ISSN: 2321-8169
6253 - 6257

Designing Change Assimilation Process using Close-up Down Graph for Switch Based Networks

Ms. Deepti Metri
Computer Science and Engineering
N. B. N. Sinhgad College of Engineering, Solapur
Solapur, India
deeptimetri@gmail.com

Prof. A. V. Mophare
Computer Science and Engineering
N. B. N. Sinhgad College of Engineering, Solapur
Solapur, India
mophare_abhi@yahoo.com

Abstract— In today's modern switch-based interconnected systems require high performance, reliability and availability. These switch based networks changes their topologies due to hot expansion of components, link or node activation and deactivation. Device failures in high-speed computer networks can also result in topological changes. Also, component failures, addition and deletion of components cause changes in the topology and routing paths supplied by the interconnection network. Therefore a network reconfiguration algorithm must be executed to reestablish the connectivity between the network nodes. Now we have two types of reconfiguration techniques and they are static reconfiguration and dynamic reconfiguration. Static reconfiguration techniques significantly reduce network service since the application traffic is temporally stopped in order to avoid deadlocks. But unfortunately this has negative impact on network service availability. Dynamic network reconfiguration is the process of changing from one routing function to another routing function while the network remains up and running. While performing dynamic network reconfiguration, the main challenge is to avoid deadlocks and provide network service availability along with reduced packet dropping rate. In this paper we demonstrate how dynamic reconfiguration is more efficient than the static reconfiguration for switch based networks.

Keywords- *Deadlock avoidance; network reconfiguration; UP/DOWN routing; high-speed networks.*

*****

I. Introduction

The communication subsystem play an important role in today's modern switch based networks. Therefore high performance subsystems use the switch based technology which have point to point link and provides very high performance. Recently, some different technologies have been introduced which are based on distributed routing like InfniBand [8] and source routing like Myrinet 2000[13].

In such environment topological change in network occurs due some failure in network, or addition or removal of nodes or links in the network. Due to this network reconfiguration is required and the new routing function is obtained according to network topology. This change assimilation process is conducted by the centralized management entity called as mapper in Myrinet 2000 [13], subnet manager in InfiniBand [8] and fabric manager in Advanced Switching Interconnect (ASI) [1]. Once the manager detects any change, it collects the information about all the active devices, end points and switches which are connected to the network. When the failure is detected a new path is established between the end points for delivering the packets. Then the final step is the replacement of old routing function to the new generated routing function. This process of detecting failure and modifying the old routing function by the new one is known as network reconfiguration.

This network reconfiguration or change assimilation process is divided in two types that are static reconfiguration and dynamic reconfiguration. Static reconfiguration is a technique in which during reconfiguration of the network, packet transmission through the network is stopped. As packet transmission is stopped, deadlock occurrence in network is found less. While in dynamic reconfiguration technique the packets are transmitted through the network which can lead to the occurrence of deadlock in the network. In static reconfiguration technique the chances of occurrence of deadlock is less as the packet transmission is stopped but network performance degrades gradually. But in dynamic reconfiguration technique chances of occurrence of deadlock in the network are more while the performance of the network increases gradually.

II. Literature Review

Deadlock is a situation in which the set of packets are blocked forever because the packets that require resources are already acquired by some another set of packets. Deadlock is not a major issue in static reconfiguration technique because packets that are routed by old routing function and those that are routed by new routing function are not simultaneously available in the network. But deadlock can be an issue in dynamic reconfiguration as packets that are routed by old and new routing function are present in network.

The Partial Progressive Reconfiguration (PPR) [5] and Skyline [9] repair uncorrected up*/down* graph including several sink nodes. The PPR scheme computes new graph using spanning tree. Skyline technique identifies the region of network that is to be configured after change. But the technique was designed for distributed routing and were not suitable for source routing.

NetRec [2] and LORE [26] are designed to reroute messages or packets for faulty nodes. In NetRec scheme every switch needs to maintain information about switches some hops away. On the other hand in LORE scheme virtual channels are used to reroute the packets. Again these two techniques are not suitable or feasible in networks with source routing.

