

Received May 8, 2020, accepted May 23, 2020, date of publication May 26, 2020, date of current version June 8, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2997764

A New Adaptive Prioritization and Fail-Over Mechanism for Ring Network Adapter

AMIN KHALILZADEGAN^{ID}, MOHAMMAD ZAREI, AND KHAYROLLAH HADIDI, (Member, IEEE)

Microelectronics Research Laboratory, Urmia University, Urmia 571478334, Iran

Corresponding author: Amin Khalilzadegan (a.khalilzadegan@urmia.ac.ir)

ABSTRACT This paper presents a new adaptive prioritization and fail-over mechanism for ring network adapters (RNA). Owing to the use of a shared medium in the structure of ring networks, management of network resources is very important and cannot completely be fulfilled by fixed priority strategies. An adaptive prioritization mechanism for efficient utilization of network resources is proposed in this paper. Hop-Count and the distance between source and destination are the key parameters that are included in the priority assignment procedure. Unlike the conventional ones, the arrived traffic is not put blindly into the queues thanks to the awareness of stations form status of each other. The newly introduced fail-over mechanism is based on the Power over Ethernet concept. Each station monitors the heartbeat of its neighbors and tries to keep them alive by providing their minimum required electrical power whenever a fault occurs. By doing so, not only single failure but also any successive double failure scenarios cannot disrupt data path continuity. Several simulations are carried out to assess the behavior and performance of the proposed methods in OPNET Modeler. Moreover, a high-speed USB test board is designed using Xilinx-Spartan6-lx9 FPGA to experimentally verify the performance of the proposed mechanisms.

INDEX TERMS Ring network, weighted fair queuing, node failure, fail-over mechanism, power over Ethernet.

I. INTRODUCTION

As resources are getting scarcer in modern networks, optimum resource allocation and congestion control have become a challenging issue [1]. Since stations in the ring structure utilize a shared medium, so this problem is more and more intricate. Although exploiting priority-based queues is the most popular approach to manage limited resources in such systems, using a fixed-priority scheme is not very efficient and fair [2], [3]. In conventional structures, the distance between source and destination and the packet hop-count has no role in prioritizing mechanism. Hence, there is a risk of discarding a packet at a congested node near the destination, compared to a recently transmitted packet with only one unit greater priority value. This functionality is not so fair and leads to waste the network resources. Based on the fact that whatever the data packet passes through the network, it consumes network resources and therefore its worthiness is increased, a new adaptive prioritization mechanism is proposed in this paper to overcome this problem. Besides, the probability of facing congestion is increased by growing the

distance between the source and destination stations. Consequently, it is wise to involve distance as one of the key factors in the priority assignment process.

Another imperfection associated with the conventional network adapters (NA) is that they put blindly arrived traffic into the queues regardless of the traffic ahead at the subsequent stations. In the proposed structure, stations have a clear vision about the status of the other nodes thanks to the information that is obtained from discovery messages and data packets. This method is very beneficial for dual-ring networks in which data packets are transmitted in clockwise or counter-clockwise ringlets. In such networks, a station may decide to change the transmitting direction of a packet by comparison of the cost of delivering the packet to its destination through both ringlets.

Survivability and availability of data networks is the other serious challenge for telecommunication customers [4]–[7] and the ability to tolerate failures and having networks with zero recovery time is becoming increasingly important in nowadays networks. In such systems, the quality of services tightly depends on the reliability and degree of fault tolerance that they offer. A fault-tolerant network is able to perform its function properly in the presence of (one or more) failures in

The associate editor coordinating the review of this manuscript and approving it for publication was Yiming Huo^{ID}.

its components without any interruption and human intervention [8]. The techniques used for fault recovery rely on the network topology and structure of its components in the first place and type of faults in the second order.

Although star topology is the most common topology among the different network topologies, there is a strong tendency to use ring-based structures for data communication owing to their salient features [9]–[13]. As nodes are serially connected to each other in the ring structure, the failure of one node can affect the entire network. Therefore, healthy working of all nodes and exploiting a rapid fault recovery mechanism is more vital in this topology with respect to the others.

According to the reports released by the Network Reliability Steering Committee [14], node failure is the most common reason causing network faults. Node failure happens when the node is unable to function properly and make the network reconnected within the prescribed time. The power outage is the primary cause that leads a data network to go down. Even having some kind of backup power, this problem is inevitable [15]. Because there always comes a time when there's maintenance, power has to be turned off. Furthermore, power supply tends to be the part that fails the most often in many devices. In addition to the power outage, computers crash due to errors in operating systems is the other devastating reason that leads nodes to fail [16].

Different schemes have been proposed in the literature for enhancing the reliability of a ring network including bypass switch, pass through, braided and mesh rings [17]. Each of these methods has its pros and cons. In the first scheme [18], the defected node is detached from the network with a simple bypass switch to preserve data-path continuity. Although this method is very convenient for implementation, it suffers from some pitfalls. Bypassing failed node increases the total cable length between two active neighbor nodes and there would be a risk of exceeding maximum cable length recommendations. Significant loss of data during reconfiguration due to the slow switching speed of bypass switch is the other fundamental problem associated with this method.

Passing through all received data to other port when the station does not function properly, is another fail-over approach. However, there is no concern about the maximum cable length violation, passing through is not possible when active repeating functionality is lost due to the power outage [19]. Mesh, torus and braided structures give a higher degree of fault tolerance at the cost of adding redundant components. Since hardware redundancy requires modifications in network nodes and topologies, so these structures are not very popular and straightforward to realize efficiently [20].

It should be kept in mind that in all situations described above that node goes down; NA is healthy and can be utilized to keep the data path connected. In this paper, a new fault recovery method is presented on the base of this theorem. In the proposed method, each station checks functionality and monitors the heartbeat of its neighbors. Whenever a fault occurs in a station, its neighbors try to keep the NA of this

station alive by providing its minimum required electrical power. This capability becomes possible for the sake of using Power over Ethernet technique (PoE) [21]. By applying the proposed method in the ring networks, not only single failure but also any successive double failure scenarios cannot disrupt data path continuity. Besides, conventional fail-over mechanisms immune the networks against single failures only. Thus, exploring a new method that protects the network against successive failures is the other motivation for the proposed method in this paper.

The rest of this paper is organized as follows: Section II introduces details of the new adaptive priority based traffic and queue management. Section III briefly reviews conventional fail-over mechanisms. In Section IV, the proposed fail recovery mechanism is presented. OPNET simulation results are given in section V to validate the performance of the proposed methods compared to those available in the literature. FPGA Implementation of new RNA is discussed in Section VI. Finally, concluding remarks are provided in Section VII.

II. NEW ADAPTIVE PRIORITY BASED TRAFFIC AND QUEUE MANAGEMENT

Unfair bandwidth allocation deteriorates the overall quality of service because it leads to increase packet transmission delay and packet loss due to congestion. Hence, fair bandwidth allocation mechanisms should be employed to resolve this anomaly by guaranteeing that all class of services can get their required resources without worrying about the starvation of others [22]–[24]. In the ring network, bandwidth allocation is a challenging issue owing to using a shared medium between all stations. In this paper, a new adaptive prioritization and queue management mechanism is proposed to efficiently allocate available bandwidth. Details of these mechanisms are explained in the following.

The first thing to be noticed is that when a data packet passes through the network, it consumes network resources and its worthiness should increase. Take for example, a packet with a predefined and fixed priority which has passed several stations toward its destination. It is no fair to discard this packet at a congested node near the destination, compared to a recently transmitted packet with only one unit greater priority value. Another point that must be considered in the ring networks is that as the distance between the source and destination nodes increases, the probability of facing congestion and consequently, packet delivery time to the destination also increases. Hence it is wise to set the priority of the packets according to the amount of the distance between source and destination. In this paper, a new adaptive prioritization mechanism is proposed to improve the performance of the ring network. In this method, 8 bit priority field is set in accordance with the combination of multiple parameters including the distance between two stations, hop-count that the data has passed through and also importance degree of source, destination, and transmitted data.

In order to facilitate the implementation of the prioritization mechanism, some assumptions are considered on the importance of each aforesaid parameter which is graphically summarized in Fig. 1.

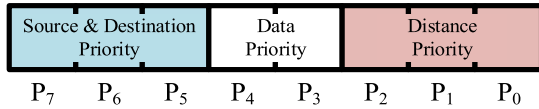


FIGURE 1. Graphical summary of the prioritization mechanism.

