Stratagems to perk up the Up*/Down* Routing Scheme

Lalit Kishore Arora*
R. K. Bhatia**

Abstract

Networks of workstations (NOWs) are often uses irregular interconnection pattern. Irregular topology provides the wiring flexibility, scalability, and incremental expansion capability required in the interconnection network. Up*/down* is the most popular routing scheme currently used in NOWs with irregular topologies. Several solutions have been proposed in order to improve the up*/down* routing scheme. In this paper some of the solutions to improve the up*/down* routing are discussed.

Keywords : Interconnection Networks, Networks Of Workstations, Irregular Topologies, Routing Algorithms, Source Routing.

1. Introduction

Networks of workstations (NOWs) are becoming increasingly popular as a cost-effective alternative to parallel computers. In these machines, the network connects processors using irregular topologies, providing the wiring flexibility, scalability, and incremental expansion capability required in this environment. Also, when performance is the primary concern, these network products are being used to build large commodity clusters with regular topologies[19]. Some commercial interconnects for NOWs are Myrinet[17], Servernet II [1], Autonet[14] Gigabit Ethernet[20], and InfiniBand[7]. And several high-performance interconnects have been recently introduced for NOWs, including the Quadrics QsNet [2], and QsNet II [18], and Sun Fire Link [23].

In some of these networks, packets are delivered using source routing. In this kind of networks, the path to destination is built at the source host and it is written into the packet header before it is transmitted. Switches route packets through the fixed path found at the packet header. One example of network with source routing is Myrinet [1].

Usually, NOWs are arranged as switch-based networks whose topology is defined by the customer in order to provide wiring

*MCA Department, AKGEC, Ghaziabad, India
**Computer Sc., Gurukul Kangri Vishva Vidyalaya, Haridwar, India

flexibility and incremental expansion capability. Often, due to building constraints, the connections between switches do not follow any regular pattern leading to an irregular topology. The irregularity in the topology makes the routing and deadlock avoidance quite complicate. In particular, a generic routing algorithm suitable for any topology is required.

Up*/Down* [14], [1] is the most popular routing algorithm used in the NOW environment. In this paper we discussed the up*/down* routing and the solutions to improve the performance of up*/down* routing. Section 2 we discussed the up*/down* routing and its drawbacks. Section 3, 4, 5, and 6 explain the methodologies to improve the performance of up*/down* routing.

2. Up*/Down* Routing

Up*/down* routing is the most popular routing scheme currently used in commercial networks, such as Myrinet [17]. It is a generic deadlock-free routing algorithm valid for any network topology.

Up*/down* is a distributed deadlock-free routing algorithm that provides partially adaptive routing in irregular networks. In order to fill the routing tables, a breadth-first spanning tree (BFS) on the graph of the network is computed first using a distributed algorithm. Routing is based on an assignment of direction labels ("up" or "down") to the operational links in the network by building a BFS spanning tree. To compute a BFS spanning tree a switch must be chosen as the root. Starting from the root, the rest of the switches in the network are arranged on a single spanning tree [14].

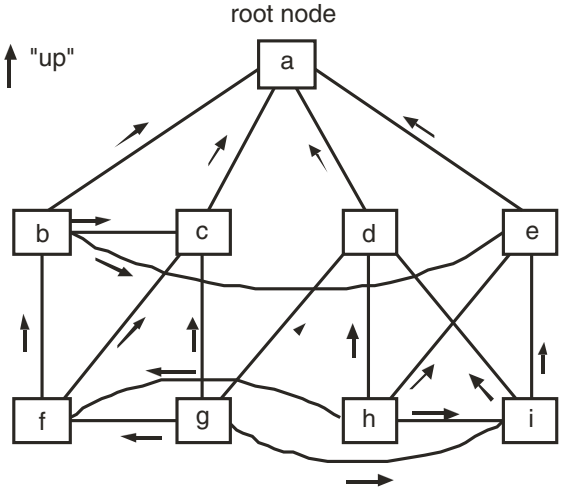

Figure 1. BFS spanning tree and assignment of directions to links for a 9 switch network

After computing the BFS spanning tree, the "up" end of each link is defined as: 1) the end whose switch is closer to the root in the spanning tree; 2) the end whose switch has the lowest identifier, if both ends are at switches at the same tree level. The result of this assignment is that each cycle in the network has at least one link in the "up" direction and one link in the "down" direction. To avoid deadlocks while still allowing all links to be used, this routing scheme uses the following up*/down* rule: a legal route must traverse zero or more links in the "up" direction followed by zero or more links in the "down" direction. Thus, cyclic channel dependencies 1 [22] are avoided because a packet cannot traverse a link in the "up" direction after having traversed one in the "down" direction.

