RESEARCH ON ENHANCING THE PERFORMANCE OF DATA SWITCHED NETWORKS USING THE KNOCKOUT PRINCIPLES AND PARALLEL FABRICS

[*1]Akpan Idongesit.T.,[2]Prof Eneh I.I.,[3]Princewill Eneh, [4]Ohagwu Walter A.
Department of Electrical and Electronic Engineering
[1,2,3,4]Enugu State University of Science and Technology, Enugu, Nigeria

[1*]Corresponding Author: idongesitakpan565@gmail.com
Tel: +234 8035529637

ABSTRACT

The accomplishment of high demand of data via the internet transmission medium when fabrics Data are of higher demand during peak hours, has led to packet retransmission. This research carries out the throughout performance of data switched network under study and developed data switch model that can reduce the packet loss by utilizing knockout principle and parallel fabrics. The knockout principle states that the likelihood of a packet being dropped due to contention can be made arbitrarily small with sufficiently large R number of packets and furthermore, for a given loss probability requirement, there exists on R number of packets that is independent of the switch size N such that the actual loss probability is not larger than the requirement. MATLAB simulation was carried out to show how the proposed technique can improve the performance of the data switched network in terms of throughout. The simulation result shows that the improvement achieve when knockout switch and routing fabrics was simulated on the data switched network is 35:027 bytes /sec as compared to the existing channel grouping technique.

Keywords: Knockout principle, parallel fabric, Data switched network.

1. Introduction

Telecommunications switched networks has been the importance of file transfer from one host connection to the other. The circuit switched telephone network has been used transport such applications resulting in the development of packet switched networks to support computer communications traffic. Packet switching has become the predominant technology, for high speed data networks and has begun to be used for applications like voice which have traditionally relied on circuit switching. A packet switches because of its significant importance consists of input and output ports connected to a switch fabric. The function of a packet switch is to transfer packets from the input ports to the appropriate output ports based on the addresses contained within the packets (in the packet headers). There are a few different options that are available for the design of the switch fabric. An option is a crossbar architecture in which one packet can be transferred to the output ports from each input port simultaneously provided the set of output ports are distinct.

The key measure of the performance of a packet switch as its aggregate throughout which is the net rate at which it switches packets from its inputs ports to its output ports. In order for the switch to support the incoming traffic without dropping packets, the aggregate throughout of the switch must be at least equal to the sum of all the packet arrival rates on the input links. The aggregate throughout of the bus (shared medium) architecture is limited by the speed of the bus. On this architecture, the bus speed must be at least equal to the sum of all the input link speeds in order for no packets to be dropped. On general, the difficulty of significantly speeding up the switch fabric when links themselves are already running at very high speeds, is one of the main reason for the renewed interest in input queuing switches (Mckneown 2016) despite their well known deficiencies.

In implementation of this design of fast packet switch one major issue Address in this dissertation is the models crossbar like switch fabrics, with and without internet buffering. As number of input increases speed of buffer at output must increase (like shared medium switch).

AIM & OBJECTIVES: The objectives of the study are to design a knockout switch for handling multiple data traffic which when multiple inputs have information for one output, crossbar controller needs to determine when each should be switched through.To design a self routing fabrics for very large high performance switches

2. Design Self Routing Fabrics for Very Large High Performance Switches

Switch fabrics are packets switched in a single step between the fabric inputs and output or multi stage where the fabrics are constructed by cascading several single-stage switching modules, called “switching elements” (SEs). Multistage switches are space division architectures ie switching are performed by interconnecting an input to an output line. Multi stage interconnection networks (MINS) constitute a large and special loss of packet switches. The simplest form (in terms of design) of on interconnection network is the crossbar fabric. A crossbar is based on a single-stage network of cross points (intersections) forming a fully interconnected grid or switching matrix. The cross points are implemented using semiconductor switches that activate a path between an input and on output port. The switching elements which constitutes the basic functional block for an interconnection network, in typically small in part-size (e.g 2 X 2) and is capable of switching packets autonomously (self routing header attached to each packet) and independently, thus introducing very high degree of parallelism. If SES has no buffering capability, we them refer to un-buffered interconnection networks. An NXN network built from dxd SES requires a minimum of n=logdN stages. Fig below depicts examples of multistage switching fabrics based on three-stage interconnection networks (N=8) that one built from 2x2 SES.