The length of this field can vary depending on the application. In this paper, size of the priority field is considered 8 bits length for ease of implementation. This field is divided into three different parts based on the importance of each priority component. Since the priority of the source and destination stations is more important in regard to the other two components, the three most significant bits are devoted to the average value of the source (P_{Source}) and destination ($P_{Destination}$) priorities. By doing so, eight levels of priority can be considered for source and destination importance, increasing from the lowest value (B'000') to the highest (B'111') one. To put these bits to their spatial position in the priority word, they should be shifted five bits to the left side or be multiplied by 32. After the station's self-priority value, priority level of the data which is transmitted between source and destination is important. Hence, the two intermediate bits are set based on the importance of the transmitted data (P_{Data}). In other words, 4 different levels of priority can be dedicated for transmitted data. In a similar manner, a multiplication by 8 is needed to take these two bits to their place as illustrated in Figure 1. Finally, the three least significant bits are determined by the distance between source and destination ($P_{Distance}$). The value of the priority byte is updated while packets are transmitted in the network. As the data passes through the stations and waits in the buffering queues, its priority value increases gradually. The general form of aforesaid parameters can be formulized as follows:

$$P_{Packet} = 32\left(\frac{P_{Source} + P_{Destination}}{2}\right) + 8P_{Data} + P_{Distance} + K \cdot Hop_count \quad (1)$$

where K is a constant that varies with the number of stations in the network. In each station, received traffic is classified and placed in one of three data queues, namely Q_{HP} , Q_{MP} and Q_{LP} , according to the level of corresponding priority. Q_{HP} , Q_{MP} and Q_{LP} are the queues with the high, medium and low priority value, respectively. In order to guaranty fairly sharing of bandwidth between different classes of priorities and preventing starvation of a particular class, an adaptive Weighted Fair Queuing (WFQ) approach is utilized [25]. The total block diagram of the queue management unit is depicted in Fig. 2. Each individual queue is considered as a priority queue in which packets with the highest priority are served at first, independent of its arrival time. The system starts to work with initial weights and thereafter weights of WFQ are

dynamically updated based on the average importance of the data and the occupancy amount of queues.

An exponential moving averaging that its period is equal to 32 is utilized to calculate average priority. Despite the simplicity of implementation, this method has sufficient accuracy in comparison to complex approaches such as reasoning based or evolutionary algorithms. By using this method, the new value of the average importance of each queue ($P_{ave(new)}$) can be obtained as follows:

$$P_{ave(new)} = \frac{31}{32}P_{ave(old)} + \frac{1}{32} \cdot Event \cdot (P_{in-data} - P_{out-data} - P_{dropped-data}) \quad (2)$$

where

$Event$: a Boolean variable which is set when the input, output or discard events occur in a queue;

$P_{ave(old)}$: the old value of the average priority;

$P_{in-data}$: priority of the newly input data to queue;

$P_{out-data}$: priority of the data that is pulled out from the queue to transmit;

$P_{dropped-data}$: priority of the data that is dropped due to queue overloading.

In order to reduce computational effort, (2) can be rewritten in the following the form by a little manipulation:

$$P_{ave(new)} = P_{ave(old)} + \frac{1}{32} \cdot Event \cdot (P_{in-data} - P_{out-data} - P_{dropped-data} - P_{ave(old)}) \quad (3)$$

Based on the (3), variation of P_{ave} in each cycle is (ΔP_{ave}) equals to:

$$\Delta P_{ave} = \frac{1}{32} \cdot Event \cdot (P_{in-data} - P_{out-data} - P_{dropped-data} - P_{ave(old)}) \quad (4)$$

An up-down counter is dedicated to each queue for indicating the occupancy amount of it. Upon event occurrence of each queue, the value of these counters is updated. (5-7) show the updating procedure of the WFQ's weights, where Q terms are determined on the base of three most significant bits of the occupancy counters. These equations are assumed such that the sum of weights remains constant. As is clear, any increment in the value of each weight leads to a reduction in the value of two others. In order to prevent starvation of the queues, variations of weights are limited to a minimum and maximum value. Accordingly, when the weight of one of the queues reaches the minimum or maximum value, coefficients of updating terms are changed to keep the sum of weights constant. Although allowed bandwidth of each queue is proportional to its assigned weight, its allocated bandwidth is divided between other queues when it goes empty. To handle all of the aforesaid updating procedures, a simple microprocessor is designed with a limited number of instructions.

$$W_{HP(new)} = W_{HP(old)} + 2Q_{HP}\Delta P_{ave-HP} - Q_{MP}\Delta P_{ave-MP} - Q_{LP}\Delta P_{ave-LP} \quad (5)$$

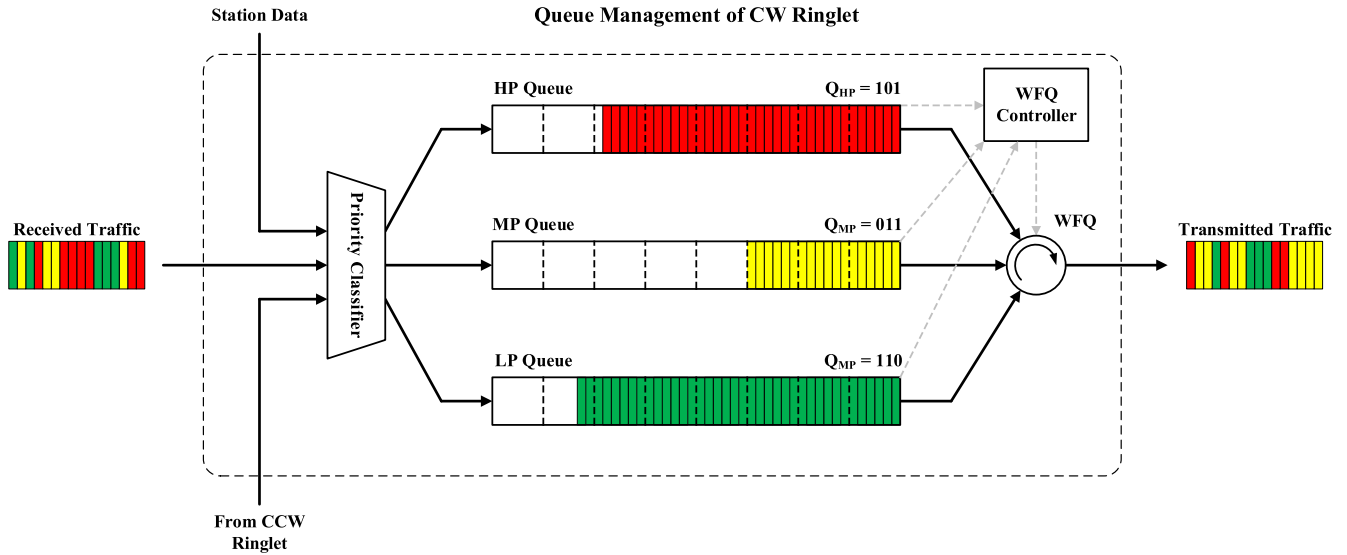


FIGURE 2. Block diagram of the queue management unit.

$$W_{MP(new)} = W_{MP(new)} - Q_{HP} \Delta P_{ave-HP} + 2Q_{MP} \Delta P_{ave-MP} - Q_{LP} \Delta P_{ave-LP} \quad (6)$$

$$W_{LP(new)} = W_{LP(new)} - Q_{HP} \Delta P_{ave-HP} - Q_{MP} \Delta P_{ave-MP} + 2Q_{LP} \Delta P_{ave-LP} \quad (7)$$

Another capability that distinguishes this network card from its counterparts is that in this network card data which is arrived to a station does not put into data queues blindly. Thanks to exchanging discovery messages, stations have a clear insight about the status of the other nodes on the network. Furthermore, each station gets additional information about network traffic by inspecting source and destination addresses of packets which only passes them to its upstream or downstream neighbors without any manipulation. Upon receiving a packet, all stations on the network calculate the total time and hardware cost of delivering the packet to its destination through both clockwise and counter-clockwise ringlets. Based on these cost values, a station may decide to change the transmitting direction of a packet due to heavy traffic ahead at the subsequent stations. According to (8), the total cost of delivering each packet is obtained by summing up all individual system costs.

$$TotalCost = \sum_i Cost_i \quad (8)$$

where $Cost_i$ is the time and hardware cost of passing the packet through i -th station. To obtain the values of individual costs, it should be noted that priority of the packet is updated while passing through the stations and it is placed in one of the queues depending on its priority at the intermediate stations. Also, whatever the priority of the packet is less than the average priority value of the queue and more data exists in the queue, the packet has to wait more time in the queue

and therefore time and hardware cost would be higher. Thus, $Cost_i$ could be formulated as follows:

$$Cost_i = \begin{cases} (P_{ave-LP-i} - P_{Packet}) \cdot Q_{LP-i}, & 0 \leq P_{Packet} < 64 \\ (P_{ave-MP-i} - P_{Packet}) \cdot Q_{MP-i}, & 64 \leq P_{Packet} < 128 \\ (P_{ave-HP-i} - P_{Packet}) \cdot Q_{HP-i}, & 128 \leq P_{Packet} \leq 255 \end{cases} \quad (9)$$

In the above equation, priority boundaries are defined according to the primary assumptions of section 4.