When a message arrives at a switch, the routing algorithm is computed by accessing the routing table. The address of the table entry is obtained by concatenating the input port number with the address of the destination node stored in the message header. If there are several suitable outgoing ports, one of them is selected.

The main advantage of using up*/down* routing is the fact that it is simple and easy to implement. However, there exist several drawbacks that may noticeably reduce network performance. First of all, this routing scheme does not guarantee all the packets to be routed through minimal paths. This problem becomes more important as network size increases. In general, up*/down* concentrates traffic near the root switch, often providing minimal paths only between switches that are allocated near the root switch [11], [10]. Additionally, the concentration of traffic in the vicinity of the root switch causes a premature saturation of the network, thus obtaining a low network throughput and leading to an uneven channel utilization.

There fore the main drawbacks of up*/down* routing are the unbalanced channel utilization and the difficulties to route most packets through minimal paths, which negatively affects network performance.

Several solutions have been proposed in order to improve the up*/down* routing scheme, such as the Minimal Adaptive routing [3], the In-transit Buffer [8], the DFS methodology [12] and the Multiple Virtual Networks [16].

3. DFS Methodology

The DFS methodology [12] proposes a new methodology to compute the up*/down* routing tables that makes a different assignment of direction ("up" or "down") to links in order to increase the number of minimal paths followed by the messages. This methodology is based on obtaining a depth-first search spanning tree (DFS) instead of the BFS spanning tree used in the original methodology of up*/down* routing.

Like in the up*/down routing with BFS spanning tree, an initial switch must be chosen as the root before starting the computation of the DFS spanning tree. The selection of the root is made by using heuristic rules [11]. For instance, the switch with the highest average topological distance to the rest of the switches will be selected as the root node. The rest of the switches are added to the DFS spanning tree following a recursive procedure. Unlike the BFS spanning tree, adding switches is made by using heuristic rules [11]. Starting from the root switch, the switch with the highest number of links connecting to switches that already belong to the tree is selected as the next switch in the tree. In case of tie, the switch with the highest average topological distance to the rest of the switches will be selected first.

Next, in order to assign directions to links, switches in the network must be labeled with positive integer numbers.

When assigning directions to links, the "up" end of each link is defined as the end whose switch has a higher label. Figure 2 shows the new link direction assignment for the same network graph depicted in Figure 1. It has been shown that the DFS methodology [12] provides more minimal paths than the BFS one, resulting in a significant increase in network performance [11].

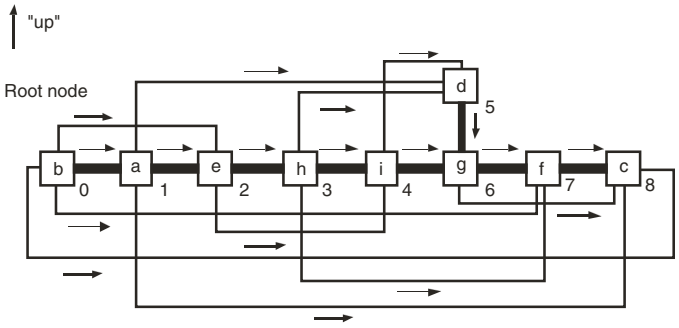

Figure 2. DFS spanning tree and assignment of directions to links for a 9 switch network

4. In-transit Buffer Mechanism

In the In-transit Buffer mechanism [8], all the minimal paths are allowed by absorbing the messages in those intermediate nodes of the path where there is a forbidden transition ("down" → "up") according to the up*/down* routing algorithm.

