Improvement in QCN with BIER for Chassis Topology

Shivani Sharma
Networking and Switch BU
NITW | Marvell Semiconductor,
Bangalore, India
shivanis@marvell.com

Ramalingaswamy Cheruku
Department of CSE
National Institute of Technology,
Warangal, Telangana, India
rmlswamy@nitw.ac.in

Yuval Peled
Networking and Switch BU
Marvell Semiconductor
kiryat Ono, Israel
yuvalp@marvell.com

Prakash Kodali*
Department of ECE
National Institute of Technology,
Warangal, Telangana, India
kprakash@nitw.ac.in

Venkatesh Gudipadu
Department of ME
National Institute of Technology,
Warangal, Telangana, India
venkatesh@nitw.ac.in

Shai Savir
Switching Architecture
Marvell Semiconductor
Santa Clara, USA
shaisa@marvell.com

***Abstract*— In the present era, networking has become an essential requirement as individuals seek interconnectivity and efficient means to share data rapidly. When discussing the swift exchange of information, a network characterized by high speed and minimal congestion is favored by all. In networking, Switches play a very important role to transfer packets of information from source to destination, as routers do in any network topology. For switches to exchange data from the source port of one switch to the destination port of another switch, it requires a special arrangement where every switch is connected to another using chassis topology. When packets transfer, in certain cases, it leads to congestion in the output port due to heavy traffic. A method for resolving the congestion is by signaling to the source port(s), sending the traffic causing the congestion, to throttle the traffic towards the congested port. One such signaling may be based on Quantized Congestion Notification (QCN). As the packets might be received from different sources, the notification needs to be replicated to a subset of the source ports sending to the congested target. This paper proposes the analysis of QCN signaling within a chassis topology, based on different cases using an advanced replication architecture i.e., BIER (Bit Index Explicit Replication) to study its effect to achieve high throughput and low latency of the whole system and proposes a novel approach called periodic QCN for handling persistent congestion.**

Keywords*— Networking, Switch, BIER, QCN, Periodic QCN, Chassis topology.

I. Introduction

As industries grow, in the technical world they require efficient networks with high speed and low latency to establish advanced connectivity. Within various network topologies, routers play a crucial role in directing multiple data packets from the source network to the destination network. On the other hand, switches are employed for internal network communication and facilitate the efficient transmission of data within a network. To better understand the difference between switch and router from a technical standpoint, switches operate at the L2 layer of OSI i.e., the Data Link layer and routers at the L3 layer of OSI i.e., the Network layer. A network switch examines the destination address of each packet of data that enters one of its ports, does the necessary checks, and then transfers the data to the appropriate destinations. A packet with a single destination is referred to as a known unicast, while multiple destinations are referred to as BUM (Broadcast, Unknown unicast and Multicast). In networking, BUM packets are replicated from the source to several output ports/routers and are forwarded with respect to multicast states achieved by tree building protocols. This process is complex and may result in consuming larger BW. A new approach got introduced called BIER (Bit Index Explicit Replication) with improvised architecture for managing BUM traffic. The absence of a sper-flow multicast state in the network core is a benefit of BIER, hence is used in this paper. While communicating between ports, many times cases arise when multiple input ports try to send data packets to a single output port when bandwidth transmitted towards a port is more than the line rate of that port which leads to congestion. Therefore, a congestion Control mechanism is introduced called QCN. In some cases, the congestion takes more time to resolve due to varied interlink speeds between switches and thus we introduce a novel approach called as periodic QCN mechanism. This paper deals with the implementation of switches in a chassis topology and analyzes how a Quantized Congestion Notification- QCN is sent to relevant source ports with and without BIER implementation and analyzes overall QCN throughput and latency.

II. Background and Related Works

A. Chassis Topology

Networking Data Center Networks (DCNs) are being widely discussed both in industry and the research community as they are the ground structure for supporting cloud computing [1]. Therefore, data center networks have increased demand for high throughput and low latency as the technical advancement aims to develop better results for these goals [2]. Hence, within a network multiple ports are connected using switches which can either be stacked switches or chassis switches. Chassis switches in contrast to the fixed configuration switch operate as a single integrated system and offer software and/or hardware features additionally, which are unavailable on a stacked switch [3].

