Bandwidth Fragmentation Avoided QoS Multicast Routing by Employing Admission Control

Bo Rong, Maria Bennani, Michel Kadoch
LAGRIT, Department of Electrical Engineering
Ecole de technologie superieure, Universite du Quebec
Montreal, Quebec, Canada, H3C 1K3

Ahmed K. Elhakeem
Department of Electrical and Computer Engineering
Concordia University
Montreal, Quebec, Canada, H3G 1M8

Abstract—**The rapid growth of multimedia group-based applications motivates research into QoS guaranteed multicast network. In the multicast network carrying traffics of different QoS requirements, bandwidth fragmentation is considered as a bottleneck for traffic admission fairness and efficient usage of bandwidth resource. When the bandwidth fragmentation happens, the chance of accepting high-bandwidth flows is greatly decreased. To prevent bandwidth fragmentation in multicast network, we propose a new approach of bandwidth fragmentation avoided QoS multicast routing in this paper. In our new approach, we integrate active admission control into traditional QoS multicast routing algorithms to gain bandwidth fragmentation immunity. Moreover, an algorithm named dynamic bandwidth allocation with adaptive constraint is proposed and investigated as an example of active admission control algorithm in this paper. Because this algorithm is nonlinear and unsolvable by analytical approach, we employ OPNET simulation to study its performance.**

Keywords-Bandwidth fragmentation, QoS multicast routing, active admission control

I. INTRODUCTION

Multicast provides an efficient way to transmit data from a sender to a group of receivers. For the multicast network that can provide QoS guaranteed service, avoiding bandwidth fragmentation and achieving traffic admission fairness have attracted more and more attention recently. In this paper, we concentrate on a new approach of integrating QoS multicast routing with active admission control to offer bandwidth fragmentation immunity.

Previous work on QoS multicast routing mainly concerned about how to develop algorithms of finding a cost optimal tree with certain QoS constraints [1,2,3,4,5]. For these algorithms, the admission control is only considered as a by-product of QoS routing and resource reservation. If the routing algorithm can find a route meeting the QoS requirements and the resource reservation is successfully done along the selected route, the connection request is accepted; otherwise, the request is rejected. The motivation of this paper is to propose a new approach of employing active admission control as bandwidth fragmentation immunizing mechanism in QoS multicast routing. By this way, the admission control is no longer a by-product of routing algorithm, it can affect the routing result significantly. This new approach is based on the preprocessing of the network graph used by traditional QoS multicast routing algorithm. First of all, a set of active links is selected in the original network graph beforehand. Other links, which are not selected as active links, are defined as non-active links. Second of all, when a new connection request comes, the original network graph is preprocessed by using an active admission control algorithm only on active links (As for non-active links, no action is taken on them.). At last, the preprocessed network graph is utilized as input for the traditional QoS routing algorithm to find a QoS guaranteed route.

The rest of the paper is organized as follows. Firstly, we discuss the integration of QoS multicast routing and active admission control for bandwidth fragmentation immunity in Section 2. Then an active admission control algorithm named dynamic bandwidth allocation with adaptive constraint is proposed in Section 3, and the performance of this algorithm is studied in Section 4. In the end, Section 5 summarizes our results.

II. INTEGRATING QOS MULTICAST ROUTING WITH ADMISSION CONTROL FOR BANDWIDTH FRAGMENTATION IMMUNITY

A. QoS Multicast Routing for Real-Time Connections

A network can be represented by a connected graph $G(V,E)$ with weights associated with edges. In the graph, the nodes stand for communication endpoints, the edges stand for communication links, and the weight on each edge $\{W_l, l\in E\}$ represents the cost of that link. A simple network represented by graph is shown in Fig.1. By treating a network as graph, a multicast session can be described as $M=(s,D)$ where s is the source node, $D=\{d_1,\ldots,d_n\}$ is a set of destination nodes. The multicast tree for M is a subtree of $G(V,E)$ rooted from s, which contains all the nodes of D as well as an arbitrary subset of $(V\text{-}D)$, and whose leaf set consists only of a subset of the nodes from D. Along multicast tree, multicast packets should be transmitted simultaneously from s to all the destinations. Parallel transmission can reduce the delay time to send messages to all recipients. Moreover, a minimal number of message copies are transmitted by copying messages only at forks in the multicast tree and it is a good way to reduce the network traffic load.

