Performance analysis of a fault tolerant multistage interconnection network with backpressure blocking mechanism

M. Amiri Zarandi [1], N. Hashemi [1, *], F. Safaei[2], S. Azizi [1]

1. Amirkabir University of Technology, Tehran, Iran
2. Sahid Beheshti University, Tehran, Iran

nhashemi@aut.ac.ir

Abstract: In recent years, multistage networks have been suggested as efficient tools in interconnection networks and have been studied by many researchers. In this direction many fault tolerant networks have been introduced and different studies have been performed in the field of evaluating performance of these networks. One of these networks is called Fault tolerant Double Tree (FDOT) networks which have proper efficiency in addition to the feature of cost-efficiency. Most studies performed on different kinds of multistage networks in the past have either been in the form of simulation or in the form of analytical modeling, but they have been performed in very special conditions. We have tried in this research to study the FDOT networks through analytical modeling. In this article a model has been provided for evaluating the efficiency of these networks that can evaluate the efficiency of different kinds of these networks having any size and buffer capacity under uniform traffic.

[M. Amiri Zarandi, N. Hashemi, F. Safaei, S. Azizi. **Performance analysis of a fault tolerant multistage interconnection network with backpressure blocking mechanism.** *J Am Sci* 2012;8(7):127-134]. (ISSN: 1545-1003). http://www.jofamericanscience.org. 20

Key words: multistage interconnection networks; irregular networks; perfarmance analysis; fault tolerance

1. Introduction

By given with developing technology and new systems with many processors and memories and also with developing the concept of parallel and distributed processing it is necessary to study and investigate mechanism for communicating and transferring information in these systems. In this direction, multistage networks have been provided as efficient mechanisms for transferring information in multiprocessor systems and also networks on chip during past years and many studies have been performed in this field.

Multistage networks are networks used for transferring pocket among some producer such as processors and some consumers such as memories. Their structure is such a way that the producers are situated in one side of the network as source and the consumers are situated in the other side of them as destination and these networks are connected to each other using switches which are situated in some levels and using links between these levels in such a way that it is possible for them to share memories among all processors. This does not need the direct connection between each processors and memories and in the result the number of links and sources used and therefore the cost of creating network will decrease seriously. Usually in these networks buffer is used in the exit or entry place of switches in order to decrease the level of missing pockets which can not enter to the considered port in the next level in a specific cycle.

Since the performance of these networks influences too much the efficiency of considered systems, the researchers and developers are always interested in introducing and studying structures and mechanisms which are able to improve efficiency of these networks and also cause to increase their reliabilities.

The multistage networks are placed in two general categories:

1- regular networks: in these networks, the number of switches of all levels is equal and in the result the number of switches from which the pocket must pass in order to move from source to destination is fix and is equal to every other path between any other pairs of sources and destinations.
2- Irregular networks: in these networks the number of existing switches in different levels and in the result the distance between different sources and destinations is variable.

One of the important problems in studying all above mentioned networks is the traffic of the network entry, meaning how is the rate of entering pockets to the network and the way of distributing availability of produced pockets by each source toward each destination. In uniform traffic each processor produces pocket with fix probability in different cycles and this probability is equal among different processors. On the other side it is hypothesized that the pocket produced by each one of processors is possible to be sent to any memories with equal probability. In this article we intend to provide an analytical model which can be used

for investigating the considered network under uniform traffic.

One of the other challenges existed in developing multistage networks is the creation of possibility of fault tolerance by the use of creating the possibility of using substituted paths in the network. So it is possible that in the case of creating problem in the network and failing of one or several links or switches the network will be able to continue its work with acceptable efficiency and performance.

In general modeling and evaluating different systems are performed by the way of simulating and analytical modeling. The selection between these two ways depends on the system investigated and the way of our treating. In the first way which is simulation by the computer tools we can gain sometimes more exact results. Also the time spent at the first for preparing a model is less, but if we want to test more it is necessary to spend much time for preparing and studying models in different cases. In analytical modeling more time is spent for preparing the first model but then the results of investigations and tests will be produced quickly. Meantime if the proposed analytical model is complete and reliable it will be possible to predict the influences of different factors and parameters on each other.