III. CONVENTIONAL FAIL-OVER MECHANISMS

A. NODAL BYPASS SWITCH

In this methodology [18], each station has a nodal bypass switch which is usually a mechanical relay as depicted in Fig. 3. Under network normal circumstances, this switch is open and two ports of NA can send and receive data independently. When a node failure occurs, this switch is

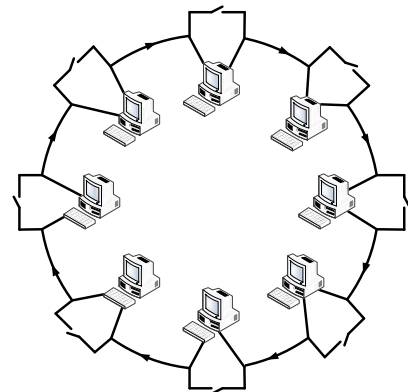


FIGURE 3. Nodal bypass switch mechanism.

closed to detach the failure node from the ring to maintain data-path continuity. By doing so, all frames received from one port are directly transmitted to the adjacent port and the communication between all other healthy operative nodes can be continued. In this situation, two cable segments between the failed node and its healthy neighbors connect together. In other words, total cable length between two active neighbor nodes increases and there would be a risk of exceeding maximum cable length recommendations. This issue is the main disadvantage associated with this method. Another problem of this fail-over mechanism appears when a failure occurs in a node that is passing traffic of the other nodes through itself. In this case, significant parts of data are lost during low-speed bypass switch reconfiguration time. This obstacle gets worse as the data rate increases.

B. PASS-TROUGH

The Pass-through mode is another fault recovery approach which allows a station to leave the ring while maintaining the topology of the ring network in the closed-form. When a station recognizes that it is no longer able to function properly, it passes through all received packets to its other port. Indeed, the station acts as a repeater and repeats each received data to its upstream or downstream neighbor without any manipulation. Although data loss of this method is less than bypass switch and there is no concern about the maximum cable length violation, the pass-through mode is not possible when active repeating functionality is lost due to power loss [19].

C. BRAIDED AND MESH RING

Fault tolerance can also be achieved by providing extra components and considering redundancy in the structure of the network [20]. Mesh and braided (Fig. 4) ring are the [26], [27] structures in which each node is connected to other neighboring nodes in addition to adjacent neighbors. In such structures, service outages due to network partitioning can be avoided by properly selecting active links.

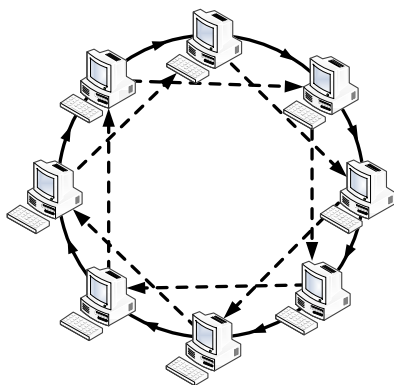


FIGURE 4. Braided ring network.

Since hardware redundancy requires modification of network nodes and topologies that may be expensive and

difficult to implement, so this method is not very popular and straightforward to realize in an efficient way [20]. Besides, redundant components are idle and unnecessary in fault-free conditions.

IV. PROPOSED FAIL RECOVERY MECHANISM

As discussed earlier, NA is healthy in most of the faulty situations and can keep the data path connected. Based on this concept, a new fail recovery mechanism is proposed in this paper for ring structure. The key principle behind this method is PoE technology. For the sake of using PoE technique in the ring network, the proposed method can be named Power over Ring or PoR in abbreviation. PoE is a technique that superimposes data and electrical power over the same network cable using a phantom feeding circuit via the center-tapped signal coupling transformers [21].

The basic principle behind the phantom circuit can be explained on the basis of the Wheatstone bridge. According to Fig. 5, suppose a special case of the Wheatstone bridge with all its bridge resistances exactly equal and derived by two voltage sources. Although each source will inject its corresponding current components through the bridge, they are mutually independent and “phantom” to each other.

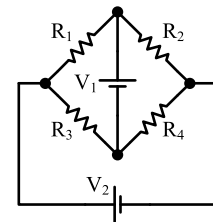


FIGURE 5. Basis of the phantom feeding circuit.

Substituting bridge resistors with transformers and wires with equal resistance does not disturb the balance of the Wheatstone bridge. Since twisted pair cable, the most common type of network medium, is made up of two intertwining wires with the same physical and electrical characteristics, it is possible to replace the wires with twisted pair cable. Also, to remove single-wire earth return (SWER) architecture [28] and provide proper return current path, two Wheatstone bridges should be used together as illustrated in Fig. 6. The advantage of using center tap transformers is that current of the PoR power source splits up in halves with equal magnitude and opposite direction. In this case,

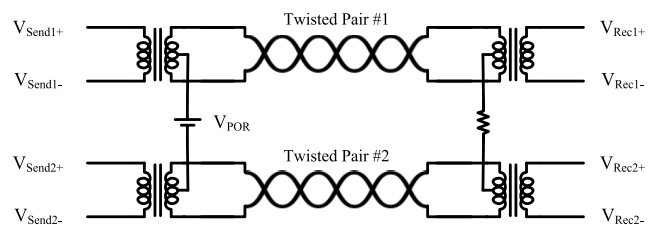


FIGURE 6. Modified phantom feeding circuit.

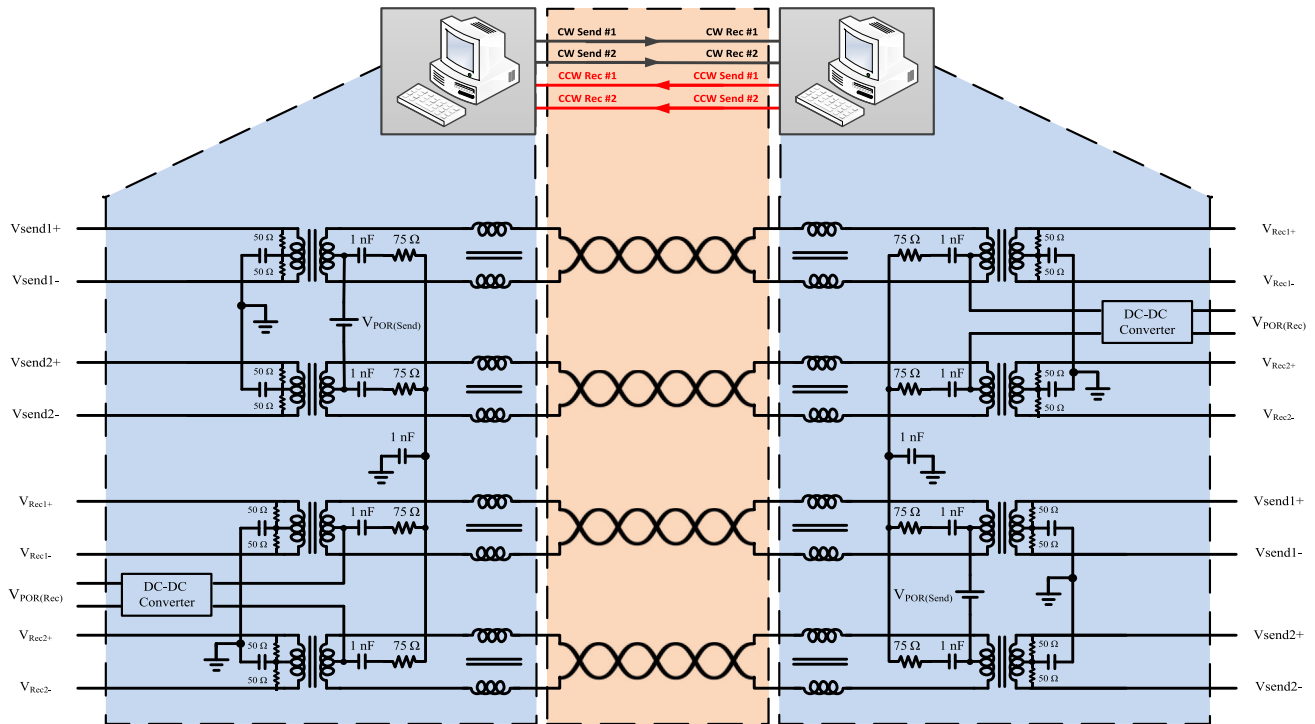


FIGURE 7. Detailed block diagram of the physical layer of the designed ring NA.