Work supported by Bell Canada and NSERC grants.

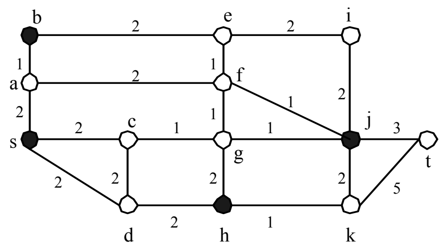

Fig. 1. An example of a network graph (s is the source node; $\{b, h, j\}$ are the destination nodes.)

Let T be a multicast tree of multicast session M, we define the cost of T as follows:

$$COST_T = \sum_{l \in T} W_l \tag{1}$$

$COST_T$ decides the network cost totally used by M. One important aim of multicast routing algorithms is to find out the tree with least cost which can save the consumption of multicast connection greatly. Finding a least cost tree in the weighted graph is regarded as Steiner tree problem[6]. Because the Steiner tree problem has been proven to be NP-complete, a lot of heuristic algorithms are proposed in previous research, most of them are centralized [7,8].

To set up real-time multicast connections, more issues have to be taken into consideration. Most real-time audio or video applications require quite stringent quality of service to provide smooth play-out at the receiver. For these applications, we can define the end-to-end QoS constraints of a multicast connection as $Q(s, D)$, where s is the source node, D is a set of destination nodes. The Steiner tree problem can be extended to constrained Steiner tree problem by including all kinds of QoS constraints, such as bandwidth, end-to-end delay, delay jitter, and so on. Under this circumstance, W_l (the weight of link l) in Fig.1 should be changed to $W_l^{QoS}=(W_l^c, W_l^b, W_l^d)$, where W_l^c stands for the cost of link l, W_l^b stands for the remaining free bandwidth on link l, and W_l^d stands for the transmission delay of link l. By using W_l^{QoS}, the network graph in Fig.1 is redrawn in Fig.2 where the units of W_l^b and W_l^d are Mbps and Millisecond respectively. The problem of constrained Steiner tree is also NP-complete, and corresponding heuristic algorithms can be found in [1,2,3,4,5]. These heuristic algorithms serve as the foundation for QoS multicast routing protocols. In general, all these heuristic algorithms can be classified into two categories, the centralized algorithms and the distributed algorithms. Until now, most published algorithms belong to centralized category. Therefore, in this paper, we only concern the centralized QoS multicast routing algorithms.

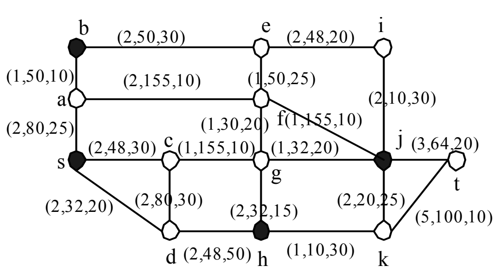

Fig. 2. The extended network graph for QoS multicast routing

B. Active Admission Control as Bandwidth Fragmentation Immunizing Mechanism in QoS Multicast Routing

Although we can use existed QoS multicast routing algorithms to find a QoS guaranteed path for a real-time multicast connection, the routing algorithms do not ensure that the network can run effectively. For instance, when the bandwidth fragmentation happens, all the connections of low bandwidth requirement can access the network easily while most connections of high bandwidth requirement can not. Therefore, in QoS guaranteed multicast network, it is urgent to integrate bandwidth fragmentation immunizing mechanism into QoS multicast routing.

Here, we propose a new approach of employing active admission control in QoS multicast routing. In this paper, admission control algorithms are classified to two categories, the passive ones and the active ones. The passive admission control algorithm uses a policy of FCFS, which treats all connection requests equally. On the other hand, the active admission control algorithm employs specific policy to control resource allocation actively, and it usually shows different priorities to connection requests of different QoS requirements. In this paper, the active admission control algorithm is mainly employed to control the bandwidth resource allocation on some selected links, and connection requests are treated differently only according to their bandwidth requirement. For convenience, we define connection requests with high bandwidth requirement as high-bandwidth connection requests, and connection requests with low bandwidth requirement as low-bandwidth connection requests. An active admission control algorithm named dynamic bandwidth allocation with adaptive constraint, which can give preference to high-bandwidth connections and limit the acceptance of low-bandwidth connections, will be studied in Section 3 in terms of bandwidth fragmentation immunity.