Double scheme uses the virtual channel and it requires additional source to transmit the message through these virtual channel. Token is used in this scheme. In this scheme, the deadlock can be avoided because here packets are first transmitted that belong to the old routing table and then that to the new routing table. But the drawback of the scheme or technique is that it could be applied where the virtual channels are used.

III. EXISTING SYSTEM

Network reconfiguration mechanisms in current high-speed LANs are based on static reconfiguration techniques. Autonet [5] is, perhaps, the most representative example. In this network, a distributed reconfiguration protocol is triggered when a significant change in the topology occurs, spreading it to the entire network, and updating the routing tables in hosts and switches. This protocol avoids deadlocks during the reconfiguration process by stopping application traffic before starting the reconfiguration process. When the reconfiguration finishes then packet transmission is allowed to resume. As a consequence, performance degradation of the interconnection network is occurred [26].

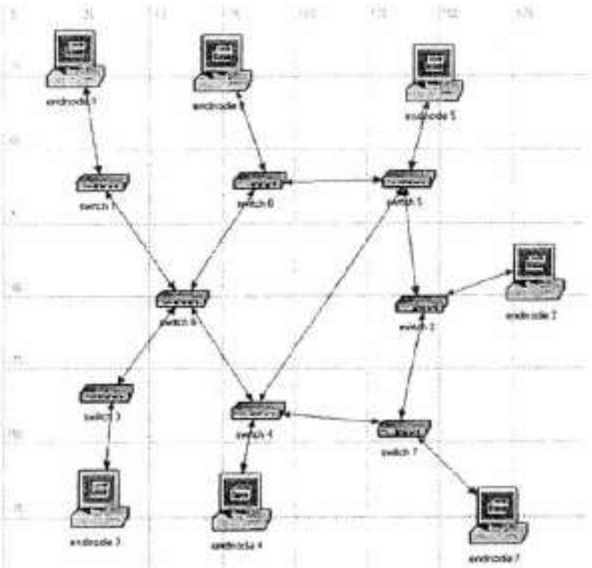


Figure 1. An IBA subnet consisting of eight switches and seven end nodes.

Above Figure 1 shows an IBA (Infiniband Architecture) consisting of eight switches and seven end nodes. Figure 3 shows the IBA subnet of Fig. 1 undergoing reconfiguration triggered at simulation time 61 seconds.. In the case of static reconfiguration (Fig.2), all network ports are brought to INITIALIZE state before the forwarding tables are updated. This results in the dropping of approximately 15,000 data packets at the load rate mentioned above. Furthermore, no data packets are injected into or delivered by the network during this period. This drop in network throughput is highlighted in Fig. 2. Total time taken for static reconfiguration is approximately 65 ms. It is important to note that as the network size or the applied load rate increases, the number of data packets being dropped by the static scheme also increases significantly [26].

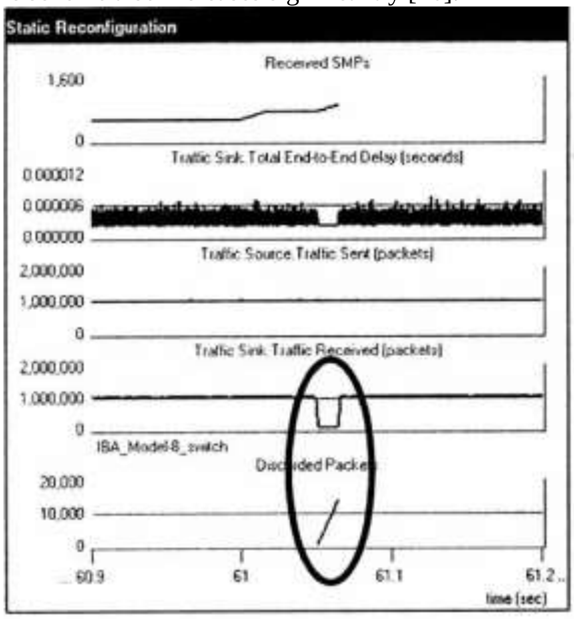


Figure 2. Static reconfiguration for IBA subnet of Figure 1

IV. PROPOSED SYSTEM

Now, we introduce a complete management mechanism which supports dynamic routing reconfiguration using close graph properties.