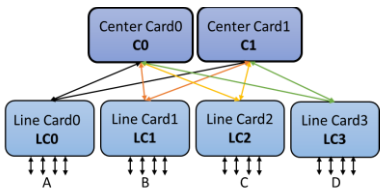

Fig. 1. Chassis topology

In a Chassis Topology, Center Cards are used to interconnect line cards. An Ingress line card refers to the line card from which a packet is received, whereas an egress line card refers to the target line card of that same packet. A Port on an Ingress Line card is referred to as Ingress port and a port on the egress line card is referred to as egress port Following Fig1. represents the basic diagram to explain the Chassis topology. In Fig1. Line Cards are mentioned as LC0, LC1, LC2, LC3 having multiple bidirectional ports A,B,C,D. These ports act as either ingress ports or egress ports based on the packet's flow in the Chassis. To forward the data packets received and processed by ingress Line Cards, Center Cards connect the line cards and forwards the traffic to egress Line Card ports. Line Cards (LC) consist of ports connected to devices known as cascade ports and the ports connected with center cards within the chassis. As we refer to chassis topology there are two main types of chassis functionality, first is distributed chassis, where the forwarding decision takes place at the LC, these center cards are referred to as Fabric cards. The second case is a centralized chassis, where forwarding decision is made in the center card, these cards typically referred as the Central Switch. Here, the paper deals with distributed chassis i.e., the packet forwarding is initiated by the Line cards. The connectivity between the LCs and the Center cards is ideally maintained through several links such that on every LC, the aggregated Network(NW) bandwidth(BW) < aggregated center links BW and for a single port: Center Card link port BW > Highest NW port rate.

B. *Multicasting*

Multicast is a method of group communication where the source/ingress port sends data one-to-many and many to-many (multiple) receivers/ egress ports simultaneously. The significance of multicasting is further supported by the sheer number of technologies that offer point-to-multipoint communication technology as a service. [4]. Multicast routing is about building forwarding trees from the sender 'S' to the group 'G' of receivers or listeners and each forwarding state has single incoming interface and outgoing interface list. Here, routing takes place based on Shortest Path Tree (SPT) or other such methods. In an SPT each path from the root to all the end nodes is the shortest possible in some sense [5]. Each Router along the Domain requires to maintain relevant states as represented below in Fig2.

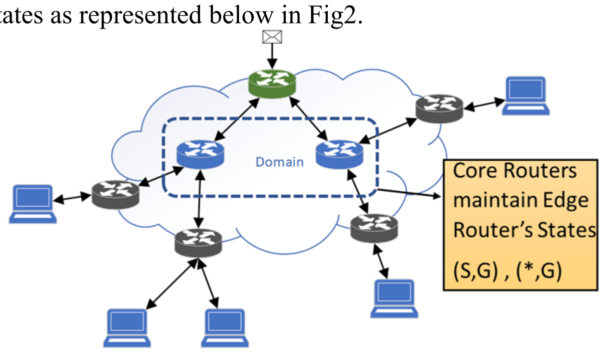

Fig. 2. Multicast Routing

It is necessary to understand that once the data packet is received by the source it forwards them to core routers which based on source and forwarding state replicates those many packets and forwards further. In Ingress Replication all the required replications take place from initial router only and consumes a lot of bandwidth, therefore BIER got introduced.