A multistage switch can be naturally augmented (in port count) if we consider additional stages. Depending on how the SEs accesses the stages of an interconnection network is linked, different types (topologies) of multistage networks can be produced. If a single path exists between an input and an output of a multistage network, packets can be self routed across disjoints paths (no sharing of inter-stage links) that are established prior to switching the packets presented to the network. In the process the interconnection network of the single plane design of switching fabric is taken into account because in this contribution we are focused primarily on the performance, the slotted traffic source model is used with a uniform random distribution of product destinations, the switching network is operated synchronously meaning that the packets are transmitted only at the beginning of a time slot given the packet clock and each input link is offered the same traffic load, the buffering is external to the switch fabric i.e from non-buffered switch fabrics formed from non-buffered SEs are under consideration. The repercussionsof queuing effects are not considered in this model.

3. To simulate the model above using MATLAB as a tool

The model simulation in Matlab demonstrates the scheduling and transmission of packet switched network a data packets are generated by the token generators. Each token has attributes representing the weight and transmission time. The data packets are then transmitted on the data network.

3.1 The switch design consists of three important parts:

Transmitted, queuing, processing and receiver the transmitter part is a set of events generated by "time based entity generator" block. These events were generated in a specific time depending on the case study requirements; in this thesis the generation time of the events set to 4ms. The events were forwarded from the transmitter to the processor for processing. The processor CPU is represented by the "single server" block. When the single server block is busy processing the incoming events and the transmitter keeps sending more events, the extra incoming events are stored in the FIFO (First Input First Output) buffer for a finite period of time until it gets permission from the CPU to release them and send the packets to the processor and then to the receiver. The receiver parts of the switch is modeled using the "entity sink" block. The scenario of the drop packets happens when the number of events that come from the input generators exceeds the overall capacity of the processor. Such an occasion reduces the reliability of the system; therefore, calculating the number of drop packets is crucial in this study and it is formulated as fellow. No of drop packets = total number of inputs events – butter capacity-service rate of CPU. This model simulates data switched network with three computers and a shared channel operated at 10mbps. The MAC controller block implements the CSMA/CD (carrier sense multiple access with collision detection) protocol. You can specify the packet generation rate and packet size range at the application blocks, the transmission buffer size at the MAC controller blocks. The top level of the model includes three computers, each consisting of the following

1. An application block that models the consumer of data.
2. A MAC (medium access control) controller that governs the computer's use of the shared channel.
3. Access point that connects the computer to the network.

Implementing the CSMA/CD protocol standard data switched network employ the CSMA/CD protocol to manage use of the shared channel. Each computer's MAC controller physically monitors the traffic on the channel and initiates a transmission only when the controller detects no other traffic on the channel. If transmissions from two computers compete for use of the channel, then a collision occurs. Packets that collide with other packets can make a fixed number of additional attempts after awaiting period elapses, using the binary exponential back-off algorithm.

3.2 Simulink Model for Knockout Switch

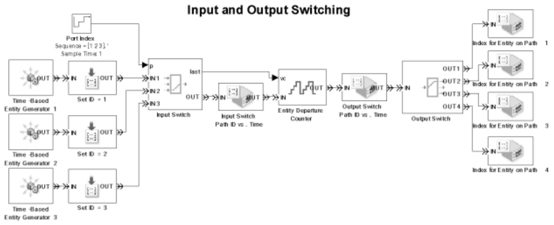

Fig. 1: MATLAB - Simulink Model for knockout switch

Inside the MAC controller subsystems a transmission buffer holds packets waiting for transmission. When the carrier sensing component indicates that the channel is ready, a gate opens and the packet advances to the subsystems TX output port. This output port permits the packet to advance to the access point. In the course of advancing pass the gate and the TX output port, the packet also passes through a replicate block that makes a copy of the packet. The subsystem uses this copy to observe the channel's state and to implement CSMA/CD. In particular, the subsystem tracks the packets that use the channel, determines when the channel is not busy, and determiner when the channel experiences a collision between two packets. For packets that experience a collision the back off controller subsystem determines whether and when a packet reattempts transmission. Data presentation and analysis the parameter used for the measurement and the result generated as shown in Table 1: Simulation Parameter.

