ISSN No. 0976-5697

Volume 2, No. 6, Nov-Dec 2011

International Journal of Advanced Research in Computer Science

REVIEW ARTICLE

Available Online at www.ijarcs.info

A Survey on Deadlock Free Network Reconfiguration

S.Mohiadeen Abdul Kadhar*
Research Scholar
Anna University of Technology Tirunelveli
Tamilnadu, India
Kader_psr@yahoo.com

Dr. T.Revathi
Supervisor
Anna University of Technology Tirunelveli
Tamilnadu, India
trevathi@mepcoeng.ac.in

***Abstract**:* In any Interconnection networks reconfiguration may occur frequently due to addition or removal of network components. Reconfiguration leads to increase in network downtime which in turn reduces the performance of the system. In general, every reconfiguration technique has the objective of maintaining the Quality of Service (QoS), reliability, availability and dependability. In this paper, a survey is made between the various deadlock free network reconfiguration techniques. Network reconfiguration and its impacts are discussed. Analysis is made over the deadlocks in interconnection networks. Finally various research issues towards network reconfiguration are also discussed.

Keywords - Network Reconfiguration, Deadlock, Interconnection Networks.

I. INTRODUCTION

Interconnection networks are used for many different applications, ranging from internal buses in very large-scale integration (VLSI) circuits to wide area computer networks. Among others, these applications include backplane buses and system area networks; telephone switches; internal networks for asynchronous transfer mode (ATM) and Internet Protocol (IP) switches; processor/memory interconnects for vector supercomputers; interconnection networks for multicomputer and distributed shared-memory multiprocessors; clusters of workstations and personal computers; local area networks; metropolitan area networks; wide area computer networks; and networks for industrial applications. Additionally, the number of applications requiring interconnection networks is continuously growing. For example, an integral control system for a car requires a network connecting several microprocessors and devices. The characteristics and cost of these networks depend considerably on the application. The lack of standards and the need for very high performance and reliability pushed the development of interconnection networks for multicomputer. This technology was transferred to distributed shared-memory multiprocessors, improving the scalability of those machines. However, distributed shared-memory multiprocessors require an even higher network performance than multicomputer, pushing the development of interconnection networks even more. More recently, this network technology began to be transferred to local area networks (LANs). Also, it has been proposed as a replacement for backplane buses, creating the concept of a system area network (SAN) [1]. Hence, the advances in interconnection networks for multicomputer are the basis for the development of interconnection networks for other architectures and environments.

Parallel computing and communication systems built from the above networks require high-performance communication services with high reliability, availability and dependability - collectively, high robustness. The performance of the interconnection network is measured, in part, by packet delivery time from source to destination (i.e., latency) and by the number of packets delivered per unit time (i.e., throughput). In essence, a high-performance network allows the maximum number of packets to make forward progress to their destinations in minimal time, preferably along shortest paths to preserve network bandwidth. Likewise, the reliability, availability and dependability of a network equally impact the overall –goodness and quality of a system. These attributes are measured, in part; by the network's ability to remain up and running at near normal levels even when events occur which change its configuration, possibly due to changes in users' needs and/or system state. Such reconfiguration events may include, for example, hot-swapping of components, failure or addition of links/nodes, activation or deactivation of hosts/routers, etc., in the network.

In recent years, research efforts have focused on improving the router as the primary means of increasing network performance. This includes efforts in such diverse areas as improving router switching, scheduling, injection limitation, flow control, and the routing algorithms. Fig. 1 shows such a simple router model. Since network resources are finite and, ultimately, are contended for, structural hazards on those resources are inevitable which delay or prevent packet transmission in the network. This occurs even in networks with advanced router architectures. Such hazards cause packets to be blocked which, eventually, can lead to network congestion and, possibly, deadlock. One of the more critical problems to be addressed in order to achieve high network performance and robustness is that of efficiently handling deadlock anomalies. Deadlock occurs when there is a circular hold-and-wait dependency relation on network resources by in-flight packets such that progress in routing those packets is indefinitely inhibited. That is, packets would block in the network forever unless some action to resolve the deadlock situation was taken. This phenomenon can result in the entire network (and system) coming to a complete stand-still, consequently degrading system reliability, availability, and dependability considerably. Thus, it is vitally important to guard against deadlock in such a way as not to impose overly restrictive measures that under-utilize network resources.

