Asian Journal of Information Technology 6 (2): 132-135, 2007

Minimum Proximity Algorithm to Minimize Delay in Networks

[1]D.S. Mahendran and [2]S. Sakthivel
[1]Department of Computer Sciences, Aditanar College of Arts and Science, Virapandianpatnam, 6282216 Tiruchendur, India
[2]Department of CSE and Engineering, Alagappa University, Karaikudi-630003, India

Abstract: This study shows a new approach to minimize delay in networks. The routing algorithm guarantees a loss-free delivery of data packets from congested sources and a deterministic bound on the route length in arbitrary topology networks. This research shows that routing decisions using Local Greedy method are not optimal and the performance of the algorithm can be improved substantially by using new look-ahead measures. The contribution of this study is to propose a new metrics to minimize delay in networks .In minimum proximity algorithm the time taken to set routing priority is less when compared to local greedy algorithm. The objective is to minimize the average packet delay. The performance is studied computationally for various networks topologies under static traffic model. In all the experiments the Minimum Proximity algorithm shows better results.

Key words: Congestion, routing, proximity, offered load, routing priority

INTRODUCTION

An efficient routing result in smaller average packet delays, which means that the flow control algorithm can accept more traffic into the network. On the other hand, an efficient flow control algorithm rejects excessive offered load that would necessarily increase packet delays by saturating network resources. Earlier models of static and dynamic routing problems have been well studied by Bertsekas and Gallager (1992), Bertsekas (1982), Segall (1977) and others.

In nondeterministic routing techniques such as hot-potato routing (Baran, 1964), deflection routing (Greenberg and Hajek, 1992) and convergence routing (Ofek and Yung, 1994; Yener *et al.*, 1994; Yoram and Modi, 1995) ensure no packet loss due to congestion inside the network. The nondeterministic routing combines, in a dynamic fashion, the on-line routing decision with the instant traffic load inside the network. The dynamic behavior of deflection routing has been studied on some regular topologies such as the Mahattan street network (Greenberg and Goodman, 1986; Maxemchuck, 1982) and the hypercube. Convergence routing (Yener *et al.*, 1994) ensures that packets will reach their destinations without being routed on the same link twice. Thus it ensures a deterministic bound on the maximum route length in an arbitrary topology network (Bao and Garcia, 2003).

Deficiencies of local-greedy algorithm: The performance of convergence routing with the local-greedy routing strategy is not necessarily the best one since it considers only the local traffic conditions. Local-Greedy routing decisions may also increase delay and congestion since it takes more time (because of more iterations) to set routing priorities and path length can also be increased due to default routing.

Related work: Global improvement and implicit self-routing is given by the method of "interval routing" of Santoro and Khativ (1985). In MetaNet routing (Yoram and Modi, 1995) algorithm the packets will reach their destinations unless physical failure has occurred. This property is not provided by deflection routing, which means that the packets can deflect indefinitely inside the network. Therefore, in Baran's Hot-Potato (Baran, 1964) routing there is hop-count field in each packet header, which is decremented by one after every hop. If the hop-count reaches zero the packet is discarded i.e., the packet may get lost due to congestion inside the network.

In Metanet Principles of an Arbitrary Topology LAN (Bao and Garcia, 2003), they assumed that the physical layout of the network is a tree and a ring is embedded into an arbitrary topology network (Yener and Ofek, 1994) using euler tree traversal without the thread links. All the links of the tree are ring links and are part of Tree Embedded Ring (TER).

In Abraham and Kumar (2001), a successful approach to deal with network delays, although it requires the computation of a large number of control parameters. The number of parameters is proportional to the round trip delays of the system. Reference (Katabi *et al.*, 2002)

Corresponding Author: D.S. Mahendran,Department of Computer Sciences, Aditanar College of Arts and Science, Virapandianpatnam, 6282216 Tiruchendur, India

acknowledges the need to use a separate set of parameters for each delay's values and the number of sources in the system. In (Dirceu *et al.*, 2004), a congestion control system is developed to motivate the handling of feedback delays.

In Virtual Node Algorithm for Data Networks (Mahendran and Sakthivel, 2007) they assumed that the physical layout of the network is a graph and a ring is embedded using the above algorithm with thread links. The traversal ring on a graph is called graph embedded ring.

Proposed algorithm: In this stduy the performance of local-greedy routing decisions can be improved substantially with a different distance measure that provides a look-ahead of the potential routes. The performance measure considered in this work is to find the loop free paths, so that it minimizes the average packet delay. Given a traffic load, optimization of a nondeterministic routing algorithm requires new techniques since actual routes cannot be fixed, but altered, based on routing priorities and the actual traffic conditions The performance improvements are studied on various network topologies.

