network. On the other hand, when a neighboring switch accepts the TCN, it sends back a TCA and stores the information about the new RB. In this case, the network separates into two segments with different RBs. In particular, the switch, which originated the TCN, is in an unstable state, since it oscillates between two possible values of the RB.

The latter case is exactly the scenario which occurs when a link or a node fail. More precisely, referring to the scenario with ring topology, a failure creates an open-chain network. If the STP parameters are too small for a network, the switch which verifies that the message age is larger than the max age, will start broadcasting a TCN. Since the nodes after that switch will not receive any other BPDU from the real RB, they will acknowledge the topology change and the new RB in the network. A solution to this problem is given by the use of more "relaxed" STP parameters, which let the BPDUs propagate into the open-chain to the most distant switch from the RB. The determination of the STP parameters through the analysis of an open-chain network with $N$ switches has a peculiar importance in network design. In fact, the STP parameters optimized for this scenario guarantee that every

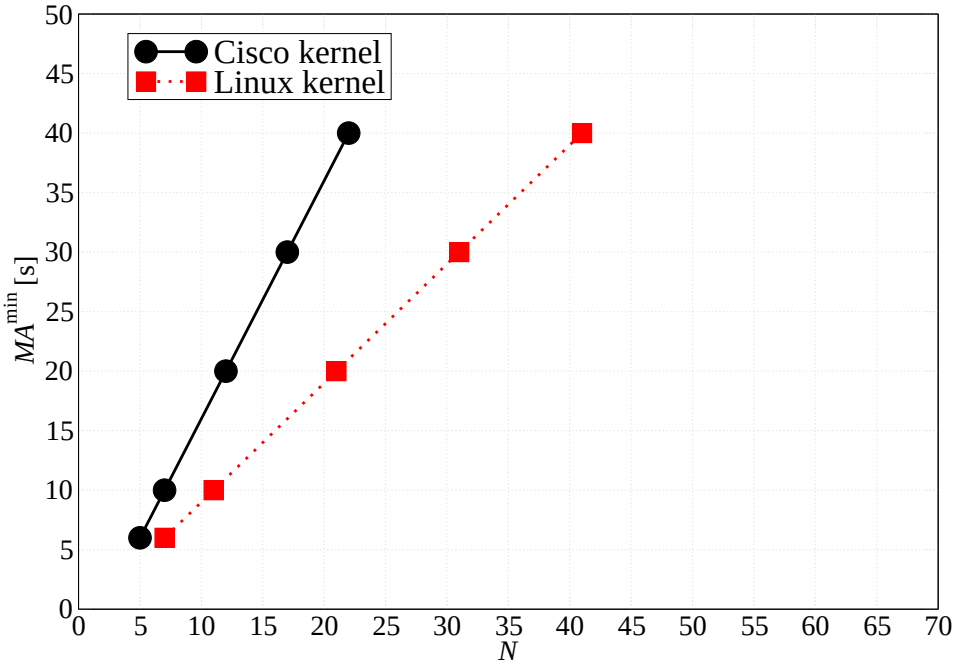


Figure 8. Minimum $MA$ considering an open-chain scenario with a single failure.

network with loops, where the distance between any couple of (source-destination) nodes is smaller than $N$ hops (i.e., $N-1$ switches must be crossed), converges.

According to the STP, a switch realizes that a reset is required when the value of $m_{\text{age}}$ of a received BPDU is equal to the value of $MA$. Assume that there is a node or a link failure at a generic instant $T^*$. After this failure, a switch which does not receive any BPDU for a period of time equal to $MA - m_{\text{age}}$, starts sending a TCN message.[3] After the TCN message has propagated through the network and some switches have reset their information, their ports are first put into listening state. Then, after a time interval equal to $FD$, these ports are switched into learning state and, after another interval of length $FD$, into forwarding state. On the other hand, when the failure is recovered and the loop is restored, as soon as a BPDU propagates through the network and updates the information stored in each node, the switch which must put a port in the blocking state, changes its port state and convergence is reached again.

