DESIGN OF MPLS BASED ATM NETWORKS

R.Tino Merlin
Francis Xavier Engineering College
Email:tinomerlin@yahoo.co.in

Abstract—The problem is to find the cheapest path that satisfies certain constraints. The efficiency of the algorithms directly relates to the magnitude of the errors introduced during discretization. In this paper, we propose two techniques that reduce the discretization errors, which allow faster algorithms to be designed. Reducing the overhead of computing constrained shortest paths is practically important for the successful design of a high-throughput QoS router, which is limited at both processing power and memory space.

1. INTRODUCTION

In practice, on-line algorithms are not always desired. When the request arrival rate is high (major gateways may receive thousands or tens of thousands of requests every second), even the time complexity of Dijkstra’s algorithm will overwhelm the router if it is executed on a per-request basis.1 To solve this problem, the second scheme is to extend a link-state protocol (e.g., OSPF) and periodically pre-compute the cheapest delay-constrained paths for all destinations, for instance, for voice traffic with an end-to-end delay requirement of 100 ms. [1-5] The computed paths are cached for the duration before the next computation. This approach provides support for both constrained unicast and constrained multicast. The computation load on a router is independent of the request arrival rate. Moreover, many algorithms, including those we will propose shortly, have the same time complexity for computing constrained shortest paths to all destinations or to a single destination. This paper studies the second scheme [6-10].

2. PROBLEM DEFINITION

We design the path discretization algorithm (PDA) based on the above intuition. The algorithm solves the ε–approximation with the same worst-case complexity as RDA. However, its average execution time is better than RDA according to our simulations.

3. PERFORMANCE ANALYSIS

Creating the Cluster Head

In this fig 1 the cluster node source node and destination node will be identified according to that the data will be transmitted from source to destination by cluster information cluster node is group controller source and destination is used to communicate between other nodes.[11-15]
Fig-2 Creating the cluster Head

Indicates the Watchdog

In this fig 3 is used to know about the watchdog. The watchdog will be placed in every group of node. In a group of node which has a higher energy is act as a watchdog in every group of nodes.

Fig-3 Indicates the Watchdog

Detecting the Malicious Node

The watchdog node will be finding about the malicious node and give alert signal to another node and cluster head. The data will not be transfer to the malicious node.

Sometimes the data will be send to malicious node at that time the will be lost in fig-2.

Fig-3 Detecting the Malicious Node

5.1.8 Moved the Malicious Node

In fig 4 every malicious node will be slowly removed from the group.

Fig-4 Moved the Malicious Node

GRAPHICAL REPRESENTATION
Throughput

In the above graph 5 shown the difference of throughput and time. The red color line will be indicates the proposed system and green color line will be indicates the existing system.

Table 1 Throughput

<table>
<thead>
<tr>
<th>S.No</th>
<th>Existing System</th>
<th>Proposed System</th>
</tr>
</thead>
</table>

All Rights Reserved © 2015 IJARBEST
In this graph is shown the difference between delay and time. The red color will indicate the node energy and green color will be indicates the watchdog energy.

<table>
<thead>
<tr>
<th>S.No</th>
<th>Node Time</th>
<th>Watchdog Delay</th>
<th>Time</th>
<th>Delay</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>2</td>
<td>10.0000</td>
<td>1.6500</td>
<td>10.0000</td>
<td>3.2000</td>
</tr>
<tr>
<td>3</td>
<td>20.0000</td>
<td>1.8000</td>
<td>20.0000</td>
<td>3.3000</td>
</tr>
</tbody>
</table>

5.2.3 Delay

In this graph is shown the difference between existing delay and proposed delay. The red color indicates the energy level of delay and the green color indicates the proposed system delay. The energy indicates the delay slowly and proposed will be color indicates the delay fastly.

4. CONCLUSION

In this paper, we proposed two techniques, randomized discretization and path delay discretization, to design fast algorithms for computing constrained shortest paths. While the previous approaches (RTF and RTC) build up the discretization error along a path, the new techniques either make the link errors to cancel out each other along the path or treat the path delay as a whole for discretization, which results in much smaller errors.

REFERENCES


Compression and Decompression for Wavelet Based Algorithm’, Wireless Personal Communications, Print-ISSN: 0929-6212, Volume. 81, No. 2, pp. 839-859, March 2015.