Table 1:Simulation parameters

Parameter	**Value / Type**
Routing	Switches
Modem	Wireless
Connection	Dial up/broadband
Protocol	TCP/IP
Network	Wifi
Server	Database Server
Channel bit rate (data)	10 Mbps
Traffic Model	CBR
Simulation Time	45 mins
No of Transactions	10
radio spectrum/bandwidth	40 MHz
MAC type	802.11

4. Simulations Result Network Characterization Result

An experiment was carried out to measure the network parameters to know their impact on the data switched network at the Akwa Ibom State Ministry of Science and Technology. The data switched network in the organization was used to monitor the network performance. The performance parameters and service parameters are monitored as outputs based on the network traffic. Four key performance indicators (KPI) are recorded as the network QOS parameters: latency, average loading (utilization) throughout and packet loss the characterization collects the KPI data in 30 days as shown in table 2;

Table 2: Results of characterization

Day	**Data Network utilization (%)**	**Packet loss (Packets/s)**	**Latency (second)**	**Throughput (Bytes/ Sec)**
Day 1	1	0	0.0204	2.44
Day 2	1	0	0.0213	3.35
Day 3	3	0.028	0.0238	87.95
Day 4	3	0	0.0243	99.16

Day 5	5	0	0.0294	216.02
Day 6	6	0.028	0.0277	238.31
Day 7	2	0	0.0208	28.81
Day 8	2	0	0.0213	48.39
Day 9	3	0.333	0.0217	65.98
Day 10	2	0	0.0208	29.31
Day 11	2	0.025	0.0213	57.54
Day 12	1	0	0.0200	2.78
Day 13	1	0	0.0200	2.66
Day 14	1	0	0.0200	2.67
Day 15	1	0	0.0200	3.030
Day 16	1	0	0.0200	2.58
Day 17	1	0	0.0204	2.37
Day 18	1	0	0.0213	2.37
Day 19	1	0	0.0238	2.37
Day 20	1	0	0.0243	2.37
Day 21	3	0.041	0.0231	11.23
Day 22	1	0	0.0220	10.81
Day 23	2	0	0.0217	4.12
Day 24	2	0	0.0200	4.56
Day 25	1	0	0.0213	7.32
Day 26	1	0	0.0271	26.21
Day 27	1	0	0.0201	2.24
Day 28	4	00.035	0.0241	27.31
Day 29	4	0	0.0213	59.12
Day 30	3	0	0.0207	27.56

Simulated data for calculating throughput using knockout switch and routing fabrics table 3.

Table 3: Simulation results

Packet Size (KB)	**Throughput (Bytes/Sec)**
10.7	29.43
14.9	29.13
50	45.1
50.3	45.3
100	50.2
115	50.5
151	60
166	62.4
191.3	62.9
210	70.2

If the probability of queuing at the inputs is small enough, perhaps we can do away with the input queues and simply drop the very few packets that have lost contention. Such switches operate as loss systems, and the underlying principle is called the knockout principle.

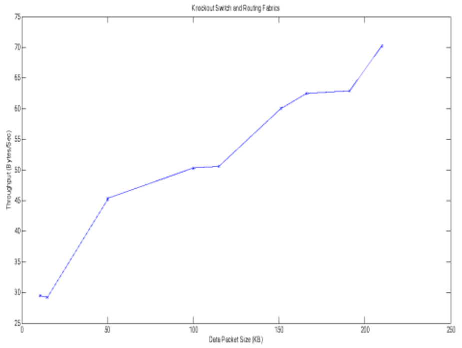

Fig. 2: Graph for Throughput using Knockout Switch and Routing Fabrics

Represents Variations in throughput with changes in file size (kb), using Knockout Switch and Routing Fabrics. From table 3, it can be seen that the throughput was relatively high. This is because in the knockout principle, the likelihood of a packet being dropped due to contention can be made arbitrarily small with sufficiently large R number ofpackets, and furthermore, for a given loss probability requirement, there exists an R that is independent of the switch size N such that the actual loss probability is not larger than the requirement.

