

## Copyright Warning & Restrictions

The copyright law of the United States (Title 17, United States Code) governs the making of photocopies or other reproductions of copyrighted material.

Under certain conditions specified in the law, libraries and archives are authorized to furnish a photocopy or other reproduction. One of these specified conditions is that the photocopy or reproduction is not to be “used for any purpose other than private study, scholarship, or research.” If a user makes a request for, or later uses, a photocopy or reproduction for purposes in excess of “fair use” that user may be liable for copyright infringement,

This institution reserves the right to refuse to accept a copying order if, in its judgment, fulfillment of the order would involve violation of copyright law.

**Please Note: The author retains the copyright while the New Jersey Institute of Technology reserves the right to distribute this thesis or dissertation**

Printing note: If you do not wish to print this page, then select “Pages from: first page # to: last page #” on the print dialog screen



The Van Houten library has removed some of the personal information and all signatures from the approval page and biographical sketches of theses and dissertations in order to protect the identity of NJIT graduates and faculty.

## ABSTRACT

### ADAPTIVE TURN-PROHIBITION ROUTING ALGORITHM FOR THE NETWORKS OF WORKSTATIONS

by  
**Aney Bhaskar Shevtekar**

Deadlock occurrence is a critical problem for any computer network. Various solutions have been proposed over last two decades to solve problem of deadlocks in networks using different routing schemes, like up/down routing algorithm used in Myrinet switches. However, most of existing approaches for deadlock-free routing either try to eliminate any possibility of deadlock occurrence, which can result in putting extra restrictions on the routing in the networks or put no restrictions on routing, which leads to other approach namely deadlock recovery. In this thesis emphasis is on developing hybrid approach for routing in wormhole networks, wherein some prohibition is imposed on routing along with some kind of deadlock recovery. This adaptive approach allows changing the amount of routing restrictions depending on network traffic, thus providing a flexible method to achieve better network performance compared to the existing techniques. The main idea of the proposed method consists in the sequential selections of some turns, which are prohibited to be selected during routing. After each additional turn is added, the probability of deadlock occurrence decreases gradually. Cost formula is proposed to estimate cost of implementing both strategies in a network which is basis of proposed adaptive model.

**ADAPTIVE TURN-PROHIBITION ROUTING ALGORITHM FOR THE  
NETWORKS OF WORKSTATIONS**

by  
**Amey Bhaskar Shevtekar**

**A Thesis  
Submitted to the Faculty of  
New Jersey Institute of Technology  
In Partial Fulfillment of the Requirements for the Degree of  
Master of Science in Telecommunications**

**Department of Electrical and Computer Engineering**

**January 2004**

Blank Page

**APPROVAL PAGE**

**ADAPTIVE TURN-PROHIBITION ROUTING ALGORITHM FOR THE  
NETWORKS OF WORKSTATIONS**

**Amey Bhaskar Shevtekar**

---

Dr. Lev Zakrevski, Thesis Advisor  
Assistant Professor of Electrical and Computer Engineering, NJIT

Date

---

Dr. Nirwan Ansari, Committee Member  
Professor of Electrical and Computer Engineering, NJIT

Date

---

Dr. Sirin Tekinay, Committee Member  
Associate Professor of Electrical and Computer Engineering, NJIT

Date

## **BIOGRAPHICAL SKETCH**

**Author:** Amey Bhaskar Shevtekar

**Degree:** Master of Science

**Date:** January 2004

### **Undergraduate and Graduate Education:**

- Master of Science in Telecommunications,  
New Jersey Institute of Technology, Newark, NJ, 2004
- Bachelor of Engineering in Electronics and Telecommunications,  
University of Mumbai, Mumbai, India, 2002

**Major:** Telecommunications

**“We go where our vision is” -Joseph Edward Murphy**

This thesis is dedicated to my mother who has given me this vision.



## ACKNOWLEDGEMENT

I would like to sincerely acknowledge the supervision of Dr. Lev Zakrevski during this work. It was his valuable guidance that provided insight of the matter leading to fruitful outcomes.

I would like to thank Dr. Nirwan Ansari for showing confidence in my work and providing support. I would like to thank Cynthia Policard of NJWINS Lab who was always available to help with any problems in Lab. I would also like to thank Dr. Sirin Tekinay to be on my thesis review committee.

I would like to appreciate help provided by Wai Hong Ho of University of Southern California in using IRFlexSim0.5 simulator for experiments conducted.

I would like to thank all my friends and roommates who were co-operative and helpful to me during this period. Finally, I would always be indebted to my parents and other family members for their encouragement and support at each step of this work.

## TABLE OF CONTENTS

<b>Chapter</b>		<b>Page</b>
1	INTRODUCTION .....	1
	1.1 Objective .....	1
2	INTERCONNECTION NETWORKS OVERVIEW .....	2
	2.1 Background Information .....	2
	2.2 Survey of Various Interconnection Networks .....	2
	2.3 Important Switching Schemes in Interconnection Networks .....	4
	2.4 Deadlocks in Interconnection Networks .....	8
	2.5 Previous Work .....	9
3	ADAPTIVE ROUTING MODEL .....	13
	3.1 Introduction .....	13
	3.2 Model Description .....	13
	3.3 Turn Prohibition Module .....	14
	3.3.1 Model Parameters .....	14
	3.3.2 Algorithm .....	16
	3.3.3 Algorithm Description .....	16
	3.4 Cost Function Based Deadlock Recovery Module .....	17
4	RESULTS & ANALYSIS .....	19
	4.1 Introduction to IRFlexSim0.5 .....	19
	4.2 Modifications to IRFLexSim0.5 and Results .....	19
	4.3 Comparison with Turn Prohibition Algorithm .....	33
	4.4 Comparison with Up/Down Routing Algorithm .....	34

**TABLE OF CONTENTS**  
**(Continued)**

<b>Chapter</b>	<b>Page</b>
4.5 Implementing Adaptive Routing .....	35
5 CONCLUSION .....	38
APPENDIX PROBABILITY ESTIMATION CODES .....	39
REFERENCES .....	42

## LIST OF TABLES

<b>Table</b>		<b>Page</b>
2.1	Classification of Multicomputers .....	4
2.2	Comparison of Switching Techniques .....	7
4.1	Values of $P_t$ .....	21
4.2	Values of $P_c$ .....	22
4.3	Values of Blocking Score .....	23
4.4	Deadlock Probability Estimation .....	24
4.5	$P_t$ after Blocking Turn 4-3-1.....	25
4.6	$P_c$ after Blocking Turn 4-3-1 .....	26
4.7	Blocking Score after Prohibiting Turn 4-3-1 .....	27
4.8	$P_t$ after Blocking Turns 4-3-1 & 4-5-2 .....	28
4.9	$P_c$ after Blocking Turns 4-3-1 & 4-5-2 .....	29
4.10	Blocking Score after Prohibiting Turns 4-3-1 & 4-5-2 .....	30

## LIST OF FIGURES

Figure		Page
2.1	Uniform memory-access (UMA) model of a multiprocessor .....	3
2.2	Multicomputer model .....	3
2.3	Router model .....	5
2.4	Simple deadlock example .....	8
2.5	Deadlock in wormhole network .....	9
3.1	Network model .....	15
4.1	Turn occupancy probability graph for the network .....	21
4.2	Graph of cycle probability for turns .....	22
4.3	Blocking score for turns .....	23
4.4	Turn occupancy probability graph after blocking turn 4-3-1 .....	25
4.5	Graph of cycle probability after blocking turn 4-3-1 .....	26
4.6	Blocking score after prohibiting turn 4-3-1 .....	27
4.7	Turn occupancy probability graph after blocking turns 4-3-1 & 4-5-2 .....	28
4.8	Graph of cycle probability after blocking turns 4-3-1 & 4-5-2 .....	29
4.9	Blocking score after prohibiting turns 4-3-1 & 4-5-2 .....	30
4.10	Deadlock probability graph .....	31
4.11	Graph showing change in average path length .....	32
4.12	Prohibited turns using adaptive model .....	32
4.13	Prohibited turns using turn prohibition algorithm .....	33
4.14	Prohibited turns using up/down routing algorithm .....	34

**LIST OF FIGURES**  
**(Continued)**

<b>Figure</b>		<b>Page</b>
4.15	Cost graph when deadlock avoidance is costly .....	35
4.16	Cost graph when deadlock recovery is costly .....	36
4.17	Cost graph for adaptive model .....	37

# CHAPTER 1

## INTRODUCTION

### 1.1 Objective

The objective of this thesis is to develop routing strategy to circumvent problem of deadlocks in irregular networks of workstations using wormhole routing.

An adaptive routing model is proposed in this thesis, which uses both deadlock avoidance and deadlock recovery method in same routing algorithm. Deadlock avoidance module is based on turn prohibition concept to remove cyclic dependencies among the channels. New parameters are proposed to select turns to block. Proposed deadlock avoidance strategy can also be used as a complete deadlock-free routing strategy.

Deadlock recovery module of the adaptive model introduces a cost formula which can be used to decide cost of implementation of the network under adaptive scheme. Results are obtained from simulations that confirm the proposed model. Algorithm is also compared with existing routing strategies like up/down, turn prohibition routing algorithms.

Proposed adaptive routing algorithm can be used in networks with medium levels of traffic where completely deadlock-free routing algorithm is not required and some kind of deadlock recovery strategy is already available.