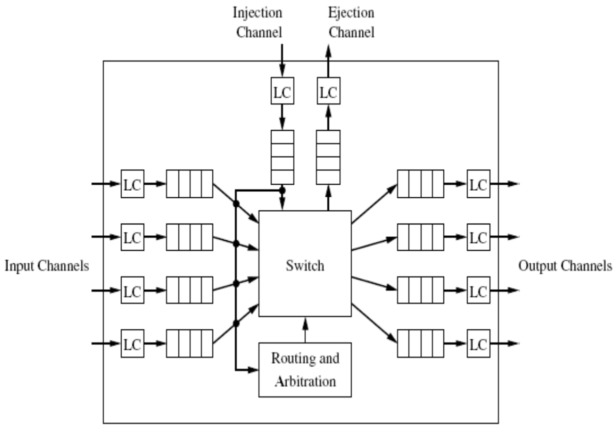

Figure. 1 Generic Router Model (LC=Link Controller)

The remainder of the paper is organized as follows. Section 2 describes about network reconfiguration and its importance. Section 3 deals with deadlocks in interconnection networks. Section 4 surveys various deadlock handling mechanisms. Section 5 explains about the possible research issues in the above topics.

II. NETWORK RECONFIGURATION AND ITS IMPACT

In some situations the premises on which the routing algorithm and/or network topology are defined may break, which affects the network's dependability.

This may happen when the topology of the network changes, either involuntarily due to failing/faulty components or voluntarily due to hot removal or addition of components. This normally requires the network routing algorithm to be reconfigured in order to (re)establish full network connectivity among the attached nodes. In transitioning between the old and new routing functions during network reconfiguration, additional dependencies among network resources may be introduced; causing what is referred to as reconfiguration-induced deadlock.

Current techniques typically handle this situation through static reconfiguration meaning that application traffic is stopped and, usually, dropped from the network during the reconfiguration process. While this approach guarantees the prevention of reconfiguration-induced deadlock, it can lead to unacceptable packet latencies and dropping frequencies for many applications, particularly real-time and quality-of-service (QoS) applications. With dynamic reconfiguration, the idea is to allow user traffic to continue uninterruptedly during the time that the network is reconfigured, thus reducing the number of packets that miss their real-time/QoS deadline.

III. DEADLOCKS IN INTERCONNECTION NETWORKS

Deadlocks in interconnection networks are classified into three basic categories, depending on the circumstances under which they form.

a. Routing-induced deadlocks
b. Message-induced deadlocks
c. Reconfiguration induced deadlocks

Routing-induced deadlocks are those caused by interactions and dependencies created within the network - between network endpoints - by the routing function which prevents packets from reaching their destinations. The routing function supplies the possible paths packets are allowed to take in the network to reach their destinations from their current locations.

Message-induced deadlocks (also called protocol-induced deadlocks) are those caused by interactions and dependencies created at the network endpoints among different message types (i.e., requests, replies, etc.), which prevent packets from sinking upon arrival at their destinations.

Reconfiguration induced deadlocks are those caused by the interactions and dependencies created through time (dynamically) in a network that undergoes reconfiguration, which prevents packets from reaching their destinations due to being routed under the influence of multiple active routing functions. This can occur even if each of those routing functions is independently deadlock-free under static conditions.

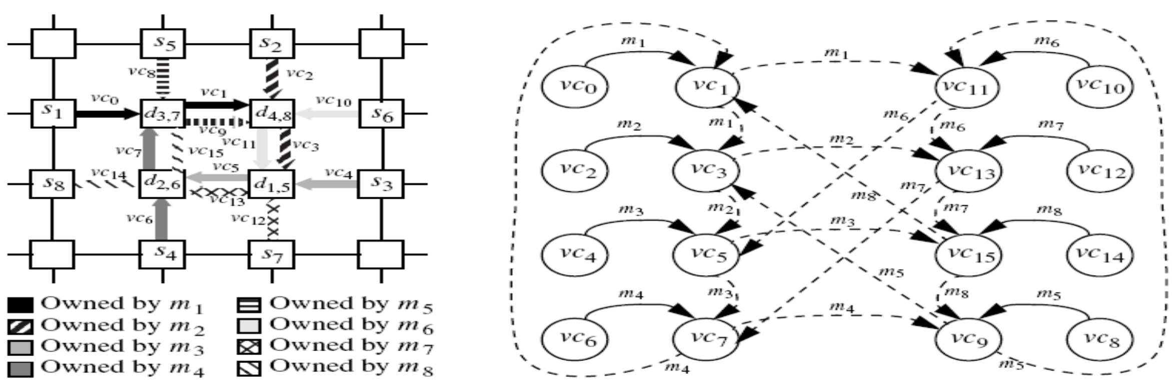

Figure. 2 Network and Channel Wait-for Graph

A. *Routing Induced Deadlocks:*

Whenever a packet reaches the router through any input channel or injection channel, it is the responsible of the routing function to show any output or delivery channel. The aggregation of these routing-induced channel dependence relations for all possible packet configurations of a network is captured by the network's CDG. Accordingly, each channel used by a packet has a dependence relation on the next channel(s) supplied by the routing function, creating a chain or path of dependencies captured by the Channel Wait-for Graph (CWG). As such, routing induced dependencies take into account only those dependencies on channel and queue resources shown in Figure 2; specifically, injection and delivery channels, edge queues, and/or central queues.