- Up*/Down* Routing

  In this work, we use up*/down* [25], a well-known deadlock free routing algorithm. Here the network topology is represented as a directed graph G (N, L), where N is set of devices and L is set of operational links. Crossing a link l = ni → nj from ni to nj is referred to as the up direction. Similarly, crossing the link from nj to ni is referred to as the down direction. Legal routes never use a link in the up direction after having used one in the down direction. In other words, messages can cross zero or more links in the up direction, followed by zero or more links in the down direction. Fig. 3[1] shows a possible assignment of directions for a particular topological graph. In this example, a possible legal route from node 2 to node 12 could be 2 → 1 → 14 ←13 ←10 ←12, but not 2 → 1← 3 ←5 ←7 → 8 → 10 ←12 [1].

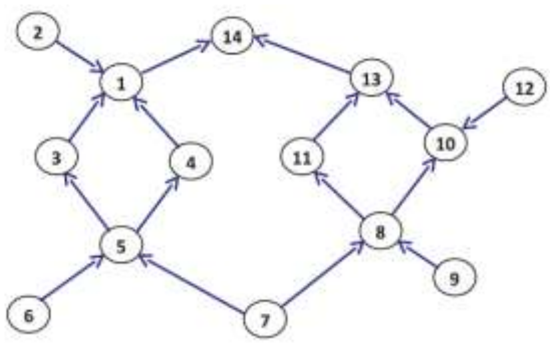


Fig. 3. Example of up*/down* link direction assignment for a particular topological graph.

- Proposed Management Mechanism Phases

Figure 4 [1] shows all steps involved in the change assimilation process. Initially the manager periodically poll the status of fabric devices to detect the topological change. When the fabric detects the change, it is notified to the manager.

Then the manager obtain information about all active devices or endpoints connected to the fabric. After thet a new up*/down* directed graph is obtained. New links are assigned such that both original and new graph maintain certain properties.

After that, manager computes path. Here Fully Explicit Routing (FER) algorithm [3] is used.

The last phase involved in the change assimilation process consists in replacing the old routing function by the new one. The manager sends the new paths to the network endpoints. Each endpoint replaces the obsolete paths by the new ones as soon as they are received [1].

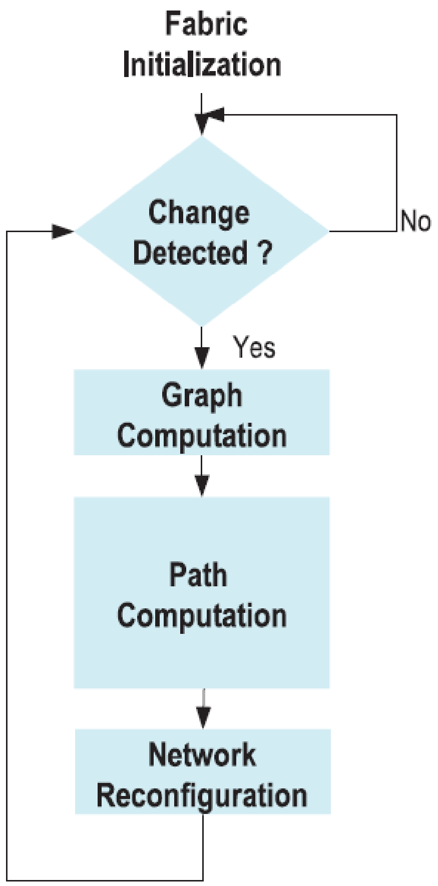


Figure 4. Fabric management mechanism

- Implementation and Results

The simulation model [20] used for this work was developed with the OPNET Modeler [14], and embodies the physical and link layers of Advanced Switching, a source routing technology.

Following Table 5 represents the simulation parameters used in this project.