## **CHAPTER 2**

### **INTERCONNECTION NETWORKS OVERVIEW**

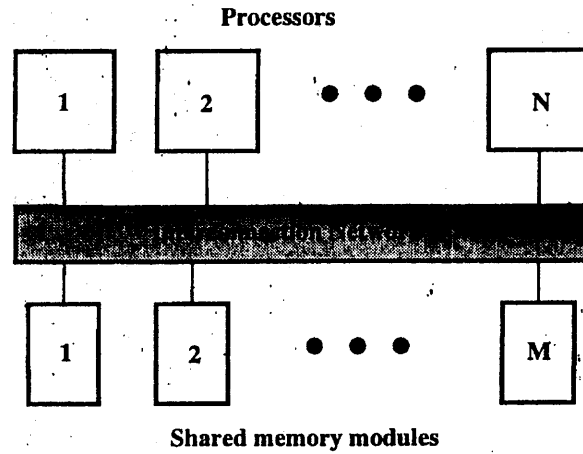
#### **2.1 Background Information**

Demand for high computational power is increasing in various defense, automotive, aerospace applications day by day. To meet this demand research efforts in terms of developing supercomputer is not a new trend, early research areas were concentrated in developing fast processors, memories etc to enhance computational power. But recent large-scale availability of personal computers and workstations have propelled the idea of achieving high performance computational power by using networks of workstations or cluster of workstations as a substitute for traditional costly supercomputers, the Berkeley NOW [1] project is a example of this idea. Interconnection network topologies, routing schemes etc plays important role in performance of these systems. A general know-how of these systems is presented in sections below.

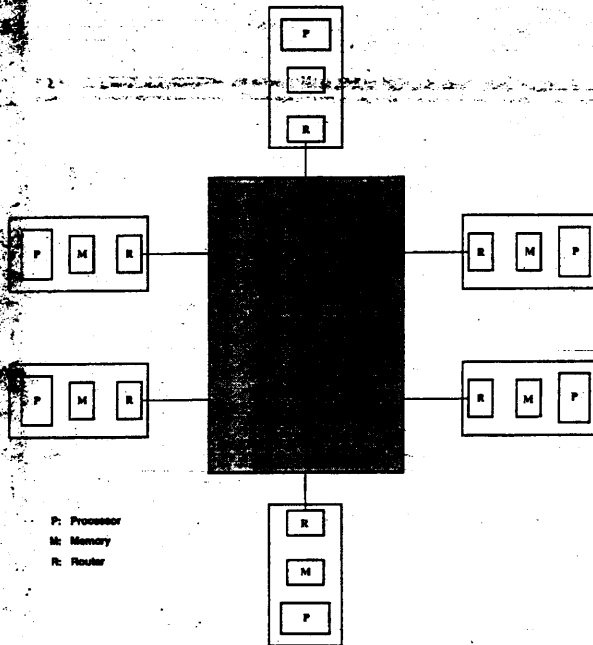
#### **2.2 Survey of Various Interconnection Networks**

Early mention of parallel architectures is found in [2] where parallelism is mentioned to occur in instruction stream, data stream, or both. These classifications are SIMD (single instruction-stream, multiple data-streams), MISD (multiple instruction-streams, single data-stream) and MIMD (multiple instruction-streams, multiple data-streams). Further MIMD is classified as multiprocessors and multicomputers shown in Figure below





**Figure 2.1** Uniform memory-access (UMA) model of a multiprocessor [3].



**Figure 2.2** Multicomputer model [3].

Basic difference between multicomputer and multiprocessor being processors share memory in multiprocessors while processors have local memory in multicomputer and inter-processor communication is done by message passing as shown in figure above. Multicomputers shown above can be classified in following major types based on interconnection network used shown in Table below.

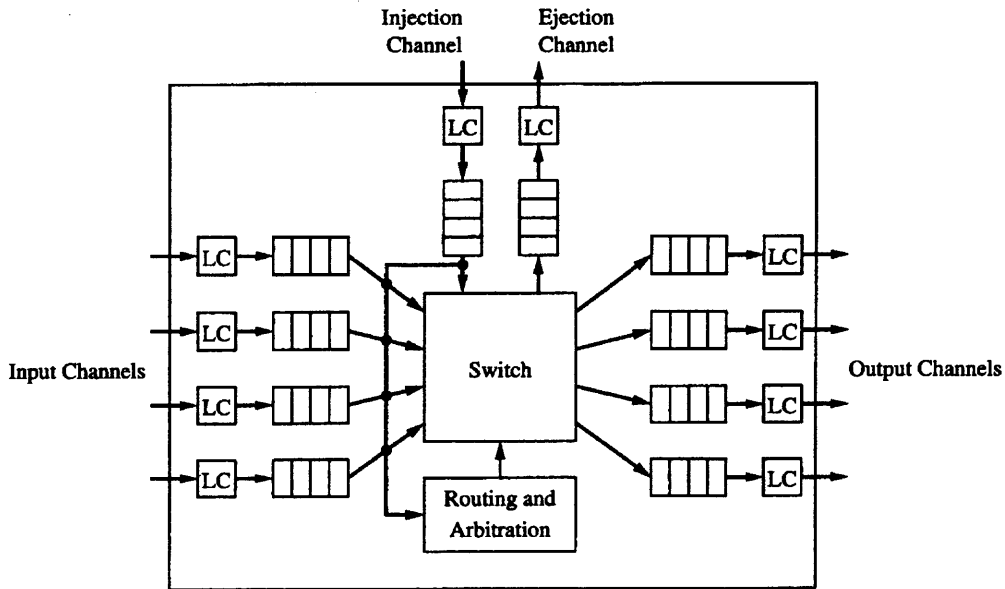
**Table 2.1** Classification of Multicomputers

<i>Shared Medium Networks</i>	Direct Networks	Indirect Networks	Hybrid Networks
LAN	Mesh	Regular topology e.g. crossbar, MIN	Multiple backplane buses
Backplane bus	Torus(k-ary n-cube)	Irregular topology e.g. autonet, myrinet	Cluster-based networks
	Hypercube		Hypergraph topologies

Source: Duato, J., Yalamanchili, S., & Ni, L. (2003). Interconnection Networks, an Engineering Approach. revised printing by elsevier science p. 8.

### 2.3 Important Switching Schemes in Interconnection Networks

Switching layer is important part of any interconnection network, initial switching methods were governed by existing LAN or WAN standards but as field of interconnection network was widely studied new techniques efficient than before came up. This subchapter deals with introduction to all switching schemes. A general router model [4] used in interconnection networks is given in Figure below.



**Figure 2.3** Router model [4].

Basic components of router model shown above are buffers to store messages in between while they are routed to destination, switches are used to connect input buffers to output buffers, routing and arbitration unit implements routing algorithms and takes corresponding routing decisions for each incoming message, link controllers controls flow of messages across physical links and processor interface is used for physical channel interface to decision making unit. Various switching techniques with their characteristics and few advantages disadvantages are discussed below.

Circuit Switching: The most widely used technique so far in which a fixed path or bandwidth is reserved for transmission of messages between source and destination. Here a routing probe contains initial routing information, which is then routed through the network, as path is established messages are sent until end of transmission. It is usually advantageous when messages are long and infrequent otherwise bandwidth remains

occupied. The base latency of a message depends on time to establish circuit and to transmit data.

Packet Switching: This technique relies on fact that messages should be broken into parts to which routing information must be attached in form of packet header and these so called packets should be then routed in the network individually. It is called store and forward switching because each packet is buffered at intermediate nodes before transmitting. The major advantage of this scheme is not a single link remains unutilized as in circuit switching where if there is no transmission link remains idle and it cannot be used by other messages. On other hand it adds additional overhead on the system. The latency of packet depends on distance between source and destination nodes.

Virtual Cut Through Switching: This technique is necessarily based on concept of packet switching but is different in one sense i.e. intermediate nodes in packet switching wait until entire packet is received before taking any forwarding decision whereas in virtual cut through switching if few bytes of packet containing routing information is received, packet is forwarded on the desired channel remaining bytes are then forwarded behind them. Thus time required to wait until entire packet is received is eliminated in this technique. Latency of messages is significantly reduced here.

Wormhole Switching: Buffer capacity problems in packet switching are solved in wormhole switching enabling use of fast, compact & small routers in the network. A message is split in small units called flits. Only the starting flit contains header information called header flit, rest data flits are pipelined through the network behind header flit. This technique helps to keep small compact buffers at routers to store just flits, thus each data flit can be just forwarded by intermediate routers. Latency hence

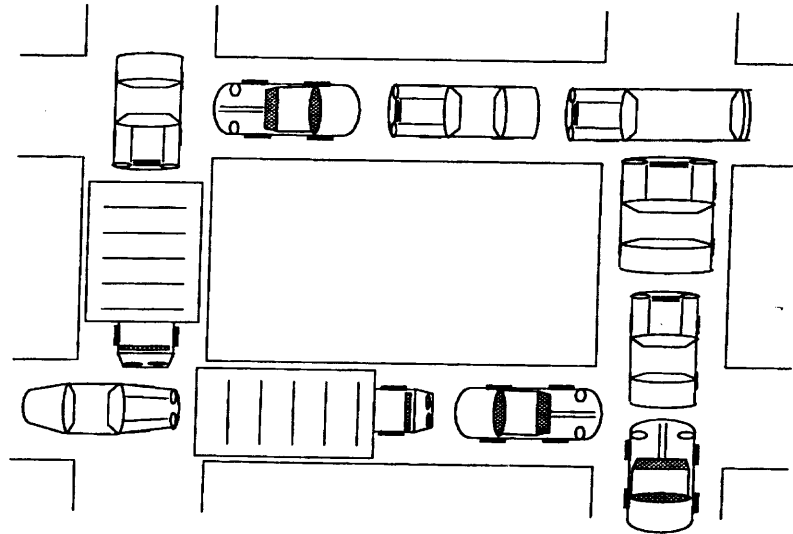
depends on intermediate switch and wire delays and is independent of distance between source and destination nodes. But once a header flit gets blocked in between flits occupy multiple routers and problem of deadlock gets aggravated because of resulting cyclic dependencies.

**Table 2.2** Comparison of Switching Techniques

	<b>Circuit Switching</b>	<b>Packet Switching</b>	<b>Worm hole switching</b>
<i>Path from Src-Dest</i>	Reserved	Not Reserved	Not Reserved
<i>Unit of Transmission</i>	Messages	Packets	Flits
<i>Header Information</i>	Not Applicable	In all Packets	Only Header flit
<i>Link Utilization</i>	Not full	Full	Full
<i>Buffer Capacity</i>	Large	Large	Small
<i>Latency</i>	High	High	Small

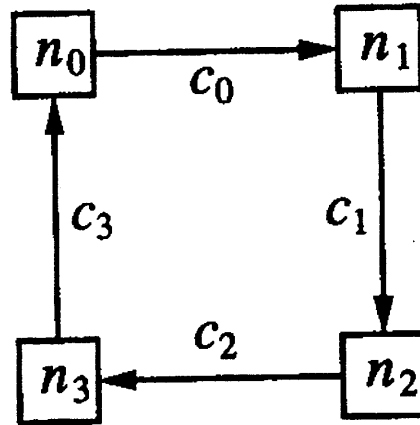
## 2.4 Deadlocks in Interconnection Networks

A simple example of deadlock in day-to-day life is shown Figure 2.4 below.



