

Adaptive Bandwidth Reservation Scheme for Multimedia Traffic Using Mobile Agents

S. S. Manvi* and P. Venkataram**

Protocol Engineering and Technology (PET) Unit, ECE Department,
Indian Institute of Science, Bangalore-560012, INDIA,

*E-mail: smanvi@ieee.org, <http://pet.ece.iisc.ernet.in/sunil>

**E-mail: pallapa@ece.iisc.ernet.in, <http://pet.ece.iisc.ernet.in/pallapa>

Abstract

The bandwidth reservation for multimedia traffic poses technical challenges due to bursty and delay sensitive nature of applications. The objectives of bandwidth reservation schemes are: optimize network utilization, minimize the packet losses and delays. The growth of multimedia services on Internet and the possible discovery of programmable networks has made us to investigate new techniques for resolving bandwidth issues in multimedia communication. Mobile agent technology seems to be the promising solution for network management and QoS control. In this paper, we consider joint problems of adaptive bandwidth reservation and link rearrangement (rerouting) for multimedia traffic under the event of congestion/failures of link, and propose a mobile agent based approach to achieve these objectives. The scheme is simulated using a multimedia traffic model. Simulation results show that the use of agents increase the network utilization, acceptance ratio of applications, flexibility and efficiency of bandwidth reservation. The flexibility in using agent technology is that the policies can be changed and implemented easily by encoding in the agents.

1 Introduction

Multimedia applications such as video-conferencing, tele-conferencing, on-demand services, etc., are less data loss sensitive, more delay sensitive and bursty in nature, and consume significant amount of network bandwidth. They require real time delivery and acceptable QoS (Quality of service). Therefore efficient bandwidth reservation schemes are needed to ensure better QoS. Static bandwidth reservation schemes reserve only once after admitting the application, such as peak rate reservation, but under-utilizes network bandwidth due to variable bit rate (VBR) services [1]. Adaptive bandwidth reservation schemes reserve bandwidth online according to the implicit or explicit feedback about the network state from intermediate nodes and receivers [2,3], pricing policies, application priorities, queuing based reservation, etc.

Since conventional IP routing uses shortest path technique for packet routing, network load will be unevenly distributed, paths get congested and degrade performance of the network. Hence, it is necessary to consider the joint problems of rerouting (rearranging flows) and adaptive bandwidth reservation under the congestion/link failure situations to optimize network utilization. Self healing and backup virtual path type schemes are used for restoration of network during failures and reserve bandwidth on the alternate routes [4,5]. Rerouting strategies that can be used are source routing, local rerouting, local destination routing. The

work given in [6] considers a new self healing scheme that re-reserves bandwidth for differentiated bandwidth requirements of applications during rerouting under link/node failure situations. Some of the works done on adaptive bandwidth reservation and rerouting under traffic load variations are based on bumping packets, least loaded paths, minimum delay routes, partition policy with reservation on a link, markov decision process and path rearrangement schemes [7]. Mobile agent based bandwidth adjustments at source for point to multipoint connections subject to network conditions and bandwidth reservation at network nodes over a given path are suggested in [8, 9]. Mobile agents are also used to learn the load on the network and update routing tables so as to balance the load in network [10].

In this paper, we will describe a mobile agent based adaptive bandwidth reservation and rerouting based on congestion occurrence or failure of a link. The scheme assumes availability of mobile agent platform and bandwidth reservation agency. The agency comprises of mobile agents, static agents and resource handling information at each node in the network. We also assume that disjoint alternate routes to patch up a link are available in the node. These alternate paths are supplied to all the nodes by a central topology manager. Mobile agent platform comprises of agents, agent server, interpreter and transport mechanisms for agents (TCP or E-mail). Agent server is responsible for receiving mobile agents and sending it for execution by interpreter.

A mobile agent at the node is triggered depending on traffic thresholds or failures monitored on the links. In the event of congestion occurrence on link (monitored traffic > threshold) or link break down, mobile agent of each attached node of link picks an disjoint alternate route for that link from alternate routing table to patch up that link, and travels to capture and reserve the required bandwidth for rerouting the applications. The scheme intuitively does load balancing and tries to evenly distribute the load in network there-by increasing network utilization. Following sections explain the mobile agents, proposed scheme, multimedia traffic model and results.