Interactions occurring at network endpoints are excluded from this set, meaning that packets are assumed always to sink upon reaching their destinations. If knotted cycles appear along a fully occupied set of these resources, routing-induced deadlock is said to form.

In figure 2, network graph and channel wait-for graph of multicycle deadlock formed under minimal adaptive routing with two virtual channels per physical channel are shown. Routing induced deadlock is influenced by three factors: routing freedom of routing algorithm, number of blocked packets within the network and the number of resource dependency cycles. Routing freedom has an opposite and more influential effect on deadlock probability than it does on the creation of cycles. As routing freedom is increased, the number of blocked packets decreases substantially. More importantly, the degree of correlation required among blocked packets to form a knot also increases substantially. This greatly decreases the likelihood of the occurrence of deadlock. Given enough routing freedom, this correlation factor offsets the opposing effect on deadlock probability caused by the potential increase in the number of cycles. Networks with minimal routing freedom may not offset the opposing effects as there may exist a one-to-one correspondence between cycles and deadlocks, e.g., single-cycle deadlocks. However, networks with greater routing freedom may offset the opposing effects as a large number of cycles can exist without deadlock formation, e.g., cyclic non-deadlocks.

B. Message Induced Deadlocks:

In any interconnection networks, to complete the transactions successfully, messages are needed. Selection of messages depends on the type of communication protocol used by the system. For example, cache coherence protocols may use messages like *request*, *reply*, etc. At any given end-node in the system, there can be a coupling between the two message types: the generation of one message type is directly coupled to the reception of another message type i.e. request reply messages. As the coupling between message types is transferred to network resources due to the finiteness of resources along the message path inside each node, additional dependencies on network resources are created, referred to as message or protocol dependencies.

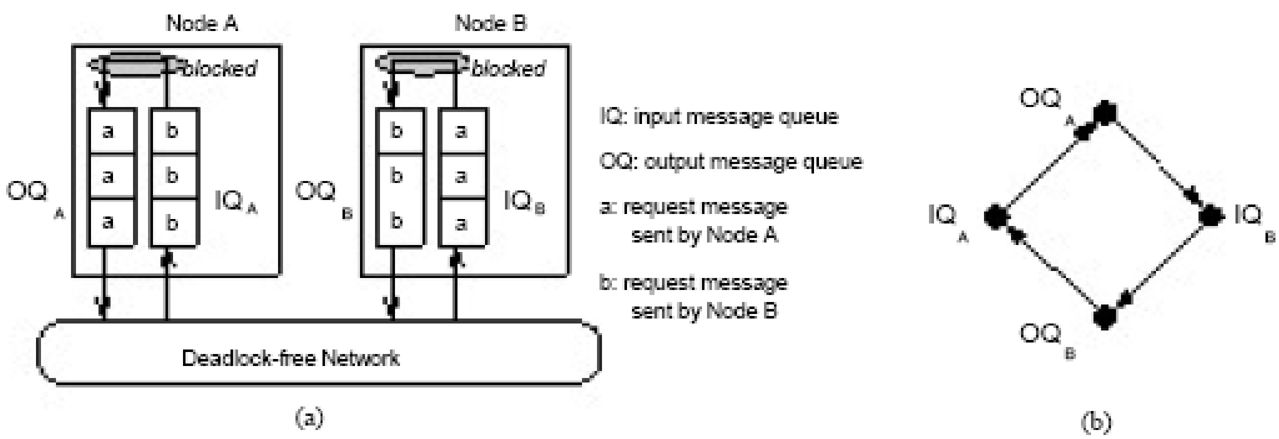

Figure. 3 (a) Message-Induced Deadlock (b) Channel dependency graph

A distinct class of message dependency is created for each pair of message types for which a direct coupling exists and is transferred to network resources. Each combination may present different kinds of message dependencies and a corresponding message dependency chain, which is a series of message types used to complete a transaction on network resources. Since message dependencies may prevent messages from sinking at their destinations, they must be added to the complete set of resource dependencies. If knotted cycles form along a set of resources when resources at the network endpoints are also taken into account, a type of deadlock called *message-induced or protocol-induced deadlock* forms once all resources comprising the knot become full.

Figure 3(a) is a example of message-induced deadlock occurring between two nodes connected by a network free of routing induced deadlock. Figure 3(b) shows the corresponding dependency graph for resources at network endpoints where dependencies form a cyclic wait-for relationship.

C. Reconfiguration Induced Deadlocks:

When a change in network topology or routing arises through time, it may be necessary to reconfigure the routing function in order to reconnect routing paths between nodes in the system. Reconfiguring a network's routing function can cause additional dependencies among network resources both during and after the reconfiguration process that are not independently allowed by either the old or new routing functions. The paths of channels occupied by some undelivered packets routed with the old routing function could be illegal under the new routing function. As a result, two adjacent units of such packets called flits, could be stored in two different channels - one allowed only by the old routing function and the other allowed only by the new routing function. This can create a set of residual dependencies, referred to as ghost dependencies that must be taken into account in the total set of resource dependencies when determining the network's deadlock properties. Ghost dependencies can interact with dependencies allowed by the new routing function to close dependency cycles on resources used to escape from deadlock, causing reconfiguration-induced deadlock.