Basically, this mechanism avoids routing restrictions by ejecting packets at intermediate hosts and later re-injecting them. This mechanism can be easily implemented in Myrinet by modifying the network control program at the network interface card without changing the network hardware. This mechanism was originally proposed to provide minimal routing to up*/down*. In this routing algorithm, ITBs are put in all the down-up transitions. The mechanism has been extensively evaluated for both irregular [8] and regular networks [9] under different traffic patterns, network topologies, network sizes, and different message sizes. Overall, this mechanism improves on the performance achieved by up*/down*. Moreover, as network size increases, more benefits are obtained since the up*/down* routing does not scale well.

The basic idea of the mechanism is to break cyclic dependences with host buffering. The paths between source-destination pairs are computed following any given rule and the corresponding CDG is obtained. Then, the cycles in the CDG are broken by splitting some paths into sub-paths. To do so, an intermediate host inside the path is selected and used as an in-transit buffer (ITB); at this host, packets are ejected from the network as if it were their destination. The mechanism works similarly to the cut-through switching technique. Therefore, packets are re-injected into the network as soon as possible to reach their final destination. Notice that the dependences between the input and output channels of the switch are completely removed because, in the case of network contention, packets will be completely ejected from the network at the intermediate host. The CDG is made acyclic by repeating this process until no cycles are found. Notice that more than one intermediate host may be needed for a particular path [24].

As an example [24], Fig. 3.a shows a network and the assignment of link directions following the up*/down* rule.

Although there is a minimal path between switch 4 and switch 1 (4 → 6 → 1), it is forbidden because it uses an up link after a down link at switch 6. However, with the ITB mechanism (see Fig. 3.b), this path is allowed by using one host at switch 6 as an in-transit host to break the dependence. By using ITBs, minimal routing can be guaranteed while keeping deadlock freedom.

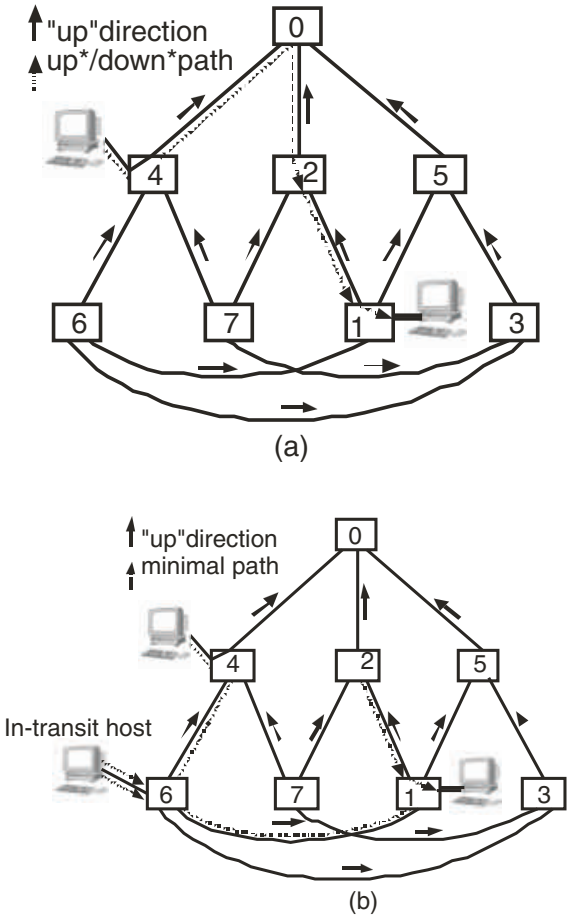

Figure 3. Link direction assignment and use of the ITB mechanism for an irregular network.

5. Minimal Adaptive Routing

The Minimal Adaptive routing [3] is only suitable for networks with distributed routing. It increases adaptivity and provides minimal paths in most cases. This algorithm requires the use of two virtual channels in order to avoid deadlocks. One virtual channel is used to route packets through minimal paths and the other one provides an escape path using the up*/down* routing.

Minimal adaptive routing [4] multiplexes each link into two virtual channels, the adaptive and escape channels, respectively. The adaptive channel is used to route packets through minimal paths without restrictions, while the escape channel is used as an escape path when the adaptive channel is busy. As proposed in [4], escape channels must be used following the up*/down* rule, which guarantees an acyclic CDG. Minimal adaptive routing is only suitable for networks with distributed routing. This routing algorithm outperforms the up*/down* routing algorithm by

providing minimal paths in most cases. However, since it cannot be applied to networks with source routing and it requires the use of virtual channels, minimal adaptive routing is not well suited for current LANs.