In Fig. 8, the minimum max age required to guarantee convergence in a open-chain network is shown as a function of the number $N$ of switches in the network. Note that an open-chain network, obtained from a ring network with $N$ nodes, upon a single node failure contains $N-1$ nodes. The performance in scenarios with Cisco and Linux kernels is evaluated. When the Cisco kernel is used, the nodes introduce $m_{\text{a-io}}$ equal to 1 s, and the $MA$ rapidly reaches the value of $m_{\text{age}}$. On the other hand, when the Linux kernel is used,

[3]Considering $T^* = 30$ s, $MA = 6$ s and a ring-topology with a failure of the switch neighboring the RB, the network becomes aware of the failure at 33 s. In fact, the BPDU at $t = 30$ s is not received by the switch, therefore the last refreshing BPDU was received at $t = 28$ s and the information conveyed is valid for $6-1 = 5$ s upon the reception of the last BPDU.

Table 1. Minimum value of $MA$ predicted by the experimental testbed with switches with Cisco kernel.

| $MA^{\min}$ | $N$ |
|---|---|
| 6 | 4 |
| 8 | 5 |
| 14 | 8 |

as mentioned above, given a value of $MA$, the number of admitted switches is larger. This consideration is confirmed by the equations which characterize the minimum values of $MA$ as functions of the number of switches in the network. For the scenario with Cisco kernel one obtains

$$MA^{\min} = 2N - 4$$

whereas for the scenario with Linux kernel it holds that

$$MA^{\min} = N - 1.$$

This performance analysis can be extended to the case of a link failure, after which there are still $N$ active nodes, unlike the previous scenario, where $N-1$ switches are active after a node failure. In the case of Cisco kernel, it holds that

$$MA^{\min} = 2N - 2$$

and in the case of Linux kernel one has

$$MA^{\min} = N.$$

The results with Cisco kernel and a link failure have been confirmed through the experimental results presented in Table 1.

According to this configuration rule, the parameters are tighter than those presented in Subsection 3.1.. However, even if this configuration leads to a longer convergence time, the stability of the network is guaranteed in all the scenarios where the paths between any couple of nodes are shorter than $N$ hops. In addition, according to [8,12], the recommended network diameter (i.e., the maximum number of switches that a packet crosses in order to link any two switches in the network) should be equal to 7. The discrepancy between Cisco recommendations and our results is mainly due to the fact that the Cisco recommendations are conservative. In fact, in order to prevent from possible STP reconfigurations due to delayed transmission of BDPUs in the presence of intense traffic, the specifications given by Cisco force the network to be small, thus guaranteeing convergence even in the presence of traffic congestion. However, our results show that the maximum number of switches which guarantees convergence is 11 in the case of Cisco kernel, and 20 in the case of Linux kernel.

# 4. Extended Networks

In the previous section, we have presented the performance of the STP both in the absence and in the presence of node or link failures in ring and double ring networks. However, with

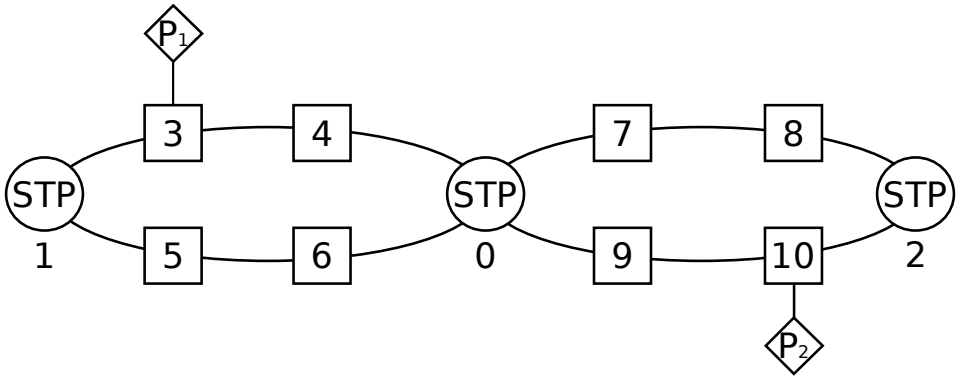


Figure 9. Extended network topology.

large networks it is necessary to configure the STP with large values of *MA* and $FD$. This solution, which refers to the configurations presented in Subsection 3.2., is feasible and reliable, but leads to a longer convergence time and a slow reaction to a failure. Therefore, the use of extended networks might be required.