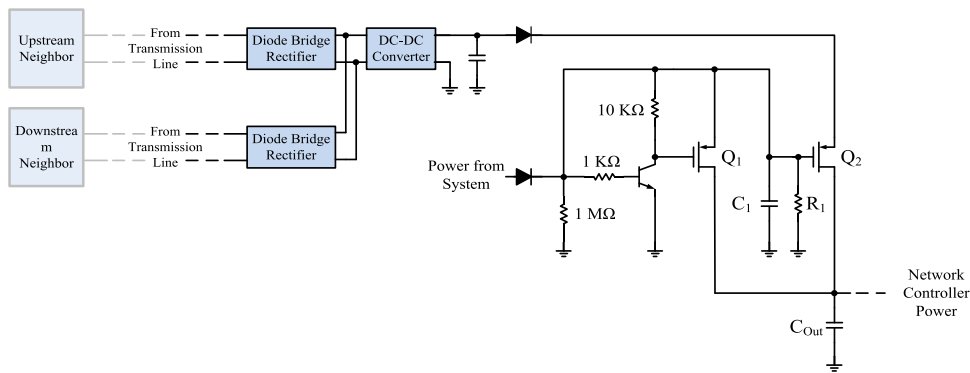


FIGURE 8. Power switching circuit.

the corresponding magnetic flux of the currents cancels out each other and leaves no net DC flux in the transformer core.

By exploiting the phantom feeding circuit in the structure of the ring NA's, each station is powered by its own power supply in normal operation. Whenever power supply crashes, the station's power is provided by its neighbors. The detailed block diagram of the physical layer of the designed RNA is depicted in Fig. 7. In this structure, two pairs of twisted pair are dedicated for sending and the other two for receiving data. In other words, there are two concentric clockwise and counter-clockwise rings that can operate independently of each other for handling different data streams or jointly together for a single one.

In order to guarantee zero fault recovery time and no data loss during fault occurrence, an immediate power switching mechanism is needed. Fig. 8 illustrates the designed circuit

for this purpose. As long as the station's power is OK, there is no need for an external power source. In this case, gates of Q_1 and Q_2 transistors are low and high, respectively. So Q_1 is turned on and NA is supplied by the station itself. But when the station's power goes down, Q_2 turns on to provide the required power from neighbors. Although in this situation the NA power is connected, it only receives data frames and acts as a repeater.

The longer the distance between the station and neighbors, the higher the voltage drop across the transmission line. So, it is wise to provide the required power from the nearest station. Besides, neighbor power sources should not be connected to each other with opposite polarities. These requirements can be fulfilled by connecting neighbor's power sources together using diode bridges, as shown in Fig. 8.

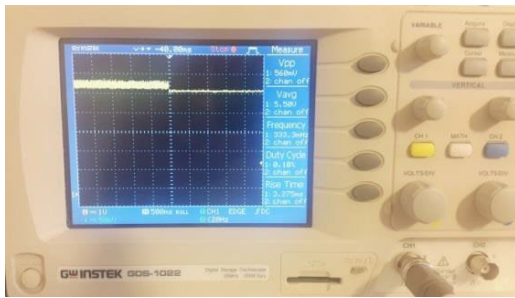
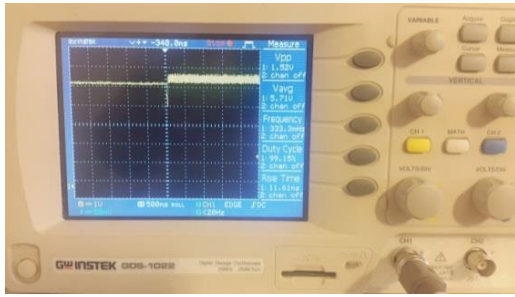


FIGURE 9. Voltage variation of the power supply of the designed NA.

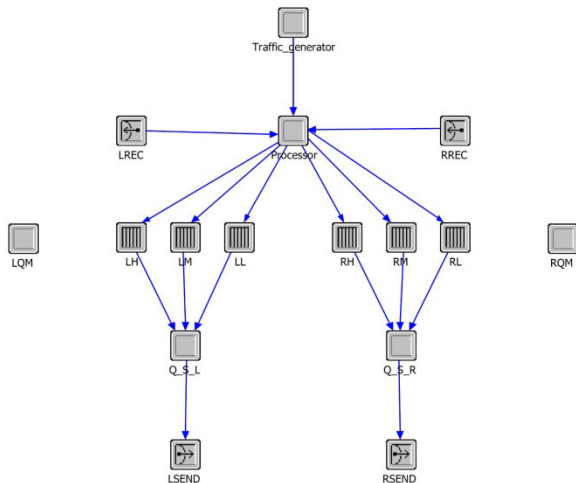


FIGURE 10. Node model of the designed ring network card.

Since electrical power is transmitted through the transmission line with a higher voltage level to reduce power losses, so a DC-DC converter is included in the design. Also, protection diodes are considered to prevent flowing current in the reverse direction. Fig. 9 depicts the voltage variation of the NA power supply during switching from its own supply to PoR or vice versa. It is obvious that the voltage drop is not so high that it can disrupt the functionality of the NA in both cases.

V. OPNET SIMULATION RESULTS

There are different tools for designing and evaluating network structures. Some of these tools are suitable for analytical modeling while others are appropriate for simulating

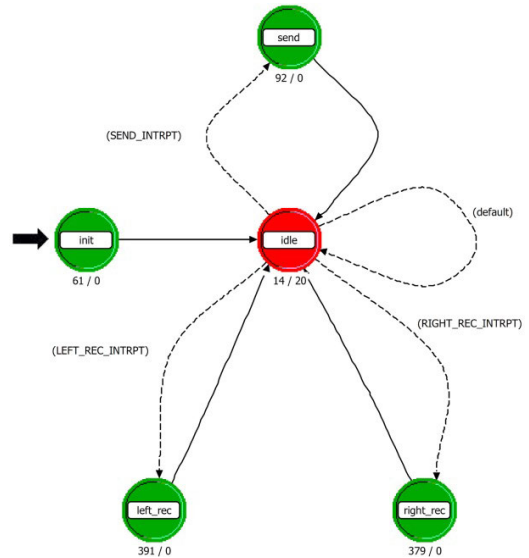


FIGURE 11. State transition diagram model of the main processor.

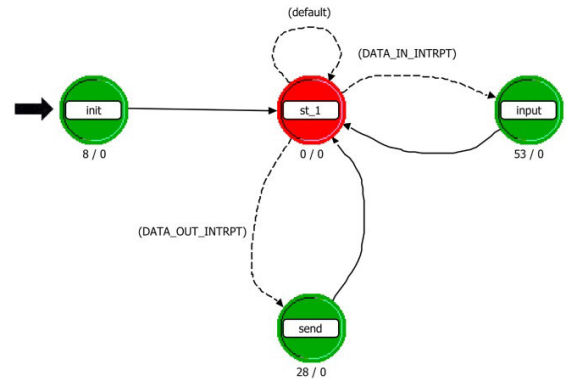


FIGURE 12. State transition diagram model of the data queues.