**Figure 2.4** Simple deadlock example [21].

In a message passing multi-computer model several nodes send and receive messages or packets, these packets are routed through the network where they are stored in intermediate node buffers if output channel is not available to use immediately. Buffer space at these nodes is finite and at some point of time there is a possibility that a packet does not get output channel as resources are blocked by other packets, if these blocking of resources is in cyclic manner, resulting in unavailability of resources to other involved packets cause deadlock. Deadlock can bring system to standstill resulting in decreased performance of the network. An example of deadlock in interconnection network is shown in Figure below.



**Figure 2.5** Deadlock in wormhole network [4].

In above example network has four nodes and channels connecting them, lets say there is message in  $C_1$  and is requesting channel  $C_2$ , there is another message in  $C_2$  and it is requesting  $C_3$  similarly there is request pattern for  $C_3$ - $C_0$  and  $C_0$ - $C_1$ . Such a cyclic dependency leads to deadlock. Strategies proposed to date to circumvent deadlocks can be classified broadly as deadlock avoidance schemes and deadlock prevention schemes. Both these schemes are widely studied for many years next section provides a glimpse of all work done in both these areas.

## 2.5 Previous Work

Among different Interconnect network topologies Hypercube, Meshes [4], etc are widely used for long period of time. Nowadays uses of network switches like MYRINET [7], AUTONET [8], and SERVERNET [9] have made possible use of NOWs (Networks of Workstations) as alternative to regular topology multiprocessors. NOWs are collections of workstations using high speed interconnect technology. Important physical constraints, which play important role on performance of the system, are arrangement of wires in

small space and number of pins per chip dedicated to communication board so irregular topologies are preferred because of greater wiring flexibility than traditional regular structures [4] [10]. However, as topology becomes irregular shortest path routing and deadlock free routing become two opposite factors whose optimization becomes a key factor while designing a routing algorithm [10]. Popular interconnect topologies using MYRINET [7] switches use up/down routing strategy to avoid deadlocks. Up/Down routing strategy was first used in AUTONET [8] where a spanning tree is generated and messages are either sent in up or down direction.

As said earlier up/down routing imposes restrictions on routing freedom so many improvements were suggested in up/down routing in [11] several link directions are analyzed and various effective heuristics are proposed.

A non-spanning tree approach is proposed in [12] where concentration is given on finding cyclic dependencies in a network at any instant and breaking those with consideration of some heuristic cost function. Another approach in this category is proposed in [13] where an algorithm is proposed to select nodes of a network following certain criteria introduced then block turns involving that node this process is iterated for all nodes until network becomes cycle free.

Study on characterization of Deadlocks by USC SMART Interconnects Group [14] shows that deadlocks are rare and occur when network is approaching to saturation. This work suggests use of deadlock recovery technique as a viable way to deal with problem of deadlocks. DISHA [15] is one of the widely considered schemes used to recover from Deadlocks. Another routing strategy proposed defines selecting a Pseudo-



Hamilton cycle of links in Irregular network that could form cycles. This path is then used as an escape path for blocked packets [16].

There are several issues with all the approaches discussed above deadlock avoidance methods are easy to implement, as we need to restrict certain paths while routing if necessary conditions are met. Also degree of adaptivity can be increased by using more virtual channels thus giving more options for routing and thereby reducing frequency of deadlocks/cycles compared to strategy without virtual channels as shown in [14]. On other hand these schemes since based on principle of restriction do not fully utilize network resources example up/down routing [8] in which routing is either only in up or down direction based on ordering of nodes. Thus we are not using shortest path for certain pairs of source destination leading to increase in average path lengths. In up/down routing there is another problem of imbalance of traffic in the network addressed by [17] which classifies turns then selects certain set of turns to block but it ignores affect on path length as it considers fully cycle free network. Also as said adaptively increasing virtual channels results in increase in complexity of router design and affects performance [18]. Algorithms proposed in [13] have drawbacks of concentration of turns in certain parts of network, which can lead to imbalance of traffic.

Deadlock Recovery strategies certainly do not impose any restrictions on routing being fully adaptive instead they allow deadlocks to occur and then remove blocked messages. These messages are then sent to there destinations by using allocated escape paths. Recovery schemes are also independent of network topology and switching techniques.

Success of recovery schemes is directly related to deadlock detection mechanisms [19] [20]. Deadlock detection should be true and fast as it is important to detect deadlocks not just blocked messages. It is important that deadlocks are recovered faster and resources are available back so that there is no stress on recovery resources used, which would otherwise limit performance of the scheme. Another disadvantage of implementing recovery schemes is excess hardware resource requirements like Deadlock Buffer in scheme DISHA [15].

This thesis makes an attempt to develop hybrid routing strategy as both deadlock recovery and avoidance approaches do not lead to complete solution as seen in strategies described above, motivation behind this hybrid model is to maximize advantages of both strategies in one algorithm. Hybrid routing strategy can be used adaptively based on cost model proposed in the thesis, which provides a good degree of freedom for network designer.

## CHAPTER 3

### ADAPTIVE ROUTING MODEL

#### 3.1 Introduction

Adaptive routing model is proposed as a third solution to problem of deadlocks in irregular networks using wormhole routing. It consists of two modules, Turn Prohibition module and Cost Function Based Deadlock Recovery module. Both modules are independent of each other, while recovery module addresses issue of implementing deadlock recovery scheme under presence of turn prohibition module. Turn prohibition module works on deciding turns to block to make the network completely cycle free. Proposed strategy as said before is a third solution to solve problem of deadlocks in wormhole networks, as deadlock avoidance and deadlock recovery strategies are already known. Future sections deal with general introduction to the subject and then end with detail explanations of both modules.

#### 3.2 Model Description

Network described in the explanation is modeled as a graph  $G$  with  $M$  nodes and  $N$  links. Link  $l$  is defined as connection between two nodes  $(n_1, n_2)$  of graph  $G$ , where  $n_1, n_2$  are subset of nodes  $M$  and link  $l$  is subset of links  $N$ . Links are considered to be bi-directional for analysis presented in thesis. Degree of node is defined as number of connections that start from particular node.

A path in graph  $G$  from nodes  $n_1$  to  $n_2$  is sequence of nodes  $\{n_1, n_3, n_9, n_7, n_2\}$ . Each node in above sequence is connected by a link. A Cycle in graph  $G$  is path where

first and last links are same, e.g. {n1, n3, n6, n9, n2, n1, n3}. Cycles in graph indicates dependencies that can form among links that can lead to deadlock. Breaking all cycles in graph is demanding requirement to solve problem of deadlocks.

*Turn (k-l-m)* - A pair of adjacent links (*k-l*) and (*l-m*) around node *l* is called a *Turn*. For bi-directional graphs that are used for analysis in thesis turn (k-l-m) is same as turn (m-l-k). The number of turns around node of degree *d* is equal to  $d(d-1)/2$ . The concept of turn prohibition can be explained as follows assume turn (k-l-m) is prohibited; it means messages if coming from node *k* are going to node *m* via node *l* are blocked and are sent using another path from node *k* instead of k-l-m. Similar blocking is done from m-l-k.

### 3.3 Turn Prohibition Module

Turn prohibition module of adaptive routing model describes algorithm to block turns to make an irregular network completely cycle free. Novel part of this turn prohibition algorithm is that it makes an attempt to balance effect on average path length after blocking of turns. Algorithm relies on sequential selection of turns to block.

#### 3.3.1 Model Parameters

*Turn Occupancy Probability ( $P_t$ )* - This term indicates occupancy of each turn. It emphasizes on use of turn for routing by messages in average hence is a factor that opposes blocking of a turn.

*Cycle Probability for Turn ( $P_c$ )* - This term indicates probability that a turn would be part of a deadlock-creating cycle. It gives the probability that particular turn is present in all

possible deadlock resulting cyclic dependencies hence is a factor that demands blocking of turn to avoid deadlocks.

Cycle probability ( $P_{c_i}$ ) for turn  $i$  is

$$P_{c_i} = \sum_j P_{c_j}$$

$$j \in \{0, 1, \dots, n\}$$

$P_{c_j}$  is  $P_{c_1}, P_{c_2}, P_{c_3}, \dots, P_{c_j}$  where  $c_1 - c_j$  indicates cycle in which turn  $i$  is involved and  $n$  indicates number of cycles in which turn  $i$  is involved.

Example of  $P_c$  calculation is shown below,

Let us consider turn 1-2-5 of network shown in Figure 3.1

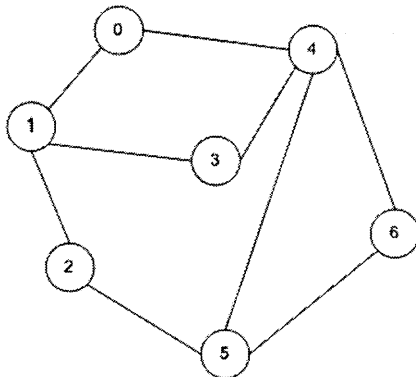
$$P_{c_{1-2-5}} = P_{c_{1-2-5-4-0-1}} + P_{c_{1-2-5-4-3-1}} + P_{c_{1-2-5-6-4-3-1}} + P_{c_{1-2-5-6-4-0-1}}$$

Each  $P_c$  depends upon on  $P_t$  of turns in the corresponding cycle, thus:

$$P_{c_{1-2-5-4-0-1}} = P_{t_{1-2-5}} \times P_{t_{2-5-4}} \times P_{t_{5-4-0}} \times P_{t_{4-0-1}}$$

Similarly  $P_{c_{1-2-5-4-3-1}}, P_{c_{1-2-5-6-4-3-1}}, P_{c_{1-2-5-6-4-0-1}}$  are calculated to get

$P_{c_{1-2-5}}$



**Figure 3.1** Network model.

*Blocking score of a turn (B)* - It is defined as ratio of  $P_c / P_t$  and is used to decide turn to block.  $P_c$  contributes towards blocking of a turn, as it would reduce expected deadlock

frequency involving that turn.  $P_t$  is a factor that is against blocking because it is indicating use of that turn by circuits and if it is blocked average path length would increase.