Network model: A computer network is modeled as an undirected simple graph G = (N, E), where N is the set of nodes and E is the set of edges or links connecting the nodes. Each node has its own unique ID, denoted by a capital letter A, B, C, D, E… etc., as in Fig. 1.

Virtual ring embedding: A virtual ring is embedded in a network by the Virtual Node Algorithm (Mahendran and Sakthivel, 2007). Such a virtual ring is called the Graph Embedded Ring or GER and the links are called ring links. The virtual ring links are numbered sequentially from 0 to m-1. The number associated with each ring link constitutes a Virtual Node (VN). Thus, m is the number of virtual nodes induced by the ring embedding (for example, in Fig. 2 m = 26).

Forward node: Let i, j and k be the virtual nodes. Then, j is called a forward node and (i,j) a forward link of i for destination k if and only if the following two conditions are true.

- Distance (in hops) from j to k must be less than i to k.
- There exists a physical link (I,J) in the network, such that i is a virtual node of I and j is a virtual node of J. This can be either a ring or thread link.

For example, in Fig. 2, the forward nodes of 2 for destination 17 are 3 and 15. The corresponding forward links are (2,3) and (2,15).

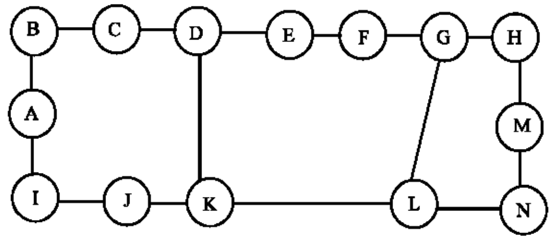

Fig. 1: Given network topology

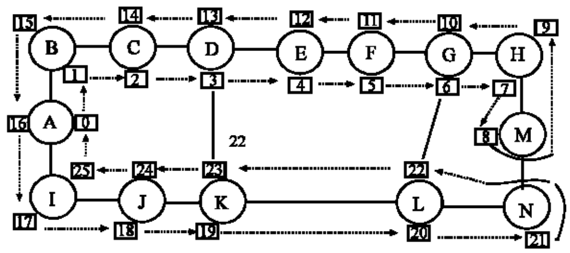

Fig. 2: An network topology with virtual rings

Let the sets N (i, k) and L (i, k) denote the forward nodes and links of virtual node i for destination k, respectively. A forward path from i to k is a chain of forward links ie. a path of the form $(v_1,v_2,\ldots v_n)$ where v_1=i, v_n=k and $(v_l,v_{l-1}) \in L(v_1,k)$, where l=1,…..n-1.

Convergence routing model: In this study a new analytical model for the behavior of convergence routing at a node (switch) has been introduced. This model will enable us to determine the routing probabilities at each node for a given destination. At each node, the routing probabilities are computed according to an ordering of the forward links of this node for the destination.

Computing routing probabilities: Given a priority assignment and utilization values on the links, the computation method for the routing probabilities is discussed here. Note that convergence routing algorithm will switch a packet designated for k to the link (i,j) with priority y if all of the links with higher priority are busy and this link is available. Precisely, the probability that edge (i,j) is selected for destination k with priority y, is

$$P_{ij}^{k} = \left(1-\rho_{ij}\right) \prod_{x:\,\pi_{ix}^{k} > y} \rho_{ix} + \left(\prod_{x=1}^{n_i^k} \rho_{ix}\right) \frac{1-\rho_{ij}}{\sum_{x=1}^{n_i^k}(1-\rho_{ix})} \quad (1)$$

Note that according to the probability law, the sum of routing probabilities for a destination at an intermediate node/switch must add up to unity (ie. $\Sigma_{ij} P_{ij}^{k}=1$ at node i for destination k).

Table 1: The performance measure used in our experiments is to find the loop free path that converges with minimum number of iterations

	Iterations	
No. of Nodes	LG-Alg	MIP-Alg
z8	10	8
14	12	10
16	14	8
18	13	11
20	13	12
22	13	9
24	10	7
26	10	7
28	13	9
30	10	7
34	13	7

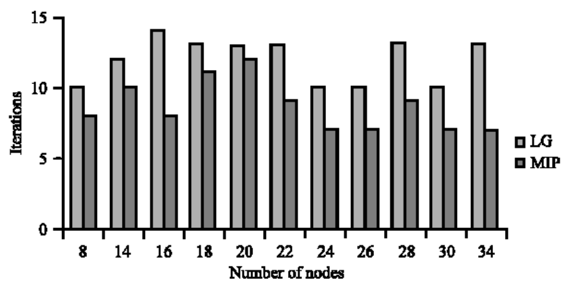

Fig. 3: Comparison of LG and MIP algorithms

Minimum Proximity (MiP): The Minimum Proximity (MiP) of node i to node k is defined as follows.

$$MiP_i^k = \min_{j \in N(i,k)} MiP_j^k (1-\rho_{ij}) \qquad (2)$$

The Minimum Proximity (MiP) routing algorithm can be defined as the convergence routing algorithm that assigns the priorities of the forward links of node i by sorting the nodes in N (i,k) in ascending order of their minimum proximities to destination k. Using this new definition, the computation of MiP can be done very efficiently. Where ρ_{ij} is the utilization of the link (i, j).