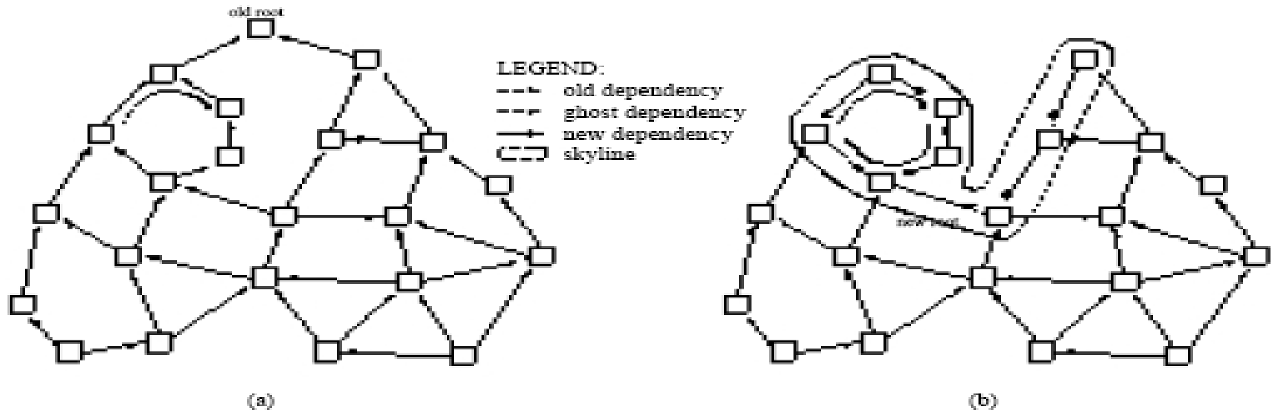

Figure 4(a & b) Reconfiguration of a network that uses Up/Down Routing

The above figure (a) shows the reconfiguration of a network that uses Up/Down Routing [2] which is free from routing induced deadlock. In that, packets route towards the destinations over paths consisting of zero or more uplink links followed by zero of more downlinks. The up direction for each link in the figure is indicated by the direction of the arrow heads. In figure (b) the old route node is removed which triggers a new route node to be discovered within the sky line of the network (enclosed by dotted lines), where the thicker lines indicate links which reverse the direction after reconfiguration completes. Ghost dependency (dashed arc) carried from the old routing function can form a dependency cycle with new dependencies (solid arc) from the new routing function, thus creating the potential for deadlock.

IV. SURVEY ON VARIOUS DEADLOCK HANDLING MECHANISMS

Handling deadlocks is essential for providing reliable communication paths between processing nodes. In this section various deadlock handling mechanism in the last decade has been studied and analyzed.

In [6], a new deadlock-free routing scheme for meshes is proposed based on a new virtual network partitioning scheme, called channel overlapping. In general a packet is delivered from a source s to a destination d along a minimum feasible path if the length of the path equals the number of hops that s and d differ, where all nodes in the path are fault-free. All neighboring nodes of the current node c in minimum feasible paths from c to the destination are the minimum next hops. A fully adaptive routing algorithm selects all minimum next hops as candidates to deliver a packet in all intermediate nodes. A packet can be delivered to an intermediate node c in a faulty mesh, where the destination and c only differ along one dimension. The packet can be blocked by a faulty node or link failure in the unique path from c to the destination. If the routing algorithm allows the packet to be delivered along one or more dimensions whose offsets have been eliminated.

This process is called misrouting. In most cases misrouting occurs because of faulty or unsafe nodes. In previous methods all these faulty nodes are disabled but in the current proposal unsafe node can either treated as source or destination node which leads to improve the performance of routing algorithms. Nearly four virtual channels are required to avoid deadlocks based on the virtual network scheme but in the current proposal only two virtual networks are used that can share some common virtual channels based on the new virtual network partitioning scheme. The deadlock-free adaptive routing method is then extended to deadlock-free adaptive fault-tolerant routing in 3D meshes still with two virtual channels. A few faulty nodes can make a higher dimensional mesh unsafe for fault-tolerant routing methods based on the block fault model, where the whole system (n-dimensional space) forms a fault block. Planar safety information in meshes is proposed to guide fault-tolerant routing and classifies fault-free nodes inside 2D planes. Many nodes globally marked as unsafe in the whole system become locally enabled inside 2D planes. This fault- tolerant deadlock-free adaptive routing algorithm is also extended to n-dimensional meshes with two virtual channels.