In this study we will try to investigate analytically one kind of multistage networks having the ability of fault tolerance called Fault tolerance Double Tree (FDOT) networks. As we will see, by investigating this network and computing the different amounts of efficiency metrics it will be indicated that besides these networks can have fault tolerance and proper efficiency in the presence of forms they can have proper efficiency in comparison with other similar networks.

2. Related works

In the past years different kinds of multistage networks have been introduced and have been evaluated by different methods. One of these networks is Banyan networks. These networks have been considered and investigated by researchers and authors due to special and remarkable features. Jeng et al. (1983) have studied the efficiency of these networks in the form of single-buffer and have analyzed them. Mun (2005) and Youn (1990) and also Tutsch and Hommel (2002) have used analytical evaluation method of studying multistage networks while Zheng et al. (2005) and Tutsch and Brenner (2003) have selected the method of simulation.

Most studies in this field such as most works introduced have been performed on networks with uniform traffic but at the same time nonuniform traffic has been studied by some researchers for example Mun (2005) has analyzed multi buffer Banyan networks under uniform traffic.

One of the important hypotheses in investigating and studying buffered multi-level networks is the treatment of network while pocket wants to enter the buffer but the related buffer can not receive it for any reason such as being full. In some studies it is hypothesized that the pocket placed in these conditions can not be entered in the buffer of this level and can not be lost such as (jenq, 1983) in which the study has been performed on this hypothesis that there is possibility of losing pockets inside the network. But some others believe that in these situations we must have blocking, meaning that the considered pocket in the previous buffer in the previous level is waiting to be able to enter its considered buffer in the next level such as (Garofalakis, 2011) . They study the model with this hypothesis that in the case a pocket enters the network it will leave the network surely and the mentioned stop in any level prevents sending the pockets in the previous levels in the form of backpressure pressure.

But in the field of fault tolerance multistage networks there are different networks that have been investigated. Choi et al. (2003) have studied one analytical model for investigating the network with considered switch structure by providing one analytical model. Sharma et al. (2008) introduce a network which has fault tolerance and cost efficiency.

DOT networks have been provided for the first time by Levitt as a kind of irregular networks. After that this network has been used by many researchers and authors. Despite many benefits of this network it was not fault tolerant network until Bansal et al. (1991) designed and provided FDOT on the basis of that network which is fault tolerant. They used the method of simulation for investigating the efficiency of their work.

FDOT networks have been studied too much so far and before that in some cases it has been tried to provide some analytical models for them but these models have been studied under specific conditions such as single buffer or network size or with pocket lost.

In this article we try to provide an analytical model for investigating these kinds of networks. This model provides the possibility of investigating performance of these networks with different buffer length. On the other hand, it is not necessary to losting packets in the network bacause we have a backpressure bloching mechanism.

In the continuation of this article and in section 3 the structure of these networks is introduced and in section 4 an algorithm is introduced through which the process of routing of pocket is done. In section 5 the primary elements of the model are introduced and in section 6 the analytical model will be introduced. In section 7 the fault tolerance level of the network will be investigated and in section 8 the results have been provided and analyzed. Finally and in section 9 the conclusions have been discussed.

3. Structure of FDOT Networks

This network is designed on the basis of the DOT networks but the aim is to add some sub networks and to use proper path seeking methods inside them so that we can create some paths between sources and destinations. In the case of necessity and when there is a fault in the pockets it is possible to use the substituted paths for getting the destination.

One FDOT-K network is a network used for transferring pockets between N processors which have been placed as source before network and N memories which have been placed at the bottom of the network as destination. The size of this network is indicated by N*N. In designing this network we place the sources in K categories (K is higher than 2). So first N/K entries are placed in the first category by this method until the last N/K entry are placed in the K category. This will be continued for destinations too. Determining the size of K depends on our aim in designing network and we can determine and select that based on this aim. In this network we will have K+1 sub networks that have similar structure completely.

The path used for transferring pockets between any pairs of sources and destinations can be placed in any one of sub networks. At the beginning of each sub network there are N/K number MUX which connect entries to the network so that through which each entry is connected to all sub networks. Also at the bottom of sub networks there is this same number DEMUX which connect the destinations to networks. Each sub network has some switches which have been placed in the form of two binary trees in two sides of one switch which is the common root of every two trees.