Thus ratio of these terms would give a value to decide whether a particular turn should be blocked. Ratio is not taken other way round because it defies the central objective of module i.e. turn prohibition.

### 3.3.2 Algorithm

Basic steps of this algorithm are presented in this section.

*Step 1:* Collect number of times each turn in the network is used for routing.

*Step 2:* Calculate  $P_t$ ,  $P_c$ , &  $B$  for all turns in the network.

*Step 3:* Select turn with highest value of  $B$  to prohibit. If many turns have same highest  $B$  value, then randomly choose any one to prohibit.

*Step 4:* Find all cycles in the network and compute Total Deadlock Probability.

*Step 5:* Block turn obtained in step 3 by recomputing routing table. Repeat steps 1 to 4.

*Step 6:* Stop iterations when Total Deadlock Probability becomes zero.

Step 1 of algorithm can be estimated by performing simulation analysis of networks with different traffic models or can be estimated from real-life traffic in the network topologies under consideration.

### 3.3.3 Algorithm Description

Turn prohibition algorithm estimates the probability values and blocking score for all turns to decide which turn to block. Before blocking total deadlock probability is estimated, it is very important parameter for the algorithm as it indicates what would be reduction in deadlock probability after blocking a turn. It is estimated in same manner as

$P_c$ , all cycles in the network are found then probability that each cycle will form is found by product of  $P_t$  of turns by which that cycle is made. Adding the formation probabilities of all cycles gives Total Deadlock Probability. Algorithm then blocks the selected turn to recompute all above parameters and stops when Total Deadlock Probability reduces to zero. Algorithm breaks all cycles in the network. It can be guaranteed that all cycles are broken because after each step of turn prohibition all new possible cycles are found and then corresponding formation probability for these cycles is found considering new  $P_t$  values of turns in it. New  $P_t$  values are important since under prohibition of selected turns occupancy probabilities for all other turns behave differently and so formation probabilities for new possible cycles change significantly. Algorithm takes all these factors into account and hence when total deadlock probability is reduced to zero it can be said that all cycles in the network are broken. One important point that should be considered in case of deadlocks is message generation rate which increases probability of deadlock occurrence. Thus to take this factor into account  $P_t$  is multiplied by generation rate of messages.

### **3.4 Cost Function Based Deadlock Recovery Module**

Integral part of Adaptive Routing Model is this recovery module; it makes this model hybrid as both strategies avoidance and recovery can be incorporated in one routing strategy with help of this module of the model. Cost of a network is a very important parameter that needs to be optimized. Thesis proposes a formula to estimate cost of implementing both deadlock avoidance and recovery strategies in a network.

$$C = \text{Cost of Deadlock Recovery per Deadlock} \times \text{Probability of Deadlock occurrence} \\ + \\ \text{Cost of Routing} \times \text{Average Path Length}$$

Above formula can be split in two parts, first considers cost of recovery from deadlock under the condition that deadlock occurs and second considers cost of implementing avoidance strategy in which cost straightforward depends upon routing messages over average path lengths in the network. The second part is present even when we consider only deadlock recovery because routing is still done and recovery comes into effect when deadlock occurs on other hand in case of only avoidance first part is ignored as we do not have recovery cost.

Above formula can be used considering first module of Adaptive Routing Model. Let's say for certain network using some kind of deadlock recovery strategy requires five turns to make total deadlock probability zero. Cost is calculated after every new turn is prohibited. Cost should go on increasing as average path length goes on increasing after each turn prohibition and deadlock probability is reduced at each stage. Cost of routing and cost of recovery are constants, as they do not change. Then after observing cost at each stage we can decide where turn prohibition can be stopped i.e. after 2 turns or 3 turns. This intermediate value of turns to be blocked thus does not make the network complete cycle free when some kind of deadlock recovery strategy is available at affordable cost.



## CHAPTER 4

### RESULTS & ANALYSIS

#### 4.1 Introduction to IRFlexSim0.5

IRFlexSim0.5 is a flit level wormhole routing simulator for arbitrary topologies of networks developed by SMART Interconnects Group at University of Southern California. It is in C language and works on UNIX platform. The simulator is used to characterize performance of various routing algorithms for different simulation parameters like number of virtual channels, message length, traffic patterns and various switch characteristics like delay, buffer size etc. It has built in deadlock detection and cycle detection module [22]. This simulator is used for analysis of the proposed adaptive model. Important characteristics of simulator are freedom in selecting irregular networks, random network generation module that supports automatic generation of routing table for different routing algorithms and network statistics such as throughput, latency, average path length etc reported to standard output. Analysis and results obtained are shown in this chapter.

#### 4.2 Modifications to IRFlexSim0.5 and Results

Turn occupancy probability was defined in Adaptive Model its estimation is done using IRFlexSim0.5. Wormhole routing can be visualized as a Circuit connection between source and destination as starting flit has header information and rest data flits just follow header through the network, as a result this connection spans multiple routers. So the simulator uses a structure called "*Circuit*" to store information of different connections

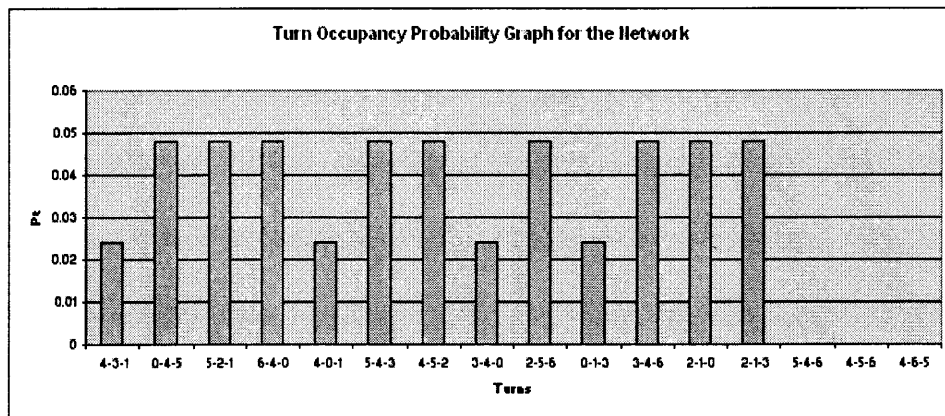
generated during the simulation run. A new variable was defined in this structure called “*previousnd*” to store all the nodes that header flit passes through while its journey from source to destination. The values of this variable are given to 2-dimensional array-defined exclusive for individual circuits generated. So for all circuits nodes used in between to travel from source to destination are obtained from which turns used by a single circuit are estimated. From the generated 2-d array estimation of how many times each turn is used by different circuits is done. This value is divided by total number of circuits generated during simulation to obtain Turn occupancy probability.

$P_t = \text{Ratio of Number of times a Turn is used by different Circuits to Total Circuits generated.}$

Simulator is used to do analysis of network parameters for randomly generated networks. For the network in Figure 3.1  $P_t$  values are generated manually based on simple calculations shown in Table 4.1 along with a graph in Figure 4.1. Example of this calculation is probability of each turn is calculated based on availability of alternate path, else for 7 x 6 source destination pairs each turn has occupancy probability 1/42. If alternate path is available occupancy probability is 1/21.

**Table 4.1** Values of  $P_t$ 

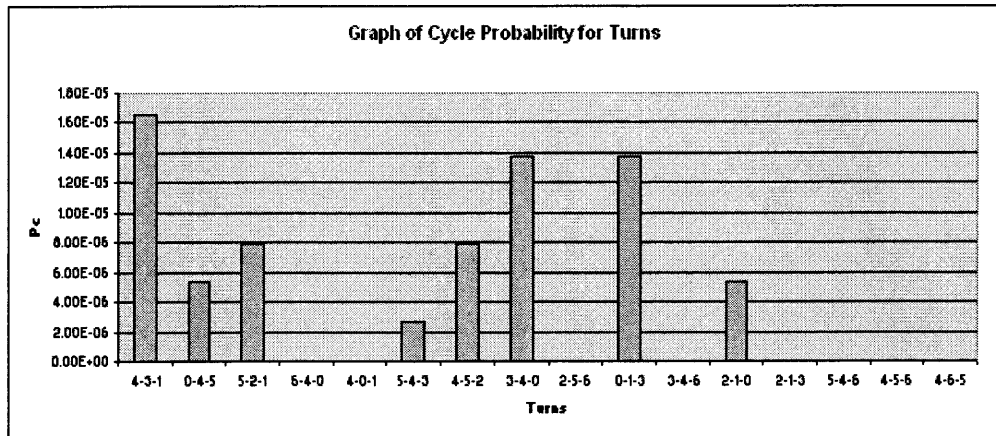
Turns	$P_t$
4-3-1	0.024
0-4-5	0.048
5-2-1	0.048
6-4-0	0.048
4-0-1	0.024
5-4-3	0.048
4-5-2	0.048
3-4-0	0.024
2-5-6	0.048
0-1-3	0.024
3-4-6	0.048
2-1-0	0.048
2-1-3	0.048
5-4-6	0
4-5-6	0
4-6-5	0

**Figure 4.1** Turn occupancy probability graph for the network.

Cycle probability for turn is then estimated by considering all cycles in which that turn is involved, this  $P_c$  values from  $P_t$  values in Table 4.1 are shown below.