In [7], they proposed and evaluated a reconfiguration method based on the up*/down* routing algorithm, which is suitable for source routing networks that does not restrict the injection of packets during the change assimilation process. Without requiring additional network resources, the proposed new scheme "*Close Graph-based Reconfiguration* (CGR)" which is able to recover topology connectivity maintaining network throughput. In this strategy a new directed graph, which is *close* to the old one, is first computed by a centralized management entity. Then, a new routing function is derived by considering simultaneously both the old and new graphs. Thereafter, it can be asynchronously distributed to the routing elements. This enables a dynamic updating of the routing function.

In [8], a new deadlock-free routing scheme is proposed for 3D meshes first, where each physical channel needs only two virtual channels to avoid deadlocks. The deadlock-free routing scheme is used to do fault-tolerant routing in meshes, where a new fault model called PN fault model is introduced to guide deadlock-free adaptive fault-tolerant routing in wormhole-routed meshes. The new PN fault model is quite suitable for the proposed PAR scheme. Safety information inside the xy plane (xy þ and xy _ subnetworks) and the zy plane (the zy þ and zy _ subnetworks) must be collected. Much less safety information must be kept in each fault-free node compared to the 3D MCC fault model. The proposed deadlock-free adaptive routing scheme is also extended to n-dimensional meshes still using two virtual channels.

Gomez et al. in [9] and [10] presented a two-phase routing scheme by selecting an intermediate node to avoid faulty nodes. The phase from the source to the intermediate node uses a virtual network, and the phase from the intermediate node to the destination uses another virtual network. A message is routed in both phases based on Duato’s fully adaptive routing protocol, where both phases share the same adaptive channel and use different escape channels. The bubble flow control mechanism or dimension-order routing is used for messages routed along escape channels. That is, three virtual channels are required to avoid deadlocks.

In [11], Olav proposed, evaluated, and proved the deadlock freedom of a new network reconfiguration protocol that overlaps various phases of “static” reconfiguration processes traditionally used in commercial and research systems to provide performance efficiency on par with that of recently proposed “dynamic” reconfiguration processes but without their complexity. Proposed Overlapping Static Reconfiguration protocol can reduce reconfiguration time by up to 50 percent, reduce packet latency by several orders of magnitude, reduce packet dropping by an order of magnitude, and provide unhalted packet injection as compared to traditional static reconfiguration while allowing network throughput similar to dynamic reconfiguration. It is generally applicable to any network topology and routing function and it does not require multiple sets of data virtual channels or packet dropping to maintain deadlock freedom. In addition, it guarantees in-order delivery of packets during reconfiguration when the old and new routing functions are deterministic and, for that reason, can offload much of the fault-handling burden from higher level network protocols.

In [12], Olav presents a methodology for devising deadlock free and dynamic transitions between old and new routing functions that is consistent with newly proposed theory. The methodology is independent of topology, can be applied to any deadlock-free routing function, and puts no restrictions on the routing function changes that can be supported. Furthermore, it does not require any virtual channels to guarantee deadlock freedom. This research is motivated by current trends toward using increasingly larger Internet and transaction processing servers based on clusters of PCs that have very high availability and dependability requirements, as well as other local, system, and storage area network-based computing systems.

In [13], Timothy proposed an efficient and deadlock-free dynamic reconfiguration schemes that are applicable to routing algorithms and networks which use wormhole, virtual cut-through, or store-and forward switching, combined with hard link-level flow control. One requirement is that the network architecture use virtual channels or duplicate physical channels for deadlock-handling as well as performance purposes. The proposed schemes do not impede the injection, transmission, or delivery of user packets during the reconfiguration process. Instead, they provide uninterrupted service, increased availability/reliability, and improved overall quality-of-service support as compared to traditional techniques based on static reconfiguration.

In [14], Mark described a simple, lossless method of preventing deadlocks and livelocks in backpressured packet networks. In contrast with prior approaches, the proposed technique does not introduce any packet losses, does not corrupt packet sequence, and does not require any changes to packet headers. It represents a new networking paradigm in which internal network losses are avoided (thereby simplifying the design of other network protocols) and internal network delays are bounded. The proposed technique failed to address the important issue of fairness in providing service to different users. One way of resolving this shortcoming is to couple this technique with end-to-end congestion control schemes that handle congestion problems on a quasi-static basis while providing the desired fairness and/or priorities in the amount of services given to different users in the long run.

In [15], author(s) characterized the frequency of message dependent deadlocks in multiprocessor /multicomputer systems. They also proposed a handling technique for message-dependent deadlocks based on progressive deadlock recovery and evaluated its performance with other approaches. Results show that message dependent deadlocks occur very infrequently under typical circumstances thus, rendering approaches based on avoiding them overly restrictive in the common case. The proposed technique relaxes restrictions considerably, allowing the routing of packets and the handling of message-dependent deadlocks to be much more efficient—particularly when network resources are scarce. When network resources are abundant, it is profitable to separate message queue resources at network endpoints according to message type—not for deadlock avoidance purposes, but rather, for performance reasons. This reduces inter-message head of-line coupling and congestion at network endpoints but does not altogether prevent deadlock from occurring. Although the performance of avoidance-based techniques improves with increased network resources, the required partitioning of network resources (i.e., virtual channels) and limited routing freedom is overly restrictive for the common case.