S.NO.	Simulation Parameters	Particulars/Specifications
1	Number of nodes	16
2	Simulator OPNET MODELER	17.5 academic edition
3	Area	1000m
4	Network Topology	Campus
5	Protocol for routing	DSR
6	Simulation Time	30 min
7	Route Expiry Time	300 Sec
8	Buffer Expiry	30 sec
9	Packet retransmission period	0.5 sec

Table 5. Simulation Parameters

Network Load: The statistic represents the total data traffic received by the entire wireless LAN from higher layers of MAC that is accepted and queued for transmission.

Throughput: The statistic represents the total number of bits forwarded from wireless LAN layers to higher layers in all wireless LAN nodes of the network.

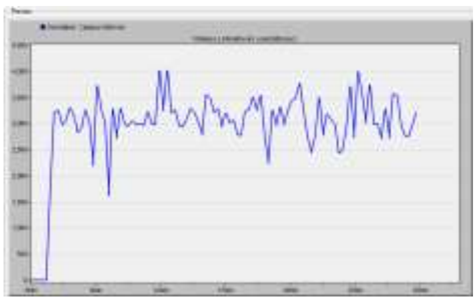
Data Dropped: Total higher layer data traffic dropped by all the wireless LAN MACs in the network as a result of consistently failing retransmissions. This statistics reports the number of higher layer packets that are dropped because the MAC couldn't receive any ACK for the (re)transmission of those packets or their fragments.

- **Results and Graphs**

Network Load Vs Throughput

Following Figure 5 shows the graph for Network Load and Throughput. As the network Load increases the throughput of the fabric also increases.

Network Load Throughput



ISSN: 2321-8169
6253 - 6257

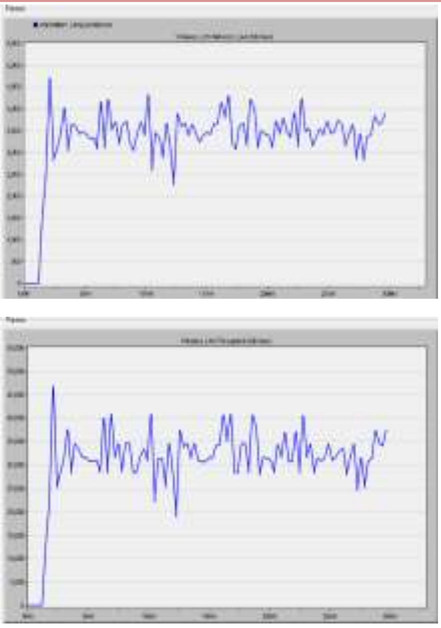


Fig 5. Graphs for Network Load and Throughput

Reconfiguration Time

Following figure shows the graph representing time taken by network to reconfigure the New graph and Close graph

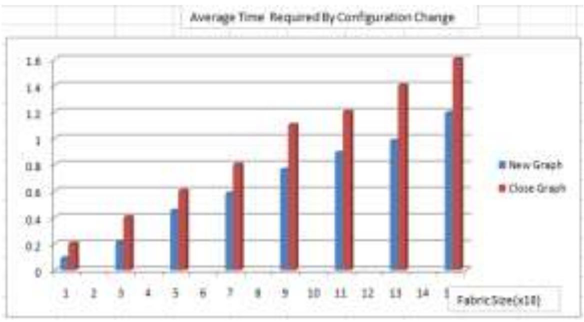


Fig 6. Network Reconfiguration Time

Traffic Patterns

a. Uniform: In this kind of pattern, source and destination is selected randomly for each message generated.

b. Shuffle: Fixed source-destination pair for every message. The node with binary value a_{n-1}, a_{n-2}, ..., a_1, a_0 communicates with the node a_{n-2}, a_{n-3}, ..., a_0, a_{n-1} (rotate left 1 bit).

c. Transpose: Fixed source-destination pair for every message. The node with binary value a_{n-1}, a_{n-2}, ..., a_1, a_0 communicates with the node $a_{n/2-1}$,..., a_0, $a_{,n-1}$, ..., $a_{n/2}$.

d. Bit Reversal: Fixed source-destination pair for every message. The node with binary value a_{n-1}, a_{n-2},..., a_1, a_0 communicates with the node a_0, a_1,..., a_{n-2}, a_{n-1}.