In order to integrate active admission control into QoS multicast routing, once a new connection request (low-bandwidth connection request or high-bandwidth connection request) comes, we use active admission control algorithm to preprocess the network graph before QoS multicast routing algorithm is started to find a QoS route. As a preparation for network graph preprocessing, some links are chosen from the original network graph as active links in advance. Other links, which are not selected as active links, are defined as non-active links. The good candidate for active link is the one who has high bandwidth capacity and plays a significant role in the network topology. For example, in Fig.2, we choose {(a,f), (c,g), (f,j)} as active links. The number of selected active links can be adjusted regarding the amount of potential high-bandwidth traffic in the network. When a new connection request comes, we use active admission control algorithm to make bandwidth admission test only on these active links (As for non-active links, no action is taken on them.). If the admission control algorithm decides that there is no available bandwidth resource on some active links for the new coming connection request, these inadequate active links are deleted from the original network graph. We define the new obtained network graph as preprocessed network graph, in which some active links may disappear while all non-active links are kept.

Only after getting the preprocessed network graph for a connection request, the QoS multicast routing algorithm is started to find a QoS route that can meet all the QoS requirements of this connection request.

To achieve bandwidth fragmentation immunity, the active admission control algorithms with preference to high-bandwidth connections should be developed for active links. By employing this kind of algorithms, connection requests of different bandwidth requirements may have different preprocessed network graphs. As a result, the traffic of high-bandwidth connections is mainly aggregated on the high capacity active links, while the traffic of low-bandwidth connections is distributed evenly in the whole network. Furthermore, with this approach, both high-bandwidth connections and low-bandwidth connections can be accepted fairly in the whole network. Although high-bandwidth connections have priority on active links, low-bandwidth connections are more competitive on non-active links as we mentioned in [9].

III. Our Active Admission Control Algorithm for Bandwidth Fragmentation Immunity

In this section, we propose the algorithm of dynamic bandwidth allocation with adaptive constraint, which can give preference to high-bandwidth connections and limit the acceptance of low-bandwidth connections. Our algorithm is suitable to be employed in network graph preprocessing for the purpose of bandwidth fragmentation immunity. The basic architecture of this algorithm is shown in Fig.3.

In this algorithm, on active link l, when the i^{th} connection request R_i with bandwidth requirement B_i arrives, the classifier should put it into one of different groups according to the value of B_i. Consider the case where bandwidth requirement of connection requests in the network can be classified by the values $b_M>\ldots> b_1>b_0=0$ (b_M is the maximum value of B_i permitted in the network), each value corresponding to a bandwidth level k (k=1,...,M). If $b_{k-1}< B_i \le b_k$, then the classifier put R_i into group k, which can be denoted by R_i , $B_i \in G_k$. Simply, a request belonging to G_k can be called a G_k request. After carefully choosing parameters such as M and b_k, we consider G_M as the group of highest-bandwidth connections while G_1 as the group of lowest-bandwidth connections. If R_i is accepted by the admission control algorithm on link l in network graph preprocessing phase, link l will be involved in the preprocessed network graph. If link l could also be chosen as a part of the QoS route for R_i, then the corresponding bandwidth resource should be allocated on link l and R_i begins to transmit data. We define the beginning time and ending time for R_i to be served on link l as $T_B(i)$ and $T_E(i)$ respectively. Then, we say R_i or B_i is alive at time $t(t>0)$, if $T_B(i)< t< T_E(i)$.