Generally, the network dimension can be extended through hubs, which relay received packets but do not participate to the operations of the STP. This solution is limited, since a hub cannot manage a VLAN. In this work, we propose the use of "transparent" switches with disabled STP. Transparent switches, unlike hubs, can still manage VLANs. In real networks, the transparent switches are implemented by disabling the STP for both VLANs and the common tree. Since the models available in the Opnet simulator do not support this functionality, we have derived a transparent switch model from the model with enabled STP. A transparent switch receives BPDUs from a port and forwards them to all the other active ports. In addition, this switch can manage the VLANs, so that it can act as access port for the VLAN activated on each port and as a trunk for the links which connect the other switches. Since the STP increases the $m_{\text{age}}$ as soon as a BPDU crosses a switch, the switches without STP must relay the received BPDU without increasing this value.

We have analyzed a network with double ring topology and three nodes with enabled STP: one is placed in the center of the double ring and the other two at the extremes of the double ring. In the middle of the STP-enabled nodes, a variable number of transparent switches can be placed. In particular, in our simulations a topology with 8 transparent switches, as shown in Fig. 9, is considered. The nodes with STP enabled (e.g., switch 0, 1, and 2) have been configured with $MA = 6$ s and $FD = 4$ s. Recalling the performance presented in Fig. 3, in a scenario with 11 switches, the minimum value of *MA* should be equal to 10 s and, consequently, the minimum value of $FD$ should be 6 s. In this scenario, we have also introduced two VLANs, named VLAN 1 and VLAN 2, on every node in the network, both for those with STP enabled and for those with STP disabled. We have then used two nodes, referred to as $P_1$ and $P_2$, which periodically send a ping message in order to trace the active path between them. In addition, we have introduce a link failure at $T^* = 30$ s and a link recovery at $T^{**} = 90$ s in order to evaluate the reaction capability of this network. These instants have been chosen to let the network converge and, subsequently, after an

alteration of the state of a node or a link, analyze its reaction speed.

The considered network converges at $t = 8$ s, as soon as the ports of the nodes running the STP switch into forwarding state. When the ping messages start to flow, the preferred route from $P_1$ to $P_2$ is 3 - 4 - 0 - 9 - 10. When a failure occurs at node 9 (we remark that the BPDU at $t = 30$ s is not delivered), the ping messages do not reach $P_2$ and BPDUs do not reach switch 2 on that side of the loop for 6 s. Then, at $t = 34$ s switch 2 sends a TCN message.[4] This message forces a change of state in the ports of the switch, so that for a period of time equal to $2FD$ switch 2 is no longer able to relay the received data. As soon as its ports are turned into forwarding state at $t = 42$ s, the new route for the ping message becomes 3 - 4 - 0 - 7 - 8 - 2 - 10. Once the failure is recovered, as soon as a BPDU propagates through the previously failed link, the information stored in the switch 2 is changed and the network converges again since switch 2 turns a port into blocking state. Referring to the failure recovery at $T^{**} = 90$ s (we remark that the BPDU at $t = 90$ s is delivered correctly), the ping message at $t = 91$ s is routed through the path 3 - 4 - 0 - 9 - 10.

Recalling the considerations presented in Subsection 3.2. and equation (3), the convergence is guaranteed when the condition $m_{\text{age}} < MA$ holds in each switch in the network, i.e., when the information conveyed by a BDPU and received by a switch is not too old. Considering two switches running STP connected through a set of transparent switches, the $m_{\text{age}}$ value conveyed by a BPDU, received by one of the STP-enabled switches, is incremented only by the STP-enabled switches. In particular, since $FD$ is kept small, the STP convergence in a network with "transparent" nodes is faster than in an STP-based network with the same number of switches. In fact, in the latter case, the convergence might be guaranteed using a higher $MA$ and, therefore, a higher $FD$. In addition, keeping $MA$ small, the network reaction capability is faster, according to the considerations carried out in Subsection 3.2..