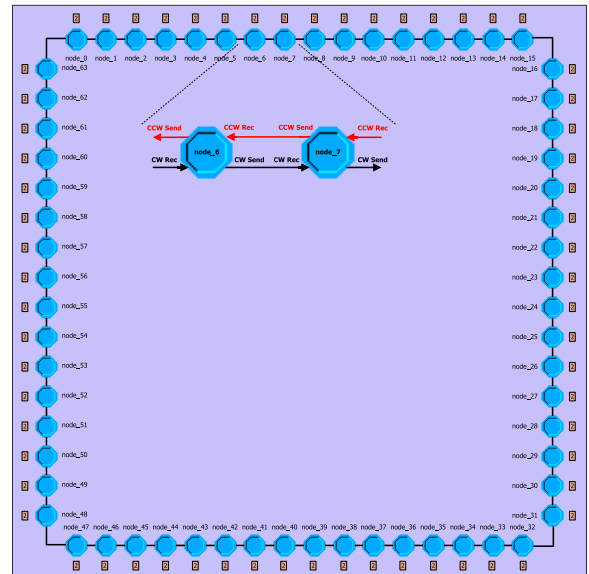
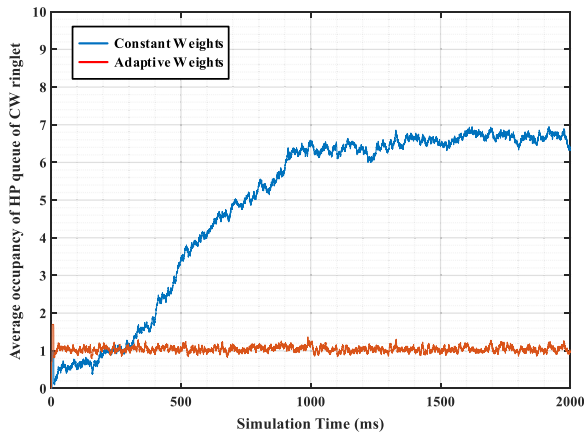
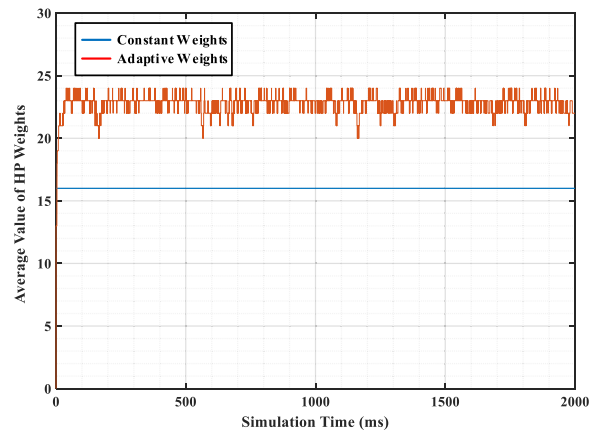


FIGURE 13. A typical ring network with 64 stations.

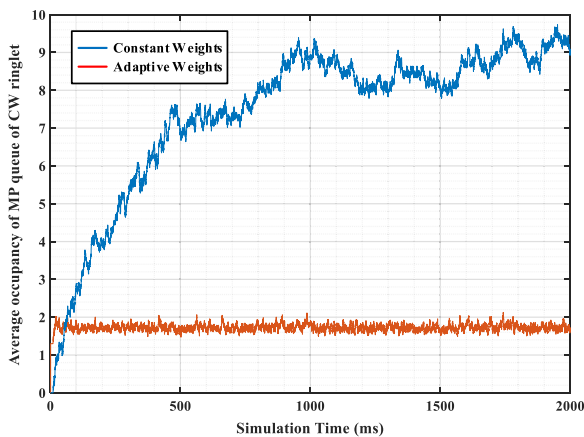
the overall behavior of the design. Petri net and finite state machine [29], [30] are the most conceptual and intuitive ones. Owing to the pictorial representation, these tools are



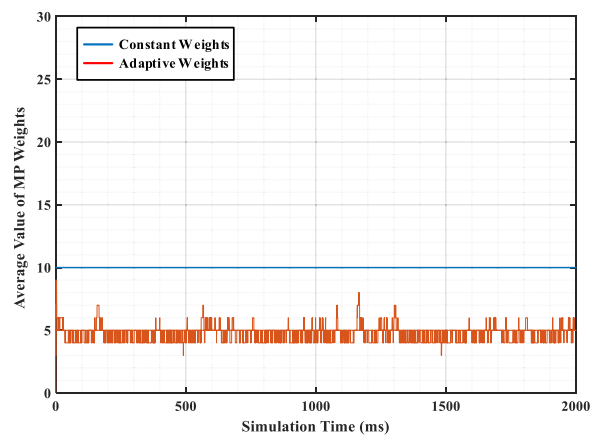
(a) Average occupancy of HP queue of CW ringlet



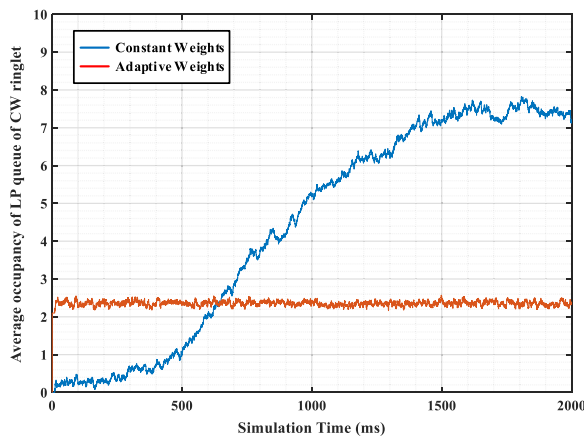
(d) Average value of HP weights



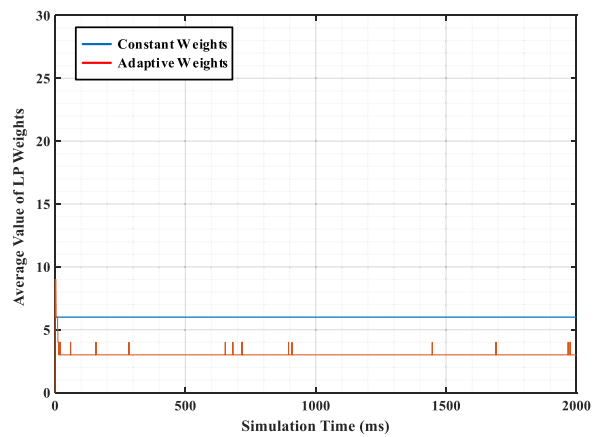
(b) Average occupancy of MP queue of CW ringlet



(e) Average value of MP weights



(c) Average occupancy of LP queue of CW ringlet



(f) Average value of LP weights

FIGURE 14. Constant weight traffic management versus the proposed adaptive queuing approach.

very appropriate to transform analytical models into simulation models or hardware implementations. Both Petri net and finite state machine evaluate states of the systems upon the occurrence of the events. The ability to model multiple activities distinguishes Petri net from finite state machine. In a finite state machine there is always a single “current” state that determines which action can occur next. While

in Petri net transitions are executed as soon as all input locations contain at least one token [31]. In general, the Petri net is more versatile with respect to finite state machine. In other words, finite state machine can be considered as a special case of Petri net [32]. However, most of the engineers are more comfortable with finite state machine because there are more tools to convert a finite state machine

to an implementation. Besides analytical modeling, network simulation is an important methodology in network research fields. OPNET Modeler [33], [34] is the leading and powerful general-purpose network simulator which is widely used by researchers, protocol designers, university teachers and students in the fields of electronic engineering and computer science for realistic simulation, modeling, and analysis of computer networks [35]–[37]. To assess the behavior and performance of the proposed adaptive queuing and fail recovery mechanisms, a new node model for RNA is developed by OPNET software in the form of state transition diagrams at first.

Fig. 10 illustrates the node model of the designed ring network card. This model includes packet generator, main processor, queue management unit, data queues, data receivers and transmitters. Packet generator is responsible for generating packet based on the traffic patterns. The generated or received traffic is directed to the clockwise or counter-clockwise ringlets based on the information of the topology table. The main processor controls and directs whole transactions. Queue management unit monitors the traffics on the network and tunes the weights of the WFQ to fairly share bandwidth between different classes of services.