This methodology starts from a deadlock-free routing algorithm, splitting all the physical channels in the network into two virtual channels. We will refer to them as the original and new channels, respectively. Next, the routing algorithm is extended so that new channels are freely used with the only restriction that they must forward messages closer to their destination, and original channels are used in the same way as in the original routing function. Additionally, when a message is injected into the network, it can only leave the source switch through new channels, since they provide a higher degree of adaptivity and, usually, a shorter path. Also, when a message arrives at an intermediate switch, it first tries to reserve a new channel. If all the suitable outgoing new channels are busy, then an original channel belonging to a minimal path is selected. If none of the original channels provides a minimal path to the destination, then one of the original channels that provide the shortest path will be used. To ensure that the new routing function is deadlock-free, once a message reserves an original channel, it can no longer reserve a new one [25], [26]. This message will be routed through original channels until it arrives at the destination switch.

The minimal adaptive algorithm [25], [26] is an application of above design methodology to the up*/down* routing algorithm. This routing algorithm provides fully adaptive minimal routing between all pairs of nodes until messages are forced to move to original channels. When a message starts using original channels, it provides the same adaptivity as the up*/down* routing algorithm.

6. Multiple Virtual Networks Methodology

Using additional virtual lanes can improve up*/down* routing performance by reducing the head-of-line blocking effect, but its use is not aimed to remove its main drawbacks. The MVN methodology, uses a reduced number of virtual lanes in an efficient way to achieve a better traffic balance and a higher number of minimal paths. This methodology is based on routing packets simultaneously through several properly selected up*/down* trees. To guarantee deadlock freedom, each up*/down* tree is built over a different virtual network.

The basic idea proposed in [16] is the following. For a given network, several up*/down* trees can be easily computed by considering different root nodes. Moreover, for a given source-destination pair, some trees may allow shorter paths than others. Indeed, some trees may offer minimal paths while others not. At first, the idea is to use two of these trees to forward packets through the network. To avoid conflicts between them, every tree will use a different virtual lane (i.e., there will be two virtual networks).

Hence, for a given source-destination pair, the tree which offers the shortest path can be used. This will mitigate the non-minimal path problem of basic up*/down* routing.

Additionally, as there are now two different trees with two different roots and the paths are distributed between them, network traffic is better balanced.

It must be noticed that this routing scheme based on the use of multiple virtual networks is deadlock-free, as packets are injected into a given virtual network and remain on it until they arrive at their destinations (i.e., packets do not cross from one virtual network to another).

The proposed routing scheme in [16] is based on the use of multiple virtual networks can be easily extended to use more than two (say n) different up*/down* trees. The possibility of having more trees may lead to reduce even more average distance (by supplying shorter paths or even minimal ones) and to achieve a better link utilization. This may be especially noticeable in large networks. The only drawback of using more than two trees is that every new tree needs a different virtual network. While this may not be a problem for some applications, it can be a limiting factor for the applications that require traffic prioritization and quality of service.

In this paper, the solutions in order to improve the up*/down* routing scheme, are discussed, such as the Minimal Adaptive routing [21], the In-transit Buffer [6], the DFS methodology [16] and the Multiple Virtual Network Methodology [16]. The Minimal Adaptive routing [21] and the Multiple Virtual Channel Methodology [16] increase adaptivity and provide minimal paths in most cases. This algorithm requires the use of two or more virtual channels in order to avoid deadlocks. On the other hand, the In-transit Buffer [6] and the DFS methodology [16] increase the number of minimal paths without requiring the use of virtual channels.

We conclude that, by using In-transit Buffers [10] on the up*/down* routing scheme, network throughput is increased. As network size increases, higher improvements are obtained. In-transit buffers avoid congestion near the root switch (in the tree-based schemes), always provide deadlock-free minimal paths and balance network traffic. On the other hand, average message latency is slightly increased, but this increase is only noticeable for short messages and small networks.

