JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 34, 225-242 (2018)
DOI: 10.6688/JISE.2018.34.1.14

A Novel Algorithm for Loop Prevention and Fast Recovery (LPFR) in Ethernet Rings

SAAD ALLAWI NSAIF[2], NGUYEN XUAN TIEN[1] AND JONG MYUNG RHEE[1]
[1]*Department of Information and Communications Engineering*
Myongji University
Yongin, 17058 Korea
[2]*Cisco Systems, Inc., USA*
E-mail: {saad.allawi1; nxtien}@gmail.com; jmr77@mju.ac.kr

Traditional Ethernet networks use the rapid spanning tree protocol (RSTP) or the enhanced RSTP (eRSTP) to ensure a loop-free topology and provide redundant links as backup paths in case an active link has failed. However, when a failure occurs, the RSTP/eRSTP require a significant amount of reconfiguration time in order to find an alternative path. The RSTP/eRSTP are also limited by the number of nodes in a ring network, and their performance degrade when the number of nodes increases; the media redundancy protocol (MRP), which is used in industrial networks, has the same issues. In this paper, we introduce a new algorithm, called loop prevention and fast recovery (LPFR), which can be applied to Ethernet ring networks to provide a loop-free topology. When a failure occurs, LPFR requires only a very short amount of time to switch to an alternative path. LPFR needs a maximum of 6.1 ms for recovery time, versus < 100 ms for the RSTP/eRSTP in a ring size of 20 nodes and < 200 ms for the MRP in a ring size of 50 nodes. The LPFR offers a reduction in recovery time of up to 93.9 % compared to the RSTP/eRSTP, and 96.9 % compared to the MRP. However, in most cases, LPFR requires zero recovery time for such switching. In addition, in most situations, no frames are lost when a node switches to an alternative path. Unlike the RSTP/eRSTP and MRP, LPFR provides a better way for distributing the sent traffic among network links, which in turn improves network performance, reduces the probability of bottleneck occurrences, and reduces the frame latency between the source and the destination. This is because LPFR always sends the frame through the fastest path. The LPFR algorithm can run on a wide variety of ring sizes, which is not the case for the RSTP/eRSTP and MRP. Finally, the LPFR will use the Ethernet standard frame layout without any modifications or changes. This will allow all of the standard Ethernet devices and networks to be directly integrated into the LPFR networks without any proxies.

Keywords: LPFR, RSTP, eRSTP, path redundancy, loop prevention, fast recovery

1. INTRODUCTION

The Ethernet, standardized by the Institute of Electrical and Electronics Engineers (IEEE) in IEEE 802.3 [1], is not capable of supporting a fault-tolerant network. The total avoidance of loops is a basic requirement for all Ethernet networks. Any loop would result in frames circulating forever, thus flooding the network. Because the Ethernet standard does not support fault tolerance, various Ethernet redundancy protocols have been developed and standardized by IEC, such as the rapid spanning tree protocol (RSTP) [2], the media redundancy protocol (MRP) [3], the cross-network redundancy protocol (CRP) [4], the beacon redundancy protocol (BRP) [5], the parallel redundancy protocol (PRP) [6], the high-availability seamless redundancy (HSR) protocol [6], and resilient

Received April 3, 2016; revised June 24, 2016; accepted July 16, 2016.
Communicated by Xiaohong Jiang.

packet ring (RPR) protocol [7].

Generally, because of the fault-tolerant feature of these protocols, industrial automation systems and smart grid networks have adopted several of them to achieve the fault-tolerant condition that required by many control messages, such as the generic object oriented substation events (GOOSE) message that needs a maximum time-out of 4 ms [8]. However, the RSTP and the HSR protocol can be applied to ring, connected ring, and mesh topologies. Other topologies are available such as the start, but some of the network terminal devices will lose the second copy from each sent frame and consequently lose their zero recovery time capability in case of the HSR protocol. The RSTP can be applied to arbitrary mesh topologies, however, it does not offer the zero recovery time feature. On the other hand, Siemens has developed a new version of the RSTP called the enhanced rapid spanning tree protocol (eRSTP) [9], which runs in a larger ring network that is beyond the capacity of the RSTP; however, eRSTP still has the same limitations as the RSTP regarding the network recovery time. In addition, the eRSTP is not a compatible protocol that can run with standard RSTP equipment.

RSTP and eRSTP implement a distributed computation of a tree based on path costs and priorities. This tree is the active topology, which is established by blocking the switch ports. If a failure occurs, the RSTP/eRSTP usually requires an upper bound of 100 ms for a ring consisting of 20 switches [9]. This is because once a link or a switch fails, the network undergoes reconfiguration to rebuild the logical paths. This makes the RSTP/eRSTP unsuitable for some message timeout requirements in time-critical applications.

The MRP is used only in a ring topology. In this protocol type, a dedicated node called the ring manager blocks one of its ring ports in order to establish a line as the active topology. If a failure occurs, this line breaks into two isolated lines that are reconnected by de-blocking the previous blocked port. The reconfiguration time, which is in the 200 ms range for a ring size of 50 nodes [10, 11], can be guaranteed. The CRP has a long reconfiguration time that might reach 1 second if the network is large. The BRP is suitable for a star topology but not for a ring topology. In contrast, the PRP does not change the active topology. It operates on two independent networks. Each frame is replicated on the sending node and transmitted over both networks. The receiving node processes the frame that arrives first and discards the second copy. The HSR protocol is almost identical to the PRP, and it is usually applied to a ring topology in order to provide two frame copies, one from each direction. However, compared to other protocols, in the HSR protocol, the available network bandwidth is halved because two copies of every frame are transmitted over the network. The PRP and the HSR protocol both provide zero recovery time, but the PRP requires a duplicated network infrastructure and the HSR protocol generates unnecessary redundant frames. Moreover, the implementation of both the PRP and the HSR protocol is complicated, and their frame format is different from the Ethernet standard (IEEE 802.3) format because of the additional header. In contrast, in ring topologies, the RPR protocol has a restoration or a recovery time of 50 ms and uses two ringlets, one in each direction, it also uses different frame format compared to the Ethernet standard frame layout. However, recently, in the IEEE Plenary meeting that held in Macau 13-18 March 2016, IEEE announced, that IEEE 802.17 protocol is transferred to inactive status because many manufactures and companies have drawn their equipment and systems from the markets [12]. The other protocols also using different frame format which make off-the-shelf Ethernet devices not able to be integrated to the networks of these tolerant protocols, except the RSTP and eRSTP.