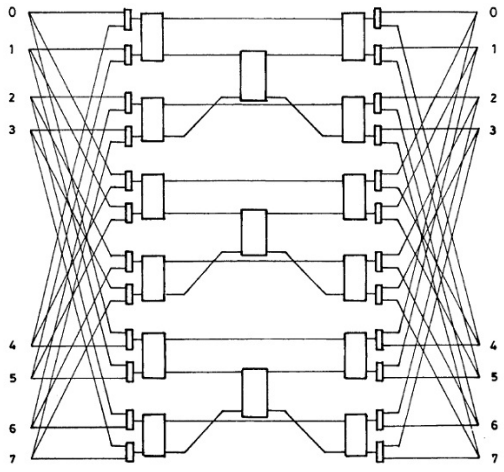

Figure 1. the structure of FDOT networks

The leaves of one of these trees are switches existed in the first level of the network which are connected to the MUXs and the leaves of other tree are switches of the last level which are connected to the DMUXs.

Since every left and right trees have a depth equal to lg(n/k) the number of levels of each sub network which is equal to the number of the levels of the whole network and it is indicated by L It is equal to (2*lg(n/k))-1 (two trees have common root). The number of switches of each sub network is also equal to (2n/k)-3 and then the total number of switches of network will be equal to (k+1)*(2n/k-3)

The network that we study has switches 2*2 and one buffer has been placed in each switch output. For determining inputs and outputs we indicate the position of each one using variable i that is the number of the level on which the switch is placed in addition to j which is the number of switch in considered level.

Each one of outputs of any switch except the switches of the last level are connected to one input of existed switches in the other level which will not be in the next level necessarily.

4. Routing Algorithm in FDOT Networks

The length of the path means the number of switches through which a pocket must pass in order to get the destination. Contrary to regular networks, in irregular networks the length of the path is not fixed. In the FDOT networks it is also possible the different paths with different lengths are existed between any pairs of source and destination. Two main problems which must be considered in this network are the way of determining the existed paths between any pair of source and destination and also the length of each one of them. In (Bansal, 1991) some algorithms have been mentioned for determining them. Here we introduce two more simple and shorter algorithms on the same basis for determining length and finding the considered path.

In order to investigate sources and destinations we number them from 0 to n-1 and represent the address of them by a string of bits (for example the source number 9 is indicated by s=1001). In this string the bit having the least value is called bit number 0 so on.

In order to understand the number of paths existed between one pair of source and destination in the network and also the length of any path we use the formula as follows :

$$\begin{cases} L & \text{if } s_{L-\frac{1}{2}} \neq d_{L-\frac{1}{2}} \\ 2R & \text{if } O.W \end{cases}$$

In this relation L is the number of levels and R is a variable which is computed by the algorithm as follows.

R = 0

for t =0 to (L-1)/2

if $s_t \oplus d_t = 1$

R = t
end
end
R = R+1

The operator $\oplus$ performs the operation "exclusive or". After computing P the length of possible paths is equal to P, P+2, P+4, … , L.

For example, if we want to compute the number and the length of possible paths between the source 0101 and the destination 0111 in one network FDOT-2 with the size 16*16 we will have R=2. Therefore there are two paths with lengths 4 and 5 between this source and destination.

After finding the length of paths we use one routing algorithm in order to determine the path between source and destination. So we will find routing tag related to each path. Routing tag is a string of bits used for routing a pocket in the network from source to destination. If the related bit was 0 the pocket will move toward the output above otherwise, it will move toward the output below.

In the continuation of this article, one algorithm will be introduced by which the routing tag of each path above will be determined. First we define one function called add-bit whose performance is such a way that adds a specific bit to the bottom of one string of bits for example add-bit (000,1) = 0001. Now we write the algorithm as follows by this function. This algorithm is used for any specified paths in the previous relation.

Tag = add-bit(' ', $s_{L+1/2}$)
For t =1 to $\left[\frac{P-1}{2}\right]$
Tag = add-bit(tag,1)
End
If $P \neq L$
Tag = add-bit(tag,0)
End
For t = $\left[\frac{P-1}{2}\right]$ to 0
Tag = add-bit(tag,d_t)
End
Tag = add-bit(tag,$d_{L+1/2}$)

Using this algorithm we will have two paths 0111110 and 010110.