**Table 4.2** Values of  $P_c$ 

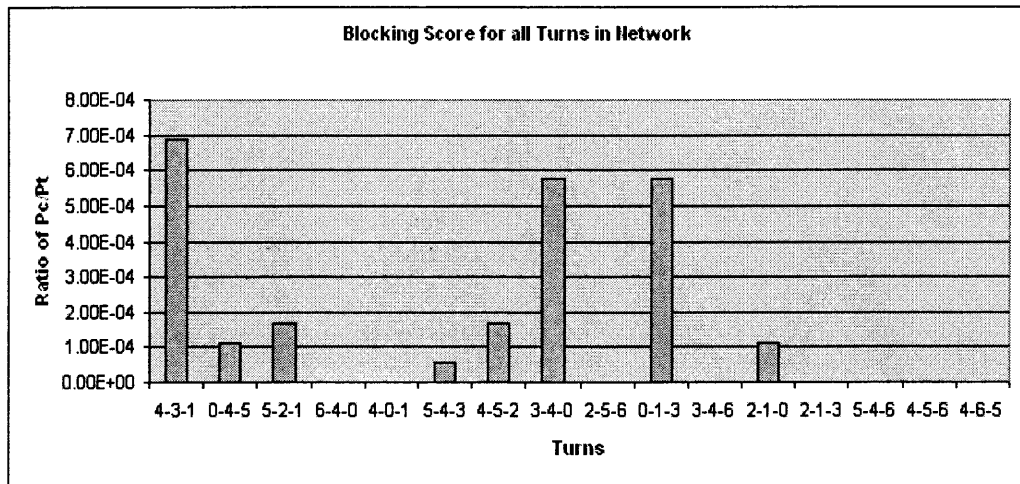
Turns	$P_c$
4-3-1	1.65E-05
0-4-5	5.31E-06
5-2-1	7.96E-06
6-4-0	0.00E+00
4-0-1	0.00E+00
5-4-3	2.65E-06
4-5-2	7.96E-06
3-4-0	1.38E-05
2-5-6	0.00E+00
0-1-3	1.38E-05
3-4-6	0.00E+00
2-1-0	5.31E-06
2-1-3	0.00E+00
5-4-6	0.00E+00
4-5-6	0.00E+00
4-6-5	0.00E+00

**Figure 4.2** Graph of cycle probability for turns.

Blocking Score now can be obtained as defined in section 3.2.3

**Table 4.3** Values of Blocking Score

Turns	$P_c / P_t$
4-3-1	6.88E-04
0-4-5	1.11E-04
5-2-1	1.66E-04
6-4-0	0.00E+00
4-0-1	0.00E+00
5-4-3	5.52E-05
4-5-2	1.66E-04
3-4-0	5.75E-04
2-5-6	0.00E+00
0-1-3	5.75E-04
3-4-6	0.00E+00
2-1-0	1.11E-04
2-1-3	0.00E+00
5-4-6	0.00E+00
4-5-6	0.00E+00
4-6-5	0.00E+00

**Figure 4.3** Blocking score for turns.

In the next step selection of turn to block is decided based on highest value of blocking score, turn 4-3-1 is selected. Turn occupancy probability and cycle probability for all turns is recomputed and new blocking score is found. This process of sequential

blocking is done till deadlock probability is reduced to zero. Calculation of deadlock probability without any blocking is shown below.

**Table 4.4** Deadlock Probability Estimation

Cycle	$P_c$
4-5-6-4	0
1-3-4-5-2-1	2.2988 E-07
0-4-5-2-1-0	9.1953 E-07
0-4-6-5-4-3-1-0	0
1-3-4-6-5-2-1	0
0-1-2-5-6-4-0-1	0
0-1-3-4-0	1.6283 E-06

*Deadlock Probability under no Blocking = Sum of  $P_c$  for all Cycles*  
 $= 2.17 E-05$

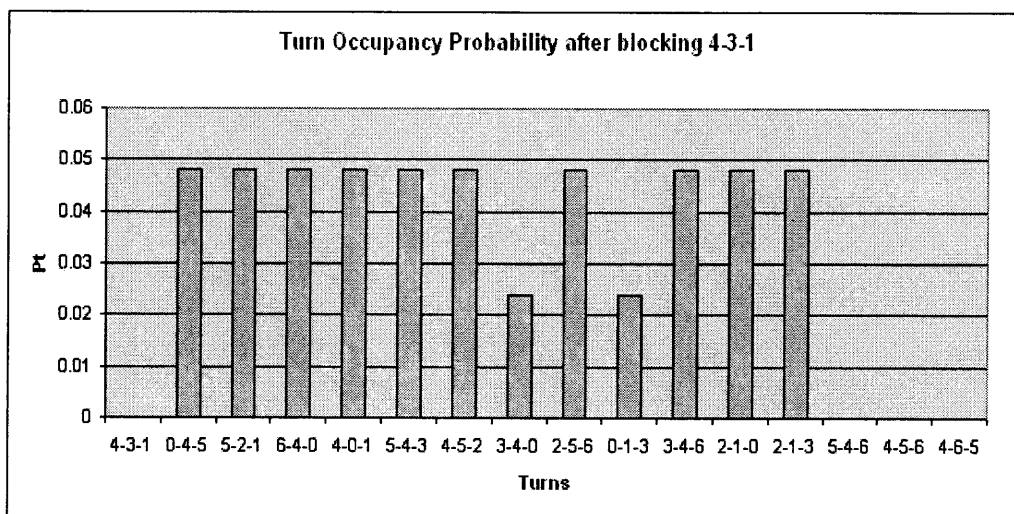
After elimination of 4-3-1

*Deadlock Probability under 4-3-1 Blocking = Sum of  $P_c$  for all Cycles*  
 $= 5.31 E-06$

After blocking, turn 4-3-1 recomputation of  $P_t$  &  $P_c$  for all turns is done i.e. with  $P_t$  value for 4-3-1 is now substituted as 0.

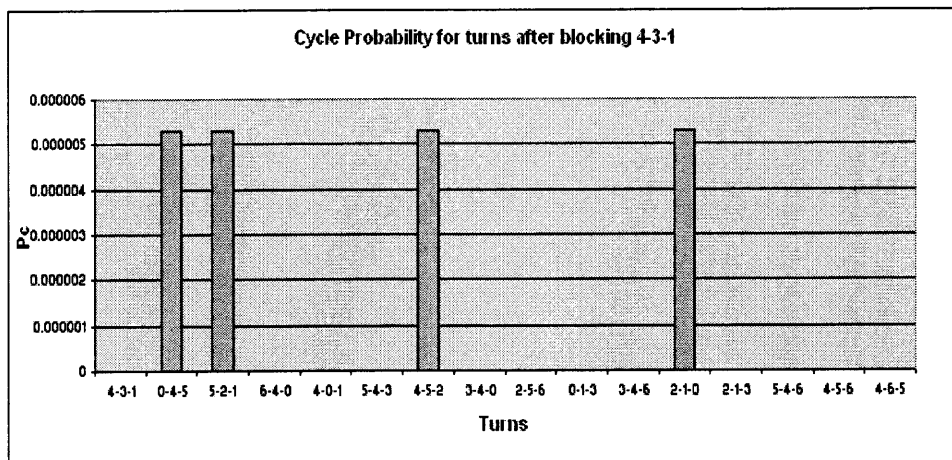
**Table 4.5**  $P_t$  after Blocking Turn 4-3-1

Turns	$P_t$
4-3-1	0
0-4-5	0.048
5-2-1	0.048
6-4-0	0.048
4-0-1	0.048
5-4-3	0.048
4-5-2	0.048
3-4-0	0.024
2-5-6	0.048
0-1-3	0.024
3-4-6	0.048
2-1-0	0.048
2-1-3	0.048
5-4-6	0
4-5-6	0
4-6-5	0

**Figure 4.4** Turn occupancy probability graph after blocking turn 4-3-1.

**Table 4.6**  $P_c$  after Blocking Turn 4-3-1

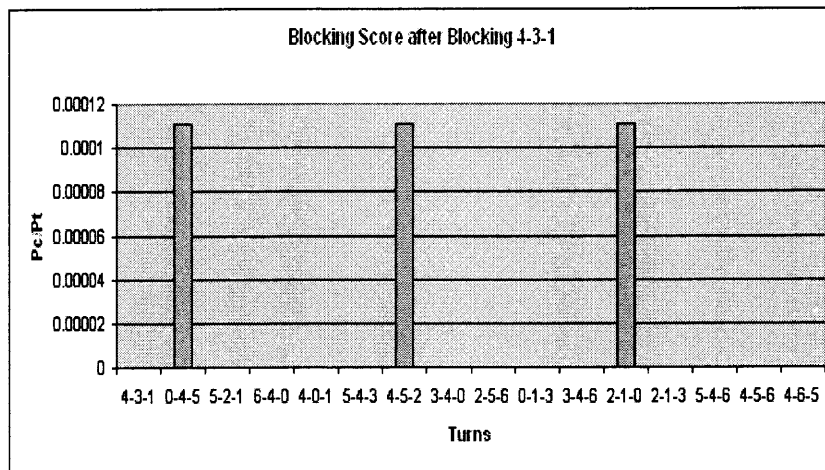
Turns	$P_c$
4-3-1	0
0-4-5	5.31E-06
5-2-1	5.31E-06
6-4-0	0.00E+00
4-0-1	0.00E+00
5-4-3	0.00E+00
4-5-2	5.31E-06
3-4-0	0.00E+00
2-5-6	0.00E+00
0-1-3	0.00E+00
3-4-6	0.00E+00
2-1-0	5.31E-06
2-1-3	0.00E+00
5-4-6	0.00E+00
4-5-6	0.00E+00
4-6-5	0.00E+00

**Figure 4.5** Graph of cycle probability after blocking turn 4-3-1.



**Table 4.7** Blocking Score after Prohibiting Turn 4-3-1

Turns	$P_c / P_t$
4-3-1	0
0-4-5	1.11E-04
5-2-1	0.00E+00
6-4-0	0.00E+00
4-0-1	0.00E+00
5-4-3	0.00E+00
4-5-2	1.11E-04
3-4-0	0.00E+00
2-5-6	0.00E+00
0-1-3	0.00E+00
3-4-6	0.00E+00
2-1-0	1.11E-04
2-1-3	0.00E+00
5-4-6	0
4-5-6	0
4-6-5	0