In this paper, we propose a novel algorithm, called loop prevention and fast recovery (LPFR). This algorithm uses the Ethernet standard (IEEE 802.3) frame layout without any modification or changes. Consequently, this allows to all off-the-shelf Ethernet devices to be directly integrated into the LPFR network. However, the LPFR will not provide seamless redundancy with zero-recovery time as it is in HSR and PRP but it offers a recovery time faster than the RSTP/eRSTP, MRP, and RPR. The LPFR also does not need to use two ringlets as it is in RPR, it only needs one full-duplex ringlet. From the other side, the RSTP/eRSTP and MRP protocols always disable one of the ring ports to avoid the looping issue, which causes some frames to take the longest path, even if the destination node is only one hop away from their source node. This may cause some congestion or bottlenecking in the ring because not all of the links are fully utilized. In contrary, the LPFR will not disable any port and it will distribute the sent traffic between both directions of the ring. In this way, the LPFR will deliver the sent frames quickly because it uses the fastest path to each destination node.

We have introduced the LPFR algorithm in a previous paper [13], but in this paper we add more analysis and demonstrate that the algorithm performs better during the recovery procedure.

The rest of the paper is organized as follows. In Section 2, we briefly introduce the concept, setup procedure, and operation of our proposed LPFR algorithm. In Section 3, the LPFR algorithm's monitoring and recovery procedure is described. Section 4 presents a procedure for avoiding looping storms under double link failure. In Section 5, we carry out a performance analysis, while Section 6 provides the numerical results of the performance analysis. Section 7 presents the simulation analysis. Finally, our conclusions and recommendations for future work are given in Section 8.

2. THE LPFR ALGORITHM

2.1 The LPFR Concept

Our LPFR algorithm is used in Ethernet network rings to ensure a loop-free network and to provide path redundancy with faster reconfiguration and recovery times than the RSTP/eRSTP, MRP, and RPR. The LPFR algorithm is based on the idea that each node in a ring topology will select one of its two ports as the fastest port leading to each destination node. This port is called the primary port (PP); the alternative port is called the secondary port (SP) and will be used if the PP of that destination fails or its path no longer leads to the required destination. In this way, the LPFR algorithm will make each node use the fastest path to connect it to each destination node in the ring. In other words, if a destination node is located one hop to the right of a source node, the source node will select the right port as the PP for that destination, whereas the left port will be the SP to be used if the PP or its path fails. In the same way, the source node will select the left port as the PP if the destination node is located to the left of the source node.

Basically, the LPFR is suitable for Ethernet nodes, such as an intelligent electronic device (IED). IED is a designation which describes various microprocessor-based protection, control, metering and monitoring devices. Compared with traditional solutions at the functional level such as individual meter or protective relay, IED functions are much more enhanced. IEDs [14] receive data from sensors and power equipment, and they can

issue control commands, such as to trip circuit breakers if they sense voltage, current, or frequency anomalies, or to raise/lower voltage levels in order to maintain the desired level. Common types of IEDs include protective relaying devices, on-load tap changer controllers, circuit breaker controllers, capacitor bank switches, recloser controllers, and voltage regulators. IEDs have two ports that share the same media access control (MAC) and Internet protocol addresses, which use the Ethernet standard. This allows the address management protocols, such as the address resolution protocol, to operate as usual without modification, which simplifies network engineering.

2.2 LPFR Setup Procedure

The LPFR algorithm uses five types of special frame. These frames will be used during the setup procedure, failure and recovery only and not for sending data. The five frames are called the port selection (PS), port down (PD), port up (PU), network monitoring status (NMS), and network fail (NF) frames.

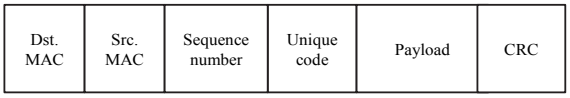

Fig. 1. Frames layouts of the LPFR special frames.

It can be seen in the special frame layout in Fig. 1 that the layout comprises a destination MAC address field, which always includes a broadcast MAC address and a source MAC address that indicates the generated node for the sent special frame. The layout also has a sequence number that increases each time a source node sends a special frame. However, each special frame for a source node will have a unique sequence number. This number will help the received node to distinguish whether the received special frame is new or has been received earlier. The unique code will be used to distinguish among the five special frames. Moreover, the payload will be a dummy to adjust the minimum size of the sent frame according to the requirements of the Ethernet standard (IEEE 802.3). The payload field also can be used for future development. Finally, the cyclic redundancy code (CRC) is an error-detecting code that is commonly used in communication networks and storage devices to detect accidental changes to raw data.