The process model of the above mentioned blocks are implemented in the form of finite state machine via state transition diagram (STD).

STD of the main processor is depicted in Fig. 11.

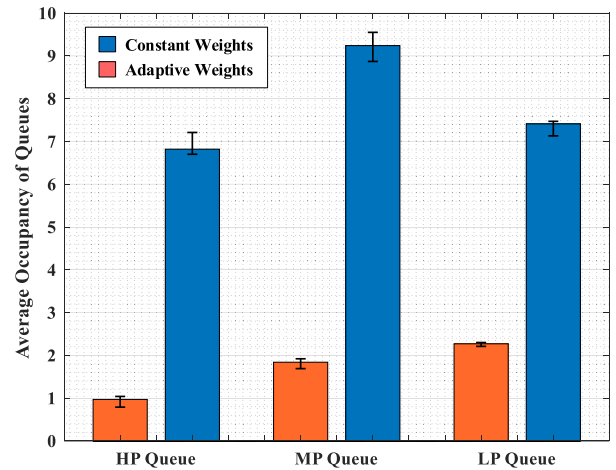
By running simulation, finite state machine starts from the state named “init” which initializes the parameter of the process. Then STD transits to “idle” state and waits in this state until a packet is arrived or generated. Upon a packet is generated or received, the process goes to other states to handle the corresponding packet and returns back to the “idle” state again.

Since operation of the all individual queues is the same, only one STD is design to model them as depicted in Fig. 12. Similar to main processor, this process also begins with “init state” to initialize the parameter and then waits in “idle” state till a packet is received or sent. A timer is set to control transmitting data rate. If the queue is not empty, a packet is sent once the timer overflows and the queue management gives the permission for data transmission to specified queue.

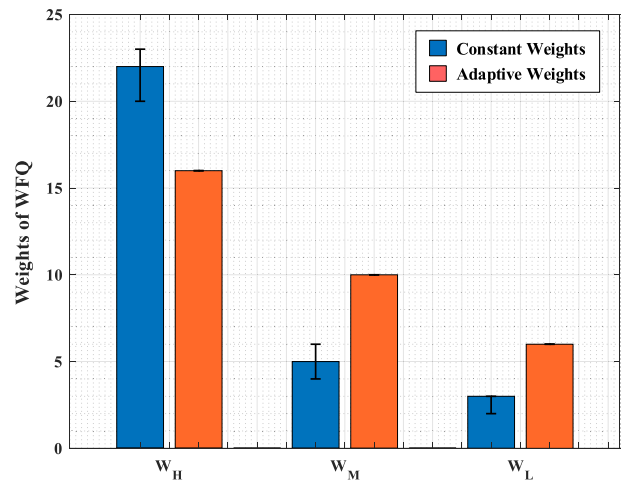
To verify the performance of the design node model, a network that consist of several numbers of stations are designed using this node model. A typical network with 64 stations is shown in Fig. 13.

In the first scenario, behavior of the stations in traffic management is examined by monitoring the average occupancy of the data queues for both cases in which queuing weights are constant or altered in an adaptive manner. Comparative results in Fig. 14 confirm that the proposed adaptive data queuing mechanism handles the incoming and outgoing traffic as well by adjusting the proper values for weights.

In order to further assess the performance of the proposed mechanism, simulations were repeated for 30 times and the mean of the results was provided in the form of error bar



(a) Error bar graph of average occupancy of queues



(b) Error bar graph of weights of WFQ

FIGURE 15. Error bar graphs of investigated parameters for 30 time iteration of simulation.

graphs to survey the mean value and error rate of the parameters under investigation as shown in Fig. 15.

In the next step, networks are simulated under the same scenarios including not only single but also successive double failures. It is worth to note that choosing an appropriate traffic model prevents under-estimation or over-estimation of network performance. Pareto based traffic models are excellent candidates for networks with the unexpected demand for packet transfers since the model takes into consideration the long-term correlation in packet arrival times [38], [39]. This traffic model is exploited to generate data packets in the stations. The simulations were aimed to evaluate average hop-count (N_H) (number of the stations that data frames pass through as they travel between source and destination) and average occupancy amount of data queues (N_Q). The values of these parameters for the proposed and resilient ring [24] networks with a different number of stations (N_S), in the presence of a double successive failure, are gathered in Table 1. As is clear, the efficiency of the proposed method becomes more prominent by increasing the number of stations. The

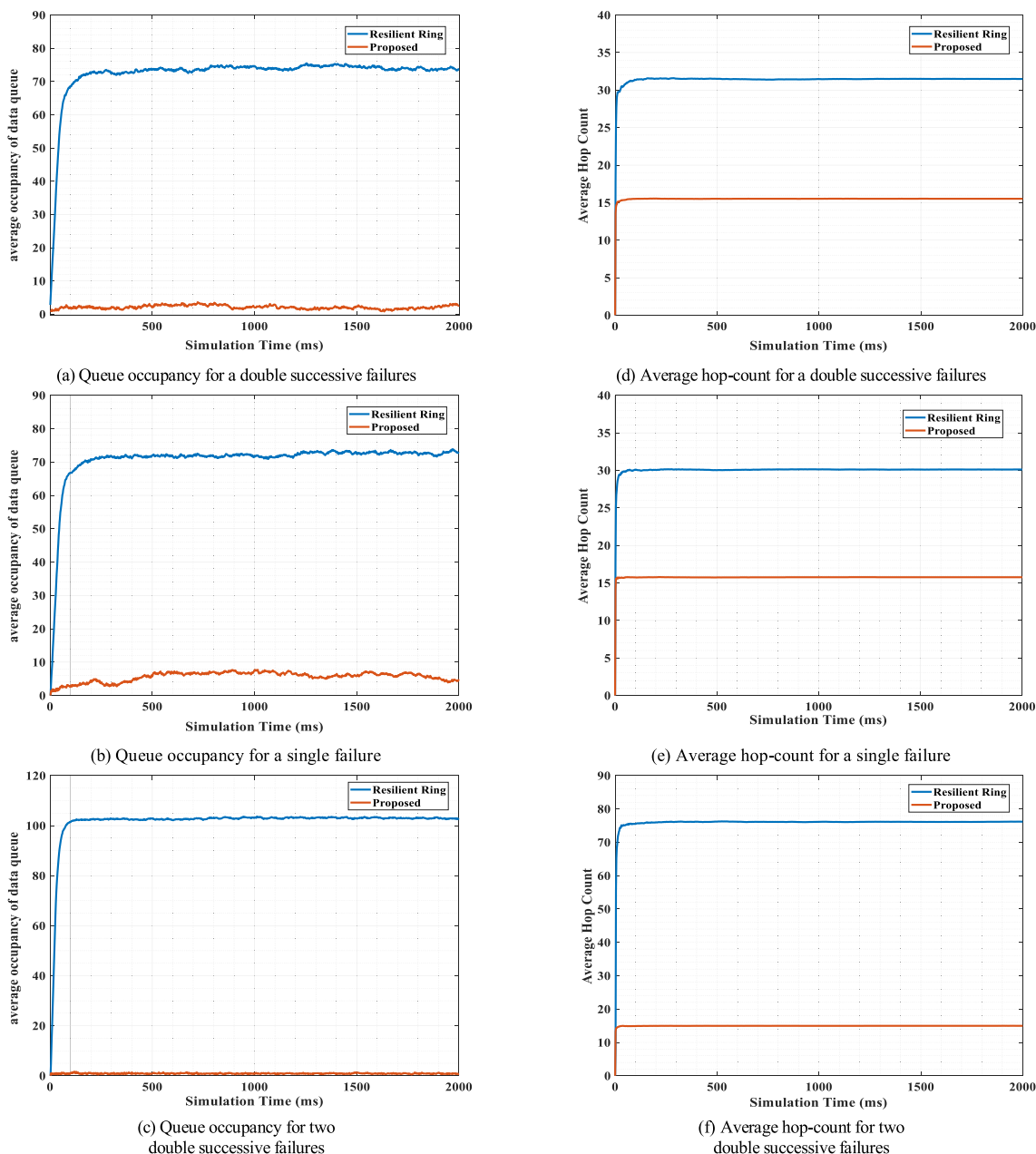


FIGURE 16. Comparative results of the proposed and resilient ring networks for different failure conditions.

interpretation of this issue is that power outage problems can easily disrupt the continuity of the ring network whenever there is no fail-over mechanism exists. In this case, stations are forced to change the direction of the transmission for delivering the packets to their destination. It is apparent that the effective length of the path that packets have to be traveled towards their destinations will be increased dramatically. While in the case of exploiting the POR fail-over mechanism, there is no need to change the direction of the packet when a single or successive double power outage occurs. A similar discussion can be held about the average occupancy of the queues.