5. Preliminaries of Model

5.1 Primary Definitions

In order to study the efficiency of network it is necessary to investigate the buffers of the network and the parameters such as the rate of entry and exit of the pockets and also their performance in confronting with blocking during the time. So we must pay attention to the case of each buffer in a specific time of starting the system. For example, how much is the possibility of this issue that the first buffer of the first level being empty after passing t seconds from the start of the system? Since the total buffers compose one network of queue which are ergodic we know that these buffers will remain in the steady state after passing a time from starting the system, meaning their case will not depend on the case of system in the first time. Thus we can investigate the considered case independent of the time of system. Regarding the problem above in a network FDOT-K having switches 2*2 and a buffer with the length z we will have the parameters as follows:

$P_n(i,j)$ = the possibility of stability case of this issue that buffer j th from the level i th of the network has n

$$\begin{pmatrix} E_0 & E_0.(1-B) & 0 & \cdots \\ E_1 & (E_0.B + E_1.(1-B)) & E_0.(1-B) & \cdots \\ \vdots & \vdots & \vdots & \end{pmatrix}$$

pockets.

$P_e(i,j)$ = the possibility of stability case of this issue that buffer j th from the level i th of the network is empty.

$P_u(i,j)$ = the possibility of stability case of this issue that buffer j th from the level i th of the network has at least one pocket.

B(i,j) = the possibility of stability case of this issue that buffer j th from the level i th of the network will remain in the case of blocking.

f1,f2 = output buffers of switches into which the output pockets from the current buffer are entered (the pocket enter them after leaving the current buffer).

q1,q2 = the buffers from which the output pockets enter the input of the switch into which the current buffer is connected (the pocket entered to this buffer has gone out of one of them).

S = the buffer into which the other input has been connected and this input in turn has been connected to the current buffer.

$E_x(i,j)$ = random variable that indicates the number of pockets which enter the buffer j th from the level i th of the network at the bottom of one cycle.

U_{ij} = the possibility of the issue that the pocket getting input j th from the level i th of the network wants to enter the deployed buffer in the output above the related switch.

5.2 hypotheses

1- Network has been composed of 2*2 switches with finite buffers in the output.

2- N processor elements have been connected to N memory elements through the network.

3- The length of the buffer is desirable but finite and the traffic of the network is uniform.
4- Each pocket is transferred from the buffer of one level to the other level during the time of one cycle.
5-All input and output operations are performed at the bottom of each cycle.
6- There is no buffer before the first level and the pockets which can not enter the network will be lost.
7- The memories are enough for receiving the pockets left the network. So there is no blocking in the last level.
8- Interference between the pockets sent to the similar output is solved in a cycle respectively and randomly.
9- If a pocket wants to enter a buffer which is full that pocket will remain in the previous buffer.

6. Analyzing The Model

6.1 Relations

In the network discussed we deal with a network of buffers whose feature is as Be/ G/ 1/ c/ FIFO in such a way that these networks are single channel queues and their service time is general. The order of queue has also the pattern FIFO, meaning the pockets entered the queue sooner will leave it sooner. The capacity of system is somehow finite that can be 0. It is obvious that the analysis of queues with unlimited capacity which can not be implemented physically has this feature that it is not necessary to consider the possibility of blocking one buffer. It means we will have for all buffers: B(ij)=0.

In order to investigate the variables related to entry and blocking we use markov chain. We can analyze the treatment of a specific buffer independent of its performance in the network using that and then we will analyze the whole network using the related information to the single buffers. The reason of establishing the conditions of markov in the studied buffers is that the number of pockets existed in one buffer in the next cycle depends only on buffer state in current cycle and there is no need to have information about the previous cycle.

Since the number of pockets and also the discussed time has discrete identity our model will be a discreet parameter markov chain. The states of markov chain discussed are the different numbers of n, meaning the number of existed pockets in the buffer.

In order to compute the transfer matrix parameters of markov chain we consider the buffer j th of the level i th of the network so:

$$E_1 = 2*u*d*p_u(q1)*p_u(q2)+u*[p_u(q1)*p_e(q2)+p_e(q1)*p_u(q2)]$$
$$E_2 = u*p_u(q1)*u*p_u(q2)$$
$$E_0 = 1-E_1-E_2$$

For example $p_u(q1)$ is the possibility of stability case of not being empty of the buffer that is connected to the input above the same switch that our current buffer is placed in one of its output.. Also for computing b for this buffer we have:

$$B = u*[p_{(z)}(f1)+u/2*p_u(s)*p_{(z-1)}(f1)]$$
$$+d*[p_{(z)}(f2)+d/2*p_u(s)*p_{(z-1)}(f2)]$$

