Preemptive Bandwidth Allocation Protocol for Multicast, Multi-Streams Environments

Nawel Chefaï
SITE, University of Ottawa
161 Louis Pasteur
Ottawa, On. K1 N 6N5
(613) 562-5800 ext: 6248
chefai@mcrlab.uottawa.ca

Nicolas D. Georganas
SITE, University of Ottawa
161 Louis Pasteur
Ottawa, On. K1 N 6N5
(613) 562-5800 ext: 6225
georganas@mcrlab.uottawa.ca

Gregor V. Bochmann
SITE, University of Ottawa
161 Louis Pasteur
Ottawa, On. K1 N 6N5
(613) 562-5800 ext: 6205
bochmann@site.uottawa.ca

ABSTRACT
In this paper, we present a protocol that allocates resources in communication networks in order to assure specific QoS characteristics as requested by new connections. The design takes into consideration the possibility for the network allocation to adapt to application requirements.

The proposed protocol uses a Bandwidth Preemptive Algorithm that permits adaptive bandwidth allocation in multicast, multi-stream environments. This design has been inspired by the one proposed by *Sakate* [I] where a centralized methodology is used. In our approach, we use a distributed methodology where we change the behavior of the communication service and allow the continuation of the service under more severe conditions. In other words, when there is a lack of bandwidth for a new connection, the communication service will try to find the missing bandwidth within the existent connections (or streams) when looking for a feasible path on a hop-by-hop basis, starting from the destination to an a on-tree node.

1. INTRODUCTION
Multicast routing is the process of finding a routing tree, which is rooted from a single source to all the destinations. Multicast routing protocols are responsible for creating multicast packet delivery trees and for performing multicast forwarding. Several multicast routing protocols exist and can be classified in two categories: source-based and shared-based multicast trees. A *source-based multicast tree* constructs the multicast tree starting at the root to reach all the destinations. A *shared multicast tree* is a mode where a "meeting place", called a *core* or *Rendezvous Point* (RP), is advertised for each multicast group, toward which sources send initial packets and receivers send explicit join messages. The above-mentioned multicast routing protocols construct only the shortest paths between the source/core and the receivers of a given multicast group without considering users' QoS requirements. Other protocols such as QoSMIC [2] and QMRP [3] have considered QoS routing by finding a feasible path from a new user to the tree that satisfies the user's QoS requirements.

For instance, QoSMIC uses flooding to investigate all possible paths from the new user to the tree. QMRP avoids the systematic use of flooding. Instead, it tries to find a path from the new user to the core of the tree; if the path does not offer sufficient QoS to the new user, flooding will be used starting from the node where the requirements couldn't be met.

In our approach to the problem of finding a feasible path to the multicast tree, we do not usc flooding but consider that we have more than one RP per multicast group. In fact, such a scheme is already used by Cisco Systems [4] where a given multicast group is characterized by more than one RP. Their goal is to split the load among different RPs and arrange RPs according to the location of group participants. In short, our protocol takes advantage of this scheme when looking for a feasible path. This approach permits a more explicit join towards the tree that avoids flooding. On the other hand, it avoids the problems related to a centralized point such as the one point of failure, distant RP dependencies, congestion towards the RP and long delays.

In this paper, we consider the problem of allocating resources in communication networks in order to assure specific QoS characteristics as requested by new connections; it is within an integrated service scheme where a per-flow treatment is needed. The design takes into consideration the possibility for network allocation to adapt to application requirements. The main function of our proposed protocol is to preempt bandwidth if necessary from the current streams along the candidate path(s) in order to meet the minimum requirements of the new connections. This preemption is done in a decentralized manner where preemption policies are applied on a hop-by-hop basis along the candidate path(s) starting from the receivers to the RP(s). For the preemption to be possible, we consider each unit of bandwidth as a separate entity as it is done in [I]. We believe that this approach introduces more fairness between the users since we do not let those users with higher priorities take over the resources within the network.

In the rest of the paper, we will describe the overall model and give details of the Bandwidth Preemption Algorithm that this protocol uses on each link along a given candidate path [1].

MM'01, Sept. 30-Oct. 5, 2001, Ottawa, Canada.

[1] The system-wide deployment is not detailed due to lack of space.