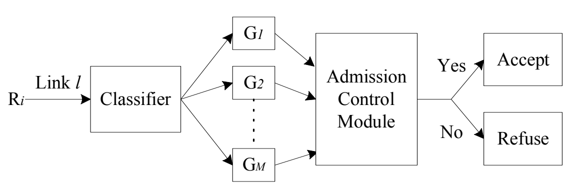

Fig. 3. The basic architecture of our algorithm

Let C_l be the whole bandwidth capacity of link l, we divide C_l into two parts, presented by $C_l=C_q+C_b$. Here, C_q stands for the bandwidth capacity of QoS guaranteed service controlled by our active admission control algorithm. C_b represents the bandwidth capacity dedicated to best-effort service which is used to support applications without QoS requirements. After the above definitions, we intend to introduce our active admission control algorithm aiming at showing preference to high-bandwidth connections. Different from the existed literature [10][11][12][13], in this paper we investigate the algorithm which cannot be studied by analytical approach. Furthermore, compared with the algorithms in [12][13], we not only concern about how to allocate resources properly according to special purposes, but also take the bandwidth utility efficiency into account to assess the performance of our algorithm. The algorithm of dynamic bandwidth allocation with adaptive constraint will be introduced in a step by step way. After discussing some basic admission control algorithms, our algorithm will be studied as an improvement to these basic algorithms.

(1) FCFS

FCFS is the simplest admission control policy, which means to accept any new coming connection request regardless of its priority type, as long as there are enough resources to accommodate it. For FCFS, only the following formula needs to be satisfied to make admission decision:

$$\sum_{\text{All alive } B_i} B_i \le C_q \tag{2}$$

Because FCFS has high bandwidth utility efficiency and is easy to be implemented, it is widely employed in existed routers and switches. However, one major drawback of FCFS is that this policy cannot show any preference to high-bandwidth connections in respect to the objective of this paper.

(2) Fixed Bandwidth Allocation

Under the fixed bandwidth allocation policy, different groups are given different fixed bandwidth resources respectively. When the bandwidth resources allocated to one group is exhausted, a new coming request of this group will be rejected by the server even though there are still available bandwidth resources to other groups. If C_q stands for the bandwidth capacity of QoS guaranteed service on link l and BF_k denotes the fixed bandwidth resources allocated to group G_k, the following formula holds:

$$\sum_{k=1}^{M} BF_k = Cq \tag{3}$$

Moreover, the formula employed to make the admission decision is shown below:

$$\sum_{\text{All alive } B_i \in G_k} B_i \le BF_k, \quad k = 1,\ldots,M \tag{4}$$

This algorithm can keep different groups isolated from each other and protect the acceptance of high-bandwidth connections from being damaged by the competition of low-bandwidth connections. Nevertheless, if the request arrival rates of these groups are different from the expected arrival rates, the bandwidth utility efficiency of link l is not very well.

(3) Dynamic Bandwidth Allocation with Adaptive Constraint

Compared with fixed bandwidth allocation policy, besides some fixed bandwidth resources to each group, the additional shared bandwidth resources C_q^{sh} is allocated for all the groups in this algorithm. This shared bandwidth resource policy can be described by the formula below:

$$C_q^{sh} + \sum_{k=1}^{M} BF_k = Cq \tag{5}$$

Moreover, in order to show preference to high-bandwidth connections, C_q^{sh} has to be shared by all groups under an adaptive constraint policy. Let U_k(t) denote the upper bound of the total alive bandwidth that G_k requests can get from C_q^{sh} at time t (t>0), then the total alive bandwidth allocated from C_q^{sh} to group G_k has to be less than U_k(t) at any time t. After this, the adaptive constraint policy can be defined as the following:

$$U_k(t) = \frac{x\left[\sum_{j=k+1}^{M} W_{k,j}\ SLB_{sh}^{j}(t)\right]}{(M-k)} \quad , \quad k = 1,\ldots,M-1 \tag{6}$$

$$U_M(t) = C_q^{sh} \tag{7}$$

where, $SLB_{sh}^{j}(t)$ is the sum of alive bandwidth occupied by group G_j in C_q^{sh} at time t, x is the upper bound coefficient and $W_{k,j}$ is the relative weight. In this algorithm, the upper bound coefficient x is designed to describe how much U_k(t) is restricted by all the connection groups whose bandwidth level is higher than k. In addition, the relative weight $W_{k,j}$ is used to control the extent that U_k(t) is restricted by a certain connection group G_j (j>k).

After these definitions, we follow the flowchart in Fig.4 to make admission decision. Because U_k(t) changes adaptively with $SLB_{sh}^{j}(t)$, this algorithm is unable to be investigated by analytical approach according to the theory of [10][11].