Minimum proximity (mip) algorithm

- Input-The given network.
- Insert virtual nodes using virtual node algorithm.
- Find the forward nodes.
- Compute the distance between each node in hops.
- Assign initial utilization value for each link in the network.
- Assign initial load/traffic to each link in the network.
- Compute the Minimum Proximity (MiP).
- Assign priority to each link.
- Set up the routing table at each node.
- Compute the routing probability for each link.
- Compute the flow on each link.
- Compute the new utilization value of the link.
- Maximum number of iterations performed or convergence, terminate the process otherwise go to step 7.
- Display the number of iterations required to set routing priorities

Performance analysis of the algorithm

Convergence property of the algorithms: In this study the proposed algorithm (MIP) is compared with Local-Greedy (LG) algorithm. The performance measure used in our experiments is to find the loop free path that converges with minimum number of iterations. So that it minimizes the average packet delay. In all the experiments the proposed algorithm converged quickly than local-greedy algorithm. Convergence performance is investigated in terms of number of iterations. The results are collected in Table 1 and shown in Fig. 3.

CONCLUSION

The local-greedy routing algorithm has certain deficiencies due to its limited use of information about the traffic load conditions across the network during the routing process. As a remedy, a new look-ahead measure is proposed. The look-ahead measure is embedded into the proposed algorithm that simulates the behavior of the local-greedy and minimum proximity routing algorithms on arbitrary networks. A series of experiments were performed for various network topologies and in all the experiments the proposed algorithm converged with minimum number of iterations, which minimizes the packet delay in the network and it is also observed that the congestion value is also minimum in the proposed algorithm than the local-greedy algorithm.

REFERENCES

Abraham, S.P. and A. Kumar, 2001. New approach for asynchronous distributed rate control of elastic session in integrated packet networks, IEEE. Trans. Networking, 9: 15-30.

Bao, L. and J.J. Garcia-Luna-Aceves, 2003. Topology Management in Adhoc Networks. Fourth ACM international symposium on mobile adhoc networking and computing.

Baran, P., 1964. On distributed Communication Networks, IEEE. Trans. Commn. Sys., pp:1-9.

Bertsekas-Dynamic D.P., 1982. Models of shortest routing algorithms for communication networks with multiple destinations IEEE. Trans. Automat. Contr., AC-27, pp: 60-74.

Bertsekas, D.P. and R.G. Gallager, 1992. Data Networks PHI.

Greenberg A.G. and J. Goodman, 1986. Sharp approximate models of adaptive routing in mesh networks, Teletraffic Ann. Comput. Performance, pp: 255-270.

DirceuCavendish, Mario Gerla and Saverio Mascolo, 2004. A control theoretical approach to congestion control in packet networks, IEEE/ACM. Trans. Networking, 12: 893-906.

Greenberg, A.G. and B. Hajek, 1992. Deflection routing in hypercube networks, IEEE. Trans. Common., 40: 1070-1081.

Katabi, D., M. Handley and C. Rohrs, 2002. Congestion control for high bandwidth delay product networks, in Proc. ACM SIGCOMM, pp: 89-102.

Mahendran, D.S. and S. Sakthivel, 2007. Virtual Node Algorithm for Data Networks, Accepted for publication in the Int. J. Soft Computing, to be Published in Vol 1.

Maxemchuck, N.F. 1982. Routing in the manhalthan Succd, IEEE. Trans., AC27: 60-74.

Ofek, Y. and M. Yung, 1994. The integrated Metanet architecture: A switch based multimedia LAN for parallel computing and real-time traffic, Presented at IEEE., INFOCOM.

Santoro, N. and R. Khativ, 1985. Labeling and implicit routing in networks, Compt. J., 28: 259-273.

Segall, A., 1977. The modeling of adaptive routing in data communication networks, IEEE. Trans. Commun., 25: 85-95.

Yener, B., Y. Ofek and M. Yung, 1994. Design and performance of convergence routing on spanning trees, IEEE. GLOBCOM, pp: 169-175.

Yener, B. and Y. Ofek, 1994. Topological design of loss-free switch-based LAN's in Proc. IEEE. INFOCOM, pp: 802-811.

Yoram Ofek and Modi Yung, 1995. Meta Net Principles of an Arbitrary Topology LAN. IEEE. Trans. Networking, 3: 169-180.