C. *BIER(Bit Index Explicit Replication)*

BIER (Bit Index Explicit Replication) multicasts a packet by assigning every edge device a Bit Mask position. It is an alternative method of multicast forwarding. It does not require any multicast-specific trees and corresponding tree-building protocols maintained in the core network. Here, instead of sending Multicast packet to each destination IP address (Receiver IP address), Domain Core routers maintain edge router's states. Basically, it sets the Bit positions and saves the amount of data plane states in the core network. It uses Unicast transport in the underlay and performs Bit Masks to avoid loops. B.I.E.R. is a multicast mechanism where each multicast data packet has a destination bit-string that the source of the data expressly defines and indicates the set of destinations. Each bit in the destination-bit-string corresponds to a network destination to which the multicast data packet will be delivered if the bit is set [6]. Overall, it makes multicast much simpler and lighter. BIER domain [7] consists of the Bit-Forwarding Routers (BFRs) along with Bit-Forwarding Ingress Router (BFIR) and Bit Forwarding Egress Router (BFER) located at the edge of BIER multicast domain. Mainly a single device may need to incorporate all 3 functionalities (BFIR. BFR and BFER) of one or multiple domains. When considering chassis topology with BIER functionality, We consider Line cards to imitate the behavior of BFIR and BFER whereas the Center cards imitate the behavior of a BFR . When a packet is to be forwarded between line cards, the forwarding logics are maintained by BIER table. In reference to Fig.1 considering the topology to be BIER capable with 8 Line Cards and 4 Center cards, if a packet is to be forwarded from Line Card7 port 0(LC7_0) to Line Card 2 port2 (LC2_2), a Bit Forwarding table is created. The operation involved to prepare the table I is done by performing AND operation on all the BFR-ID with same BFR-NBR (Neighboring BFR) which results in Forwarding-Bit Mask (F-BM).

TABLE I. F-BM TABLE FOR BIER

BFR-ID	F-BM	BFR-NBR
1(00000001)	00000011	C0
2(00000010)	00000011	C0
3(00000100)	00001100	C1
4(00001000)	00001100	C1
5(00010000)	00110000	C2
6(00100000)	00110000	C2
7(01000000)	11000000	C3
8(10000000)	11000000	C3

D. *Congestion in ports.*

During the process of sending data packets from ingress ports to egress ports many times condition arises, when multiple ingress ports try to send data packets to a single port of a Line card, in such situation the queue present in the egress port gets overfill and the fill level gets beyond a set threshold value which triggers the presence of congestion on the egress port. Once a congestion occurs a QCN- Quantized Congestion Notification is relayed to the source ports sending traffic towards the congested point. At the ingress LC, the congestion is handled by issuing PFC- Priority Flow Control [8] which temporarily stops the link partner from sending traffic of the

flow causing the congestion. PFC has a fixed standard which pauses the flow of ingress data packets for a minor duration of time, so that congestion gets resolve and ingress port can again start sending the data packets to egress port. As per standard, PFC message has a pause quanta of 16 bits and each quanta translates the time taken to transmit 512 bits depending upon the port speed-
–10Mbps port, Max PFC pause time is ~1.3s
–25Gbps port, Max PFC pause time is ~1.3ms
–100Gbps port, Max PFC pause time is ~335µs
Congestion Control Mechanism- QCN is transmitted back to the ingress Line Card (iLC) where it is terminated, and a PFC(Priority Flow Control) is generated towards the port(s) from which the congested traffic originated from. In a Distributed Chassis, QCN will be sent from the egress Line Card (eLC) towards the ingress Line card (iLC). A component experiencing the congestion will generate a QCN message containing iLC Source Device and iLC Source Port. The iLC receiving the QCN will terminate the QCN and create a PFC message towards the iLC Source Port stopping the data packet flow from ingress port. Following Fig3 represents the flow of complete process from the point when congestion takes place to the point when it is resolved.

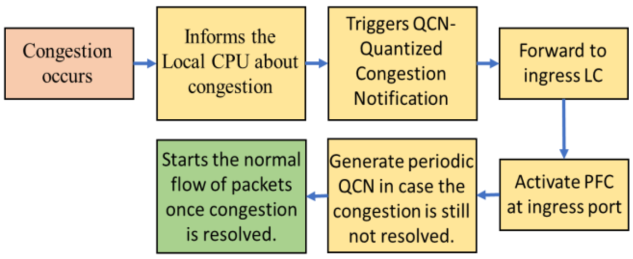

Fig. 3. QCN resolution flow chart

This paper deals with a distributed chassis topology with Inevitable congestion i.e., when data packets from multiple ports of Line card(s) try to access a single egress port. There are 2 main cases which comes into comparison to understand the throughput variation when a QCN is sent back to ingress ports from the Local CPU of Line card of egress port-

- Without BIER implementation
- With BIER implementation
  – Both Line Card and Center Card are BIER capable.
  –Only Line Card is BIER capable.

