Introduction and Analysis of Optimal Routing Algorithm in Benes Networks

Article *in* Procedia Computer Science · December 2014

DOI: 10.1016/j.procs.2014.11.068

5 authors, including:

Abbas Karimi
Islamic Azad University of Arak

Kiarash Aghakhani
Islamic Azad University of Arak

Faraneh Zarafshan
Islamic Azad University of Ashtian

Syed Abdul Rahman Al-Haddad
Universiti Putra Malaysia

Available online at www.sciencedirect.com
ScienceDirect
Procedia Computer Science 42 (2014) 313 – 319
Procedia Computer Science

International Conference on Robot PRIDE 2013-2014 - Medical and Rehabilitation Robotics and Instrumentation, ConfPRIDE 2013-2014

Introduction and Analysis of Optimal Routing Algorithm in Benes Networks

Abbas Karimi[a,*], Kiarash Aghakhani[b], Seyed Ehsan Manavi[a], Faraneh Zarafshan[d], S.A.R Al-Haddad[c]

[a]*Department of Computer Engineering, Faculty of Engineering, Arak Branch, Islamic Azad University, Arak, Iran*
[b]*Young Researchers and Elite Club, Arak Branch, Islamic Azad University, Arak, Iran*
[c]*Department of Computer and Communication Systems Engineering, Faculty of Engineering, UPM, Malayasia*
[d]*Department of Computer Engineering, Faculty of Engineering, Hamedan Branch, Islamic Azad University, Hamedan, Iran*

Abstract

In a total grouping of dynamic interconnection networks, safe and quick routing is so important. The Benes network is one of the dynamic interconnection networks that are good for telephone networks, multi-processor systems, parallel computers, ATM switches and Navigation and radio communication between robots. In this paper, two models of Benes routing algorithm is introduced then compare them with looping; Hassan-José and fast algorithms in the speed of running time and implementation then introduce the optimum algorithm.

Peer-review under responsibility of the Center for Humanoid Robots and Bio-Sensing (HuRoBs)

Keywords: Benes Network, Interconnection Networks, Matrix-based Algorithm, Robotics, Routing Algorithm

1. Introduction

Multi-stage interconnection networks are as one of the fastest communication channels that are using nowadays and they are so useful in designing the parallel computers. .indeed in these networks the connection between the source and destination nodes is formed of multiple switching layers[1,15]. the way the layers connect to each other provide different kinds of networks. In one grouping the multi-stage interconnection networks device to two group of static and dynamic group. The dynamic ones are based on switching's that are applied in run-time to make the route between the source and destination nodes[2,16]. Dynamic multi-stage interconnection networks device into 4 groups: Blocking Network, non-blocking, rearrange able non-blocking and strictly non-blocking networks[3]. Benes

* Corresponding author. Tel.: :(+98)916-3113863, (+60)17-3051064.
E-mail address: Akarimi@iau-arak.ac.ir

Peer-review under responsibility of the Center for Humanoid Robots and Bio-Sensing (HuRoBs)
doi:10.1016/j.procs.2014.11.068

network is a kind of dynamic non-blocking interconnection network after resorting. In this network we won't face the obstruction or if we face it we can close the switches to resolve this problem[12].

A Benes network with N input and N output is shown Benes (N×N) which is made of a 2×2 switch. This network has 2logN-1 switch layer and each layer has N/2 switch. Figure 1 illustrates a Benes network (8×8).

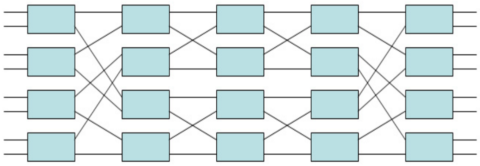

Fig. 1. A Benes network (8×8).

In Benes Network the most important issue is quick and safe routing. Determination of the state of switches to transform the data from input to output is the goal of routing in this network[9,13].

Various routing algorithm for Benes network are offered like looping algorithm that introduced by tsao-wu and opferman in 1971, that starts from outer layer and moves to inner layer determines the way from input to output, however it might needs to return and change switches[4]. Fast method is based on analyze of permutation in every step and determine the state of switches[5]. Hassan-José method is based on bits and determines the state of every switches by use of coloring graph[6,8].

In 2009 chakrabarty and his colleagues introduced the matrix-based algorithm[7] that for each layer, there is one matrix and with simple algorithms determine the state of LogN-1 switches of the first layer and the other layers control with self-routing algorithm, but the problem of this algorithm is the back warding that make it less effective[11,14] .