For example $p_z(f1)$ is the possibility of stability case of filling of buffer that is placed in the output above the switch into which the pockets are entered. Transfer matrix of the chain above which is indicated by P is as follows:

Because the buffer is ergodic we have: $\begin{cases} \pi = \pi P \\ \sum_j \pi_j = 1 \end{cases}$

or : $\begin{cases} \pi Q = 0 \\ \sum_j \pi_j = 1 \end{cases}$

In this relation P is the transfer matrix of markov chain and π is the state vector and Q is defined as P-I and I is unit matrix. So by solving the linear equations system above we can compute the cases of markov chain related to the considered buffer. We must perform this for all buffers of the network so that we can evaluate the efficiency of the network. Thus we must solve the linear equations system for computing the efficiency and analyzing the network above. Of course the main problem in our model is that after solving this system of equations, the amounts related to the buffer of any level depend on the related amounts of buffers of the previous levels. It means solving a returned equation and we need the amounts of the previous and next step in order to gain the amount of variable in the current step. The reason is that in order to compute the amounts of mentioned equations for buffer j th from the level i th we need to compute E_x. To compute this variable it is needed to have $p_u(q1)$ and other similar amounts. They are attained by the previous level. In addition we must have the amount of B(i,j). To compute this variable we must have $p_z(f1)$ and the similar amounts. They are attained from the next level. So practically we need a returned two-step equation which needs solving itterative method.

Itterative method

In this method for the needed amounts which are used for solving equations we need some parameters.

Then by this information we will solve the equations related to the network buffers respectively for all levels. In this case we will first investigate the first buffer from the first level and then we solve the second buffer from this level so on. After that we will move towards the second level. This work will be continued for all levels so that we can get the new amounts for the parameters related to each buffer. Then we repeat this work on the basis of these new amounts and this work is repeated until the obtained amounts have a difference in two successive repeats. This difference is small enough and provides the considered precision.

In order to compute the considered amounts through this method in the first land last level we need some boundary amounts. So for the last level we put the amount of $p_z(f1)$, $p_z(f2)$ equal to 0 (we do not have blocking in the last level). For the first level $p_u(q1)$, $p_u(q2)$ is equal to 1-p . p is the rate of producing the considered processor in any cycle.

6.2 Permormance metrics

For evaluating the efficiency of different kinds of considered networks there are important and influential parameters which will be discussed in different studies. The most important parameters are throughput and Delay in the network. Throughput of the network indicates the number of the pockets which leave the network in a specific time and reach the destination. In order to compare better we prefer to use normalized throughput which is the result of dividing throughput of the network by the number of network output. In this study we present this variable by T. On the basis of the above mentioned materials we can compute this parameter on the basis of variable p_u in such a way that because we do not have blocking in the last level the possibility of not being empty of the last level and at the result leaving of the pockets from them in the considered time will be equal to throughput of the network. So we have:

$$T = \frac{p_u(L)}{N}$$

The other parameter which will be discussed here will be normalized Delay. It will be indicated by D and is defined as the average time of passing a pocket from the network divided by the number of network levels. As we saw in (Garofalakis, 2010) this parameter can be computed on the basis of possibility of being empty of the buffer in different buffers as follows:

$$D = \frac{\sum_{i=1}^{L} p_u(i)}{p_u(L) * L}$$

7. Fault Tolerance

One K- FDOT network has K+1 independent sub network and there is a path in every sub network for any pair of source and destination. Adding this sub network causes increasing the ability of fault tolerance in the network and this network can have proper efficiency even in the presence of faults. A network is called fault tolerant x when a network can continue its activity in the presence of x faults. On the basis of this one FDOT-K network is a K-fault tolerant network because in the case of expressing at most K faults in the network there is yet a path in the network between any pair of source and destination. But if the number of faults is more than this number the performance of the network depends on the place of occurring faults. With regard to the possibility of this issue that the faults are distributed in all sub networks so that there are faults in all possible path between a pair of source and destination we can say that these reliable networks can continue their work with proper efficiency in the case of creating faults.

8. Result Analysis

After completing the presented model it is necessary to draw and analyze the graphs related to the efficiency evaluation parameters. These parameters are parameters that have been introduced before, meaning throughput of the network and Delay of the pockets. Then we investigate the graphs and compare them with other works and then we will investigate the correctness of this method that has been analyzed before.