The only limitation to the convergence in the considered extended network scenarios is given by the fact that the number of transparent switches crossed by a BPDU introduces a delay on this BPDU such that the information stored by the receiving BPDU is no longer valid. In fact, as mentioned in Subsection 2.1., the information stored in a switch is valid for a time interval equal to $MA - m_{\text{age}}$, after which, if no updating BPDUs are received, the switch is reset and the convergence is no longer guaranteed. However, this delay is not critical, especially for realistic networks, where the number of switches ranges between 3 and 30. In fact, the delay introduced by each switch is small, as shown in Fig. 10, where, referring to the topology presented in Fig. 9, we have shown the queueing delay on the links between (i) node 3 and node 4 and (ii) node 9 and node 10, respectively. In particular, we focus on the case of the failure of node 9 after $t = 120$ s and its recovery at $t = 360$ s. Approximatively, the delay is in the order of $10^{-5}$ s. The links from node 3 to node 4 and from node 10 to node 9 experience a higher delay since node 3 and node 10 have to manage the reception of packets, i.e., BDPUs and ping messages, from a larger number of ports. After the node failure, node 9 stops transmitting packets, so that its queueing delay is 0. On the other hand, the delay on the link from node 10 to node 9 reduces, since node 10 relays only BPDUs to node 9, while the ping messages are routed towards switch number 2. The other observed nodes, instead, basically maintain their delay, but for fluctuations due

[4] In this case, the failure affects a transparent switch, which does not take part to the STP. Therefore, the $m_{\text{age}}$ of a BPDU received by the switch 2 is equal to 0 and its stored information are valid for $MA$ seconds.

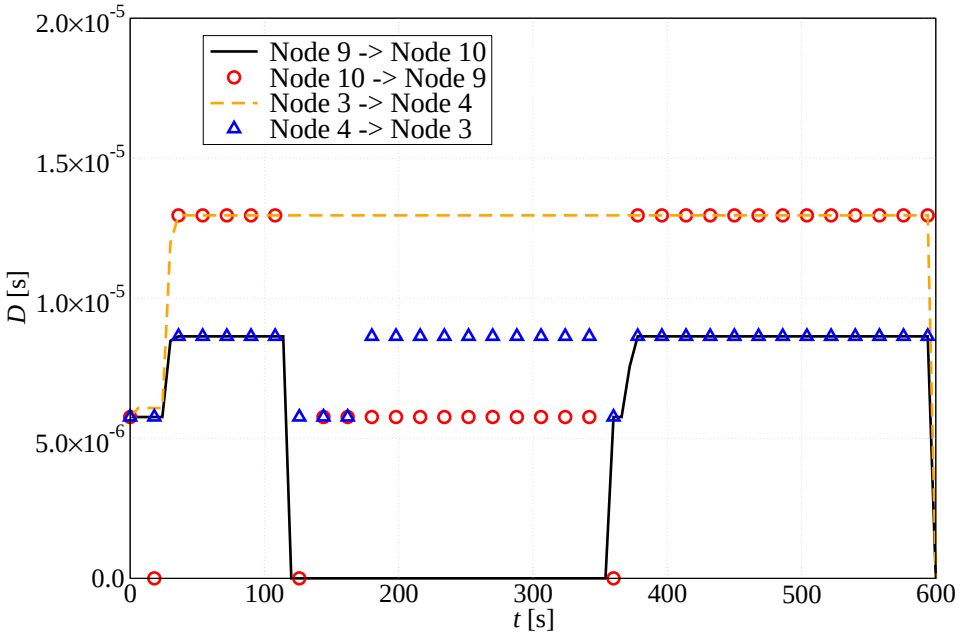


Figure 10. Delay over links between (i) node 3 and node 4 and (ii) node 9 and node 10, both considering the failure of node 9 after $t = 120$ s and its recovery at $t = 360$ s.

| | FE | Switch 100 | Switch 10 | DSL |
|---|---|---|---|---|
| Port speed (Kbps) | 100,000 | 100,000 | 10,000 | $N \cdot 64$ |
| Port delay ($\mu$s) | 0 | 500 | 500 | 300 |

Table 2. Fast Ethernet (FE), Switch 100, Switch 10, and DSL ports specifications.

to transitory traffic following the node failure and recovery. After that node 9 recovers, the delay at the nodes return to the initial values.