Since the chance of failure grows up by increasing the number of nodes, so it should be better to compare fail-over

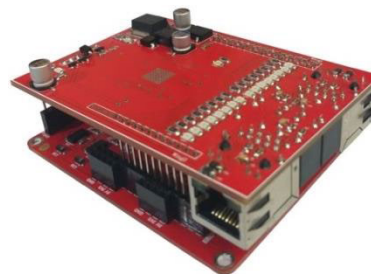


FIGURE 17. Designed test board using Xilinx Spartan 6 1x9 FPGA.

mechanisms in the networks with a higher number of nodes. Fig. 16 depicts comparative results of proposed and resilient networks under different failure conditions in a network

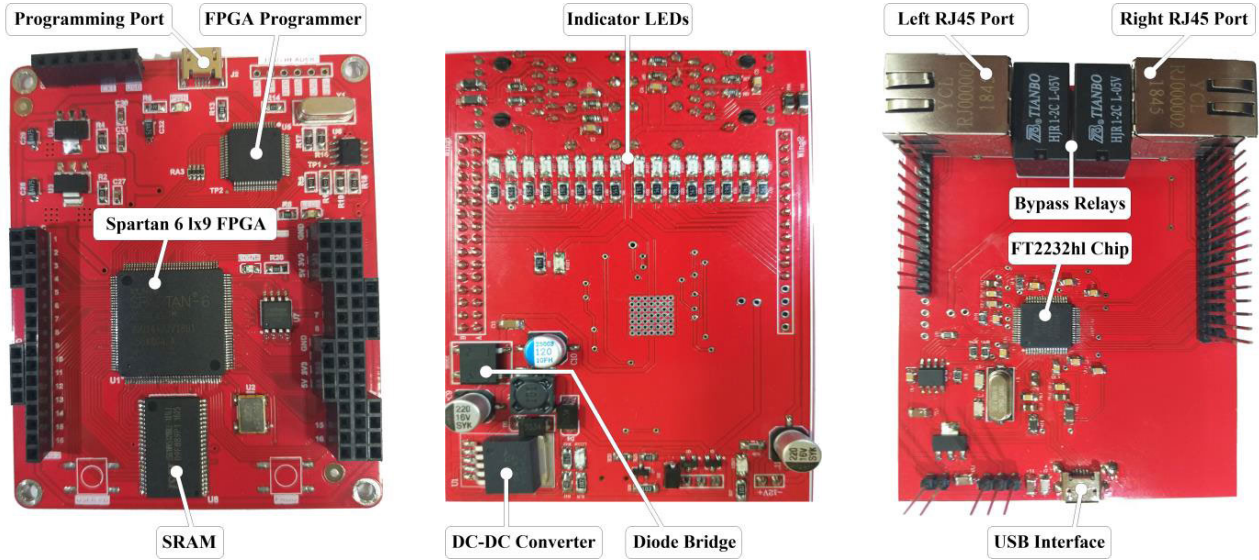


FIGURE 18. Different parts of the designed test board.

TABLE 1. Network Parameters for resilient ring NIC.

N_s	Resilient ring NA		Proposed ring NA	
	N_Q	N_H	N_Q	N_H
8	0.25	5.44	0.18	3.88
16	0.37	8.07	0.18	5.49
24	0.7	10.7	0.47	6.97
32	0.9	13.37	0.55	8.25
40	1.12	16.08	0.63	9.4
48	3.9	19.5	2.18	10.76
56	4.9	24.6	2.55	12.88
64	17.2	31.46	8.91	15.7

which is comprised of 64 nodes. As the proposed method prevents single or successive double failure from partitioning the network, it is expected that this method will gracefully handle faulty conditions. The simulation results are evidence of this claim.

VI. FPGA IMPLEMENTATION

To verify the performance of the previously described fail-over mechanism, a test board is designed using Xilinx-Spartan6-1x9 FPGA which is shown in Fig. 17. Physical and MAC sub-layers of RNA are implemented on this board.

The major parts of the test board are illustrated in Fig. 18. Two RJ-45 connectors are considered for data transferring with left and right neighbors. The PoE transformers and common mode filters are integrated into this connector. Also, two N.C. relays are included as a bypass switch. Moreover, a DC-DC converter is embedded for reducing PoR high-level voltage to the desired level. For data communication between computer and FPGA, an FT2232hl high-speed USB chip is used as a data interface.



FIGURE 19. A simple setup for qualifying the performance of the proposed fail-over mechanism.

In order to examine the functionality of the test board, a simple setup containing five stations is prepared as shown in Fig. 19. In the intended test scenario, stations are connected together through a 5 meters CAT6 patch-cord cable. Besides, only stations 2 and 5 are considered as a healthy system with their power connected, while the rest are faulty. Despite the middle failures between system 1 and 5, these systems can exchange information with each other. Thanks to the PoR fail-over mechanism, the required power of each faulty system is provided from a healthy neighbor.

VII. CONCLUSION

A new adaptive prioritization and fail-over mechanism were presented for ring network adapters. To overcome deficiencies of the fixed priority strategies, hop-count and the distance between source and destination have been included in the priority assignment procedure to manage network resources efficiently. In the proposed mechanism, each station gets an overview of the whole network topology and traffic status by inspecting the discovery messages and data packets. Thus,

the arrived traffic does not blindly put into the queues and transmission direction may be changed according to data packet delivering costs. Besides, a simple adaptive and efficient approach was used to guaranty fairly sharing of bandwidth between different classes of priorities and preventing starvation of a particular class. Also by applying the new fail-over mechanism, each station checks the functionality of its neighbors and provides the required electrical power of faulty neighbors to keep them alive for preserving the data path continuity. Several comparative OPNET simulations were done to assure the performance of the proposed mechanisms. The comparisons reveal that an excellent traffic management is performed and not only single failure but also any successive double failure scenarios cannot disrupt data path continuity by applying the proposed mechanisms in the ring networks. Moreover, a high-speed USB test board was designed using Xilinx-Spartan 6-lx9 FPGA to experimentally validate the efficiency of the proposed mechanism.