Following representation gives a theoretical understanding of the cases mentioned above. Hence, the 3 scenarios in Fig. 4 depict when congestion is experienced at the egress line card as Q_fill (queue fill) level becomes more than the queue threshold level. Therefore, a Queue Congestion Notification is sent to all the ports which are trying to send data packets from ingress ports of Line cards. In Fig4 data packets from 4 ports, 2 from LC0 and 2 from LC1 are sent to a port in the egress Line Card via center card. Fig 4(A) represents the scenario without BIER implementation.
Here, 4 replications to each ingress port are made from the eLC (egress Line Card) as congestion is triggered, where all the replications are forwarded to ingress port via center card one after another. As, all the notifications are sent to the ingress ports at different times, packet flow from one port is analyzed and QCN is sent then the next port is analyzed. Fig 4(B) represents the scenario considering with BIER approach (Both Line Card and center card are BIER capable). Here as eLC is BIER capable so it sends only one QCN to Center card because all the data packets from iLC have nearest neighbor as Center card. From center card 4 replications of the QCN is made and the replicated QCN from center card are all forwarded at a same time to all ingress ports. Fig4(C) represents the scenario with BIER approach (Only Line Card is BIER capable). Here, 4 replications to each ingress port are made from the eLC when congestion is triggered as Center card acts dumb and does not have capability to perform replications to respective ingress ports. All the replications are directly forwarded to ingress port via center card and all the notifications sent to ingress ports are done at same time, hence saving time. QCN flow from center card to Ingress Line Card is depicted by the green dots in Fig4(a),(b),(c).

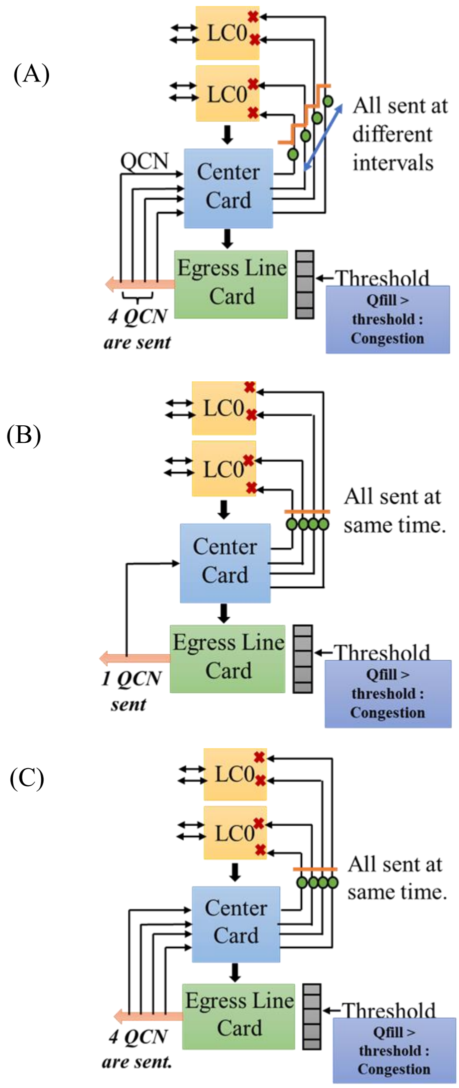

Fig. 4. QCN Flow Diagram

E. Periodic QCN(Quantized Congestion Notification)

This paper proposes periodic QCN in place of event driven QCN. As, QCN is generated based on sampled packets, an issue arises when congestion may not be resolved by event driven QCN mechanism. For example, when packet flow takes place from fast port to slow port i.e., when fast port receiving PFC might not stop for long enough for congestion of the slow port to be resolved. Periodic QCN is proposed to avoid these issues by sending additional QCN periodically until the congestion subsides.