For sake of completeness, we now present the crossing delay results given by the use of different port speeds. We considered four different scenarios, listed in Table 2: (i) Fast Ethernet (FE) port, (ii) 100 Mbit switch port, (iii) 10 Mbit switch port, and (iv) Digital Subscriber Line (DSL) port. In the latter case, the transmission speed is expressed as a function of the parameter $N$ according to the relation

$$\text{Port speed}_{\text{sDSL}} = N \cdot 64 \text{ (dimension: [Kbps])}.$$

In Table 3, the crossing delays results are shown related to the transmission of a ping message at $t = 10$ s. The crossing delay depends on both the port speed and the port delay. However, in all scenarios the crossing delay is low, so that it can be concluded that the use of transparent switches does not affect the convergence of the STP.

We finally introduce the OSPF protocol and the HSRP in the network and evaluate their impact on the STP performance. The reference network topology is shown in Fig. 11. The nodes R0, R1, R2 are routers with both L2 (STP) and L3 (HSRP and OSPF) capabilities.

| | FE | Switch 100 | Switch 10 | DSL ($N = 1$) |
|---|---|---|---|---|
| Reception instant in 3 ($T_{r3}$) (s) | 10.008692 | 10.021692 | 10.022233 | 10.119993 |
| Reception instant in 4 ($T_{r4}$) (s) | 10.009159 | 10.023159 | 10.023817 | 10.141297 |
| Transit time ($T_4 - T_3$) (s) | 0.000467 | 0.001467 | 0.001584 | 0.021304 |
| Difference from the case with FE (s) | | 0.001000 | 0.001117 | 0.020837 |

Table 3. Comparison between transmission instants of FE, Switch 100, Switch 10, and DSL ports.

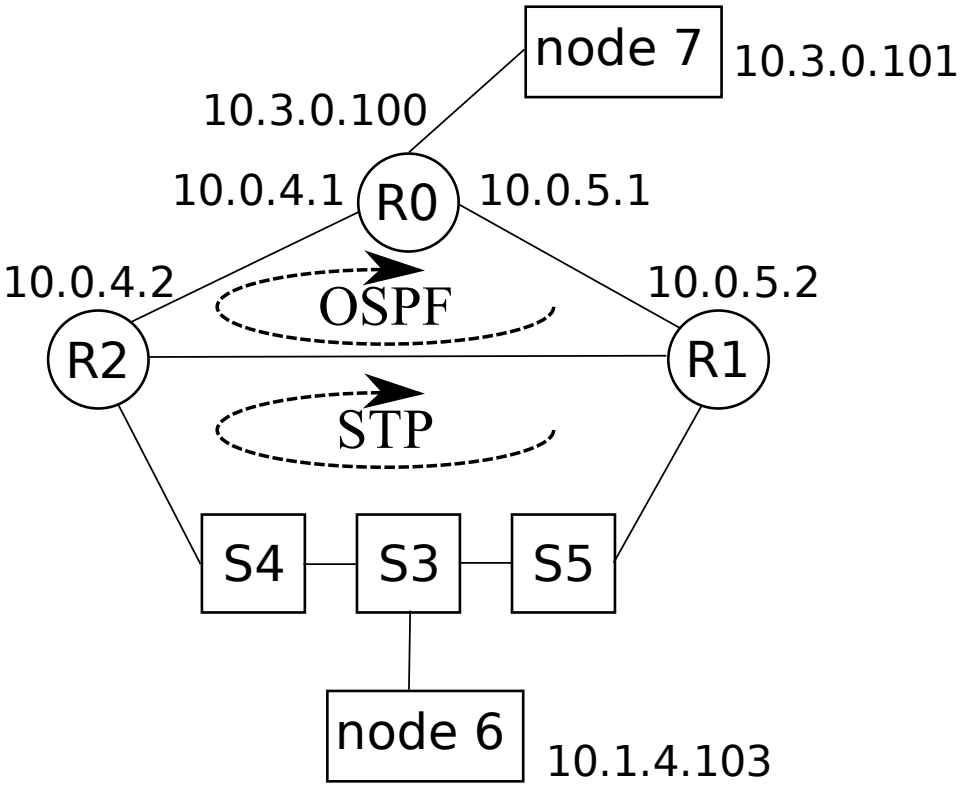


Figure 11. Considered network topology with the use of STP and OSPF.