The Throughput of single Buffer Networks:

At first we assume that we face with a network having buffers with capacity 1. As we said before the provided model can be used for network with any length of buffer. In the primary case that the buffer has the length 1 we gain the Troughput of the network having uniform traffic and represent it in the graph 2.

As it is indicated in this figure this network has been investigated with different sizes. With studying the results that have been determined in the graph and with comparing them to the results of (Kumar, 2007) we can find that the results of two studies are completely similar and confirm each other. As we predicated before in the networks having small size the amount of output parameter is higher especially in the higher rate.

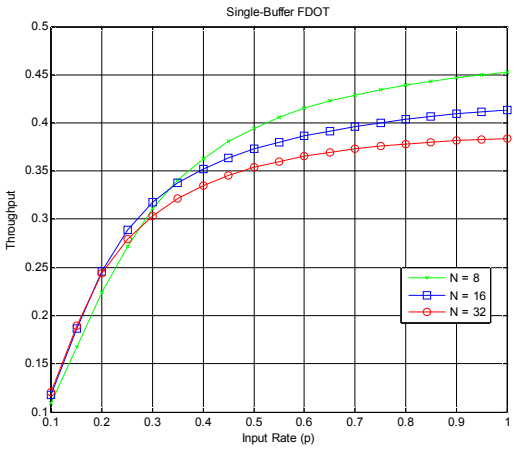

Figure 2. the Troughput of dsingle buffer networks

The Throughput of Double Buffer Networks:

Figure 3 indicates the graph related to the throughput of the networks that have buffer with capacity 2.

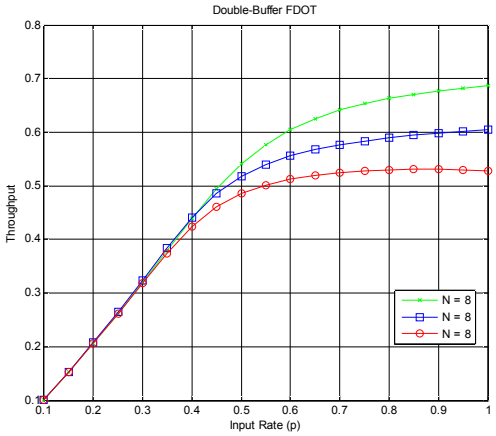

Figure 3. the throughput of double buffer networks

As we see in this figure in the networks having small size the amount of throughput of the network is higher. Of course here when the rate of entrance is less than 0.4 there is not high difference between graphs but when the amount of the entrance become higher this difference will become more too. The other important point is that in general in this graph the size of throughput of the network has been increased in proportion with previous graph. The reason is that with increasing the size of buffers the possibility of interference between pockets which causes blocking will decrease too. So the network will have higher throughput.

The Delay of Single Buffer Networks:

The other parameter which must be evaluated here is the Delay of network pockets that indicates the number of cycles of passing and leaving the pockets through the network. In the graphs below the normalized amount of this parameter has not been used and the average amount of Delay has been studied. By comparing these graphs with (Kumar, 2007) we can understand the correctness of obtained results.

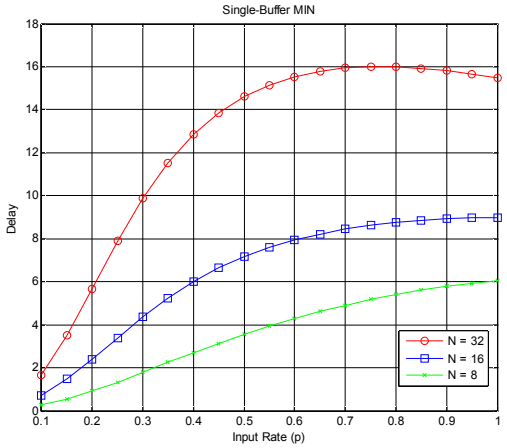

Figure4. the Delay of single buffered networks

As we can see in this figure when the size of the network increases the amount of Delay of pockets in the network will increase too seriously. One of the reasons of this issue is that contrary to the regular multistage networks, with doubling of the number of inputs the number of passed cycles for getting the destination will increase too. But it must be mentioned that since all pockets are not obliged to pass through all levels this increase in Delay is not specific and fixed, meaning even in very low entrance rate this amount is less than the amount of increasing of Delay due to the increasing of the number of levels. Higher is the entrance rate of the pockets higher is this difference.