F. Related Works

Numerous efforts have been undertaken to enhance the throughput of switch-based systems to reduce latency, and develop effective solutions for managing congested ports. Many researches have aimed to do so with varied approaches. Chang Ruan analyzes QCN in Data center networks for congested switches using NonLinear Control theory [10]. They have used it to analyze queue oscillation when congestion takes place. However, it only proposes functional method focusing only on queue oscillation during congestion. Jungie Jang with his co-researchers' state that PFC being coarse-grained protocol may lead to head of line blocking and proposed P4QCN based on P4 framework [11]. Even though it proposes a flexible data plane compatible with IP-routed network but need to evaluate congestion and packet drops while dealing with switches with varied link speed. Daniel Merling discusses appropriately about Switch throughput analysis with respect to 100 Gbps ports comparable to the condition maintained for analysis in this paper and performs testing on a Tofino board to reveal the challenges faced during performance evaluation because of recirculation due to P4 program [8]. He successfully describes the BIER based switch and its throughput but does not focus about the congestion which may take place within a switch due to inevitable congestion. Bong-Hwan Oh elaborates about a novel flow control mechanism called PFC- Priority Flow Control to resolve transmission disruption and packet loss in SDN hardware switches [7]. It gives detailed explanation about the PFC Operations and modes while dealing with traffic flows, but it is seen that when the link speed between the devices varies and PFC is triggered due to congestion at egress port/device, traditional approach would fail to deliver 100% packet flow i.e., in certain cases loss due to packet drop may occur. Olufemi Komolafe in his paper have described about the evolution of data center architectures for Layer 2 networks i.e., Switches along with an innovative approach to circumvent issues due to multicasting [12]. It talks about BIER being a better approach theoretically which as a proof has been applied in this paper. Mohammad Alizadeh in his paper have described about congestion control mechanism QCN from basic and how it proves to be a good solution to the congestion issue as per IEEE standards [9]. It even articulates method to stabilize control loops as lag increases, however it is practically infeasible in ethernet context. P. Bhagwat with co-authors has examined performance (mainly throughput) by delivering dependability and multicast delivery across a fast wide area network [13]. Whereas, this paper validates BIER approach as better than traditional multicast.

III. QCN ANALYSIS

A. Specification and Constraints

1) *PFC max pause time for 100Gbps port for pause quanta of 16 bits., provided each quanta translates to the time taken to transmit 512 bits on port.

Time = (data size in bits)/ (Link speed in bits per second)
= 512/ 10^{11} bits per sec
= 5.12 nsec (nano second)

Therefore, PFC pause time = 5.12nsec * 0xffff(i.e.,2^{16}) = 335.5μsec (micro second)

2) For congested queue [9] let congestion measure be represented as **Q_c**, **Q** denote the instantaneous queue-size, **Q_{old}** denote the queue-size when the last feedback_message was generated, **Q_{eq}** denotes desired operating point in a queue without congestion and **Fb** in a feedback message to the source of the sampled packet.

Therefore, $Q_{off} = Q - Qeq$ $Q\delta = Q - Qold$

$$Fb = -(Q_{off} + wQ\delta). \quad (1)$$

In Equation (1) 'w' is a Nonnegative constant and Fb captures a combination of queue-size excess (Q_{off}) and rate excess ($Q\delta$).

Therefore, Q_c is given by the formula -

$$Q_c = -(Q_{off} + wQ\delta) \quad (2)$$

$Q_c < 0$: CN is present
$Q_c >= 0$: No CN
Fb < 0: Congestion message is reflected back to the source as a function of |Fb|. The feedback message contains the value of Fb, quantized to 6 bits.
Fb ≥ 0: No congestion and no feedback messages are sent.
Following table II provides all the specifications considered for the analysis.

TABLE II. SPECIFICATION AND CONSTRAINTS

S. No	Specification and Constraints
1	Chassis- 8 Line Cards (16 ports each- towards center card and deice port)
2	Data packets from a Line Card are sent from ingress ports to center cards in following manner- Port0-Port3: C0 (Center Card0) Port4-Port7: C1 (Center Card1) Port8-Port11: C2 (Center Card2) Port12-Port15: C3 (Center Card3)
3	Queue depth in Line cards- 100
4	Queue Threshold value (To detect congestion in port) - CN_{Th} = 20.
5	Port Speed = 100 Gbps
6	Lines per port = 9(each port transmits 512 bits/sec)
7	PFC pause time = 335.5μs*
8	Packet flow is considered with sufficient interpacket gap such that QCN received can be monitored accurately.