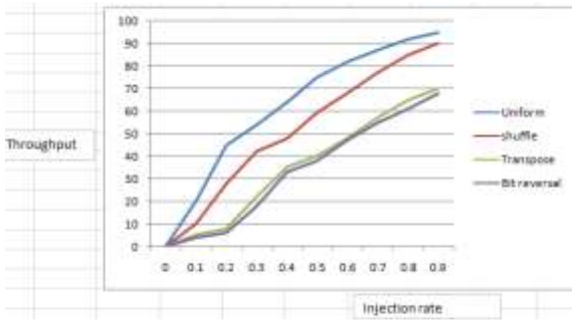


Fig. 7. Traffic Patterns

V. CONCLUSION

This work proposes and evaluates a new dynamic reconfiguration mechanism for source routing networks using the up*/down* routing algorithm. The new routing function is designed to ensure that packets routed according to both the old and new routing functions can unrestrictedly coexist in the network, without the risk of forming deadlocks.

The simulation graphs show the result of dynamic reconfiguration implemented in this paper. In static reconfiguration technique as the network size or the applied load rate increases, the number of data packets being dropped also increases significantly. But this is not in the case of dynamic reconfiguration technique. As it can be seen from the result graph as the network load of the fabric increases the throughput of the system also increases. During the simulation of the system we have used Dynamic Source Routing (DSR) Protocol which is a source routing protocol.

REFERENCES

[1] Antonio Robles-Go´mez, Aurelio Bermu´dez and Rafael Casado, “A Deadlock-Free Dynamic Reconfiguration Scheme for Source Routing Networks Using Close Up*/Down* Graphs” IEEE Trans. Parallel and Distributed Systems, vol. 22, no. 10, pp. 1641-1652,Oct. 2011.

[2] D. Avresky and N. Natchev, “Dynamic Reconfiguration in Computer Clusters with Irregular Topologies in the Presence of Multiple Node and Link Failures,” IEEE Trans. Computers, vol. 54, no. 5, pp. 603-615, May 2005.

[3] A. Bermu´ dez, R. Casado, F.J. Quiles, and J. Duato, “Fast Routing Computation on InfiniBand Networks,” IEEE Trans. Parallel and Distributed Systems, vol. 17, no. 3, pp. 215-226, Mar. 2006.

[4] N.J. Boden, D. Cohen, R.E. Felderman, A.E. Kulawik, C.L. Seitz, J.N. Seizovic, and W. Su, “Myrinet: A Gigabit-per-Second Local Area Network,” IEEE Micro, vol. 15, no. 1, pp. 29-36, Feb. 1995.

[5] R. Casado, A. Bermu´ dez, J. Duato, F.J. Quiles, and J.L. Sa´nchez, “A Protocol for Deadlock-Free Dynamic Reconfiguration in High- Speed Local Area Networks,” IEEE Trans. Parallel and Distributed Systems, vol. 12, no. 2, pp. 115-132, Feb. 2001.

[6] W.J. Dally and C.L. Seitz, "Deadlock-Free Message Routing in Multiprocessor Interconnection Networks," IEEE Trans. Computers, vol. C-36, no. 5, pp. 547-553, May 1987.

[7] J. Duato, O. Lysne, R. Pang, and T.M. Pinkston, "Part I: A Theory for Deadlock-Free Dynamic Network Reconfiguration," IEEE Trans. Parallel and Distributed Systems, vol. 16, no. 5, pp. 412-427, May 2005.

[8] InfiniBand Trade Assoc., InfiniBand Architecture Specification Release 1.2, http://www.infinibandta.org/, 2004.

[9] O. Lysne and J. Duato, "Fast Dynamic Reconfiguration in Irregular Networks," Proc. IEEE Int'l Conf. Parallel Processing (ICPP '00), 2000.