2. Division routing algorithm

This algorithm determines the state of switches of two layers and in the end just the central layer remains.
The process is in 4 steps:

Step 1: form the table of number of switches that it exists in rows and columns.
Put two port switch numbers to create corresponding binary numbers {0, 1, 2, 3} and to set table and put them in the table, which called signed cells.

Step 2: upper port is 0 and lower port is 1. You can extract rows from input and columns from output.

Step 3: scale the table from the first row, the first signed cell named A and the cell with the same row or column named B. because one switch equal to one row or column, and we cannot link two switches to one half-network.

Step 4: in this stage, define the state of input and output layer, for example if it is 0 in a cell then the first port of input and output switch in the first and last layer will link to the half network that is named something.

Step5: continue all these stages except the center stage.

Example 1:

The permutation that is shown below will be routed with the help of routing algorithm.

$$PI=\begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 3 & 0 & 7 & 1 & 4 & 6 & 2 & 5 \end{pmatrix}$$

The table has 4 columns and 4 rows for the Benes network (8×8). (Table.1)

Since Permutation is $\begin{pmatrix} 0 \\ 3 \end{pmatrix}$, 0 input (upper port of 0 switch) will reach the 3 output (lower port of 1 switch).
Put the binary number in the conflict place of [0, 1]. Continue this process to complete the table.

Table1. For the first layer

	0	1	2	3
0	2A	1B		
1	3B			1A
2			0A	2B
3		0A	3B	

According to Table 1, Figure 2 Shows The state of the first and the last layers is determined then table the signed elements, determine the connected table to the A and B networks.

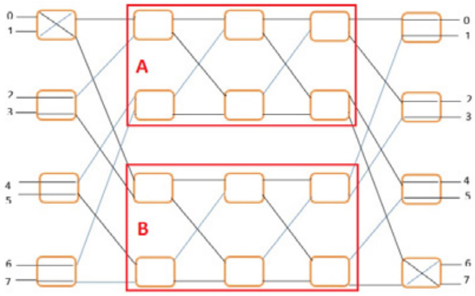

Fig. 2. Status of the first and the last layers is determined

Set the table for the second and forth layers. (Table.2)

Table2. For the first layer, A block and B block

	0	1		0	1
0	0A	3B	0	1B2A	
1	3B	0A	1		1A2B
	Block A			Block B	

According to Figure 3 Shows the Status of the second and forth layers is determined.

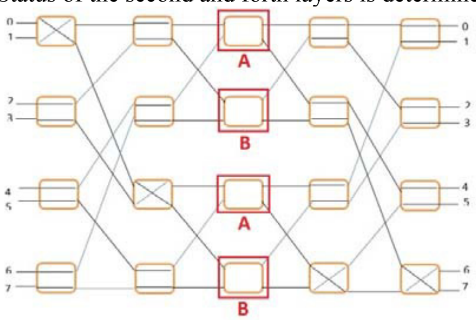

Fig. 3. Status of the second and forth layers is determined

And finally the state of switches is as shown below: (Figure 4)

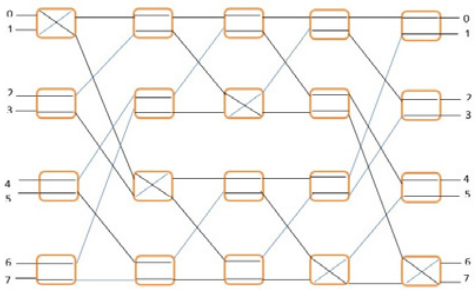

Fig. 4. Status of network switches

3. Input and output routing algorithm

This algorithm has 3 stages:

1) Select the input and output corresponding pairs.

2) Label 0 for the output corresponding pair and label 1 for input corresponding pair.
3) Label 0 to the upper port and 1 to the lower port. Assume the Permutation is

$$PF=\begin{pmatrix} i1 & i2 & i3 & i4 & i5 & i6 & i7 & i8 \\ o1 & o2 & o3 & o4 & o5 & o6 & o7 & o8 \end{pmatrix}$$

Consider the first member o1 ,i1 find the corresponding pair with the output o1from the output switch, that assume to be o3, so the o3,i3 are extracted then correspond to the input from the i3 switch, for example i6,then i6,o6 are selected in this stage.