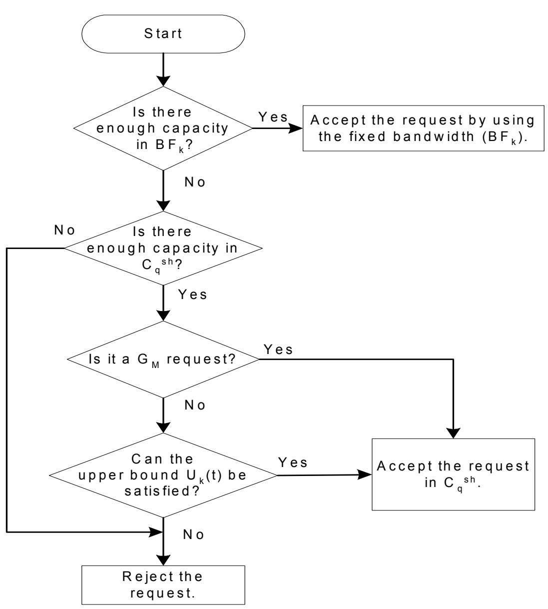

Fig. 4. Flowchart of admission decision (the algorithm of dynamic bandwidth allocation with adaptive constraint)

Due to the introduction of C_q^{sh} and U_k(t), this algorithm can offer more adaptation to traffic variance. Firstly, some fixed bandwidth resources allocated to each group guarantee the basic acceptance of a certain number of connection requests for each group. Secondly, the shared bandwidth resource C_q^{sh} is introduced to increase the flexibility and bandwidth utility efficiency on link l in case of frequently changing request arrival rates. Thirdly, in order to give high-bandwidth connection groups higher priority in C_q^{sh} area, the adaptive constraint is added. Furthermore, for the sake of real efficiency, we have to choose appropriate value for each parameter of this algorithm according to the need of a certain network.

IV. Simulation Results of Our Active Admission Control Algorithm

In this section, the numerical results are presented to demonstrate the performance of our active admission control algorithm. Although analytical approach can be applied to study some active admission control algorithms, it quickly becomes too complicated and intractable when the source model and the admission control algorithm become complex. As a result, we use OPNET Modeler to study the algorithm of dynamic bandwidth allocation with adaptive constraint, which cannot be investigated by analytical approach.

We suppose there are 4 groups of connection requests on link l, and in each group there are different bandwidth requirement classes.

(a)group G_1: 56Kbps (class1);
(b)group G_2: 200Kbps(class1), 500Kbps(class2);
(c)group G_3: 1.5Mbps(class1);
(d)group G_4: 4Mbps(class1), 6Mbps(class2).

Obviously, G_1 and G_2 are groups of low-bandwidth connection requests, while G_3 is the medium-bandwidth connection group and G_4 is high-bandwidth connection group. Moreover, requests of each class in every group are assumed to arrive according to Poisson process independently, and the service times are exponentially distributed. The parameters used in this model are defined below:

C_q(Mbps)**:** bandwidth capacity for QoS guaranteed service on link l.
C_q^{sh}(Mbps)**:** The shared bandwidth capacity for all groups.
BF_k(Mbps)**:** Fixed bandwidth resources allocated to G_k.
x**:** Upper bound coefficient in the algorithm of dynamic bandwidth allocation with adaptive constraint.
$W_{k,j}$**:** Relative weight in the algorithm of dynamic bandwidth allocation with adaptive constraint.
$\lambda_{k,m}$(calls/hour)**:**average arrival rate of class m requests in G_k.
$1/\mu_{k,m}$(hours/call)**:**average service time for class m requests in G_k.
Pb_k**:** blocking probability for G_k requests.
$Thrput_k$(Mbps)**:** throughput for G_k requests.
Eff**:** the bandwidth utility efficiency of a certain algorithm computed by $Eff=(Thrput_1+Thrput_2+Thrput_3+Thrput_4)/C_q$.