In [16], Juan proposed a novel deadlock detection mechanism (FC3D) that uses only local information to detect all true deadlocks while considerably reducing the probability of detecting false deadlocks with respect to previously proposed techniques. The mechanism is based on the use of the flow control information available at each router. In particular, a message is presumed to be involved in a deadlock if all the channels provided by the routing function are busy, no flits are being transmitted, and an additional condition is met. In particular, when a deadlocked configuration is reached, the mechanism tries to mark as deadlocked as few messages as possible in each cycle of blocked messages, significantly reducing recovery overheads. This has been done by using flow control information and a few counters and flags to guess whether a message is the first one that blocked in a deadlocked configuration, and labeling only that message as being deadlocked. The mechanism uses two thresholds, only one of them has to be empirically tuned. Once the threshold has been tuned, this detection mechanism works correctly, regardless of message length and message destination distribution. Moreover, the FC3D mechanism is able to reduce deadlock detection rates by two orders of magnitude with respect to traditional approaches based on the use of crude time-outs, thus making the use of progressive deadlock recovery techniques viable, despite the low recovery bandwidth provided by those techniques.

In [17], Jose Duato presented a theoretical framework for the design of deadlock-free fully adaptive routing algorithms for a general class of network topologies and switching techniques in a single, unified theory. A general theory is proposed that allows the design of deadlock avoidance-based as well as deadlock recovery-based wormhole and virtual cut-through adaptive routing algorithms that use a homogeneous or a heterogeneous (mixed) set of resources. The theory also allows channel queues to be allocated nonatomically, utilizing resources efficiently. A general methodology for the design of fully adaptive routing algorithms applicable to arbitrary network topologies is also proposed that allow the design of efficient network routers that require minimal resources for handling infrequent deadlocks.

In [18], Sugath presented a formal model of resource allocation and dependencies within an interconnection network. Distinguishing messages based on their relationship to knots, as is done in this formal framework, allows the occurrence of deadlocks to be defined precisely, various types of deadlock dependencies to be classified, and necessary conditions for deadlock resolution in interconnection networks to be identified. This provides a powerful framework for evaluating the correctness, accuracy, and efficiency of proposed deadlock detection and resolution schemes and for proposing methods of reducing the probability of the occurrence of deadlock. The proposed technique can be used to empirically evaluate the frequency and characteristics of deadlocks in both regular and irregular interconnection networks.

In [19], Eric presented two performance models for multihop networks under non-uniform traffic pattern. The models are a generalization of Greenberg–Goodman and Brassil–Cruz models which were designed specifically for Manhattan Street Networks. The proposed model, on the other hand, can be applied to an arbitrary network topology of arbitrary degree. Furthermore, by considering packets with non null states only, proposed model is computationally more efficient than Greenberg–Goodman and Brassil–Cruz direct implementations.

Table 1. Consolidated Survey Report on various Deadlock handling Mechanisms

Sl. No.	Reference No.	Problem Analyzed	Technique Proposed	Features
1	6	Deadlock free Routing in mesh Network	channel overlapping	Deadlock-free adaptive routing method is extended to deadlock-free adaptive fault-tolerant routing in 3D meshes with two virtual channels
2	7	Network Reconfiguration	Close Graph-based Reconfiguration	Able to recover topology connectivity maintaining network throughput. It enables a dynamic updating of the routing function.
3	8	Deadlock free Routing	PN fault model	Deadlock-free adaptive fault-tolerant routing in wormhole-routed meshes. Uses only two virtual channels for each physical channel can support minimal and non-minimal adaptive routing in meshes. It can be extended to n-dimensional meshes with two virtual channels.
4	9, 10	Deadlock free Routing	Two-phase routing scheme	Duato’s fully adaptive routing protocol and bubble flow control mechanism or dimension-order routing are used. Uses only 3 Channels.
5	11	Network Reconfiguration	Overlapping Static Reconfiguration protocol	Reduce reconfiguration time by up to 50 percent, reduce packet latency by several orders of magnitude, reduce packet dropping by an order of magnitude, and provide unhalted packet injection. It is applicable to any network topology and routing function. It guarantees in-order delivery of packets.
6	12	Deadlock free Routing	Olav’s dynamic and deadlock-free reconfiguration algorithm	First completely general methodology for developing deadlock-free, dynamic reconfiguration processes. Independent of topology and routing functions. It does not require any virtual channels.
7	13	Deadlock free Routing	Deadlock-free dynamic reconfiguration scheme	Can be used in networks which uses wormhole, virtual cut-through, or store-and forward switching, combined with hard link-level flow control.