2 Mobile Agents

Agents are the autonomous programs situated within an environment, which senses it and acts upon it using its knowledge base, and learns so as to act in future. They have certain special properties (mandatory and orthogonal) which make them different from the standard programs. Mandatory properties of the agents are: *autonomy, reactive, proactive and temporally continuous*. The orthogonal properties are: *communicative, mobile, learning and believable* [11]. Mobile agents are the multiagent systems which possess the mandatory properties and some or all of the orthogonal properties specified for an agent. It is an itinerant agent dispatched from source host which contains program, data and execution state information, migrates from one host to another host in the heterogeneous network and executes at remote host until they accomplish their task.

The mobile code should be platform independent, so that, it can execute at any remote host in the heterogeneous network environment. They communicate and cooperate with other agents to achieve its goals. Agent can update its information base while interacting with other agents during its travel. Inter-agent communication can be achieved by message passing, RPC or common knowledge base (black board). Agents can be written in Tcl, Perl, Java and C++ languages [12]. In general, there are several good reasons for using mobile agents: they reduce network load; overcome latency; encapsulate protocols; execute asynchronously and autonomously; adapt dynamically.

3 Mobile agent based bandwidth reservation scheme

The scheme assumes the availability of mobile agent platform, bandwidth reservation agency and resource handling information at the nodes in the network. Mobile agent platform is responsible for receiving agents, agent execution, transporting agents and inter-agent communication. The details of bandwidth agency and proposed scheme are given in the following subsections.

3.1 Bandwidth reservation agency

Bandwidth reservation agency comprises of several agents and resource handling information. Figure 1 depicts the agents and their interactions. It consists of manager agent, link monitoring agents, scheduler agents, bandwidth (BW) capture mobile agents and profiles.

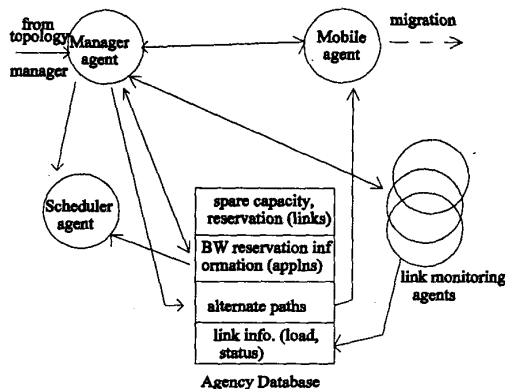


Figure 1. Bandwidth reservation Agency.

The functions of the agents and agency database are as follows:

- **Link monitoring agent:** it is a static agent responsible for monitoring the traffic and link failures, measurement is done using time window mechanisms, stores the monitored data (incoming and outgoing traffic) in link information profile and exchanges the information about buffer occupancy in attached nodes of link. It also computes the link status by comparing sum of reserved/allocated (monitored) bandwidth, buffer occupancy of attached nodes of link with traffic threshold and stores status as congested, normal, fail in link information profile.
- **Manager agent:** it is a static agent which triggers monitoring agents, scheduler agents, mobile agents and interacts with topology manager to get new alternate routes during topological changes. This agent continuously monitors status of each link, if congested/failed, calculates bandwidth requirements of excess traffic, triggers mobile agent to capture and reserve the bandwidth.
- **Scheduler agent:** it is triggered by manager agent, after mobile agent captures bandwidth. It schedules bandwidth for the applications to be rerouted on alternate route.
- **BW capture mobile agent:** it is triggered by manager agent during congestion/failures. This is responsible for reserving bandwidth on the alternate route by interacting with the nodes in

the path through spare capacity and reservation tables. During onward journey, tentative reservation is done and confirmation is given during return journey.

- **Agency database:** it comprises of spare capacity and reservation tables, link information, alternate paths, and reservation for applications on links. Spare capacity is the residual capacity of link after reserving bandwidth for applications on that link. Alternate path database consists of patch up disjoint routes for all the links of a node. Link information database comprises of monitored data (traffic load, congestion status, failure status, buffer occupancy of node and adjacent node of the link) of all the links of a node. Reservation database contains information about the applications running on the links and their bandwidth scheduled. The reservation database will be modified by scheduler agent during rerouting.

3.2 Bandwidth reservation by mobile agents

The working of the scheme is illustrated by considering a partial network topology as shown in Fig 2. Let us say that, at certain instant of time, the link 1-4 may get congested or fail (congestion occurs if bandwidth reserved + buffer occupancy for link > threshold; failed, if no traffic on link). Manager agents in node 1 and 4 trigger their mobile agents to capture and reserve bandwidth on the alternate paths for rerouting. These alternate paths are supplied by the centralized *topology manager* that also sends updated alternate paths for link patch up in case of addition/deletion of nodes. The changes in topology are fed from the network nodes/administrators.