The layer 3 has been introduced in order to evaluate the performance of an Ethernet network in a realistic scenario. The nodes S3, S4, S5 are nodes which run only the STP, i.e., they are CS2948 switches, whereas the node 6 and 7 are generic workstations used to transmit and receive ping messages—in particular, these messages are expedient to verify correct network functioning. In particular, in the lower part of the figure, i.e., between nodes S3, S4, S5 R0, and R1, the STP is running, whereas in the higher part of the network, the OSPF protocol is running. The HSRP is enabled on both routers R0 and R1: R0 is configured as active router, whereas R1 is the backup router.

The network is able to forward the ping messages to destination only after 65 s. This is due to the sum of two delay contributions: (i) the random delay (uniformly distributed between 5 and 10 s) at the beginning of the set-up phase of the OSPF protocol and (ii) the delay given by the transmissions of Hello messages (every 10 s) which notify the other routers of the active network topology.[5] In particular, since R2 is set as active router according to the HSRP, the ping message flows from node 6 to node 7 through S3-S4-R2-R0.

Once a failure occurs on the link between R2 and R0, the ping messages are lost for a time interval equal to 40 s, i.e., the router dead interval. However, after R2 realizes that the selected router is no longer active, it starts forwarding the ping messages along an alternative path, i.e., through R1, still guaranteeing the reachability of the destination.

## 5. Design Guidelines

The performance analysis presented in the previous sections is useful to derive a few design guidelines for STP-based networks. Recalling the results in Figs. 3-6, one can note that the maximum number of switches that a network can tolerate, while implementing STP successfully, is limited. In particular, this number depends on the kernel used by the switches in the network. In fact, the use of the Linux kernel, given a specific value of the $MA$, allows to create a network with the largest number of switches. For example, considering $MA = 20$ s, the maximum number of nodes using the Linux kernel is 40, whereas the maximum number of nodes using the Cisco kernel is 20. However, the use of the same $MA$ has a different impact on the network reaction capability. In fact, as explained in Section 3.1., the Linux kernel introduces a lower $m_{\text{age}}$, so that the information stored in the switches is valid for a longer period of time from the reception of the last BPDU.

This consideration can be reversed in order to determine the STP parameters that, given a fixed number of nodes $N$, guarantee fastest convergence and reaction against network failures. For example, given a network with $N = 20$, in the case of the Cisco kernel, the minimum value of the max age is 20 s, whereas in the case of Linux kernel the minimum value is 10 s. In the latter case, the network capability reaction is still under analysis. In fact, with the Linux kernel $MA$ is half of that with the Cisco kernel. However, in the former case the $m_{\text{age}}$ increases slowlier than in the latter case, and its impact on the recovery capability needs to be investigated.

In order to have correct network operations, the best solution is configuring the parameters considering an open-chain network with $N$ nodes. In this way, the convergence is

[5] These messages allow to create a forwarding table which contains the "route" from a source to a destination.

guaranteed for every network where the distance between any couple of nodes is lower than $N$ hops. This solution is less efficient in terms of convergence speed, but it assures a high network reliability.

In order to overcome the intrinsic limitations on the number of nodes of the STP, speed the convergence, and increase the failure reaction capability of a network, an appealing solution is the use of "transparent" switches. This solution, in fact, allows to extend the network dimension, still guaranteeing the use of VLAN tagging.

The considerations presented above are still valid when the OSPF protocol and the HSRP are used jointly with the STP. In particular, since the HSRP and OSPF introduce additional delays, the convergence of the network is guaranteed, but for a longer waiting time. In addition, since the routers detect a link or node failure only after some time (for instance, 40 s in the considered illustrative example in Fig. 11), it takes more time to react to the failure and a larger number of data will be lost, with respect to networks which run only STP.