				They provide uninterrupted service, increased availability/reliability, and improved overall quality-of-service support.
8	14	Deadlock and livelock free Routing	Mark's Method	Can be used in backpressured packet networks. Does not have packet losses, corrupt packet sequence, and does not require any changes to packet headers
9	15	Deadlock handling and recovery	Avoidance based Technique	Handles message-dependent deadlocks based on progressive deadlock recovery. Reduces inter-message head of-line coupling and congestion at network endpoints.
10	16	Deadlock Detection	FC3D	Uses only local information to detect which reduces the probability of detecting false deadlocks. Reduces recovery overheads.
11	17	Deadlock avoidance and recovery	Jose Duato's deadlock-free fully adaptive routing algorithm	Utilizes network resources efficiently. Requires minimal resources for handling infrequent deadlocks.
12	18	Resource allocation and dependency	Sugath's Model	Evaluates the correctness, accuracy, and efficiency of proposed deadlock detection and resolution schemes. Can be in both regular and irregular interconnection networks.
13	19	Eric models	Greenberg–Goodman and Brassil–Cruz models	Used in multihop networks under non-uniform traffic pattern. Can be applied to an arbitrary network topology of arbitrary degree More efficient.

As an application, a comparison is made between our models against simulation for a 8X8 Manhattan Street Network subject to uniform traffic, of a 9X11 Toroidal Network subject to single node accumulation traffic, and of a (3, 3) ShuffleNet Network subject to random traffic. The model provides good agreement with simulation. By incorporating event-driven simulation methodology and considering packets with non-null states only, proposed model implementations have improved time efficiency. For example, with the 8X8 Manhattan Street Network, proposed model provides several orders of magnitude run time improvement over the Greenberg– Goodman and Brassil–Cruz implementations.

V. RESEARCH ISSUES

Avoiding deadlocks after network reconfiguration is a major challenge in any network system. There are lots of research issues hide behind the various proposals discussed in earlier section. A various number of methods for relaxing routing and drainage restrictions during dynamic reconfiguration can be explored with the goal of further increasing network dependability and performance predictability of networks in faulty and otherwise dynamic environments. Methods that allow for the elimination of only harmful ghost dependencies that may occur on escape resources in order to guarantee that all potentially harmful ghost dependencies are eliminated can be investigated. Researches can be made to build hybrid approaches that use an appropriate combination of both spatial and temporal separation of resource allocations on a single escape resource. The development of more general and powerful theory that can facilitate the design of more relaxed dynamic routing reconfiguration strategies is also an interesting topic of research.

There are only a few proposals concentrates that the reconfiguration could be made to balance dynamically the utilization of network resources (virtual channels, buffers, physical links, etc.) to alleviate congested routes. Researchers can show more interest in this area. The most important challenges lie in network reconfiguration is its application areas. Researchers can focus on topology changes induced by faults, but other important application areas are adaptations to varying traffic loads and planned component replacement. A particularly interesting example of an emerging application area comes from the NoCs. In the relatively near future, see processors with tens or even hundreds of cores on them and, most likely, with an on-chip interconnection network connecting the cores [42]. This will most likely mean that future programs running on these chips will be parallel [43] and that the hardware must accommodate several independent parallel processes. The separation of coexisting processes on the same hardware has been done by timesharing, scheduling, and context switches on single-core CPUs and with virtualization of addresses on memory. Mechanisms for ensuring that different processes use separate resources in the network must therefore be developed and fast network reconfiguration will be an important ingredient in such a set of mechanisms.

VI. CONCLUSION

Various techniques for deadlock free network reconfiguration have been surveyed. The techniques discussed here are based on deadlock detection, avoidance and recovery. Deadlock can be avoided in the strict sense, in the wide sense, in the weak sense, or by a combination of these approaches. What distinguishes one approach from another mainly has to do with the allowed degree of routing freedom and the possible manifestations of resource dependencies: the more routing freedom allowed, the more complex the resource dependencies are that can be manifested. By allowing maximum routing freedom on normal network resources, knotted dependencies on fully occupied resources can form, from which there is no way of escape. The outcome of this is deadlock which must be recovered from. Deadlocks can be resolved regressively, deflectively, or progressively, depending on how deadlock resolving resources are supplied to recovering packets.

In most cases, fundamental aspects of deadlock handling approaches are universally applicable to all classes. Theses approaches have the objective of maintaining the Quality of Service (QoS), reliability, availability and dependability while network reconfiguration. But in most circumstances all these factors are not maintained equally leads to an uncomfortable situation in the network. Techniques need to be developed to match the speed up of fast network reconfiguration with maximum performance.

VII. REFERENCES

[1]. R. Horst. ServerNet deadlock avoidance and fractahedral topologies. In Proceedings of the International Parallel Processing Symposium, pages 2752 280. IEEE Computer Society, April 1996.