B. Throughput analysis

To analyze the throughput and latency, study is done. Following shows the number of QCN messages required for a chassis implemented without BIER mechanism and for a chassis implemented with BIER mechanism. To make a comparative analysis, 3 cases with varying number of ingress Line card ports transmitting the packets to One egress port of egress Line Card is considered. All the cases are studied once the queue in line card with egress port gets congested and CN_{Th} >20.

1) Case(A): When from all the 8 LC's 0-7, 3 ports from each LC transmits packet to only one output i.e., port 0 of 7th LC.

Data packets from three ports from each line cards from LC0-LC6 are sent to egress Line card LC7.

Ingress Line card ports are as follows-
LC0_1, LC0_2, LC0_5, LC1_0, LC1_1, LC1_8, LC2_1, LC2_2, LC2_13, LC3_0, LC3_12, LC3_2, LC4_1, LC4_3, LC4_5, LC5_10, LC5_6, LC5_14, LC6_5, LC6_11, LC6_9
Egress Line Card Port-LC7_3.

2) Case(B): When few LC's from 0-7 with 3 ports from each LC transmits packet to only one output i.e., port 3 of 7th LC.

Data packets from three ports from each line cards from LC1, LC2, LC5, LC6 are sent to egress Line card LC7.
Ingress Line card ports are as follows-
LC1_0, LC1_1, LC1_2, LC2_1, LC2_2, LC2_3, LC4_5, LC5_10, LC5_6, LC5_14, LC6_5, LC6_11, LC6_9
Egress Line Card Port-LC7_3

3) Case(C): When 6 ports from only one LC transmit packets to one output i.e., port 0 of 7th LC.

Data packets from six ports from only one line card- LC0 are sent to egress Line card LC7.
Ingress Line card ports are as follows-LC1_0, LC1_1, LC1_2, LC1_11, LC1_9, LC1_15
Egress Line Card Port-LC7_3
Now, based on above 3 cases throughput is analyzed for 2 major criteria's i.e., With BIER implementation and Without BIER Implementation.

CASE1: Without BIER –

In this case once congestion is triggered, for every congested packet in all the 3 cases, separate notification is sent from local CPU to every ingress port. Therefore, for case (a) 21 notifications are forwarded, case (b) 12 notifications and for case (c) 6 notifications one after another. Hence, if congestion is faced by the egress port, then only the incoming first packet will get know about the congestion to which QCN will be processed by Local CPU of egress Line Card and remaining ports will be unknown about the congestion happening at the egress port till the data packet sent is processed by the local CPU. This acts as a disadvantage and takes a lot of time to notify every port about the congestion as it analyzes the input packet one by one.

CASE2.1: With BIER (Both Line card and Center card are BIER capable) –

Here, QCN is sent to source ports with BIER capable LC's and Center cards. Once congestion takes place, only necessary replications are made to center cards which replicates to the ingress ports. Here-

1) case(a): 4 replications(reps) to center cards which further replicates as- C0: 10 reps, C1: 4 reps, C2: 4 reps, C3: 3 reps.
2) case(b): 4 replications to center cards which further replicates as- C0: 4 reps, C1: 2 reps, C2: 4 reps, C3: 2 reps.
3) case(c): 3 replications to center cards which further replicates as- C0: 3 reps, C1: 0 reps, C2: 2 reps, C3: 1 reps.

CASE2.2: With BIER (Only Line Card is BIER capable)-

In this case once the egress Line Card faces congestion then the Local CPU in Line Card with egress port analyzes all the ingress Line Card ports trying to send data packets at once and make respective number of QCN replications which further is forwarded to respective Center cards whose function is simply to forward the notification to respective ingress port. Advantage of this case over non-BIER based implementation is that the number of Notifications to be replicated are known at once and sent at once. This reduces the time and number of cycles consumed to send out all the notifications to the required ports.

