International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-4 Issue-4, April 2015

The Selection of Mesh in Networks

Attipalli Avinash, Tarik Eltaeib

Abstract - The mesh network structure is vastly implemented in Multi-computers and networks for parallel and distributed computing. In mesh network each computer or processor have maximum of four neighbors. This kind of architecture is implemented in dedicated supercomputers. But, the problem occurs when it comes to the practical implementation in networks. In networks each computer is connected to each other. In this we implement an algorithm to make the mesh network more efficient. The efficient mesh structure is possible in between the best nodes in a network. The proposed algorithm in this mesh network is AGMS ie., Adaptation Genetic Algorithm.

Keywords- **AGAMS algorithm, cost of link, topology.**

I. Introduction

The focus of this paper is to build a mesh structured network, to minimize the total cost of a structure. A computer network with mesh structure has increasing popularity over the regular topology structures [13]. The advantages in mesh structured networks are -they don't need to turn-off the whole network to maintain we just add computers and processors. Super computers are very powerful and they are more expensive. There is cheaper way to build them by using computer network and personal computers.

II. Topology

Computers that are in the mesh network are becoming popular in these days due to their simple configuration. In mesh network the parameters are height *h* and width *w*. In a 2-D network it look like as

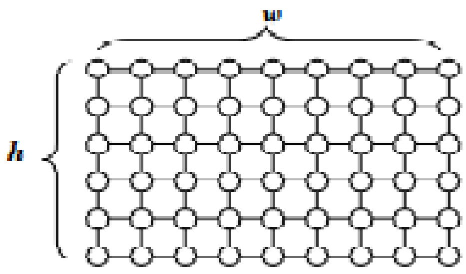

Figure 1. 2D Mesh M(6,9) [16]

a. Problem stage:

Assume that, there are n nodes and we need n(n-1)/2 links to make it a network. For instance, the figure (1.1) shows that we have n=9 and then we need 36 links to make it as a network.

Manuscript published on 30 April 2015.
***** Correspondence Author (s)
Attipalli Avinash*, Department of computer Science University of Bridgeport, USA.
Tarik Eltaeib, Department of computer Science University of Bridgeport, USA.

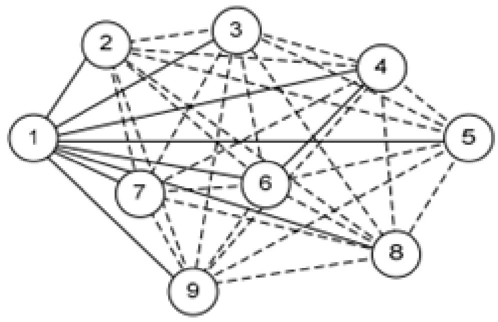

FIGURE 2. Example of nine computers network [4]

There is difficult in designing the mesh network. If there is a data that we want to transfer from one node to the other node, we actually check for the possible free links. To arrange such a possible free links together from source to destination there are n! Possibilities (For n available networks). To overcome this we actually use algorithms [21].

b. Link efficiency:

Link efficiency depends upon the cost of link that means it is just the routing of traffic in a network[15]. Basically, it depends upon the speed of link and the quality of the link etc., based on user. Assume that, we know the each link cost and their normalized values between 0to1.

Index of node	1	2	3	4	5	6	7	8	9
1	x								
2	.23	x							
3	.43	.22	x						
4	.98	.23	.88	x					
5	.53	.23	.74	.64	x				
6	.75	.32	.33	.22	.72	x			
7	.24	.42	.64	.14	.72	.63	x		
8	.23	.23	.23	.62	.31	.64	.34	x	
9	.45	.72	.62	.67	.15	.12	.23	.32	x

Table 1. Cost Array $L_{N\,(A,B)}$ [10]

It shows up smaller the cost of link gives better link.

Assume that the connection from source to destination is likely same from destination to source[18]. Suppose if there are 9 computers we can build a mesh network by considering figure(1), from that we have 12 links and M(3,3). It is shown below,

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

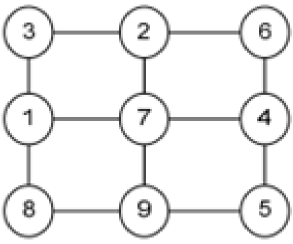

FIGURE 3. A structure of a M(3,3) in example

From that problem we can generate a solution by formulating it by,

> *Given:* n number of computers connected each other and the cost array, size of mesh is
>
> *M(h,w) and n=h*W;*
>
> *Find:* nodes that are assigned and minimize the cost of links [2].