The setup procedure of LPFR algorithm can be summarized as follows, where the PS frame is used to select a PP that will quickly lead to each destination:

a) At the beginning of the network run, each node broadcasts a PS frame from both of its ports.
b) Each node reads all the fastest PS frame copies per each sending node that will pass through it. The node will read the source MAC address, the sequence number, and the port number from which the fastest PS frame per each sending node is delivered.
c) This information will be tabulated into a table called a node primary port (NPP) table. In this table, each node identifies which port has first delivered each PS frame copy per each sending node. Consequently, each node will know which of its ports is fastest in terms of reaching each neighbor node.
d) Each node will ignore the second PS frame copy per each sending node that will later be passed through it from the second port. However, after each PS frame copy

is read, it will be forwarded to the opposite direction until it reaches the source node that is going to delete it.

e) Any additional Ethernet node that might be added to the ring during the network run will broadcast a PS frame, and the other nodes will consequently understand that there is a new node, because there is a new PS frame with a new source MAC address that is not listed in the NPP table. Therefore, the nodes will update their NPP tables and then broadcast their PS frames to enable the new node to build its NPP table.

At this point, the setup procedure of the first approach is complete, and all the nodes are ready to send their data frames to neighbor nodes on the basis of their NPP tables.

2.3 LPFR Operation

According to the network traffic type and the network status, the LPFR algorithm will operate as described in the following sections. First, we consider the example network shown in Fig. 2.

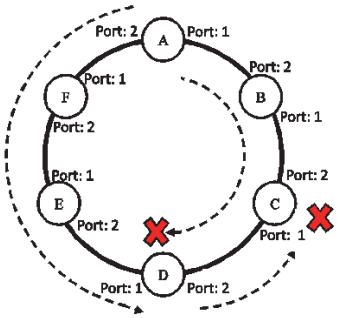

Fig. 2. Operation of LPFR under multi/broadcast traffic type.

2.3.1 Unicast traffic under the failure-free case

In Fig. 2, assume node A has an NPP table like that shown in Table 1. When node A needs to send frames to node B, it will first check its NPP table to determine which port is the PP for the destination of node B. In this case, port 1 is the PP; therefore, node A will send the frames into port 1. The same steps will be followed by node A when it needs to send frames to node F. This time, node A will use port 2 as the PP, as shown in Table 1. As a result of this form of traffic distribution, once right and once left, the network traffic will be distributed almost equally among all the network links.

Table 1. Node primary port (NPP) table for node A.

Destination MAC address	Sequence number	Primary port number
Node B	XX	1
Node C	XX	1
Node D	XX	2
Node E	XX	2
Node F	XX	2

XX: Any sequence number

2.3.2 Multi/broadcast traffic under the failure-free case

In this situation, the LPFR algorithm will operate as follows:

a) If node A broadcasts a frame, then the first copy will travel through port 2 toward nodes F, E, and D, and each of these nodes will take a copy from the sent frame, then forward it from the opposite direction to the next node. Then, node D will forward the copy to node C. The second copy will travel through port 1 toward nodes B and C, and both of those nodes will take a copy and forward it. Later, node C will forward the copy to node D.
b) Node D will receive two frame copies, one from each direction; however, according to node D's NPP table, it is assumed that the PP of node D for the destination of node A is port 1. Therefore, node D will take a copy from that frame and then forward it to node C through port 2. Later, when node D receives the second copy from port 2, it will read the frame's source MAC address, which is node A in this case, and node D will then delete that frame after checking its NPP table and determining that the PP for the destination of node A is port 1, not port 2.
c) Suppose that node C's NPP table shows that the PP of node C toward node A is port 2; in that case, node C will take a copy and then forward it to node D. As shown in Fig. 2, node C will delete the second copy that is delivered to it through port 1 as long as its source MAC address is node A. This network behavior will continue as explained above as long as the network is failure free.

2.3.3 Unicast traffic under the failure case

For the network shown in Fig. 2, assume a physical failure for the link between nodes E and D occurs. In this case, nodes D and E will detect that physical failure within 4-6 ms [9], and the following procedure will then take place:

a) In its NPP table, node D will switch all the PPs using port 1 to port 2.
b) Node D will then broadcast a PD frame to inform the network nodes that a physical failure is occurred.
c) Node E will perform the same steps undertaken by node D, but instead it will switch all the PPs using port 2 to port 1.
d) As soon as the remaining nodes determine that a PD frame is travelling inside the network, they will broadcast their PS frames in order to make the network nodes reselect the PPs for each destination node. In other words, the LPFR algorithm will be restarted. The nodes will treat any PD frame as a PS frame, read its information, and then insert it into their NPP tables.

However, during this time, the nodes will keep sending normal traffic without any interruption. For example, if node C sends frames to node E through node D, and node D has detected the failure before receiving node C's frames, node D will send these frames back to node C, and the following procedure will take place:

- Node C will read the source and destination MAC addresses. It will then realize that this frame was sent from itself to node E. Therefore, node C will understand that the path through port 1 to node E no longer exists.

- Node C will append a special tag (1 byte in size) to the next frame (only to one frame) frame) going to node E. In other words, if node C still has a stream of data that needs to be sent to node E, after recognizing the failure situation, node C will append a special tag to only a single frame from the sent stream. Next, node C will send the appended frame toward node E through port 1.
- After that, node C will not send any more frames toward node E through port 1.
- Node C will switch the PP of the destination node E to port 2.
- As soon as node C starts receiving its frames back (the previously sent frames and the appended frame) from port 1's direction because node D sent them back due to the link failure, node C will forward these frames through port 2 toward node E.
- However, node C will know that it receives back all of its frames when it observes the appended tag in the last received frame, or in other words, when node C receives the appended frame back. At that time, node C will remove that tag, and send the frame through port 2 toward node E.
- Eventually, node C will continue to send frames to node E through port 2 if it exists, and it will also redirect all future frames destined for node E through port 2.

However, node C used the appended tag approach to know when it will receive back all of its frames in order to redirect them with the same sequence and then it can continue the sending process for the remaining frames if they exist. Consequently, node C has avoided sending out-of-sequence frames to node E. Note that these steps are done before receiving the PD frame of node E through port 2, however, during that process, if node C receives the PD frame of node E, nothing will be changed in the steps of this process because node C already changed the PP for the destination node of E to port 2 and the received PD is also delivered through port 2.