## 6. Conclusion

In this work, we have first analyzed, through the Opnet simulator, the performance of an Ethernet network running the STP, with switches equipped with either Cisco or Linux kernels. For each type of device, optimizing rules for setting the STP parameters has been derived, in order to speed network convergence. In addition, the presence of VLANs has been taken into account and the STP parameters have been optimized also in this scenario. Our simulation results have also been confirmed by experimental results. In order to provide a complete set of rules for network configuration, an open-chain network has been considered. From our analysis, it turns out that a proper configuration of STP parameters should guarantee that the STP works with every possible network configuration such that the distance (in terms of number of hops) between any couple of nodes is smaller than $N$, even if this leads to a longer convergence time and reduced reaction capability. In addition, the extension of the network through "transparent" switches has been considered as a mean to overcome intrinsic limitations of the STP. The use of this type of switches allows to significantly extend the the maximum acceptable dimension (in terms of number of nodes) of an STP-based network. Finally, the impact of failures in a realistic network, running both L2 and L3 protocols, has been evaluated. In this case, it has been shown that STP, OSPF protocol, and HSRP can successfully coexist, even if a reduced reaction capability to failures in layer 3 links and a longer initial delay before the beginning of ping messages transmissions must be taken into account.

## Acknowledgments

We acknowledge useful discussions with and continuous support from A. Cavagna and A. Pasino (Selta spa).

## References

[1] "IEEE standards for local area networks: supplements to carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications," *ANSI/IEEE Std 802.3a,b,c, and e-1988*, 1987.

[2] Radia Perlman, "An algorithm for distributed computation of a spanning tree in an extended LAN," *SIGCOMM Comput. Commun. Rev.*, vol. 15, no. 4, pp. 44–53, September 1985.

[3] M. Wadekar, "Enhanced Ethernet for Data Center: Reliable, Channelized and Robust," *15th IEEE Workshop on Local & Metropolitan Area Networks* (LANMAN'07)., pp. 65–71, June 2007.

[4] K. Segaric, P. Knezevic, and B. Blaskovic, "An approach to build stable spanning tree topology," *Int. Conf. on Trends in Communications* (EUROCON'01*)*, vol. 2, pp. 400–403, July 2001.

[5] K.H. Yeung, F. Yan, and C. Leung, "Improving Network Infrastructure Security by Partitioning Networks Running Spanning Tree Protocol," *Int. Conf. on Internet Surveillance and Protection (ICISP'06)*, August 2006, 4 pages.

[6] Opnet website, "`http://www.opnet.com`," .

[7] W. Stallings, "IEEE 802.11: wireless LANs from a to n," *IT Professional*, vol. 6, no. 5, pp. 32–37, 2004.

[8] "IEEE Standard for Information technology- Telecommunications and information exchange between systems- Local and metropolitan area networks- Common specifications Part 3: Media Access Control (MAC) Bridges," *ANSI/IEEE Std 802.1D, 1998 Edition*, pp. i–355, 1998.

[9] "IEEE standard for local and metropolitan area networks - common specification. Part 3: media access control (MAC) bridges - amendment 2: rapid reconfiguration," *IEEE Std 802.1W-2001*, 2001.

[10] "IEEE Standards for Local and metropolitan area networks - Virtual Bridged Local Area Networks - Amendment 3: Multiple Spanning Trees," *IEEE Std 802.1S-2002 (Amendment to IEEE Std 802.1Q, 1998 Edition)*, 2002.

[11] Understanding Spanning tree protocol, Cisco website, "`http://www.cisco.com/univercd/cc/td/doc/product/rtrmgmt/sw_ntman/cwsimain/cwsi2/cwsiug2/vlan2/stpapp.htm`," .

[12] Understanding and Tuning Spanning tree protocol, Cisco website, "`http://www.cisco.com/en/US/tech/tk389/tk621/technologies_tech_note09186a0080094954.shtml`," .

[13] "IEEE standards for local and metropolitan area networks. Virtual bridged local area networks," *IEEE Std 802.1Q, 2003 Edition (Incorporates IEEE Std 802.1Q-1998, IEEE Std 802.1u-2001, IEEE Std 802.1v-2001, and IEEE Std 802.1s-2002)*, 2003.

[14] J. Moy, "RFC2328: OSPF Version 2," *RFC Editor United States*, 1998.

[15] E. W. Dijkstra, "A note on two problems in connexion with graphs," *numerische Mathematik*, vol. 1, no. 1, pp. 269–271, 1959.

[16] T. Li, B. Cole, P. Morton, and D. Li, "RFC 2281: Cisco Hot Standby Router Protocol," *RFC Editor United States*, 1998.

[17] The Linux Kernel Archives, `http://www.kernel.org`.