The above generated formula is for the standard or non-uniform case. For a uniform case we have to have knowledge on *weighted coefficients of matrix*[5]. To solve this problem we use SBA-algorithms that actually allocate some new tasks in networks from one fixed point to follow traffic in certain route. When at the initial every nodes check whether the link is busy or available for certain chances is nodes is looking for it at the same time. To reduce the problem we go for the AGAMS algorithm[8].

III. AGAMS Algorithm

It is an extension of a genetic algorithm[9]. These kind of special algorithms are helpful to set combinatorial difficult problems.

a. *Fitness function*

In this algorithm problem is a kind of permutation encoding. The below figure, shows up the number string in a mesh as follows,

3	2	6	1	7	4	8	9	5

FIGURE 4. The Chromosome related to the example of Mesh structure[11].

If joint is successful then nodes are in mesh altogether.

Fitness function is used to analyze the population of a network. If the network satisfies the fitness function then only the network able to work. Because, mesh network is dynamic [16].

$$F_{fitness}=\sum_{i=0}^{h-1} H_i \sum_{j=0}^{w-2} W_j L_{N(i\,w+j;\,i\,w+j+1)} + \sum_{j=0}^{w-1} W_j \sum_{i=0}^{h-2} H_i L_{N(i\,w+j;\,i\,w+j}$$

b. *Crossover*

Correction in encoding is based on classic crossover for permutation encoding. To ensure the chromosome string we actually go for the algorithm "uniform order-based crossover"[6]. The AGAMS algorithms depend on the uniform order-based algorithm [3].

Crossover in nodes is performed in 3 steps:

Step 1: generate a random, mask of 0 and 1.

Step 2: The next chromosome or child is created based on the parent.

Step 3: The permuted nodes are entered in the genes of the children.

For instance, the crossover process in figure(5),

Step 1 – Creation a mask:

0	1	1	0	0	0	1	1	1

Parents:

3	2	6	1	7	4	8	9	5
2	4	7	8	1	9	5	6	3

Step 2:

-	2	6	-	-	-	8	9	5
-	4	7	-	-	-	5	6	3

Step 3:

4	2	6	7	1	3	8	9	5
2	4	7	1	8	9	5	6	3

FIGURE 5. Example of Cross-over process[3]

c. *Mutation and Selection*

Mutation is changing the nodes order. It is shown as

3	**2**	6	1	7	**4**	8	9	5

3	**4**	6	1	7	**2**	8	9	5

FIGURE 6. Example of Mutation Process[7].

Roulette wheel method **is used for the selection.**

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

IV. System Designing

The system design is known as "*Make mesh*". We improve this system module to "*Mesh Application system*". In this we find solution to the problem by

1. Reviewing
2. AGAMS algorithm

Make mesh also have output and input systems.

Input system:

Sort out useful problem parameters like mesh size and cost of arrays etc.,

The parameters of AGAMS algorithm like cross-over probability etc.,

Output system:

At local level is L_{used}.

At global level is E_{ff}.

$$Eff = \left(1 - \frac{E(L_{Used})}{E(L_{All})}\right) 100\%$$

V. Investigation

Probability is used for mutation and selection of Make mesh.

a. Link cost and calculation time

Review the relation between link cost and calculation time of mesh size. Below are the 4 cases for mesh based networks,

	Average cost of all links $E(L_{All})$	Average cost of used links $E(L_{Used})$	Efficiency Eff [%]
$M(2,2)$	0.30	0.22	**25.7**
$M(2,3)$	0.56	0.27	**52.4**
$M(3,3)$	0.50	0.24	**51.3**
$M(3,4)$	0.55	0.26	**53.4**

Table 2. Efficiency of AGAMS[2].

In the case M(2,3) with the help of AGAMS algorithm we obtain the efficiency of 50%. Finally, we conclude that the cost of th link and efficiency is increased by half in AGAMS algorithm.

b. Efficiency and size of the network

In this, the influence is based on the parameters that are in the inner AGAMS algorithm. Two kinds of cases are as,

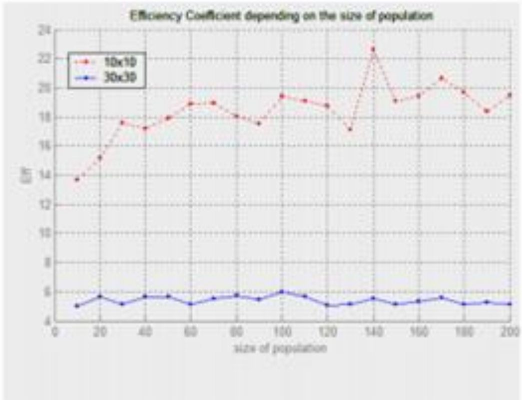

FIGURE 7. Efficiency in relation to the population size[15]

c. Analyze the AGAMS algorithm

AGAMS algorithm takes very less run-time. The two main constraints for this algorithm is,

1. Goal in this to maximize the total speed of a links.
2. Goal was to minimize the cost.

d. Advantages:

The advantage of building such a network is a kind of resource utilization i.e., so many of us have seen that the computers in college and banks that are wasting power in idle situations. We make them to be able to be able to perform the complex computations.