[2]. M.D. Schroeder et al. Autonet: A High-Speed, Self-Con9guring Local Area Network Using Point-to-Point Links. IEEE Journal on Selected Areas in Communication, 9(8):131821335, October 1991.

[3]. W. Dally and B. Towles. Route Packets, Not Wires: On-Chip Interconnection Networks. In Proceedings of the Design Automation Conference (DAC), pages 6842689. ACM, June 2001.

[4]. E. Baydal, P. Lopez, and J. Duato. A Simple and Ef9cient Mechanism to Prevent Saturation in Wormhole Networks. In Proceedings of the 14th International Parallel and Distributed Processing Symposium, pages 6172622, 2000.

[5]. M. Thottethodi, A.R. Lebeck, and S.S. Mukherjee. Self-Tuned Congestion Control for Multiprocessor Networks. In Proceedings of the 7th International Symposium on High Performance Computer Architecture, January 2001.

[6]. Dong Xiang, “Deadlock-Free Adaptive Routing in Meshes with Fault-Tolerance Ability Based on Channel Overlapping,” IEEE transactions on dependable and secure computing, vol. 8, no. 1, january-february 2011.

[7]. Antonio Robles-Gomez Aurelio Bermudez and Rafael Casado, “A Deadlock-free Dynamic Reconfiguration Scheme for Source Routing Networks using Close Up*/down* Graphs,” IEEE Transactions on Parallel And Distributed Systems, In Press.

[8]. Dong Xiang, Yueli Zhang, and Yi Pan,” Practical Deadlock-Free Fault-Tolerant Routing in Meshes Based on the Planar Network Fault Model,” IEEE Transactions on Computers, vol. 58, no. 5, May 2009.

[9]. M.E. Gomez, J. Flich, P. Lopez, A. Robles, J. Duato, N.A. Nordbotten, O. Lysne, and T. Skeie, “An Effective Fault-Tolerant Routing Methodology for Direct Networks,” Proc. 33rd Int’l Conf. Parallel Processing (ICPP ’04), pp. 222-231, 2004.

[10]. M.E. Gomez, N.A. Nordbotten, J. Flich, P. Lopez, A. Robles, J. Duato, T. Skeie, and O. Lysne, “A Routing Methodology for Achieving Fault Tolerance in Direct Networks,” IEEE Trans. Computers, vol. 55, no. 4, pp. 400-415, Apr. 2006.

[11]. Olav Lysne, Jose Miguel Montanana, Jose Flich, Jose Duato, Timothy Mark Pinkston, and Tor Skeie, “An Efficient and Deadlock-Free Network Reconfiguration Protocol,” IEEE Transactions on Computers, Vol. 57, NO. 6, June 2008

[12]. Olav Lysne, Timothy Mark Pinkston, and Jose´ Duato, “Part II: A Methodology for Developing Deadlock-Free Dynamic Network Reconfiguration Processes,” IEEE Transactions on Parallel And Distributed Systems, Vol. 16, No. 5, May 2005

[13]. Timothy Mark Pinkston, Ruoming Pang, and Jose Duato, “Deadlock-Free Dynamic Reconfiguration Schemes for Increased Network Dependability,” IEEE Transactions on Parallel And Distributed Systems, Vol. 14, NO. 8, August 2003.

[14]. Mark Karol, S. Jamaloddin Golestani, and David Lee, “Prevention of Deadlocks and Livelocks inLossless Backpressured Packet Networks,” IEEE/ACM Transactions on Networking, Vol. 11, NO. 6, December 2003.

[15]. Yong Ho Song and Timothy Mark Pinkston, “A Progressive Approach to Handling Message-Dependent Deadlock in Parallel Computer Systems,” IEEE Transactions on Parallel And Distributed Systems, Vol. 14, No. 3, March 2003

[16]. Juan-Miguel Martınez Rubio, Pedro Lopez, and Jose Duato, “FC3D: Flow Control-Based Distributed Deadlock Detection Mechanism for True Fully Adaptive Routing in Wormhole Networks,” IEEE Transactions on Parallel And Distributed Systems, Vol. 14, NO. 8, August 2003

[17]. Jose Duato, Member and Timothy Mark Pinkston,” A General Theory for Deadlock-Free Adaptive Routing Using a Mixed Set of Resources,” IEEE Transactions on Parallel And Distributed Systems, Vol. 12, No. 12, december 2001

[18]. Sugath Warnakulasuriya and Timothy Mark Pinkston,” A Formal Model of Message Blocking and Deadlock Resolution in Interconnection Networks” IEEE Transactions 0n Parallel And Distributed Systems, Vol. 11, NO. 3, March 2000

[19]. Eric Noel and K. Wendy Tang, “Performance Modeling of Multihop Network Subject to Uniform and Non-uniform Geometric Traffic,” IEEE/ACM Transactions on Networking, Vol. 8, NO. 6, December 2000.