Table 4: Simulated data for calculating Throughput using Knockout Switch and Routing Fabrics

Packet Size (KB)	**Throughput (Bytes/Sec)**	
	Knockout Switch and Routing Fabrics	**Channel grouping**
10.7	29.43	13.61
14.9	29.13	13.45
50	45.1	13.53
50.3	45.3	14
100	50.2	15.7
115	50.5	15.9
151	60	16
166	62.4	16.3
191.3	62.9	16.8

210	70.2	19.6

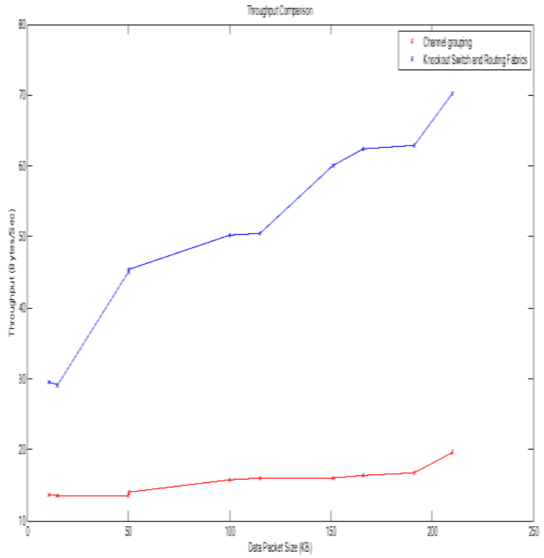

Fig. 3: Shows the comparison of both models.

The figure 3 shows the comparison of the Channel grouping and the proposed Knockout Switch and Routing Fabrics. It shows a great improvement in the throughput as the average throughput on the data network was 50.5 Bytes/Secwhen usingKnockout Switch and Routing Fabrics while the average throughput achieved when using channel grouping is 15.45 Bytes/Sec. so the improvement achieved when Knockout Switch and Routing Fabrics was simulated on the data switched network is 35.027 Bytes/Sec.

5. CONCLUSION

This research presented performance comparison of channel grouping and the proposed knockout switch and routing fabrics based on the measurement data generated by the Matlab simulation. The KPI used in the simulation was throughout so as to show how the proposed technique way able to enhance the performance of the data switched network.

6. REFERENCES

Allman, M. and Paxon. V. 2012 “On Estimating End-to-End Network path properties

Hayder, N. Y, Zuriati. A. Z, mohammed, O. and Shamala, S. (2019) “The New AMD Congestion Control Alogorithm proceedings of world academy of science, engineering and technology.

Jacobson, V. (2008) Congestion Avoidance and Control.

Katabi, D., Handley, M. and Rohrs, C. (2012). Congestion Control for High Bandwidth delay Product Networks, ACM Special Interest Group on Communication, Pittsburgh Supercomputing Center, U.S.A.

Mamatas, L., Harks, T. and Tsauossidis, V. (2007), Approaches to Network Congestion in Packet Switched Networks; Journal of Internet Engineering, Volume 1, Number 1.

Morris. R. T. (2009) Scalable TCP Congestion Control. PhD thesis, Harvard University, 2009.

Nagle, J.B. (2007), 'On Packet Switches with Infinite Storage', *IEEE Transactions on Communications*, Vol. 35, No.4, April, pp. 435-438.

Ramakrishnan, K. and Floyd, S. (2009) "A Proposal to add Explicit Congestion Notification (ECN) to IP", RFC 2481, January 2009.

Stevens, W.R. (2011) “TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms”, RFC .