References

1. S. B. Yoo and C. R. Das , "A Fast and Efficient Processor Allocation Scheme for Mesh-Connected Multi-computers", IEEE Trans. on Computers, vol. 51, January 2002.
2. J. Kim, C. R. Das and W. Lin, "A Top-Down Processor Allocation Scheme for Hypercube Computers", IEEE Trans. on Parallel and Distributed Systems, vol. 2, January 1991.
3. S. M. Yoo, H. Y. Youn and B. Shirazi, "An Efficient Allocation Scheme for 2D Mesh Architectures",IEEE Trans. on Parallel and Distributed Systems., vol. 8, September 1997.
4. D. D. Sharma and D. K. Pradhan, "A Fast and Efficient Strategy for Submesh Allocation in Mesh-Connected Parallel Computers", Proc. 5th IEEE Symp. on Paral. and Distr. Proc., 1993.
5. Y. Y. Yang and J. C. Wang, "Pipelined All-to-All Broadcast in All-Port Meshes and Tori", IEEE Trans. on Computers, vol. 50, October 2001.
6. D. Das, M. De and B. P. Sinha, "A New Network Topology with Multiple Meshes", IEEE Trans. on Computers, vol. 48, May 1999.
7. T. Liu, F. W. Huang, Lombardi and L. N. Bhuyan, "A Submesh Allocation Scheme for Mesh-Connected Multiprocessor Systems", Parallel Processing, 1995, pp. 159-163.
8. D. Lisowski and L. Koszalka, "Fast and Efficient Processor Allocation Algorithms for Mesh-Connected Multicomputers", Proc. of 16th Intern. Conf. on Systems Engineering, vol. 2, Coventry, 2003, pp. 231-235.
9. M. De, D. Das and B. P. Sinha, "An Efficient Sorting Algorithm on the Multi-Mesh Network", IEEE Trans. on Computers, vol. 46, October 1997.
10. L. O. Zhang, "Fault-Tolerant Meshes with Small Degree", IEEE Trans. on Computers, vol. 51, May 2002.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

11. D. Lisowski and I. Pozniak-Koszalka, "Efficiency Analysis for Task Allocation Algorithms in Mesh Networks", Sci. Report, Faculty of Electronics, WUT, July 2004 (in Polish).
12. Karbowski, A., and E. Niewiadomska, Parallel and Distributed Computing, WUT, 2001 (in Polish).
A. Agarwal, "The MIT Alewife Machine: Architecture and Performance", Computer Architecture,1995, pp. 2-13.
13. http://www.hp.com.
14. http://www.top500.org.
15. http://www.beowulf.org.
16. http://www.gomez.com.
17. http://www.seti.org.
18. Davis, L., Handbook of Genetic Algorithms, Van Nostrand Reinhold, New York, 1991.
19. Michalewicz, Z.,Genetic Algorithms + Data Structure = Evolution Programs, Springer Verlag, New York, 2001.
20. M. Gen. and Cheng R., Genetic Algorithms and Engineering Design, John Wiley & Sons, New York, 1999.
21. K. T. Ko and S. Kwong , "Using Genetic Algorithms to Design Mesh Networks", Computer, vol. 30, August 1997.
22. Kasprzak, A., Design of Wide Area Networks,Wyd. Pol. Wroc., Wroclaw, 1999 (in Polish).
23. L. Koszalka, D. Lisowski and I. Pozniak-Koszalka, "Comparison of Task Allocation Algorithms for MeshStructured Networks", Lecture Notes in Computer Science, 3886, Springer, 2006, pp. 21-30.

Avinash Attipalli, I am pursuing master's in USA. Have a good skills and certification in CCNA and more on programming skills.

Tarik El Taeib, As a profession, working as an Associate professor in UNIVERSITY OF BRIDGEPORT,USA. Research in areas of interest Networking and router protocols and it's programming. Received PhD in computer scienc.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)