However, if failure occurs when node D is sending frames to node E, then node D will lose frames for an interval of 4-6 ms or until it detects the failure.

It is important to note that in this type of failure case, nodes D and E will not need to check their NPP tables before sending their traffic, because they know that port 1 for node D and port 2 for node E are down due to the physical failure.

2.3.4 Multi/broadcast traffic under the failure case

In Fig. 2, if the link between nodes E and F is failed, and node A broadcasts frames, node E will lose the fastest copies through port 1, but it will keep receiving the second copies originating from node D as long as node E has detected the physical link failure, broadcasted its PD frame, and changed all the PPs of all its destination nodes. In other words, node E will not delete the second copy of the sent frames, because the transmission of the first copy, which was supposed to be delivered through node F, has stopped and node E has detected that event.

It is important to mention before concluding this section that with the unicast traffic type, the node will use its NPP table to select the proper PP only for sending its frames; it will not do so to receive frames from other nodes. This is because there is only one copy from each sent unicast frame; because it is not duplicated, the receiving node does not need to consider from where it will receive the frames. In another case, if a node, such as node F in Fig. 2, has received a frame from node A going to node E, node F will

not check its NPP, and it will forward the received frame from the opposite direction only if the received frame has a destination MAC other than node F. However, the situation is different in a multi/broadcast case because there are duplicated frame copies for each sent frame – one copy from each direction; therefore, the received node will only receive from one direction and delete the copies that will be delivered from the second direction. Therefore, in this case, the received node will check its NPP table to identify the PP that is associated with the sending node. Consequently, the received node will only consume the frames of the PP and delete the other copies of the SP.

3. MONITORING AND RECOVERY PROCEDURE

As soon as the failed link is repaired and its adjacent ports are back up again, both of the nodes that are connected by that link will broadcast PU frames, which the other nodes will treat these two PU frames similar to PS frame. In turn, all the remaining nodes will broadcast their PS frames, and update their NPP tables. In other words, LPFR algorithm will be restarted.

The LPFR algorithm also has the ability to detect nonphysical failures. This is achieved by sending an NMS frame. However, because the NPP table contains the MAC addresses for all the nodes, each node will know which node has the lowest MAC address value. That node is called the LPFR root node. The root node will perform the following procedures:

- It will periodically broadcast an NMS frame to check whether the network has a failure. This frame will be sent through both ports of the root node, and it will be received by the opposite ports.
- The NMS frame will be sent at 100 ms intervals, as we assumed. However, if the root node does not receive both of the NMS frame copies within 90 ms, it will know that there is a failure in the network. In that case, the root node will broadcast an NF frame, which all the nodes will treat like a PS frame. In turn, all the remaining nodes will broadcast their PS frames and update their NPP tables.
- However, if the root node receives only one NMS frame, then the node will send another NMS frame and wait another 90 ms. If both copies are received, then the root node will know the network has no issues; otherwise, the root node will broadcast an NF frame, and the remaining nodes will broadcast their PS frames to update their NPP tables.
- If any of the network nodes does not receive a new NMS frame within 100 ms after receiving the last NMS frame, that node will broadcast an NF frame, and the remaining nodes will broadcast their PS frames to update their NPP tables.
- Under the failure case, the root node will continue to periodically send the NMS frame without sending an NF frame, because the node sent that frame earlier, that is, after the failure was discovered.
- Under a physical link failure, the root node will receive two PD frame copies from the two nodes that detected the failure, one from each direction. Therefore, it will not need to broadcast its NF frame, but will continue to periodically send the NMS frame until the failure is repaired and it receives PU frames from both directions.

4. AVOIDING A LOOPING STORM UNDER DOUBLE LINK FAILURE

The LPFR algorithm will deal with the following case to avoid creating a looping storm, as described below. Assume that in the network shown in Fig. 3, node C is sending frames to node F through port 2, which is the PP for this destination. Assume that during this period, the link between nodes F and A undergoes a physical failure and node A detects the failure. In this case, node A will send these frames back to node C, which will switch its PP for that destination to port 1 and then forward these frames toward node F. Assume that, later, the second link of node F, which connects it with node E, also fails and node E detects the failure. Node E will send the frames back to node C, which in turn will delete them because they have passed through it twice, once when they returned from node A and once when they returned from node E. This means that there is no path to node F from both directions and that if those frames keep going back and forth, they will generate a looping storm. For this reason, node C deletes them and it will not send any future frames to node F. However, during that period, node C will know that the link between nodes A and F is still down, because nodes A and E have not yet detected the link restoration and sent their PU frames.

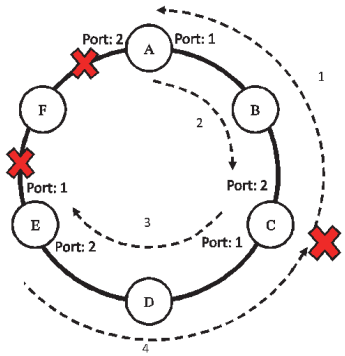

Fig. 3. Loop avoidance in LPFR when double link failure occurs.

5. LPFR PERFORMANCE ANALYSIS

In this section, the LPFR's performance is discussed from the perspectives of frame distribution, frame latency, and recovery time.

5.1 Frame Distribution

Although the main purpose of the LPFR algorithm is to provide path redundancy with fast recovery time and without a looping issue, the LPFR offers better distribution for the sent frames among the network links compared to the RSTP/eRSTP and MRP protocols. This is because LPFR uses all the network links during each sending process. As explained above and as shown in Fig. 2, node A sends frames to node B through port 1, and it sends frames to node F through port 2; thus, the sending node has selected the fastest path to each destination, according to the selection of PS frames or according to the NPP table. This procedure is performed by each node. Consequently, LPFR provides

a type of better distribution of network traffic among the network links that is not provided by the RSTP/eRSTP or MRP, because such protocols must disable a port or cut the ring to avoid the looping issue, as a result of which the sent frames always follow the longest path and consequently increase the number of frames per each link segment.