Note: if during the selection of corresponding pair, reach to repetitive value then maybe the work is done or all values are not scaled and we should reach the first value that we start from it before.

Step 2: in this stage in scaling Permutation for the output corresponding pair label 0 and label 1 for the input corresponding pair.

Step 3: in this stage we select the output corresponding pair and link it to upper connection in switch and the input corresponding pair which is label 1 link to lower connection.

Note: each Permutation that has input and output determines base on the type of 0 label (upper) and 1 label (lower). Continue the upper stages for inner layers.

Example 2: The presented algorithm will be verified with an eight permutation. The permutation that is shown below will be routed with the help of routing algorithm.

$$PF=\begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 3 & 0 & 7 & 1 & 4 & 6 & 2 & 5 \end{pmatrix}$$

Step one: select the spiral permutation and label them:

Select the first permutation $\begin{pmatrix} 0 \\ 3 \end{pmatrix}$,now consider the output port(3),find the port 3 in 8 channel Benes network (Figure 1),the corresponding pair is port 2,select the permutation that 2 is output, $\begin{pmatrix} 6 \\ 2 \end{pmatrix}$,while select the input permutation, repeat this operation, consider 6,then find the port 6 in input 8 channel Benes network, the corresponding pair is 7,select the permutation that the input is 7, $\begin{pmatrix} 7 \\ 5 \end{pmatrix}$.

Continue the above process till all elements order as below:

$$\begin{pmatrix} 0 \\ 3 \end{pmatrix} -> \begin{pmatrix} 6 \\ 2 \end{pmatrix} -> \begin{pmatrix} 7 \\ 5 \end{pmatrix} -> \begin{pmatrix} 4 \\ 4 \end{pmatrix} -> \begin{pmatrix} 5 \\ 6 \end{pmatrix} -> \begin{pmatrix} 2 \\ 7 \end{pmatrix} -> \begin{pmatrix} 3 \\ 1 \end{pmatrix} -> \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

Now label the elements by 0 and 1.

So these elements label by 0,

$$\begin{pmatrix} 3 \\ 1 \end{pmatrix} , \begin{pmatrix} 5 \\ 6 \end{pmatrix} , \begin{pmatrix} 7 \\ 5 \end{pmatrix} , \begin{pmatrix} 0 \\ 3 \end{pmatrix}$$

And these elements label by 1,

$$\begin{pmatrix} 1 \\ 0 \end{pmatrix} , \begin{pmatrix} 2 \\ 7 \end{pmatrix} , \begin{pmatrix} 4 \\ 4 \end{pmatrix} , \begin{pmatrix} 6 \\ 2 \end{pmatrix}$$

2-determine the state of 2 layer switches correspond and repeat the operation in backward mode till all switches are determined.

According to Figure 5 Shows the state of the first and last switches defines in this way that connects the input and output labeled 0 to the upper switches.

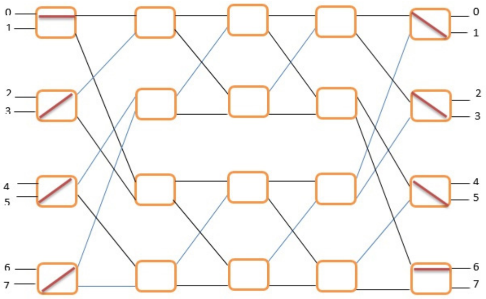

Fig. 5. The way input and output ports are connected to each other.

And the same happens, connect the input and output labeled 1 to the upper switches.
Also The way the first and last layers are set. (Figure 6)

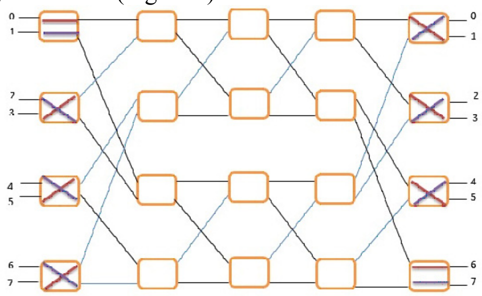

Fig. 6. The way the first and last layers are set.