With OPNET simulation, numerical results are calculated to illustrate the performance of our algorithm named dynamic bandwidth allocation with adaptive constraint. The first three numerical results are shown in Fig.5, Fig.6 and Fig.7 to demonstrate the blocking probabilities, throughputs and bandwidth utility efficiency of our algorithm under the condition of varying the constraint (upper bound coefficient x) while keeping other parameters constant. The parameters of our simulation are set as the following:

C_q=500Mbps, C_q^{sh}=250Mbps,
$BF_1=5\%*(C_q-C_q^{sh})$,$BF_2=30\%*(C_q-C_q^{sh})$,
$BF_3=35\%*(C_q-C_q^{sh})$, $BF_4=30\%*(C_q-C_q^{sh})$, x=25%~300%,
$W_{k,j}$=1 (any k or j), $1/\mu_{k,m}$=1hour/call (any k or m),
$\lambda_{1,1}$(56Kbps)=1200(calls/hour),$\lambda_{2,1}$(200Kbps)=600(calls/hour),
$\lambda_{2,2}$(500Kbps)=400(calls/hour), $\lambda_{3,1}$(1.5Mbps)=60(calls/hour),
$\lambda_{4,1}$(4Mbps)=39(calls/hour), $\lambda_{4,2}$(6Mbps)=16(calls/hour).

From Fig.5 and Fig.6, we can conclude that when upper bound coefficient x takes a small value, the group of high-bandwidth connection requests can be protected very well. On the other hand, Fig.7 shows that when upper bound coefficient x takes a high value, a good bandwidth utility efficiency can be achieved. Therefore, the upper bound coefficient x is crucial to both protecting high-bandwidth connections and improving bandwidth utility efficiency, a medium value should be chosen to compromise these two effects.

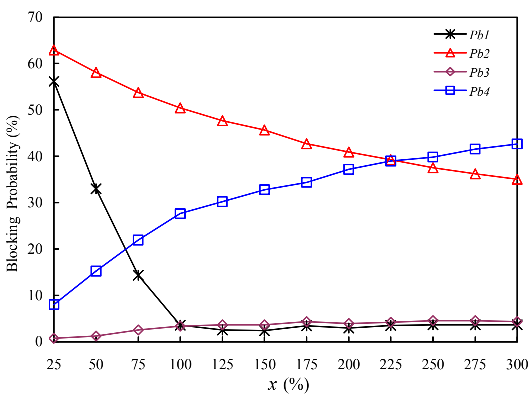

Fig. 5. Blocking probability Pb_k versus upper bound coefficient x (the algorithm of dynamic bandwidth allocation with adaptive constraint)

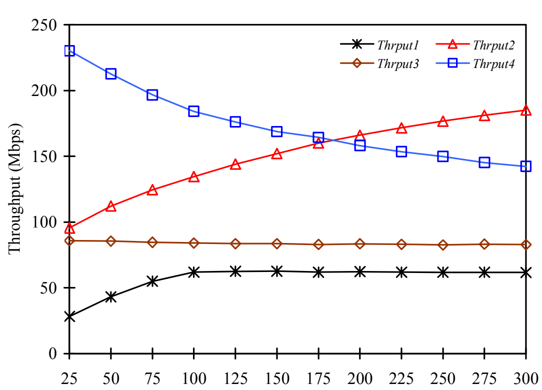

Fig. 6. Throughput $Thrput_k$ versus upper bound coefficient x (the algorithm of dynamic bandwidth allocation with adaptive constraint)

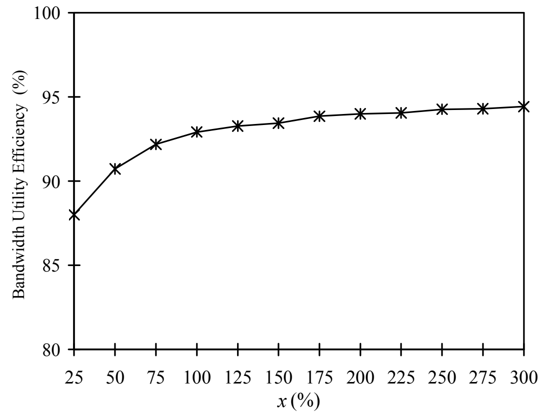

Fig. 7. Bandwidth utility efficiency Eff versus upper bound coefficient x (the algorithm of dynamic bandwidth allocation with adaptive constraint)

More numerical results are shown from Fig.8 to Fig.11. Fig.8, Fig.9 and Fig.10 demonstrate the blocking probabilities obtained by the algorithms of dynamic bandwidth allocation with adaptive constraint, fixed bandwidth allocation and FCFS respectively, while Fig.11 shows the efficiencies of all three algorithms. All these four figures try to get the numerical results under the condition of varying the arrival rate $\lambda_{2,2}$ while other parameters are held constant. The parameters are set as the following:

Common part for all three algorithms:
C_q=500Mbps, $1/\mu_{k,m}$=1hour/call (any k or m),
$\lambda_{1,1}$(56Kbps)=1200(calls/hour), $\lambda_{2,1}$(200Kbps)=300(calls/hour),
$\lambda_{2,2}$(500Kbps)=100~500(calls/hour), $\lambda_{3,1}$(1.5Mbps)=60(calls/hour),
$\lambda_{4,1}$(4Mbps)=16(calls/hour), $\lambda_{4,2}$(6Mbps)=12(calls/hour).