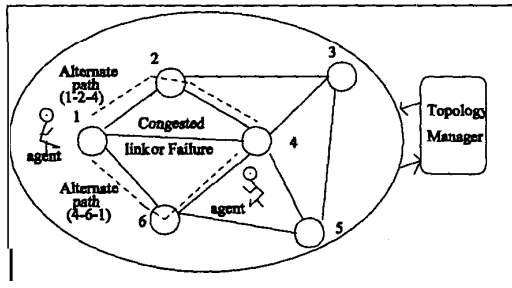


Figure 2. Bandwidth reservation and routing by Mobile agents.

The mobile agent on node 1 (assume that the node with **lower** numbered address of the affected link will choose first path) chooses the first alternate path (1-2-4) and mobile agent on node 4 will choose second alternate path

(4-6-1) for their travel. Then, mobile agents inform to the sender nodes about reserved bandwidth.

Mobile agents initially reserve spare bandwidth available on links of alternate routes (minimum of required and spare bandwidth). On reaching destination (4 or 1) they will pick the minimum of bandwidth reserved, confirms bandwidth reservation while returning to sender node. Agent releases the excess bandwidth reserved. After reservation at node, mobile agents update routing tables of the nodes in alternate path. Once bandwidth reservation is confirmed, nodes will schedule applications of failed/congested link on alternate routes. Every node contains a database of spare capacity available on its links and, spare capacity reserved. This will facilitate mobile agents to capture bandwidth.

The scheduler agents at the nodes will schedule the applications on the alternate paths. The scheduling can be based on priority of classes of traffic, if bandwidth captured is not enough to support the overloaded traffic. During such situation, some applications may be rejected or delayed. The following pseudo-code (algorithm) gives description about the functioning of the scheme. The algorithm is repeated periodically at each node in the network. The symbol l_i represents the i^{th} link of a node.

Algorithm: Adaptive Bandwidth Reservation

Begin

1. Initialize $i=1, n=$ number of links of a node;
2. Manager agent at a node checks a link (l_i) status using link information profile;
3. If (Congestion/failure of link l_i C TRUE) go-to step 7;
4. Manager agent sends mobile agent on alternate routes to capture and reserve bandwidth (uses spare capacity and reservation tables of node);
5. Mobile agent confirms reservation on return travel, update routing tables and inform the manager agent;
6. Scheduler agent schedules bandwidth for applications on alternate route, if sufficient bandwidth is available, else reject some of the applications;
7. $i=i+1$; If ($i \leq n$) repeat steps 2-5 for link l_i of node;
8. STOP.

End.

4 Multimedia traffic model

We assume two traffic classes which are more prominently used in network modeling. They are continuous bit rate services (CBR) and variable bit rate services (VBR). VBR sources are modeled as on-off models. The network model considered is a collection of 'J' links, $\{l_1, l_2, \dots, l_j\}$, where each link has a capacity of C bandwidth units. The network supports two classes of traffic. Associated with each traffic class there is an arrival rate λ_c and λ_v respectively, such that $\lambda_c + \lambda_v = 1$, large holding time and bandwidth requirement. Suppose, if there are 100 sources contributing to the traffic, if λ_c

$=0.4$, then $\lambda_v=0.6$, thus the number of class 1 traffic calls will be $0.4*100=40$ and class 2 will be 60. A route for an arrived application/call is a subset of links. An arriving call/application is admitted into network with its route, r , such that r is a subset of L , which is a predetermined route randomly selected from route set $R=\{r_1, r_2, \dots, r_n\}$ where r_i is a subset of L , such that $L=\{l_1, \dots, l_j\}$. These routes are uniformly distributed. Probability of a link being chosen for the route in the route set is given as, $p = 1/(\text{number_of_links_in_route})$, and the offered load on a link is product of p and bandwidth units requested by source. This model has been used in simulation to generate the background traffic. The bandwidth units for class 1 (CBR) traffic is assumed to be uniformly distributed within some range (a,b). Similarly bandwidth units for class 2 (VBR) traffic is uniformly distributed in the on period within some range (x,y). The burstiness of VBR traffic is given by on/(on+off). We have considered probability of being in on period as Bernoulli distributed with value P_b . Traffic is generated for an VBR call/application, if it is in on period.

5 Results and discussion

We have considered a partially .connected network topology of 16 nodes for simulation as shown in Figure 3 under congestion and link failure situations. Simulations have been carried out extensively with different random number seeds. The results presented here are an average of them.