**Figure 4.6** Blocking score after prohibiting turn 4-3-1.

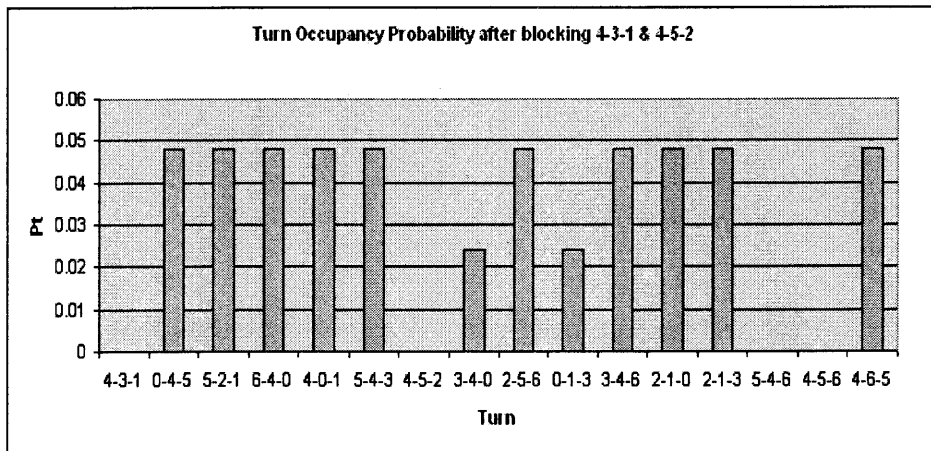
After elimination of 4-3-1, either of turns 0-4-5, 4-5-2 and 2-1-0 can be selected to block. 4-5-2 is blocked.

$\text{Deadlock Probability when 4-3-1 and 4-5-2 are Blocked} = \text{Sum of } P_c \text{ for all Cycles}$ $= 2.55 E-07$
--

From Figure 4.6 turn 4-5-2 is selected to be blocked in next step and new blocking score values are recomputed.

**Table 4.8**  $P_t$  after Blocking Turns 4-3-1 & 4-5-2

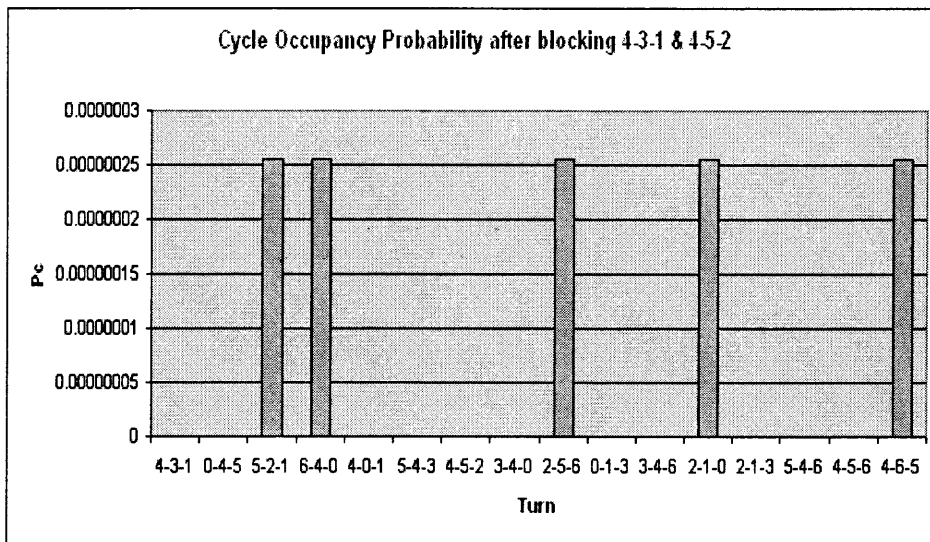
Turns	$P_t$
4-3-1	0
0-4-5	0.048
5-2-1	0.048
6-4-0	0.048
4-0-1	0.048
5-4-3	0.048
4-5-2	0
3-4-0	0.024
2-5-6	0.048
0-1-3	0.024
3-4-6	0.048
2-1-0	0.048
2-1-3	0.048
5-4-6	0
4-5-6	0
4-6-5	0.048



**Figure 4.7** Turn occupancy probability graph after blocking turns 4-3-1 & 4-5-2.

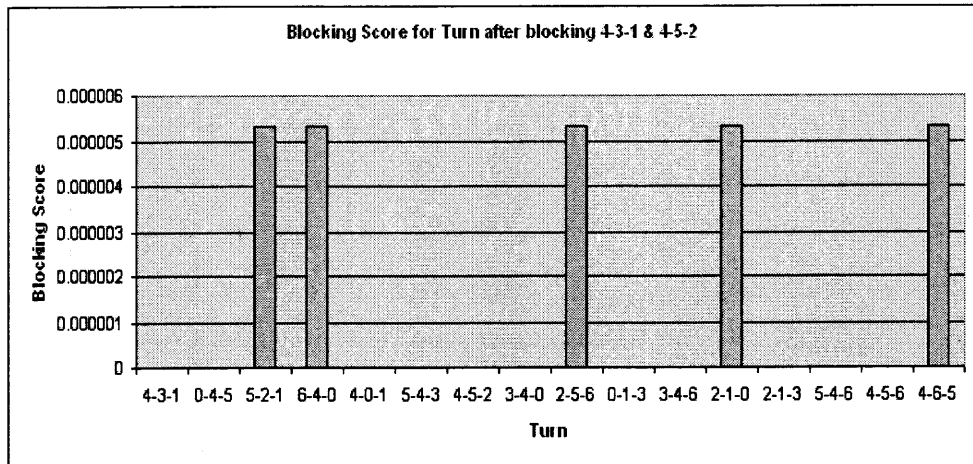
**Table 4.9**  $P_c$  after Blocking Turns 4-3-1 & 4-5-2

Turns	$P_c$
4-3-1	0
0-4-5	0
5-2-1	2.55E-07
6-4-0	2.55E-07
4-0-1	0
5-4-3	0
4-5-2	0
3-4-0	0
2-5-6	2.55E-07
0-1-3	0
3-4-6	0
2-1-0	2.55E-07
2-1-3	0
5-4-6	0
4-5-6	0
4-6-5	2.55E-07

**Figure 4.8** Graph of cycle probability after blocking turns 4-3-1 & 4-5-2.

**Table 4.10** Blocking Score after Prohibiting Turns 4-3-1 & 4-5-2

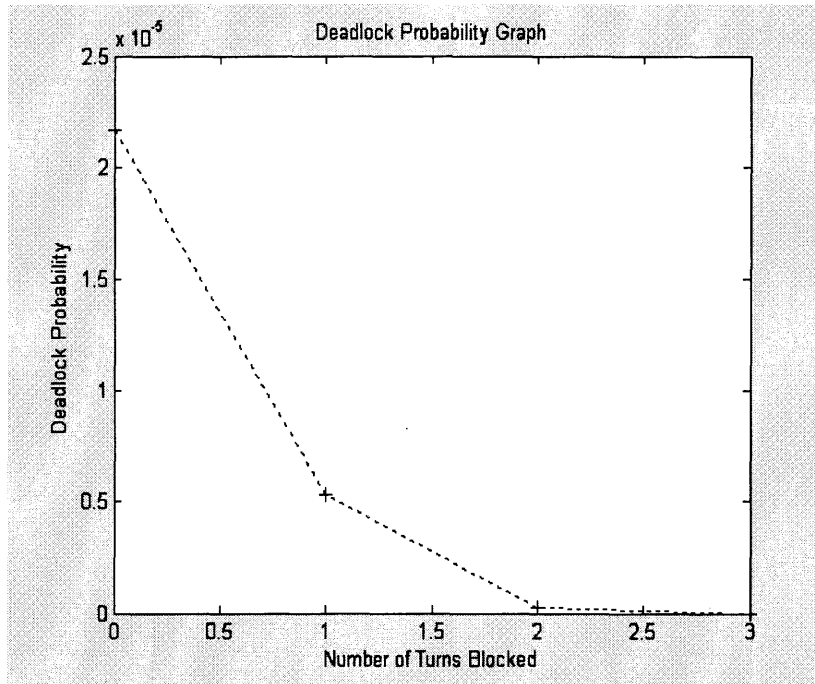
Turns	$P_c / P_t$
4-3-1	0
0-4-5	0
5-2-1	5.31E-06
6-4-0	5.31E-06
4-0-1	0
5-4-3	0
4-5-2	0
3-4-0	0
2-5-6	5.31E-06
0-1-3	0
3-4-6	0
2-1-0	5.31E-06
2-1-3	0
5-4-6	0
4-5-6	0
4-6-5	5.31E-06

**Figure 4.9** Blocking score after prohibiting turns 4-3-1 & 4-5-2.

From Figure 4.9 either of turn 5-2-1, 6-4-0, 2-5-6, 2-1-0 and 4-6-5 can be blocked. Turn 2-1-0 is randomly chosen to be blocked in next step and deadlock probability is recomputed.

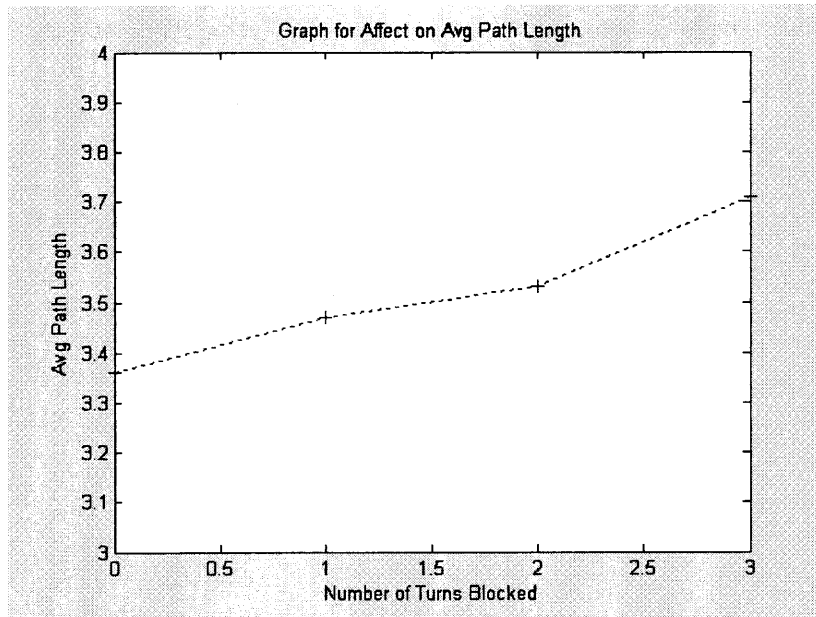
*Deadlock Probability when 4-3-1, 4-5-2 and 2-1-0 are Blocked = Sum of  $P_c$  for all cycles  
= 0*

As deadlock probability reduces to zero, algorithm does not perform next iteration. Following Figure shows graph of deadlock probability.