5.2 Frame Latency

Let us consider the network shown in Fig. 4 as an example for performance analysis from the frame-latency perspective. Assume that node 5 sends a unicast frame to node 10. The expressions presented below represent the frame latency of that frame under the cases described in the subsections below.

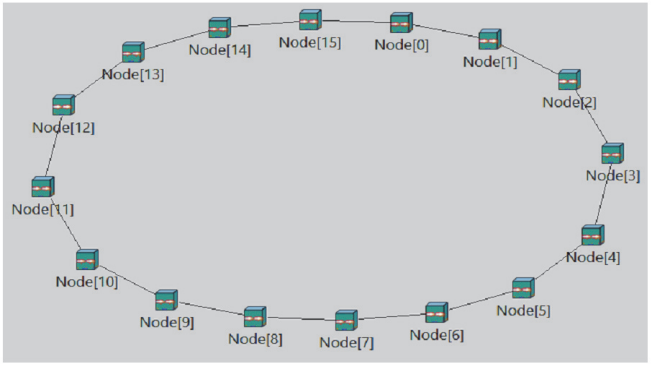

Fig. 4. An example network used in the performance analysis.

5.2.1 Unicast traffic under the failure-free case

If all the nodes and links are working properly without any errors or failures, then the frame latency for the above scenario can be determined as follows, assuming that the latency will be determined from the time the frame leaves the source node until the time it is received by the destination node:

$$t = \tau_r + \tau_p + \tau_q + \tau_n$$

where t is the frame latency under the failure-free unicast traffic case, τ_r is the transmission delay time, τ_p is the propagation delay time that assumed to be zero because the network example has short link lengths, τ_q is the queuing delay time, assume to be zero, and τ_n is the node processing delay time. The above expression can be written as follows:

$$t = l\left(\frac{f}{\beta}\right) + \left[(n-1)\left(\frac{f}{\beta}\right)\right] \tag{1}$$

where l is the number of failure-free links between the source and destination nodes from the PP's direction, f is the frame size in bits, β is the link capacity in bits per second, n is the number of nodes starts from the source to the destination node from the PP's direction, and ρ is the node processing rate.

The term (n−1) represents the exclusion of the source node from the calculation of the path latency for the sent frame, because we assume the latency will be calculated from the time the frame leaves the source node.

5.2.2 Unicast traffic under the stable failure case

If a link or node undergoes failure in the path of the PP and the source node has already started sending through the SP and remained in this situation (stable), then the following expression will determine the frame latency from the time the frame leaves the source node (SP) until it is received by the destination node. This case can be called frame latency in the stable failure case:

$$t_S = l_S\left(\frac{f}{\beta}\right)+\left[\left(n_S-1\right)\left(\frac{f}{\rho}\right)\right] \tag{2}$$

where t_s is frame latency under the stable failure case with unicast traffic, l_s is the number of failure-free links between the source and the destination nodes from the SP port's direction, and n_s is the number of nodes starts from the source node till the destination node from the SP's direction.

5.2.3 Unicast traffic under the return failure case

Under a failure, the time that elapses between a frame's departure through the PP, its reception by the node that detected the failure (or the node that's updated its NPP table before receiving that frame) and its return to its source node will be called the return failure case. Under this case, the source node has not yet started to send its frame through the SP. The frame latency for this case can be determined as follows:

$$t_r = 2\left(l_r\left(\frac{f}{\beta}\right)+\left[\left(n_r-1\right)\left(\frac{f}{\rho}\right)\right]\right) \tag{3}$$

where t_r is frame latency under the return failure case with unicast traffic, l_r is the number of failure-free links between the source node and the node that detected the failure, and n_r is the number of nodes starts from the source node till the node that detected the failure from the PP's direction. Note that the number 2 represents the number of times the sent frame traversed the path between the source node and the node that detected the failure.

5.2.4 Multi/broadcast traffic

Frame latency in the case of multi/broadcast traffic is equal to that under unicast traffic. In the failure-free case, the node will always consume the delivered frames of the PP associated with each sending node and will delete the other copies that deliver through the SP. Consequently, the frame latency will only be considered for the received frame through the PP. In this case, Eq. (1) can be used to determine the frame latency. However, in case of failure, the received node will use the SP to receive the sent frames. In this case, Eq. (2) can be employed to determine the frame latency.

5.3 Recovery Time

Assume that a ring consists of 20 nodes and all the nodes are connected sequentially. If node 1 is communicating with node 11, and the link between nodes 10 and 11 is down during that time, then node 10 needs 4-6 ms to detect the physical link failure; once it does so, it broadcasts its PD frame and returns the sent frames from node 1. Node 1 will keep sending its frames to node 11 because it has not yet received the PD of node 11 or its returned frames from node 10. As soon as node 1 receives the PD or the returned frames, it will update its NPP table and switch its PP to the SP for node 11. Therefore, the duration of the returned frames' journey from node 10 to node 1 plus node 10's detection time related to the link failure can be called the recovery time, when node 1 change its PP and begins to send its traffic from the opposite direction, or in other words, from the SP path. Note that the return frames have a shorter path to node 1 than the PD of node 11's path. The recovery time under the LPFR can be determined as follows:

$$t_{rec} = \left(l_r \left(\frac{f}{\beta} \right) + \left[\left(n_r - 1 \right) \left(\frac{f}{\rho} \right) \right] \right) + t_d \tag{4}$$

where t_d is the time required by the node to detect a physical link failure, which in this case we assume to be equal to 6 ms.

6. NUMERICAL RESULTS

In this section, the numerical results of the cases discussed in Section 5 are illustrated.