C. Timing Analysis

After examining various scenarios to analyze throughput, the time required for transmitting a congestion notification from a congested port to the source port attempting to send packets has been evaluated here. It is observed from all the three cases that time required for one QCN notification to reach ingress Line Card Port from Egress Line Card port is -
T_{QCN} = 12µs (micro second.)
(Assuming no extra time delay between sending QCN)

1) Without BIER and without Periodic QCN:

Time consumed in sending a QCN to source port in all the sub cases are as follows. As, it takes 12µs for one notification to reach, then for 'n' packets it will take (n * 12)
Therefore,
case(a)- Total Time = 21 * 12µs = 252µs
case(b)- Total Time = 12 * 12µs = 144µs
case(c)- Total Time = 6 * 12µs = 72µs
PFC pause time is 335µs for 100Gbps so, after this time ingress port again starts sending the packets to egress port and just in case the egress port remains congested and has not resolved it again needs to send the QCN to all the ports which will take additional time as mentioned above for every case.

2) With BIER and with Periodic QCN:

Time consumed in sending a QCN to source port in all the cases remains same. As, it takes 12µs for one notification to reach, therefore here irrespective of number of QCN message to send to ingress ports, time remains same.
Therefore, Time taken for QCN message to send in all three cases a, b, c is 12µs.
Same results are seen in both scenarios when both Line Card and Center card are BIER capable and when only Line Card is BIER capable. With the proposed idea of periodic QCN it is captured that depending upon how fast the egress port congestion is resolved, the ingress port resumes forwarding the packets. Therefore, faster the Congestion subsides sooner the packet flow initiates. In periodic QCN there is no need for the entire PFC pause time to end to initiate the packet flow but, can start sending the ingress packets once the congestion subsides.

IV. Validation

This section validates the overall analysis done for QCN throughput and latency over two cases i.e., with BIER approach- Both LC and Center card are capable and Without BIER approach.

TABLE III. THROUGHPUT ANALYSIS

Center card	**With BIER**			**Without BIER**		
Case->	A	B	C	A	B	C
0	1	1	1	10	4	3
1	1	1	0	4	2	0
2	1	1	1	4	4	2
3	1	1	1	3	2	1

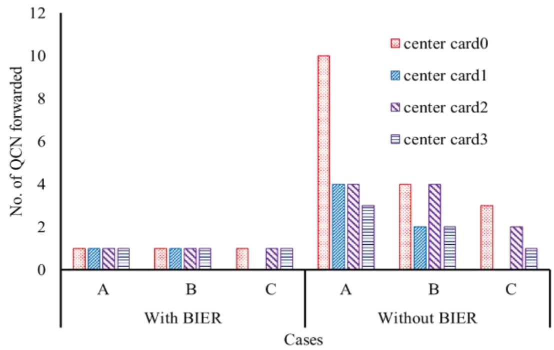

Fig. 5. Throughput analysis graph

The investigation of how QCN throughput is impacted is tabulated in the Table III. and graph in Fig.5 represent for 3 cases. With respect to all the cases with and without BIER, following is the table formed which describes the number of QCN messages sent when congestion is reported and triggered by local CPU of line card with congested port to all the ingress ports which were trying to send packets via center card. Fig. 5 graph represents the table clearly that in case with BIER approach for all the cases, number of QCN sent remains constant and irrespective of number of source ports to send the notification, only 1 notification per center card is forwarded. Whereas, in Without BIER multiple notifications are forwarded resulting to increase in throughput. Latency or time taken to transmit a QCN to source ports is captured in the following table IV and its respective graph in Fig.6.