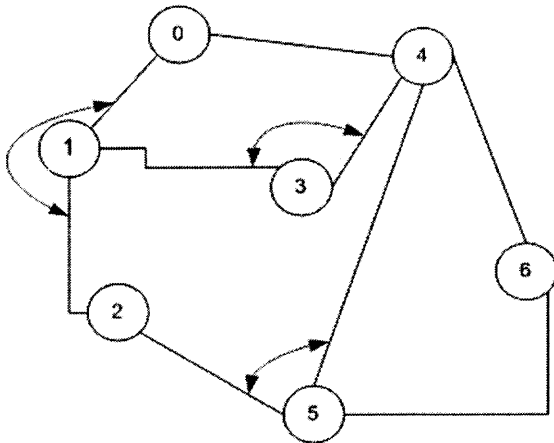


**Figure 4.10** Deadlock probability graph.

Another parameter that is affected after turn prohibition is average path length, which should obviously increase as routing restriction, is imposed. Following Figure shows graph of change in average path length.



**Figure 4.11** Graph showing change in average path length.



**Figure 4.12** Prohibited turns using adaptive model.

### 4.3 Comparison with Turn Prohibition Algorithm

Turn prohibition algorithm [13] is completely cycle free.

Basic steps

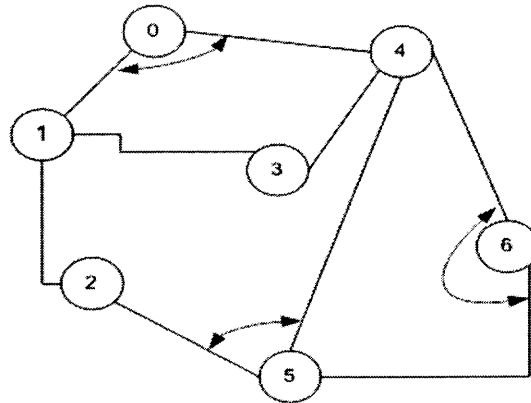
Step 1: Select node with lowest degree.

Step 2: Prohibit all turns around that node.

Step 3: Delete that node and links incident on it.

Step 4: Repeat the procedure until all nodes are deleted.

TP-Algorithm for our network prohibits 3 turns shown in figure below. Key observation being performance of proposed Module 1 of adaptive algorithm is close to TP-Algorithm.

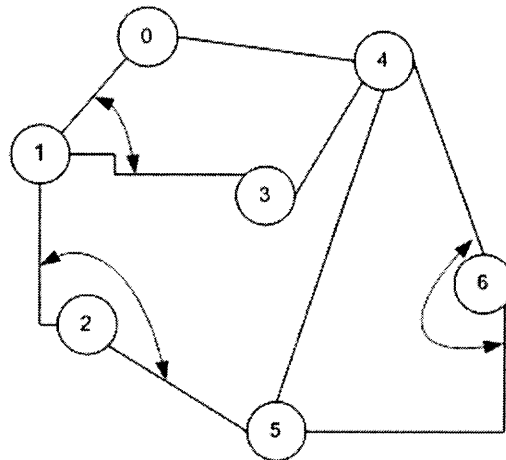


**Figure 4.13** Prohibited turns using turn prohibition algorithm.

#### 4.4 Comparison with Up/Down Routing Algorithm

Up/Down routing algorithm [8] is completely deadlock free routing algorithm. In Up/Down routing messages can be routed either in up or down direction, direction is decided by node numbers e.g. valid routing path can be  $4 - 6 - 7 - 11$ ,  $9 - 5 - 3 - 1$  but  $4 - 6 - 2 - 9$  is invalid path. Turn  $0-1-3$ ,  $1-2-5$  and  $5-6-4$  is blocked because prohibiting down/up turns eliminates all cycles, where down/up turn  $(a b c)$  follows condition  $a < b$  &  $c < b$ .

Up/down algorithm has a problem of root selection as number of turns prohibited directly depends on it. It also suffers from congestion near the root. Key observation being performance of proposed Module 1 of adaptive algorithm is close to Up/Down routing algorithm.



**Figure 4.14** Prohibited turns using up/down routing algorithm.



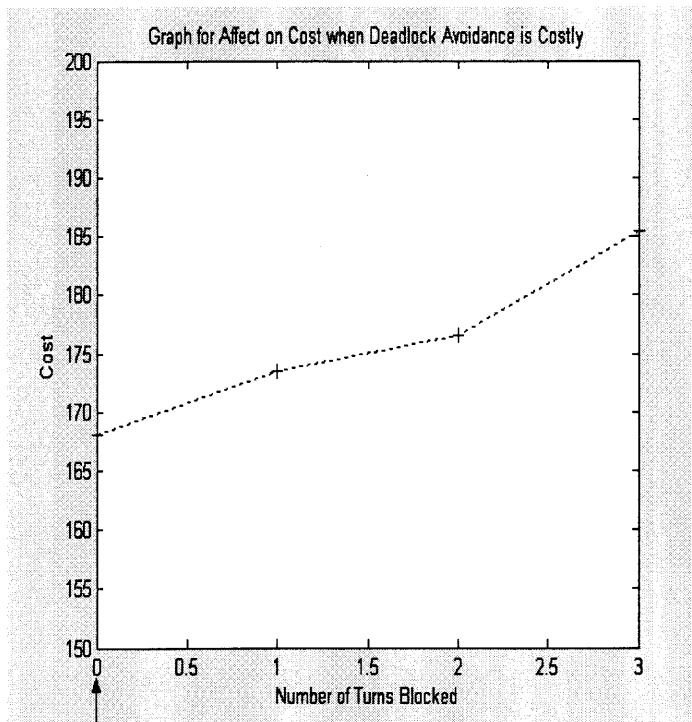
### 4.5 Implementing Adaptive Routing

Cost formula proposed in deadlock recovery module is used to estimate cost of the network at various stages i.e. after each turn is prohibited and for different costs of routing and deadlock recovery methods. Consider the formula

$$C = \text{Cost of Deadlock Recovery per Deadlock} \times \text{Probability of Deadlock occurrence} + \text{Cost of Routing} \times \text{Average Path Length}$$

Three different scenarios are explained below,

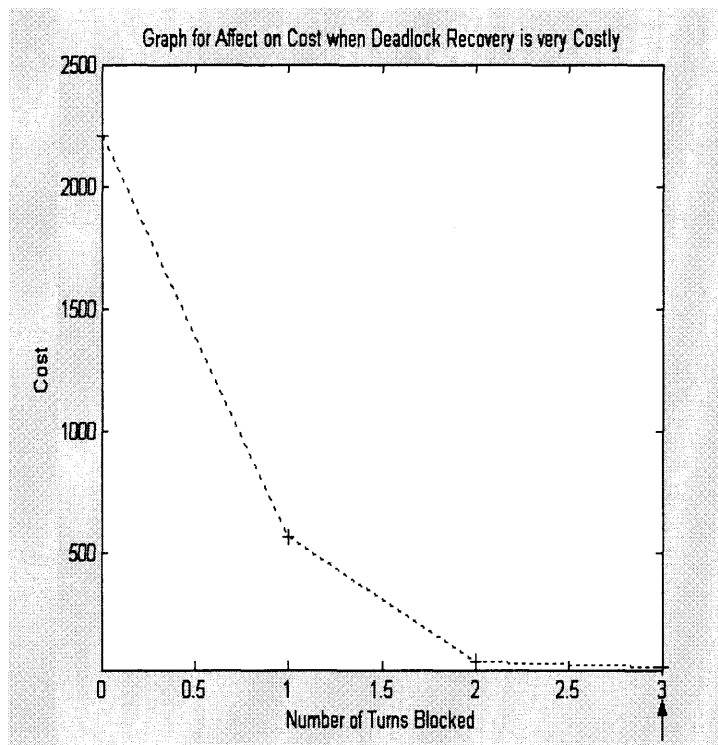
*Case 1:* Experimental value of cost of deadlock recovery per deadlock is considered 100 and cost of routing is 50. Average path length and probability of deadlock values are used from results shown before.



**Figure 4.15** Cost graph when deadlock avoidance is costly.

In this case it can be observed that deadlock avoidance approach is costly and recovery is the cost-effective option.

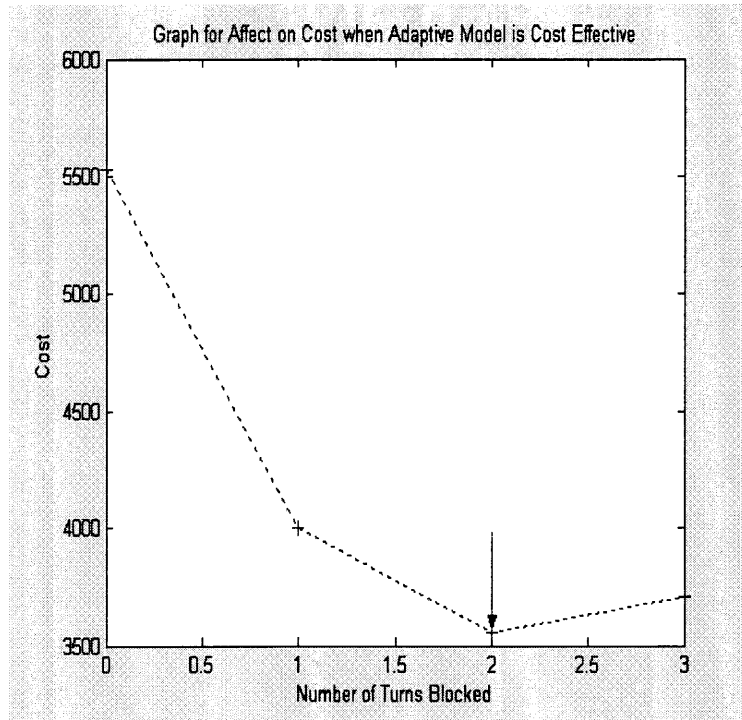
*Case 2:* Experimental value of cost of deadlock recovery per deadlock is considered  $10E+08$  and cost of routing is 10. Average path length and probability of deadlock values are used from results shown before.