6.1 Frame Distribution

To illustrate the advantage in frame distribution, the LPFR and RSTP were applied separately to the network shown in Fig. 2, and each of the nodes was allowed to send one frame to the others. We then counted the number of frames that passed through each network link. The result is shown in Figs. 5 and 6. Fig. 5 shows the number of frames for each ring's link, and Fig. 6 shows the mean values for the network traffic under both the LPFR and the RSTP. Note that the network traffic under the eRSTP and MRP is equal to the network traffic under the RSTP because all of these protocols will disable one port of the ring to avoid the looping issue; consequently, the ring will logically be converted into a bus. For the LPFR algorithm, the mean value was 9 frames per link; for the RSTP/eRSTP/MRP, the mean value was 11.6 frames per link.

6.2 Frame Latency

For the example of sending a single unicast frame from node 5 to node 10 with a frame size of 64 bytes, a link capacity of 100 Mb/s, and a node processing rate of 100 Mb/s, the frame latency of each case is calculated as described in the subsections below.

6.2.1 Unicast traffic under the failure-free case

Using Eq. (1), the frame latency will equal the following:

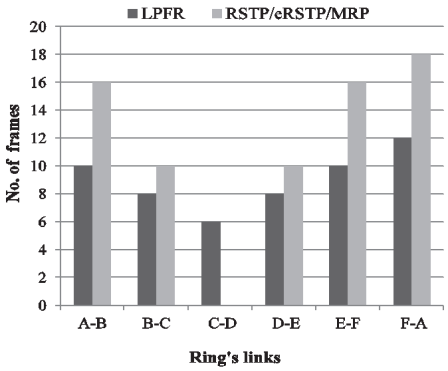

Fig. 5. Frame distribution among the ring's links of Fig. 2.

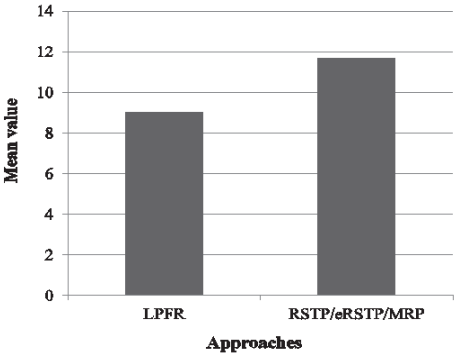

Fig. 6. Mean values of the frame distribution.

$$t = 5\left(\frac{64\times 8}{100\times 10^6}\right)+\left[(6-1)\left(\frac{64\times 8}{100\times 10^6}\right)\right].$$

6.2.2 Unicast traffic under the stable failure case

Assume a physical link failure between nodes 8 and 9 occurs, and all of the nodes update their NPP tables; therefore, in the future, node 5 will use the SP to send traffic to node 10 through node 4. The frame latency is calculated using Eq. (2), as follows:

$$t_f = 11\left(\frac{64\times 8}{100\times 10^6}\right)+\left[(12-1)\left(\frac{64\times 8}{100\times 10^6}\right)\right] = 112.64\ \mu s.$$

This latency represents the time the sent frame will take to leave the SP of the source node and reach its destination.

6.2.3 Unicast traffic under the return failure case

Using Eq. (3), the frame latency will equal to the following:

$$t_r = 2\left(3\left(\frac{64\times 8}{100\times 10^6}\right)+\left[(4-1)\left(\frac{64\times 8}{100\times 10^6}\right)\right]\right) = 61.44\ \mu s.$$

However, during a failure case, the sent frame may pass through the return case before reaching the stable case. At this time, the total frame latency will be determined as follows:

$$t_{to} = t_r + t_s. \tag{5}$$

For the same scenario for nodes 5 and 10, the total frame latency will be equal to:

$$t_{to} = 61.44 + 112.64 = 174.08 \mu s.$$

6.3 Recovery Time

For the mentioned scenario, the recovery time will be equal to:

$$t_{rec} = \left(9\left(\frac{64\times 8}{100\times 10^6}\right)+\left[(10-1)\left(\frac{64\times 8}{100\times 10^6}\right)\right]\right)+6m \approx 6.1\ ms.$$

By contrast, the RSTP (802.1D) will need < 100 ms to recover and find an alternative path for a ring size of 20 nodes. The eRSTP will also need < 100 ms to recover for the same ring size. The MRP needs < 200 ms for a ring size of 50 nodes.

Note that the RSTP can be applied to a maximum ring size of 40 nodes, while for the eRSTP, the maximum ring size is 160 nodes; the LPFR has no such limitation. Fig. 7 shows the recovery time of both the LPFR algorithm and the RSTP/eRSTP with respect to the number of nodes in a ring. The test demonstrates that for a ring size of 300 nodes, the recovery time for LPFR will be equal to 7.52 ms, whereas that for the RSTP/eRSTP will be equal to at least 1.5s if it is assumed that the time delay under these protocols increases linearly. This shows that the LPFR algorithm can be used for a wide variety of ring sizes and a variety of applications that require short recovery time.

7. SIMULATION ANALYSIS

The network shown in Fig. 4 is used in the simulation analysis to simulate the frame latency for the aforementioned scenario of nodes 5 and 10 under the cases described in Section 5. For this purpose, we use OMNeT++ Simulator version 4.6 [15]. In the simulation's scenario, node 5 sends 50 frames to node 10; during that time, a link failure occurs between node 8 and node 9, which causes node 8 to redirect the received frames that were sent from node 5 and send them back to node 5. In addition, all future frames whose destination is node 10 will be sent into the SP of node 5 through node 4. Later, the link failure is repaired and the LPFR algorithm sets node 5 to reuse the PP (fastest) path through node 8 toward node 10. During that simulation scenario, the frame latency is calculated through all the simulation steps, and the results are shown in the graph in Fig. 8.