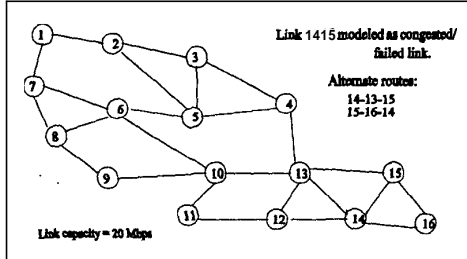


Figure 3. Network topology for simulation

The input parameters used for simulation are: link capacity= 20 Mbps; VBR traffic sources peak rate are uniformly distributed in the range (0.2,0.4) Mbps; P_b , Probability of being in on period (utilization of source) for VBR source is 0.7; CBR traffic is uniformly distributed in the range (0.3,0.5) Mbps; $\lambda_c=\lambda_{ca}=0.5$, probability of class 1 (CBR) traffic arrivals for background and foreground traffic; $\lambda_v=\lambda_{va}=0.5$, probability of class 2 (VBR) traffic arrivals; p_{mc} (probability of agent capturing bandwidth on alternate route) values considered are 1.0, 0.8 and 0.6; threshold = 25.0 Mbps; background traffic sources (N) are 300 and 400; number of additional sources on congested/failed link are varied from 20 to 200 sources.

Congestion is triggered on link 14-15, after background traffic is generated by N sources. Figures 4 and 5 depict the simulation results under congestion situations. We observe that the network utilization increases with use of mobile agents. After certain number of applications, it saturates due to unavailability of bandwidth on alternate routes (14-13-15, 15-16-14). We also observe that percentage of applications rejected for rerouting increases when utilization saturates. This is quite obvious, because, residual/spare capacity is not sufficient on alternate route to divert all the existing applications. It is observed that network utilization decreases and rejections increase with the decrease in probability of mobile agent ($p_{mc}=1.0, 0.8, 0.6$) capturing bandwidth, these detailed results are not presented due to page limitations.

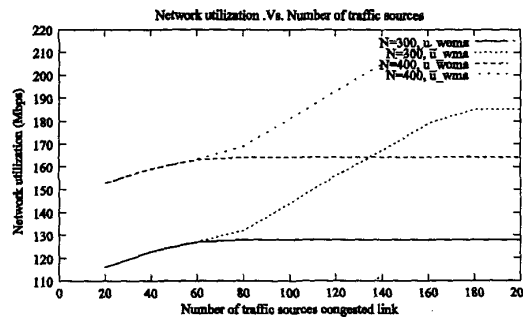


Figure 4. Network utilization with and without mobile agents (u_{woma}). Vs. Number of traffic sources with $P_{mc}=1$ and $N=300$ and 400.

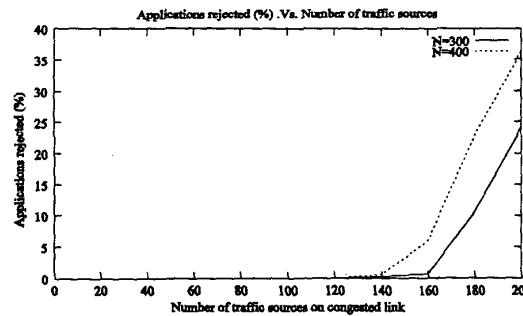


Figure 5. Applications rejected (%) Vs. Number of traffic sources with $N=300$ and 400 and $P_{mc}=1$.

Similar results are shown in figures 6 and 7 for link failure. These figures indicate that the proposed scheme increases network utilization and decreases rejections of applications. The reserved bandwidth depends on existing traffic and hence we observe that rejections will increase when existing traffic is raised from 300 to 400 sources for both congestion and link failure situations. It is also observed that as threshold value is increased, utilization decreases and rejection ratio reduces.

We have also computed the time complexity of mobile agents, i.e., time required by mobile agent to dynamically allocate bandwidth. The assumptions made for computation time are: agent migration time is uniformly distributed between (0.3, 0.5) seconds, and agent execution time is exponentially distributed with mean execution time=0.1 seconds. The simulation results show that the computation time varies in the range of 2.2 to 2.5 seconds for increasing number of applications on congested or failed link with constant values of background sources.