**Figure 4.16** Cost graph when deadlock recovery is costly.

In this case it can be observed that deadlock recovery approach is costly and avoidance is the cost-effective option.

*Case 3:* Experimental value of cost of deadlock recovery per deadlock is considered  $10E+08$  and cost of routing is 1000. Average path length and probability of deadlock values are used from results shown before.



**Figure 4.17** Cost graph for adaptive model.

In this case, it can be observed that both deadlock recovery and avoidance approach is not very useful and cost effective solution is to prohibit only two turns and then allowing some form of deadlock recovery. Thus, by using proposed Adaptive Model both avoidance and recovery strategy can be used in a network simultaneously.

## CHAPTER 5

### CONCLUSION

In this work a novel approach is developed for a problem in computer network using wormhole routing. Proposed solution is a third solution for problem of deadlocks. Traditional methods are deadlock avoidance and deadlock recovery strategies. The proposed algorithm can be used sequentially, each time selecting next turn to be prohibited in the computer network. As the result, the amount of routing restrictions can be changed based on the network traffic. Deadlock avoidance strategies work well when incoming message rate is high but for low network traffic deadlock recovery is better option.

In real life scenarios often the incoming message rate is somewhere in between, so it is not necessary to use a complete deadlock free routing algorithm but an adaptive approach can be preferred if some form of deadlock recovery is already available. The efficiency of the proposed algorithm has been compared with the deadlock-free routing (up/down routing algorithm has been used) and routing without restrictions (using deadlock recovery scheme). Preliminary estimations show, that depending on the network traffic the proposed hybrid approach provides about 15% performance improvement compared to both other methods.

Direction of future work is to collect more experimental data. Namely, comparing the average delay in randomly generated networks (up to 200 nodes), using 3 considered approaches (depending on average message incoming rate).

## APPENDIX

### PROBABILITY ESTIMATION CODES

Matlab Code to generate  $P_t$  and  $P_c$  from Simulation Results

```
A = [ 4 3 1 0; 0 4 5 0; 5 2 1 0; 6 4 0 0; 4 0 1 0; 5 2 1 0; 5 4 3 0; 4 5 2 0; 3 4 0 0; 2 5 6 0; 2
5 6 0;
    0 1 3 0; 3 4 6 0; 5 4 0 0; 5 2 1 0; 1 2 5 0; 2 5 6 0; 2 1 0 0; 0 4 6 0; 1 2 5 0; 0 1 2 0; 6 4 0
0;
    3 4 6 0; 2 5 4 0; 1 0 4 0; 1 2 5 0; 2 5 6 0; 2 1 3 0];
prob = 0;

for i = 1 : 28,%change
    if( A(i,4) == 0 )
        k = A(i,1);
        l = A(i,2);
        m = A(i,3);
        A(i,4) = 1111;
        for j = 1 : 28,%change
            if( ( l == A(j,2) ) & ( ( k == A(j,1) ) | ( m == A(j,1) ) ) & ( ( k == A(j,3) ) | ( m ==
A(j,3) ) ) ),
                prob = prob + 1;
                A(j,4) = 1111;
            end
        end
        for p = 1 : 28,%change
            if( A(p,4) == 1111 )
                A(p,4) = prob;
            end
        end
        prob = 0;
    end
end
%disp(A);

for h = 1 : 28,%change
    A(h,4) = A(h,4)/85;%change
end
%disp(A);

B(1,1)=A(1,1);
B(1,2)=A(1,2);
B(1,3)=A(1,3);
```

```

B(1,4)=A(1,4);
cnt=1;
checkcnt=0;

for i = 2 : 28,%change
    k = A(i,1);
    l = A(i,2);
    m = A(i,3);
    for j = 1 : cnt,
        B(j,3) ) ),
        if( (l == B(j,2) ) & ( (k == B(j,1)) | (m == B(j,1)) ) & ( (k == B(j,3)) | (m ==
            checkcnt = checkcnt+1;
        end
    end
    if(checkcnt == 0)
        cnt = cnt + 1;
        B(cnt,1)=A(i,1);
        B(cnt,2)=A(i,2);
        B(cnt,3)=A(i,3);
        B(cnt,4)=A(i,4);
    end
    checkcnt=0;
end
disp(B);

cparr=[3 4 6 5 4 3]; % change as per needed

mulval = 1;

for g = 1 : 4,% Change size of cparr-2

    turn1=cparr(1,g);
    turn2=cparr(1,g+1);
    turn3=cparr(1,g+2);

    for f = 1 : 13,%size of B
        k = B(f,1);
        l = B(f,2);
        m = B(f,3);

        if( (l == turn2) & ( (k == turn1) | (m == turn1) ) & ( (k == turn3) | (m ==
turn3) ) ),
            val = B(f,4)
            mulval=mulval * val;
            break
        end
    end
end

```

```
    if (f==13)%change size of B
        val=0;
        mulval=mulval * val;
    end
end
end
disp('Final Pc Value for cycle in Cparr');
disp(mulval);
```

## REFERENCES

- 1 <http://now.cs.berkeley.edu/> [October 10, 2003].
- 2 M. J. Flynn, "Some Computer Organizations and Their Effectiveness," IEEE transactions on Computers, vol 21, no 9, September 1972, pp. 948 –960.
- 3 A. Varma, & C. S. Raghvendra, "Interconnection networks for multiprocessors and multicomputers theory and practice," IEEE Computer Society Press 1994.
- 4 J. Duato, S. Yalamanchili and L. Ni, "Interconnection Networks, an Engineering Approach", Revised Printing 2003 by Elsevier Science.
- 5 W. J. Dally & C. L. Seitz, "Deadlock free message routing in multiprocessor interconnection networks," IEEE transactions on computers, vol c-36, no 5, pp. 547-553, May 1987.
- 6 T. Cormen, C. Leiserson, and R. Rivest, Introduction to Algorithms, McGraw-Hill, 1990.
- 7 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, 15(1): pp. 29-36, February 1995.
- 8 M.D. Schroeder et al., "Autonet: A high-speed, self configuring local area network using point to point links" Technical Report SRC Research Report 59, DEC, April 1990.
- 9 D. Garcia and W. Watson, Server Net II, in Proceedings of the 2<sup>nd</sup> PCRCW, Lecture Notes in Computer Science 1417, pp. 119 – 135, Springer – Verlag, Berlin, 1997.
- 10 Tor Skeie, Ingebjørg Theiss and Olav Lysne "Evaluation of Minimal Deterministic Routing in Irregular Networks" In Proceedings of the SSGRR International Conference on Infrastructure for e-Business, e-education, e-Science, and e-Medicine (SSGRR) 2002.
- 11 J. S. Yang and C. T. King, "Designing Deadlock-Free Turn-Restricted Routing Algorithms for Irregular Wormhole-Routed Networks" In Journal of Information Science and Engineering 17, pp. 575-594 (2001).



- 12 L. Cherkasova, V. Kotov, and T. Rockicki, "Fibre channel fabrics: Evaluation and design," in 29<sup>th</sup> Hawaii international conference on system sciences, 1995.
- 13 D. Starobinski, M. Karpovsky, and L. Zakrevski, "Application of network calculus to general topologies using turn-prohibition" In IEEE/ACM Transactions on Networking Volume: 11, Issue: 3, pp. 411-421 June 2003.
- 14 Timothy Mark Pinkston and Sugath Warnakulasuriya "Characterization of Deadlocks in Irregular Networks" "Journal of parallel and Distributed Computing 62(1): pp. 61-84 (2002).
- 15 Anjan K.V and Timothy Mark Pinkston, "An Efficient, Fully Adaptive Deadlock Recovery Scheme: DISHA" In Proceedings of the 22<sup>nd</sup> International Symposium on Computer Architecture, pp. 201-210, June 1995.
- 16 V. Puente, J.A Gregorio, R. Beivide, F. Vallejo and A. Ibañez," A New Routing Mechanism for Networks with Irregular Topology" In Proceedings of the 2001 ACM/IEEE conference on Supercomputing pp. 31-31.
- 17 Akiya Jouraku, Michihiro Koibuchi, Hideharu Amano and Akira Funahashi," Routing Algorithms Based on 2D Turn Model for Irregular Networks" In 2002 International Symposium on Parallel Architectures, Algorithms and Networks (ISPAN'02) May 22-24, 2002.
- 18 Andrew A. Chien. "A Cost and Speed Model for k-ary n-cube Wormhole Routers" In Proceedings of the symposium on Hot Interconnects, IEEE Computer Society, August 1993.
- 19 Yong Ho Song and Timothy Mark Pinkston, "A New Mechanism for Congestion and Deadlock Resolution" In International Conference on Parallel Processing August 18-21, 2002 pp. 81.
- 20 F. Petrini and M. Vanneschi, "Performance Analysis of Minimal Adaptive Wormhole Routing with Time-Dependent Deadlock Recovery" In 11<sup>th</sup> International Parallel Processing Symposium (IPPS' 97) April 01 - 05, 1997 pp. 589.

- 21 [http://www.usc.edu/dept/ceng/pinkston/people/publications/sugath\\_thesis\\_ab.htm](http://www.usc.edu/dept/ceng/pinkston/people/publications/sugath_thesis_ab.htm) 1 [November 2, 2003].
- 22 <http://www.usc.edu/dept/ceng/pinkston/FlexSim/IRFlexSim0.5.pdf> [March 9, 2003].