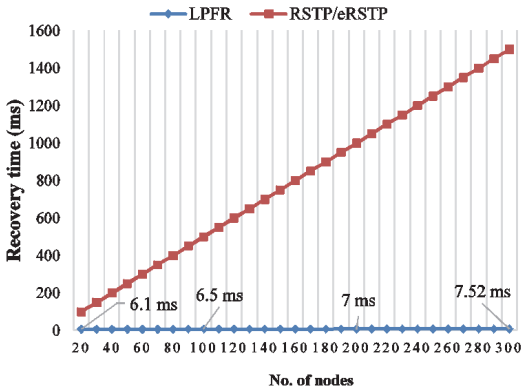

Fig. 7. Recovery time of LPFR compared to the RSTP/eRSTP.

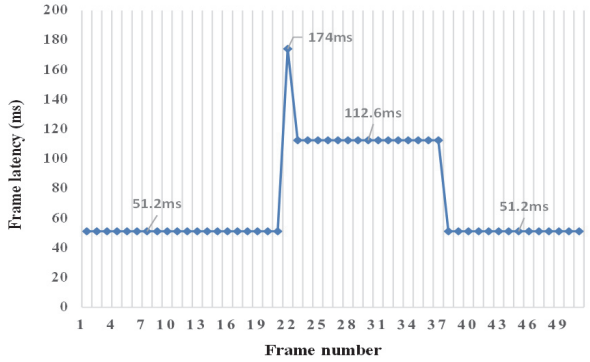

Fig. 8. Frame latency under different cases for the first simulation's scenario.

Fig. 8 shows that during the operation of the LPFR, the sending process did not stop and there was no loss in the sent frames as long as the node that detected the failure received the frames after the detection. This is different from the RSTP process during failure, as in this case, an alternative path must be found before the sending process can continue. The MRP also needs time to switch to another alternative path. The RSTP and the MRP will result in the loss of all of the frames that have been sent because the frames cannot find a path to the destination node. The graph in Fig. 8 shows a spike of about 174μs for the return failure case, whereas, under the stable failure case, node 5 will use the SP and the latency will revert to the value of 112.6μs. When the failure is repaired, the latency reverts to 51.2μs and node 5 again uses the fastest path of the PP through node 8.

Another simulation scenario was conducted for the same network shown in Fig. 4; this is illustrated in Table 2. The number of frames per link was calculated under the failure and failure-free cases. The mean values for these cases were also calculated and compared. The results are illustrated in Figs. 9 and 10. From these two figures, the LPFR shows better performance than the RSTP/eRSTP and MRP. The LPFR shows a frame mean value of 228.8 frames/link, whereas for the RSTP/eRSTP and MRP, the mean value is 634.4 frames/link, as shown in Fig. 10.

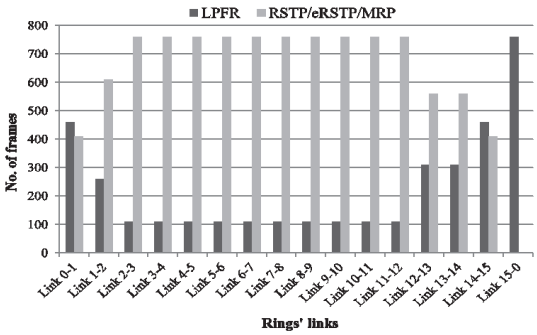

Fig. 9. Frame distribution among the network links of Fig. 4 during the second simulation's scenario.

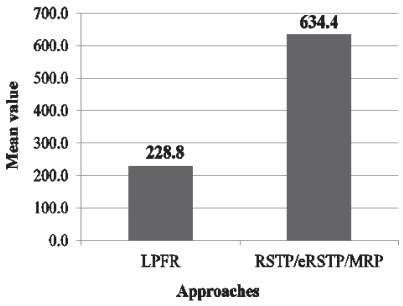

Fig. 10. Mean values of the frame distribution for the network shown in Fig. 4 during the second simulation's scenario.

Table 2. The second simulation scenario.

Source MAC address	Destination MAC address	No. of sent frames
Node 0	Node 15	300
Node 2	Node 14	150
Node 12	Node 1	200
Node 4	All nodes (broadcast)	50
Node 8	All nodes (broadcast)	60

8. CONCLUSIONS

This paper proposed a novel algorithm, called LPFR, to provide path redundancy in Ethernet ring networks with fast recovery and without the looping issue; LPFR also offers better traffic distribution among the network links because it uses the fastest path to deliver the sent traffic and does not disable any port in the network. In most cases, the LPFR provides zero recovery time or seamless redundancy if the node that has detected the failure received the frames after the detection; in this situation, the frames will not be lost because that node will send them back to their source node to redirect them from the opposite direction. However, if the node is unable to detect the failure, the received frame may be lost if it is forwarded in the failed direction. The LPFR uses the Ethernet standard frame layout; therefore, the frame layout does not require any modification. Thus, it will be easy to integrate the LPFR network into the Ethernet standard networks. The LPFR exhibits better performance than the RSTP/eRSTP and the MRP from the perspectives of frame distribution, frame latency, and recovery time. The LPFR algorithm offers reduced frame latency because it always sends the frames through the fastest path. The mean value for the frames per link shows that the LPFR has less traffic mean value than the RSTP/eRSTP or even the MRP. Lastly, the LPFR offers very short recovery time compared to both the RSTP/eRSTP and the MRP. In a ring size of 20 nodes, the LPFR algorithm needs only 6.1 ms, whereas the RSTP/eRSTP needs < 100 ms for a ring size of 20 nodes and the MRP needs < 200 ms for a ring size of 50 nodes. In other words, the LPFR offers a reduction in recovery time of up to 93.9 % compared to the RSTP/eRSTP and up to 96.9 % compared to the MRP. Moreover, LPFR can be applied to a wide variety of ring sizes, and it can work in a ring size of 300 nodes with a recovery time of 7.52 ms, whereas the RSTP/eRSTP needs at least 1.5 seconds under the same circumstances if it assumed that the time delay increases linearly. The other advantage of using the LPFR algorithm is that during the failure period, no frames will be lost as long as a node located in the primary port path has detected the failure before passing the sent frames through it toward the destination.