Pass the input and output port numbers from the connected switches to determine the state of next layers. (Figure 7)

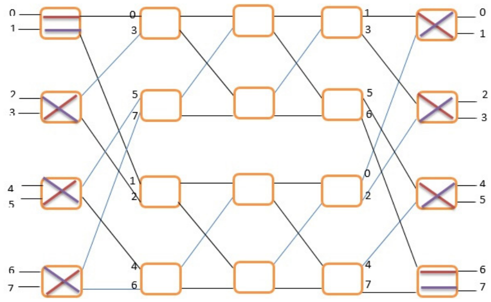

Fig. 7. Transmit addresses through the set switches in the first and last layers.

Now it is the time to do the same process for the second and the last but one layer.

Select the first element of second layer$\begin{pmatrix}0\\3\end{pmatrix}$, now consider the output port 3, in output find the 3 (the last but one layer) the corresponding pair is 1,$\begin{pmatrix}3\\1\end{pmatrix}$. Now consider the input port 3 and the corresponding pair in second layer is 0, the corresponding element is $\begin{pmatrix}0\\3\end{pmatrix}$,that selected before, now gauge the permutation of second layer till reach the first unvisited element $\begin{pmatrix}5\\6\end{pmatrix}$.repeat the process till the elements order as below:

$\begin{pmatrix}0\\3\end{pmatrix}\text{->}\begin{pmatrix}3\\1\end{pmatrix}\text{->}\begin{pmatrix}5\\6\end{pmatrix}\text{->}\begin{pmatrix}7\\5\end{pmatrix}\text{->}\begin{pmatrix}1\\0\end{pmatrix}\text{->}\begin{pmatrix}6\\2\end{pmatrix}\text{->}\begin{pmatrix}4\\4\end{pmatrix}\text{->}\begin{pmatrix}2\\7\end{pmatrix}$

The same as previews step label every other element by 0 and 1, and determine the state of second and last but one element as Figure 8.

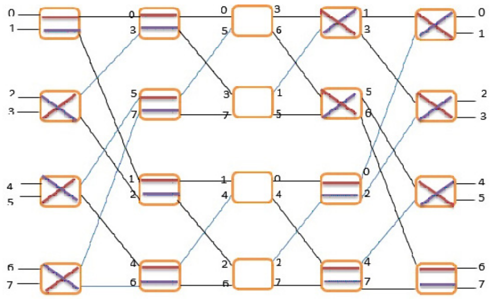

Fig. 8. The way the second and the one to last switches are set.

The state of last switch is as shown in Figure 9.

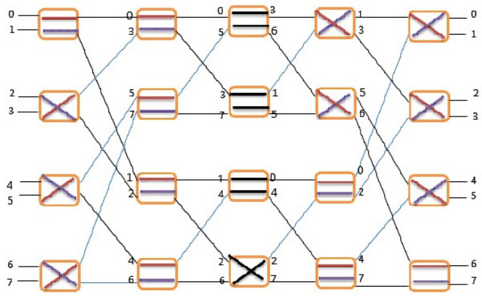

Fig. 9. Define the state of all switches and complete the routing.

4. Analyze of algorithms

Comparing the algorithm with the 3 other known algorithms by the means of implement and the run time and the result of survey has shown in Fig. 10. the horizontal axis shows the different scales of the network, that for the network with the scale of 4 the Hassan-José run time is the most and others are almost the same with each other, however for the network with 8 and 16 scale the run time for input and output is less than the others. The implement of looping algorithm is complicated in real systems because a record of travelled states should be kept, That not only make overlap and the complicity of circuit, but also looping will utilize the serial method to show the state of switches, but the introduced algorithms do not have this complication since in each layer the routing tags and the state of switches is determined and Use the parallel method.

In division algorithm there is no need to backtrack and also it determine the state of 2 layers which optimize the time and storage.

According to Table 3 Shows the compare Two models of Benes routing algorithm is introduced with looping, Hassan-José and fast algorithms in the speed of running time.

Table3. Compared algorithms in terms of runtime

N=16	N=8	N=4	Size / Algorithm
0/0000779	0/0000239	0/0000057	Division
0/0000839	0/0000256	0/0000061	Input and output
0.0000861	0.0000262	0.0000063	Looping
0.0000893	0.0000272	0.0000066	Fast
0.0001145	0.0000349	0.0000093	Hassan-José