Special part for the algorithm of dynamic bandwidth allocation with adaptive constraint:
C_q^{sh}=250Mbps, $BF_1=5\%*(C_q-C_q^{sh})$, $BF_2=30\%*(C_q-C_q^{sh})$,
$BF_3=35\%*(C_q-C_q^{sh})$, $BF_4=30\%*(C_q-C_q^{sh})$, x=50%, $W_{k,j}$=1 (any k or j).

Special part for the algorithm of fixed bandwidth allocation:
$BF_1=5\%*C_q$, $BF_2=30\%*C_q$, $BF_3=35\%*C_q$, $BF_4=30\%*C_q$.

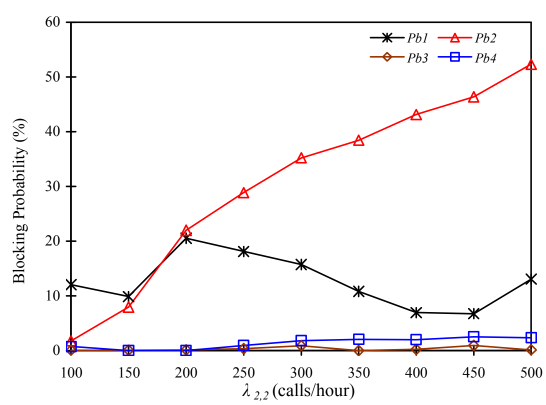

Fig. 8. Blocking probability Pb_k versus arrival rate $\lambda_{2,2}$ (the algorithm of dynamic bandwidth allocation with adaptive constraint)

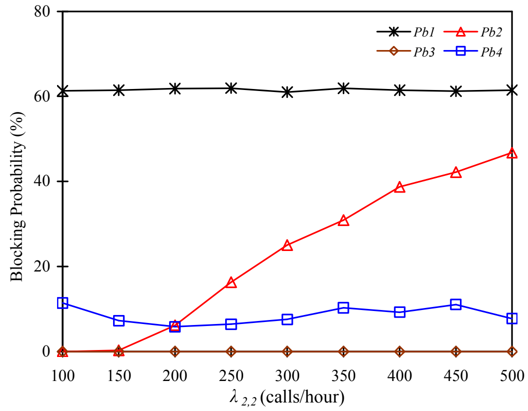

Fig. 9. Blocking probability Pb_k versus arrival rate $\lambda_{2,2}$ (the algorithm of fixed bandwidth allocation)

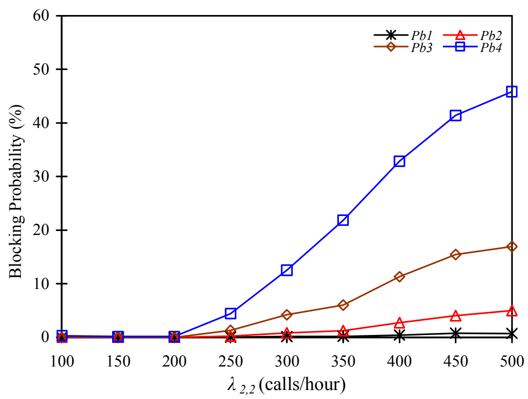

Fig. 10. Blocking probability Pb_k versus arrival rate $\lambda_{2,2}$ (FCFS)