TABLE IV. QCN LATENCY ANALYSIS

Cases	With BIER (In µsec)	Without BIER (In µsec)
A	12	252
B	12	144
C	12	72

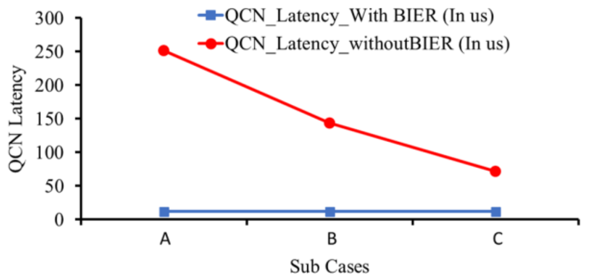

Fig. 6. Latency Analysis Graph

V. Conclusion

QCN analysis is discussed as a congestion control mechanism whose throughput analysis is done in this paper by considering BIER architecture in a distributed chassis topology for switches. It can widely be used in Data Center Networks (DCN) to have faster exchange of data packets between ethernet switches in place of traditional approach to resolve congestion. Through the utilization of appropriate tables and graphs, this paper substantiates the superiority of the proposed implementation, which leverages BIER (Bit Index Explicit Replication), over the traditional non-BIER architecture in achieving enhanced throughput and effectively managing unavoidable congestion. Both approaches have been implemented and thoroughly compared in terms of throughput and latency. In the context of BIER (Bit Index Explicit Replication), it can be inferred that the configuration combining both LC (Line Card) and Center card capable architecture yields the most favorable outcome. However, it is worth noting that in practice, there arises a requirement to replace generic center cards in the network topology with BIER-capable ones, which may entail a higher investment. On the other hand, a setup consisting solely of LC-capable architecture also demonstrates its advantages by achieving low latency without compromising on throughput with respect to traditional approach. In the current market, the proposed architecture presented in this paper offers significant advantages over the traditional approach. This is primarily due to the utilization of BIER for chassis topology and the incorporation of periodic QCN which represents a novel and innovative approach.

References

[1] Zhu, Y., Eran, H., Firestone, D., Guo, C., Lipshteyn, M., Liron, Y., Padhye, J., Raindel, S., Yahia, M.H. and Zhang, M., 2015. Congestion control for large-scale RDMA deployments. *ACM SIGCOMM Computer Communication Review*, *45*(4), pp.523-536.

[2] Handley, M., Raiciu, C., Agache, A., Voinescu, A., Moore, A.W., Antichi, G. and Wójcik, M., 2017, August. Re-architecting datacenter networks and stacks for low latency and high performance. In *Proceedings of the Conference of the ACM Special Interest Group on Data Communication* (pp. 29-42).

[3] Chabarek, J. and Barford, P., 2013, June. Energy Audit: Monitoring power consumption in diverse network environments. In *2013 International Green Computing Conference Proceedings* (pp. 1-10). IEEE.

[4] Vella, J.M. and Zammit, S., 2012. A survey of multicasting over wireless access networks. *IEEE Communications Surveys & Tutorials*, *15*(2), pp.718-753.

[5] Nguyen, U.T. and Xu, J., 2007. Multicast routing in wireless mesh networks: Minimum cost trees or shortest path trees?. *IEEE Communications Magazine*, *45*(11), pp.72-77.

[6] Desmouceaux, Y., Clausen, T.H., Fuertes, J.A.C. and Townsley, W.M., 2018. Reliable multicast with BIER. *Journal of Communications and Networks*, *20*(2), pp.182-197.

[7] Merling, D., Lindner, S. and Menth, M., 2020, April. Comparison of fast-reroute mechanisms for BIER-based IP multicast. In *2020 Seventh International Conference on Software Defined Systems (SDS)* (pp. 51-58). IEEE.

[8] Oh, B.H., Vural, S., Wang, N. and Tafazolli, R., 2018. Priority-based flow control for dynamic and reliable flow management in SDN. *IEEE Transactions on Network and Service Management*, *15*(4), pp.1720-1732.

[9] Alizadeh, M., Atikoglu, B., Kabbani, A., Lakshmikantha, A., Pan, R., Prabhakar, B. and Seaman, M., 2008, September. Data center transport mechanisms: Congestion control theory and IEEE standardization. In *2008 46th Annual Allerton Conference on Communication, Control, and Computing* (pp. 1270-1277). IEEE.

[10] Ruan, C., Zhang, T., Li, H. and Xi, Y., 2020. usinglinear control analysis of quantized congestion notification in data center networks. *IEEE Access*, *8*, pp.125401-125411.

[11] Geng, J., Yan, J. and Zhang, Y., 2019. P4QCN: Congestion control using P4-capable device in data center networks. *Electronics*, *8*(3), p.280.

[12] Komolafe, O., 2017, May. IP multicast in virtualized data centers: Challenges and opportunities. In *2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)* (pp. 407-413). IEEE.

[13] Bhagwat, P., Mishra, P.P. and Tripathi, S.K., 1994, June. Effect of topology on performance of reliable multicast communication. In *Proceedings of INFOCOM'94 Conference on Computer Communications* (pp. 602-609). IEEE.