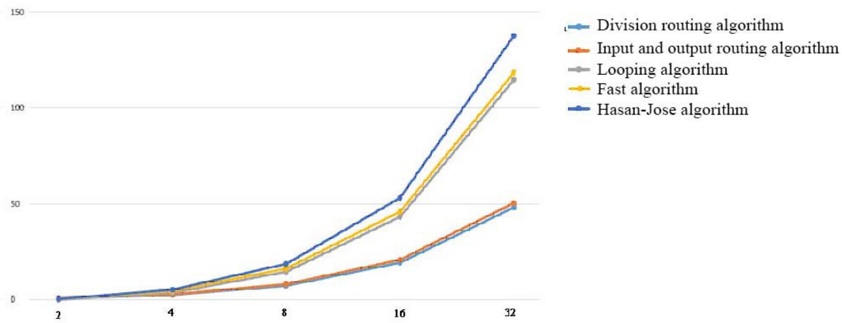

Fig. 10. Compared algorithms in terms of runtime

5. Conclusion

In this paper, we introduced 2 matrix-bases in Benes network, and solve the problems of division algorithm, such as determining the state of switches and the backtracking problem and a new and simple routing method without collision has been shown in Benes network. In input and output algorithm the strength is to reduce the run time since in each chapter of running algorithm the state of switches of two layers is determined at the same time. The blind spot is the overflow of caring addresses in lines that may less the permittivity of system.

Comparing these 2 algorithm with similar algorithms we conclude that the new algorithms in compare with the 3other algorithm has less run time and also between the two algorithms division algorithm has the less run time.

References

1. A Waksman, *A Permutation Network*. Journal of ACM; 1968; 15:159-163.
2. K Yangon, F Tsc-Yun, S. Seung-Woo, *On a Class of Concatenated (2log2n-1)-Stage Interconnection Network*. IEEE Conferences; 1996.p.537-543.
3. CS Yang, LP Zu, *Two Expansible Multistage Interconnection Network*. IEEE Conferences; 1994.p.373-378.
4. D Opferman, NT Tsao-Wu, *On a Class of Rearrangeable Switching Networks*, *Part I: Control Algorithm.* Bell System Technical J; 1971; 50: 1579.
5. Enyue Lu, SQ Zheng, *Fast Reconfiguration Algorithms for Time, Space, and Wavelength Dilated Optical Benes Networks*. International Journal of Parallel, Emergent and Distributed Systems; 2007; 22.
6. H Cam, J Fortes, *Work-Efficient Routing Algorithms for Rearrangeable Symmetrical Networks*. IEEE Trans on Parallel and Distributed Systems; 1999; 10:733-741.
7. A Chakrabarty, M Collier, S Mukhopadhyay, *Matrix-Based Nonblocking Routing Algorithm for Benes Networks*. Computation World: Future Computing, Service Computation, Cognitive, Adaptive, Content, Patterns; 2009.
8. KY Lee, *A New Benes Network Control Algorithm*. IEEE Trans on Computers; 1987; 36:768-772.
9. CP Lee, CP Chang, JS Deng, MHLI, MF Lu, YT Huang, PY Kuei, *A strictly nonblocking network based on nonblocking 4×4 optical switches*. Optical Switching and Networking 9; 2012.
10. N Das, K Mukhopadhyaya, J Dattagupta, *O(n) routing in rearrangeable networks*. Journal of Systems Architecture 46; 2000.p.529-542.
11. AS Almazyad, *Optical omega networks with centralized buffering and wavelength conversion*. Journal of King Saud University, Computer and Information Sciences 23; 2011.p.15–28.
12. A Thamarakuzhi, JA Chandy, *2-Dilated flattened butterfly: A nonblocking switching topology for high-radix networks*. Computer Communications 34; 2011.p.1822–1835.
13. X Jia, F Xin, WR Chuan, *Adaptive Spray Routing For Opportunistic Networks*, International Journal On Smart Sensing and Intelligent Systems; 2013.
14. A Karimi, F Zarafshan, A Jantan, AR Ramli, M Saripan, *A new fuzzy approach for dynamic load balancing algorithm*. arXiv preprint; 2010.p.0910-0317.
15. F Zarafshan, A Karimi, SAR Al-Haddad, MI Saripan, S Subramaniam, *preliminary study on ancestral voting algorithm for availability improvement of mutual exclusion in partitioned distributed systems*. Proceedings of the International Conference on Computers and Computing (ICCC); 2011.
16. A Karimi, A Afsharfarnia, F Zarafshan, SAR Al-Haddad, *A Novel Clustering Algorithm for Mobile Ad Hoc Networks Based on Determination of Virtual Links' Weight to Increase Network Stability*. The Scientific World Journal; 2011.