[10] O. Lysne, J.M. Montan˜ ana, J. Flich, J. Duato, T.M. Pinkston, and T. Skeie, "An Efficient and Deadlock-Free Network Reconfiguration Protocol," IEEE Trans. Computers, vol. 57, no. 6, pp. 762-779, June 2008.

[11] O. Lysne, T. Pinkston, and J. Duato, "Part II: A Methodology for Developing Deadlock-Free Dynamic Network Reconfiguration Processes," IEEE Trans. Parallel and Distributed Systems, vol. 16, no. 5, pp. 428-443, May 2005.

[12] J.M. Montan˜ ana, J. Flich, and J. Duato, "Epoch-Based Reconfiguration: Fast, Simple, and Effective Dynamic Network Reconfiguration," Proc. Int'l Parallel and Distributed Processing Symp. (IPDPS '08), Apr. 2008.

[13] Myrinet, Guide to Myrinet-2000 Switches and Switch Networks, http://www.myri.com/, 2011.

[14] OPNET, http://www.opnet.com, 2011.

[15] T. Pinkston, R. Pang, and J. Duato, "Deadlock-Free Dynamic Reconfiguration Schemes for Increased Network Dependability," IEEE Trans. Parallel and Distributed Systems, vol. 14, no. 8, pp. 780- 794, Aug. 2003.

[16] A. Robles-Go´mez, A. Bermu´ dez, R. Casado, and F.J. Quiles, "Implementing the Advanced Switching Fabric Discovery Process," Proc. Seventh Workshop Comm. Architecture for Clusters (CAC '07), Mar. 2007.

[17] A. Robles-Go´mez, A. Bermu´ dez, R. Casado, F.J. Quiles, T. Skeie, and J. Duato, "A Proposal for Managing ASI Fabrics," J. Systems Architecture, vol. 54, no. 7, pp. 664-678, July 2008.

[18] A. Robles-Go´mez, A. Bermu´ dez, R. Casado, and _A.G. Solheim, "Deadlock-Free Dynamic Network Reconfiguration Based on Close Up*/Down* Graphs," Proc. Euro-Par Conf., Aug. 2008.

[19] A. Robles-Go´mez, A. Bermu´ dez, R. Casado, _A.G. Solheim, T. Sødring, and T. Skeie, "A New Distributed Management Mechanism for ASI Based Networks," Computer Comm., vol. 32, no. 2, pp. 294-304, Feb. 2009.

[20] A. Robles-Go´mez, E.M. Garcı´a, A. Bermu´ dez, R. Casado, and F.J. Quiles, "A Model for the Development of AS Fabric Management Protocols," Proc. Euro-Par Conf., Aug. 2006.

[21] T.L. Rodeheffer and M.D. Schroeder, "Automatic Reconfiguration in Autonet," Proc. 13th ACM Symp. Operating Systems Principles, 1991.

[22] M. Rooholamini, "Advanced Switching: A New Take on PCI Express," http://www.edn.com/article/CA468416.html, Oct. 2004.

[23] M. Rooholamini and R. Kaapor, "Fabric Discovery in ASI," http://www.networksystemsdesignline.com/, Oct. 2005.

[24] J.C. Sancho, A. Robles, and J. Duato, "A New Methodology to Compute Deadlock-Free Routing Tables for Irregular Networks," Proc. Seventh Workshop Comm., Architecture, and Applications for Network-Based Parallel Computing, Jan. 2000.

[25] M.D. Schroeder, A.D. Birrell, M. Burrows, H. Murray, R.M. Needham, T.L. Rodeheffer, E.H. Satterthwate, and C.P. Thacker, "Autonet: A High-Speed, Self-Configuring Local AreaNetwork Using Point-to-Point Links," IEEE J. Selected Areas in Comm., vol. 9, no. 8, pp. 1318-1335, Oct. 1991.

[26] Bilal Zafar, Timothy M. Pinkston, Aurelio Bermu ´ Dezb and Jose´ Duatoc, "Deadlock Free Dynamic Reconfiguration over InfiniBand Networks" Parallel Algorithms and Applications, Vol. 19 (2–3) June–September 2004, pp. 127–143.