The Delay of Double Buffer Networks:

Figure 5 indicates the Delay of the pockets which pass through the network while the capacity of the buffers of the network is 2. These networks have different sizes.

We can see again that with increasing the number of the network levels there is not serious change in the average amount of Delay of the pockets at first, but little by little and with increasing the rate of entrance of the pockets to the networks we can observe much increase in the amount of this parameter.

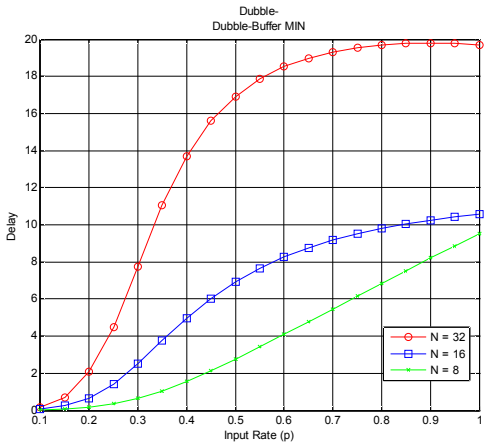

Figure 5. the Delay of double buffer networks

9. Conclusions:

In this study we tried to provide a new analytical method in order to evaluate one of the irregular multistage networks. The provided method is proper for analyzing different networks of FDOT and it can investigate this kind of networks in different states. On the basis of the obtained results and with comparing them we could indicate that the provided model represents the proper and exact responses. Meantime this model never have the limitations of the other studies such as the limitations of buffer size or losting pockets and can evaluate the studied networks in different states. In any case after analyzing FDOT networks we can find that these kinds of networks with low cost can play a suitable role in interconnections networks due to having the possibility of fault tolerance.

Correspondence to:
N. Hashemi
Amirkabir University of Technology, Tehran, Iran
Email : nhashemi@aut.ac.ir

References

[1] Bansal, P.K., et al. Fault tolerant double tree multistage interconnection network. in INFOCOM '91. Proceedings. Tenth Annual Joint Conference of the IEEE Computer and Communications Societies. Networking in the 90s., IEEE. 1991.

[2] Garofalakis, J. and E. Stergiou, An approximate analytical performance model for multistage interconnection networks with backpressure blocking mechanism. Journal of Communications, 2010. 5(3): p. 247-261.

[3] Garofalakis, J. and E. Stergiou, Analytical model for performance-evaluation of multistage interconnection networks supporting multi-class priority service. International Journal of Parallel, Emergent and Distributed Systems, 2011. 26(5): p. 399-427.

[4] Jenq, Y.C., Performance analysis of a packet switch based on single-buffered banyan network. 1983, IEEE. p. 1014-1021.

[5] Kumar, S. Mathematical Modelling and Simulation of a Buffered Fault Tolerant Double Tree Network. 2007: IEEE.

[6] Minsu, C., N. Park, and F. Lombardi, Modeling and analysis of fault tolerant multistage interconnection networks. Instrumentation and Measurement, IEEE Transactions on, 2003. 52(5): p. 1509-1519.

[7] Mun, Y., Performance Analysis of Banyan-Type Multistage Interconnection Networks Under Nonuniform Traffic Pattern. 2005, Springer. p. 33-52.

[8] Sharma, S., P.K.Bansal, and K. Kahlon, On a class of multistage interconnection network in parallel processing. International Journal of Computer Science and Network Secrity, 2008. 8(5): p. 287-291.

[9] Tutsch, D. and G. Hommel, Generating Systems of Equations for Performance Evaluation of Multistage Interconnection Networks. Journal of Parallel and Distributed Computing, 2002. 62(2): p. 228-240.

[10] Tutsch, D. and M. Brenner, MIN simulate. A multistage interconnection network simulator. 2003, Citeseer.

[11] Yoon, H., K.Y. Lee, and M.T. Liu, Performance analysis of multibuffered packet-switching networks in multiprocessor systems. Computers, IEEE Transactions on, 1990. 39(3): p. 319-327.

[12] Zheng, G., et al., Simulation-based performance prediction for large parallel machines. 2005, Springer. p. 183-207.

12/21/2011