LPFR will be suitable for smart grid, substation, and industrial applications that need a very short time-out. In the future, we plan to work on extending the LPFR algorithm so it can be applied to any type of network topology.

ACKNOWLEDGMENTS

This work was supported by the Technology Innovation Program, (No. 10058109) funded by the Ministry of Trade, industry and Energy (MI, Korea). It is also supported

by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (No. NRF-2015R1D1A1A02059506}

REFERENCES

1. IEEE 802.3 standard, "IEEE standards for Ethernet," http://standards.ieee.org/getieee 802/download/802.3.1-2013.pdf.
2. IEEE 802.1D-2004 standard, "IEEE standard for local and metropolitan area networks: Media Access Control (MAC) bridges," http://standards.ieee.org/getieee802/ download/802.1D-2004.pdf.
3. IEC 62439-2 standard, "Industrial communication networks: High-availability automation networks, Part 2: Media Redundancy Protocol (MRP)," Geneva, Switzerland, 2010.
4. IEC 62439-4 standard, "Industrial communication networks: High-availability automation networks, Part 4: Cross-Network Redundancy Protocol (CRP)," Geneva, Switzerland, 2010.
5. IEC 62439-5 standard, "Industrial communication networks: High-availability automation networks, Part 5: Beacon Redundancy Protocol (BRP)," Geneva, Switzerland, 2010.
6. IEC 62439-3 standard. "Industrial communication networks – High-availability automation networks, Part 3: Parallel Redundancy Protocol (PRP) and High-Availability Seamless Redundancy (HSR)," Geneva, Switzerland, 2010.
7. IEEE 802.17-2011 standard, "IEEE standard for Resilient Packet Ring (RPR) protocol," https://standards.ieee.org/getieee802/download/802.17-2011.pdf.
8. IEC 61850-90-4, "Network engineering guideline for communication networks and systems in substations," Geneva, Switzerland, 2013.
9. M. Pustylnik, M. Zafirovic-Vukotic, and R. Moore, "Performance of the rapid spanning tree protocol in ring network topology performance of the rapid spanning tree protocol in ring network topology," White paper by Siemens, 2007.
10. IEC 61158-5-10, "Industrial communication networks-Fieldbus specifications-Part 5-10: Application layer service definition-Type 10 elements," Geneva, Switzerland, 2013.
11. A. Giorgetti, F. Cugini, F. Paolucci, L. Valcarenghi, A. Pistone, and P. Castoldi. "Performance analysis of media redundancy protocol (MRP)," *IEEE Transactions on Industrial Informatics*, Vol. 9, 2013, pp. 218-227.
12. https://mentor.ieee.org/802-ec/dcn/16/ec-16-0031-00-00EC-802-17-to-inactive.pptx.
13. S. A. Nsaif, N. X. Tien, and J. M. Rhee, "LPLB: A novel approach for loop prevention and load balancing in ethernet ring networks," *Advances in Computer Science and Ubiquitous Computing*, 2015, pp. 683-691.
14. N. Kezunovic, F. Xu, B. Cuka, and P. Myrda, "Intelligent processing of IED data for protection engineers in the Smart Grid," in *Proceedings of the 15th IEEE Mediterranean Electrotechnical Conference*, 2010, pp. 437-442.
15. OMNeT++ Simulator version 4.6, http://www.omnetpp.org/.

Saad Allawi Nsaif received his B.Sc. degree in Electrical Engineering and M.Sc. degree in Computer and Control Systems from University of Baghdad in 1999 and 2002 respectively. After graduation, he joined University of Baghdad as an Assistant Lecturer, later he joined the Iraqi Ministry of Defense in 2004. He was the Director of the Command and Control Systems (C2) for 7 years. His contribution in designing, developing and establishing the Iraqi C4I systems is well known, especially with the Iraqi Defense Network (IDN). In February 2015, he received his Ph.D. degree in Information and Communications Engineering from Myongji University, South Korea. Currently he is working for Cisco Systems, Inc., USA. His current research interests are in ubiquitous networks, layer 2 switching, industrial Ethernet, ad hoc networks, and smart grid communications. He's also a member of IEEE.

Nguyen Xuan Tien received his B.S. degree in Electronics and Telecommunications Engineering and M.S. degree in Information Technology from Danang University of Technology (DUT) in 1997 and 2004, respectively. After graduation, he joined VNPT Group and GTEL Mobile Company in 1997 and 2009, respectively. He served as Deputy Head of Technical Department at VNPT and Head of O&M Department at GTEL Mobile. He received his Ph.D. degree in Information and Communications Engineering at Myongji University in 2017. He is currently an Assistant Professor in the Information and Communication Engineering Department at Myongji University, Korea. His current research interests include fault-tolerant networks, optimal routing algorithms, and ad-hoc wireless networks.

Jong Myung Rhee received his Ph.D. from North Carolina State University, USA, in 1987. After 20 years at the Agency for Defense Development in Korea, where he made noteworthy contributions to C4I and military satellite communications, he joined DACOM and Hanaro Telecom in 1997 and 1999, respectively. At Hanaro Telecom, which was the second largest local carrier in Korea, he served as chief technology officer (CTO), with a senior executive vice-president position. His main duty at Hanaro Telecom was a combination of management and new technology development for high-speed Internet, VoIP, and IPTV. In 2006 he joined Myongji University and is currently a full professor in the Information and Communications Engineering Department. His recent research interests are centered on military communications and smart grid, including ad-hoc and fault-tolerant networks. He is a member of IEEE and IEICE.