REFERENCES

- [1] K. McLaughlin, D. Burns, C. Toal, C. McKillen, and S. Sezer, "Fully hardware based WFQ architecture for high-speed QoS packet scheduling," *Integration*, vol. 45, no. 1, pp. 99–109, Jan. 2012.
- [2] K. S. Kahlon, "An embedded fuzzy expert system for adaptive WFQ scheduling of IEEE 802.16 networks," *Expert Syst. Appl.*, vol. 41, no. 16, pp. 7621–7629, Nov. 2014.
- [3] T. Balogh and M. Medvecky, "Performance evaluation of WFQ, WF²Q+ and WRR queue scheduling algorithms," in *Proc. 34th Int. Conf. Telecommun. Signal Process. (TSP)*, Aug. 2011, pp. 136–140.
- [4] R. R. Iraschko and W. D. Grover, "A highly efficient path-restoration protocol for management of optical network transport integrity," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 5, pp. 779–794, May 2000.
- [5] P. Ezran, Y. Haddad, and M. Debbah, "Availability optimization in a ring-based network topology," *Comput. Netw.*, vol. 124, pp. 27–32, Sep. 2017.
- [6] N. Du, H. Hu, and M. Zhou, "A survey on robust deadlock control policies for automated manufacturing systems with unreliable resources," *IEEE Trans. Autom. Sci. Eng.*, vol. 17, no. 1, pp. 389–406, Jan. 2020.
- [7] N. Du, H. Hu, and M. Zhou, "Robust deadlock avoidance and control of automated manufacturing systems with assembly operations using Petri nets," *IEEE Trans. Autom. Sci. Eng.*, early access, Apr. 30, 2020, doi: 10.1109/TASE.2020.2983672.
- [8] R. Jhavar and V. Piuri, "Fault tolerance and resilience in cloud computing environments," in *Computer and Information Security Handbook*, San Mateo, CA, USA: Morgan Kaufmann, 2017, pp. 165–181.
- [9] J. H. Saltzer and K. T. Pogran, "A star-shaped ring network with high maintainability," *Comput. Netw.*, vol. 4, no. 5, pp. 239–244, Oct. 1980.
- [10] R. Dixon, N. Strole, and J. Markov, "A token-ring network for local data communications," *J. Telecommun. Netw.*, vol. 3, no. 2, pp. 69–88, 1984.
- [11] J. H. Saltzer, K. T. Pogran, and D. D. Clark, "Why a ring," *Comput. Netw.*, vol. 7, no. 4, pp. 223–231, 1983.
- [12] W. Li, B. Guo, X. Li, Y. Zhou, S. Huang, and G. N. Rouskas, "A large-scale nesting ring multi-chip architecture for manycore processor systems," *Opt. Switching Netw.*, vol. 31, pp. 183–192, Jan. 2019.
- [13] I. V. Zhukovyts'kyi and V. M. Pakhomova, "Research of token ring network options in automation system of marshalling yard," Dnipropetrovsk Nat. Univ. Railway Transp., Dnipro, Ukraine, Tech. Rep., 2018, doi: 10.20858/tp.2018.13.2.14.
- [14] F. Zhao, N. Huang, and J. Chen, "Impact analysis of communication network reliability based on node failure," in *Proc. 2nd Int. Symp. Comput., Commun., Control Autom.*, 2013, pp. 0239–0242.
- [15] P. Gill, N. Jain, and N. Nagappan, "Understanding network failures in data centers: Measurement, analysis, and implications," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 41, no. 4, pp. 350–361, Aug. 2011.
- [16] A. L. Reibman and M. Veeraraghavan, "Reliability modeling: An overview for system designers," *Computer*, vol. 24, no. 4, pp. 49–57, Apr. 1991.
- [17] O. C. Ibe, "Reliability comparison of token-ring network schemes," *IEEE Trans. Rel.*, vol. 41, no. 2, pp. 288–293, Jun. 1992.
- [18] D. L. Boatwright, "Automatic bypass system for ring type local area network," U.S. Patent 4956836 A. Washington, DC, USA. Patent and Trademark Office, Nov. 9, 1990.
- [19] *IEEE 802.17 Resilient Packet Ring Working Group. 'Part17: Resilient Packet Ring (RPR) Access Method and Physical Layer Specifications [J/OL]*, Standard 802.17, 2004.
- [20] H. Gu, J. Zhang, K. Wang, Z. Liu, and G. Kang, "Enhanced fault tolerant routing algorithms using a concept of 'balanced ring,'" *J. Syst. Archit.*, vol. 53, no. 12, pp. 902–912, Dec. 2007.
- [21] S. Maniktala, *Power over Ethernet Interoperability Guide*. New York, NY, USA: McGraw-Hill, 2013.
- [22] J. Domzal, "Congestion control in flow-aware resilient multi-ring networks," in *Proc. Int. Conf. Comput., Netw. Commun. (ICNC)*, Feb. 2014, pp. 927–931.
- [23] H. Hu, M. Zhou, and Z. Li, "Supervisor design to enforce production ratio and absence of deadlock in automated manufacturing systems," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 41, no. 2, pp. 201–212, Mar. 2011.
- [24] H. Hu, M. Zhou, and Z. Li, "Liveness and ratio-enforcing supervision of automated manufacturing systems using Petri nets," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 42, no. 2, pp. 392–403, Mar. 2012.
- [25] R. Hammami, Y. H. Kacem, S. Souissi, H. Bellaaj, and A. H. Kacem, "Weighted priority queuing: A new scheduling strategy for Web services," *Int. J. Inf. Technol. Comput. Sci.*, vol. 9, no. 2, pp. 11–17, Feb. 2017.
- [26] M. Nurujjaman, S. Sebbah, and C. M. Assi, "Design of resilient Ethernet ring protection (ERP) mesh networks with improved service availability," *J. Lightw. Technol.*, vol. 31, no. 2, pp. 203–212, Jan. 2013.
- [27] B. Hall, K. Driscoll, M. Paulitsch, and S. Dajani-Brown, "Ringing out fault Tolerance. A new ring network for superior low-cost dependability," in *Proc. Int. Conf. Dependable Syst. Netw. (DSN)*, Jun. 2005, pp. 298–307.
- [28] M. S. Nirere, "Design of a single wire earth return (Swer) power distribution system and improvement of its voltage prole using capacitors," Ph.D. dissertation, Dept. Inf. Technol., JKUAT-PAUSTI, Juja, Kenya, 2018.
- [29] H. Hu, M. Zhou, and Z. Li, "Liveness enforcing supervision of video streaming systems using non-sequential Petri nets," *IEEE Trans. Multimedia*, vol. 11, no. 8, pp. 1457–1465, Dec. 2009.
- [30] H. Hu, M. Zhou, and Z. Li, "Liveness enforcing supervision of video streaming systems using nonsequential Petri nets," *J. Inf. Sci. Eng.*, vol. 25, no. 6, pp. 1–22, Nov. 2009.
- [31] A. H. Shuaib, T. Mahmoodi, and A. H. Aghvami, "A timed Petri net model for the IEEE 802.15.4 CSMA-CA process," in *Proc. IEEE 20th Int. Symp. Pers., Indoor Mobile Radio Commun.*, Sep. 2009, pp. 1204–1210.
- [32] M. Thirumaran, P. Dhavachelvan, D. Aishwarya, and K. Rajakumari, "Conventional usage of finite state machine over Petri net in Web service change management framework," *IERI Procedia*, vol. 4, pp. 99–109, Jan. 2013.
- [33] Z. Lu and H. Yang, *Unlocking the Power of OPNET Modeler*. Cambridge, U.K.: Cambridge Univ. Press, 2012.
- [34] M. Chen, Y. Miao, and I. Humar, *OPNET IoT Simulation*. Singapore: Springer, 2019.
- [35] X. Li, M. Ye, and W. Hao, *Network Modeling and Simulation With OPNET Modeler*. Xi'an, China: Xidian Univ., 2006.
- [36] M. Z. Shahabuddin and H. Hasbullah, "Experimental simulation of ZigBee's multi-hops wireless sensor network using OPNET," in *Proc. Int. Conf. Comput., Math. Statist. (iCMS)*. Singapore: Springer, 2017, pp. 37–44.
- [37] N. I. Sarkar, S. Gul, and B. Anderton, "Gigabit Ethernet with wireless extension: OPNET modelling and performance study," in *Proc. Int. Conf. Inf. Netw. (ICOIN)*, Jan. 2019, pp. 216–221.
- [38] A. Adas, "Traffic models in broadband networks," *IEEE Commun. Mag.*, vol. 35, no. 7, pp. 82–89, Jul. 1997.
- [39] B. Chandrasekaran, *Survey of Network Traffic Models*. St. Louis, Missouri: Washington Univ. St. Louis CSE, 2009.



AMIN KHALILZADEGAN was born in Urmia, Iran, in 1987. He received the B.S. and M.Sc. (Hons.) degrees in electrical engineering from Urmia University, Urmia, in 2009 and 2012, respectively, where he is currently pursuing the Ph.D. degree in electrical engineering. His research interests include analog and digital integrated circuit designs, computer networks, fuzzy systems, and neural networks.



MOHAMMAD ZAREI was born in Urmia, Iran, in 1987. He received the B.S. and M.Sc. degrees in electrical engineering from Urmia University, Urmia, in 2009 and 2012, respectively, where he is currently pursuing the Ph.D. degree in electrical engineering. His research interests include analog and digital integrated circuit designs, computer networks, fuzzy systems, and neural networks.



KHAYROLLAH HADIDI (Member, IEEE) received the B.S. degree from the Sharif University of Technology, Tehran, Iran, the M.S. degree from Polytechnic University, New York, and the Ph.D. degree from the University of California at Los Angeles, Los Angeles, all in electrical engineering. He is currently with the Electrical Engineering Department and also with the Microelectronics Research Laboratory, Urmia University, Urmia, Iran. He holds 4 U.S., U.K., and German patents (issued) and 12 Japanese patents (pending). His research interests include high-speed high-resolution data converter designs, wide-band integrated filter designs, and nonlinearity analysis and improvement in analog circuits.

• • •