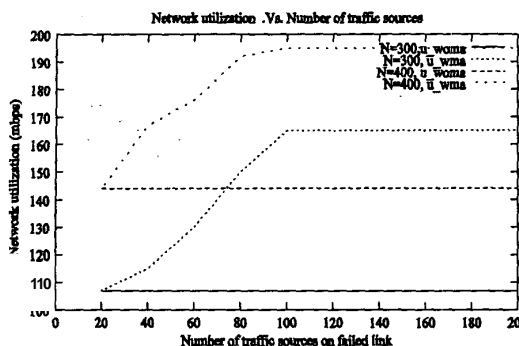


Figure 6. Network utilization with and without mobile agents(u_woma). Vs. Number of traffic sources on failed link with Pmc=1 and N=300 and 400.

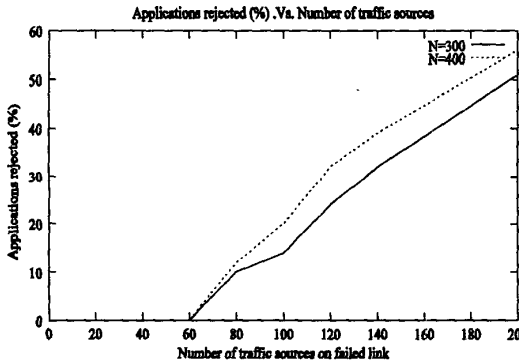


Figure 7. Applications rejected (%) Vs. Number of traffic sources on failed link with N=300 and 400 and Pmc=1.

6. Conclusions

The paper considered joint problem of adaptive bandwidth reservation and rerouting to achieve optimal network utilization and exploited uneven network load distribution. Proposed Mobile agent based adaptive bandwidth reservation scheme is a flexible and scalable

approach. The scheme triggers adaptive bandwidth reservation based on traffic load variations and link failure. Several other factors such as, policy changes by administrator, delays cropping up, priority can also be considered. Agents allow encoding of all these factors in them for bandwidth reservation, thus providing flexibility and scalability. Simulation results demonstrated the optimal use of network bandwidth and the feasibility of using mobile agents especially in heterogeneous nodes in the networks. However, several overheads are associated with the scheme such as: agent transport time and bandwidth; memory space for agent platform; CPU slots for agent execution; security to hosts and agents. But, these overheads are nullified owing to advantages of using mobile agents for adaptive bandwidth reservation.

References

- [1] I.A. Sariot Ismail, "Bandwidth problems in high speed networks", *IBM J.Res. Develop*, Vol. 44,2000.
- [2] M. Mowbray, G. Karlsson, and T. Kohler, "Capacity reservation for multimedia traffic", *Distr. Syst. Eng.*, Vol. 5, 1998, pp. 12—18.
- [3] Y. Thomas Hou, Bo Li, Shivendra S. Panwar, Henry Tzeng, "On network bandwidth allocation policies and feedback control algorithms for packet networks", *Computer networks*, Vol. 34,2000, pp. 481—501.
- [4] Komut Wipusitwarakun, Hideki Tode, Hiromasa Ikeda, "VP's priority based restoring function enhanced self healing algorithm", *IEICE trans. Comm.*, Vol. E81-B, No. 11,1998, pp. 2100-2109.
- [5] Ayano Yamashita, R. Kawamura, Hisaya Hadama, "Dynamic VP rearrangement in ATM networks", *Proc. IEEE Globecom*, 1995, pp. 1379-1383.
- [6] Taishi Yahara, Ryutaro Kawamura, Satoru Ohta, "New self healing scheme that realizes differentiated bandwidth requirements on ATM networks", *IEICE Trans. Comm.*, Vol. E83-B, No. 3,2000, pp. 672-679.
- [7] Eric W.M.Wong, Andy K.M.Chan, Sammy Chan, King Tim Ko, "Bandwidth allocation for virtual paths in ATM Networks with dynamic routing", *IEICE Trans. Comm*, Vol. E83-B, No. 3,2000, pp. 626-637.
- [8] S.S. Manvi and P. Venkataram, "QoS management by mobile agents in multimedia communication", *Proc. DEXA (Workshop on ABIS)*, Sep. 2000, pp. 407—411.
- [9] S.S. Manvi and P. Venkataram, "Mobile agent based online bandwidth allocation for multimedia communication", *IEEE Globecom*, Nov. 2001, pp. 2622-2625.
- [10] Steffen Liperts and Birgit Kreller, "Mobile agents in telecommunications -- A simulative approach to load balancing", *Proc. ISAS*, 1999.
- [11] <http://www.cetus-links.org/oo-mobile-agents.html>.
- [12] <http://www.cs.umbc.edu/agents>