7. References

1. *Horst, R., "ServerNet deadlock avoidance and fractahedral topologies", in Proc. of the Int. Parallel Processing Symp., 1996*
2. *Petrini,F. et.al, "Performance Evaluation of the Quadrics Interconnection Network", Journal of Cluster Computing, pp. 125-142, 2003.*
3. *Silla, F. and Duato, J., "High-Performance Routing in Networks of Workstations with Irregular Topology", IEEE Trans. on Parallel and Distributed Systems, vol. 11, no. 7, 2000.*
4. *Silla, F. and Duato,J.,"Improving the Efficiency of Adaptive Routing in Networks with Irregular Topology", in 1997 Int. Conference on High Performance Computing, 1997.*
5. *Silla, F. and Duato, J. , "On the Use of Virtual Channels in Networks of Workstations with Irregular Topology", in 1997 Parallel Computer Routing and Communication Workshop,1997.*
6. *Silla, F. and Duato, J., "Tuning the Number of Virtual Channels*

in Networks of Workstations", in Proc. of the 10th International Conference on Parallel and Distributed Computing Systems (PDCS'97) ,1997.

7. *InfiniBandTM Trade Association, InfiniBandTM architecture. Specification Volume 1. Release 1.0.a. Available at http://www.infinibandta.com.*
8. *Flich, J. et.al, "Performance Evaluation of a New Routing Strategy for Irregular Networks with Source Routing", Proc. Int'l Conf. Supercomputing,2000.*
9. *Flich, J. et.al, "Improving the Performance of Regular Networks with Source Routing", Proc. Int'l Conf. Parallel Processing,2000.*
10. *Flich, J. et.al, "Combining In-Transit Buffers with Optimized Routing Schemes to Boost the Performance of Networks with Source Routing", Proc. of Int. Symp. on High Performance Computing,2000.*
11. *Sancho, J. and Robles, A., "Improving the Up*/Down* Routing Scheme for Networks of Workstations", in Proc. of Euro-Par 2000, 2000.*
12. *Sancho, J. et.al,"New Methodology to Compute Deadlock-Free Routing Tables for Irregular Networks", in Proc. of 4thWorkshop on Communication, Architecture and Applications for Networkbased Parallel Computing, 2000.*
13. *Cherkasova, L. et.al, "Fibre channel fabrics: Evaluation and design", in Proc. of 29th Int. Conf. on System Sciences, 1995.*
14. *Schroeder, M. et al., "Autonet: A high-speed, self-configuring local area network using point-to-point links", SRC research report 59,1990.*
15. *Sancho, J. et.al, , "Effective Strategy to Compute Forwarding Tables for InfiniBand Networks", in Proc. of 2001 International Conference on Parallel Processing (ICPP'01), 2001.*
16. *Flich, J. et.al, "Improving InfiniBand Routing through Multiple Virtual Networks", in Int. Symp. High Performance Computing, 2002.*
17. *Boden,N.J. et al., "Myrinet - A gigabit per second local area network", IEEE Micro, vol. 15, 1995.*
18. *Quadrics. Available: http://www.quadrics.com.*
19. *Riesen, R.et al, "CPLANT", in Proc. of the 2nd. Extreme Linux Workshop, June 1999.*
20. *Sheifert, R., "Gigabit Ethernet", Addison-Wesley, 1998.*
21. *Dally, W.J. and Seitz,C.L., "Deadlock-free message routing in multiprocessors interconnection networks", IEEE Transactions on Computers, vol. C-36, no. 5, pp. 547-553, 1987.*
22. *Qiao, W. and Ni, L.M., "Adaptive routing in irregular networks using cut-through switches," in Proc. of the 1996 International Conference on Parallel Processing,1996.*
23. *Qian, Y. et.al, "Performance Evaluation of the Sun Fire Link SMP Clusters", 18th International Symposium on High Performance Computing Systems and Applications, HPCS 2004, pp. 145-156,2004.*
24. *Flich, J. et.al, "Applying In-Transit Buffers to Boost the Performance of Networks with Source Routing", IEEE Transactions On Computers, Vol. 52, No. 9,2003.*
25. *Silla, F. et.al, "Efficient Adaptive Routing in Networks of Workstations with Irregular Topology," in Workshop on Communications and Architectural Support for Network-based Parallel Computing,1997.*