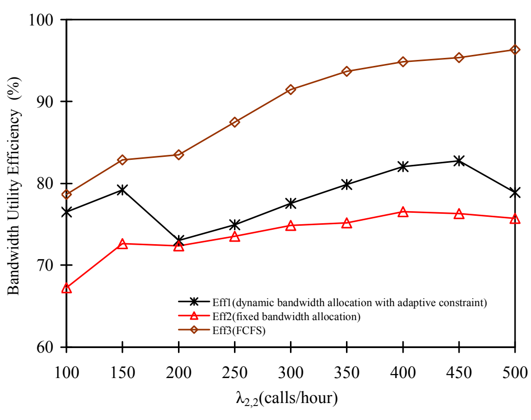

Fig. 11. Bandwidth utility efficiency *Eff* versus arrival rate $\lambda_{2,2}$ (the results of all three algorithms)

From Fig.8 to Fig.11, it can be seen clearly that the algorithm of dynamic bandwidth allocation with adaptive constraint has the best performance among all three algorithms. Although FCFS has the highest bandwidth utility efficiency, it cannot offer any protection for high-bandwidth connection requests. The algorithm of fixed bandwidth allocation can isolate different groups of requests very well, but its bandwidth utility efficiency is low. The algorithm of dynamic bandwidth allocation with adaptive constraint not only can show the preference to high-bandwidth connection requests, but also has higher bandwidth utility efficiency than the fixed bandwidth allocation algorithm. In a word, the algorithm of dynamic bandwidth allocation with adaptive constraint is a good candidate for network graph preprocessing in terms of bandwidth fragmentation immunity.

V. Conclusions

To avoid bandwidth fragmentation in the IP multicast network is an urgent task for ISPs. In this paper, a new approach of integrating QoS multicast routing with active admission control is studied to solve this problem. Furthermore, an active admission control algorithm named dynamic bandwidth allocation with adaptive constraint is proposed and investigated to offer bandwidth fragmentation immunity while keeping high bandwidth utility efficiency. From the simulation results, we can conclude that the algorithm of dynamic bandwidth allocation with adaptive constraint can control traffics of different bandwidth requirements effectively, and fulfil bandwidth fragmentation immunizing task successfully.

References

[1] V. P. Kompella, J. C. Pasquale, and G. C. Polyzo, “Multicast routing for multimedia communication,” *IEEE/ACM Transactions on Networking*, pages 286-292, June 1993.

[2] V. P. Kompella, J. C. Pasquale, and G. C. Polyzo, “Two distributed algorithms for multicasting multimedia information,” *Proceedings of ICCCN’93*, pages 343-349, 1993.

[3] Q. Zhu, M. Parsa, and J. J. Garcia-Luna-Aceves, “A source-based algorithm for delay-constrained minimum-cost multicasting,” *Proc. of IEEE INFOCOM’95*, pages 377-385, 1995.

[4] G. N. Rouskas and I. Baldine, “Multicast routing with end-to-end delay and delay variation constraints,” *IEEE Journal on Selected Areas in Communications*, pages 346-356, April 1997.

[5] X. Jia, “A distributed algorithm of delay bounded multicast routing for multimedia applications in wide area networks,” *IEEE/ACM Transactions on Networking*, pages 828-837, December 1998.

[6] Gilbert and H. O. Pollack, “Steiner minimal tree,” SIAM J. Appl. Math., Vol.16, 1968.

[7] L. Kou, G. Markowsky, and L. Berman, “A fast algorithm for Steiner trees,” *Acta Informatica*, pages 141-145, 1981.

[8] H. Takahashi and A. Matsuyama, “An approximate solution for the Steiner problem in graphs,” *Math. Japonica*, pages 573-577, 1980.

[9] Bo Rong, M. A. Breton, M. Bennani, and M. Kadoch, "Simulation of Active Admission Control Algorithms with OPNET," *OPNETWORK* 2003, Session 1559, August 2003, Washington D.C.

[10] J. M. Aein, “A multi-user-class, blocked-calls-cleared demand access mode,” *IEEE Transactions on Communications*, vol. COM-26, pages 378–385, Mar. 1978

[11] E. Arthurs and J. S. Kaufman, “Sizing a message store subject to blocking criteria,” in *Performance of Computer Systems*, pages 547–564, 1979.

[12] J. S. Kaufman, “Blocking in a shared resource environment,” *IEEE Transactions on Communications*, pages 1474-1481, October 1981.

[13] P. Mundur, A. Sood, and R. Simon, “Threshold-based admission control for multi-class Video-on-Demand systems,” In *Proceedings of the IEEE IPCC*C, pages 154–160, February 1998.