2. THE MODEL DESCRIPTION

To facilitate preemption, end-users should describe their needs by specifying minimum and maximum requirements in addition to a priority level. This priority level can either be negotiated at the beginning of the session (with the service provider, for example) or can initially be set to the lowest value.

We consider the situation where the multicast tree evolves dynamically, depending on end-user requirements. The tree construction is based on the shared tree paradigm in the sense that the receivers do not get their connection from the source but from an intermediate node in the tree, called the Rendezvous Point (RP) (fig.1). Furthermore, each new receiver will get connected to the multicast tree gradually; from the point of view of the routing problem, we will consider each new request for a connection as a problem of resolving a unicast QoS routing.

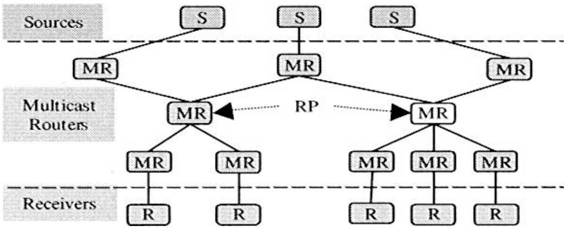

Figure 1: A shared-tree with multiple Rendezvous Points.

Our approach to the multicast routing problem is the link-constrained problem. A link constraint, which consists of bandwidth availability on a given link, is imposed to construct a feasible path from a source or Rendezvous Point (RP) to a receiver. Our protocol proposes a distributed model that looks for a feasible path by applying some admission controls on a hop-by-hop basis. The section below describes the algorithm that the protocol deploys along each candidate path.

3. THE PREEMPTIVE ALGORITHM

On each link that belongs to a given candidate path, two admissions controls will be applied to verify if enough bandwidth is available to satisfy a new user's minimum requirements.

In the following sections, we will describe how the two admission controls are held on each link when looking for a feasible and how the problem is formulated when there is a necessity for bandwidth preemption.

3.1 First Admission Condition

The first admission condition will check if there is enough available bandwidth on a given link along a given candidate path that satisfies a new user's minimum bandwidth requirements. This condition is expressed by a mean of comparing user R_n's minimum bandwidth requirements, $Bmin(R_n)$, to bandwidth availability, $Bavailable(L_j)$, of each link L_j that belongs to a given path (this set of links is denoted $link\_set(R_n)$). The link L_j's available bandwidth, $Bavailable(L_j)$, is equal to the unused bandwidth on this link, $Bunused(L_j)$, added with $Cum\_prep(L_j)$, the marked preemptive bandwidth from those streams that have traversed the previous link and are still traversing the current link and from where some bandwidth has been marked preemptable. The first admission condition is then expressed as follows:

$$\forall L_j \in link\_set(R_n)\ [Bmin(R_n) \leq Bavailable(L_j)] \qquad (1)$$

Where $Bavailable(L_j) = Bunused(L_j) + Cum\_prep(L_j)$

The calculation of $Cum\_prep(L_j)$ necessitates the knowledge of the m-1 streams, denoted $St_{i(j-1)}$, $1 \leq i \leq m\text{-}1$, that are traversing the previous link L_{j-1} and from where some bandwidth $\delta_{i(j-1)}$ has been marked preemptable. This set, denoted by $Mark\_prep(L_{j-1})$, is defined as follows:

$$Mark\_prep(L_{j-1}) = \{St_{i(j-1)}\ from\ where\ \delta_{i(j-1)}\ bandwidth\ is\ marked\ preemptable,\ 1 \leq i \leq m\text{-}1\}$$

Therefore, as specified in equation (I), the link L_j on the path will have to include the already marked preemptable bandwidth from the previous link to the already unused bandwidth. This statement is expressed by the following two equations:

$$Mark\_Prep(L_j) = Mark\_prep(L_{j-1}) \cap \{St_{kj}, 1 \leq k \leq m\text{-}1\}$$

$$Cum\_prep(L_j) = \Sigma\, \delta_{i(j-1)},\ 1 \leq i \leq m\text{-}1$$

Where $St_{i(j-1)} \subset Mark\_prep(L_j)$

if this first admission condition does not succeed, meaning that not enough bandwidth is available on the current link, the second admission control will have to be checked.

3.2 Second Admission Condition

If not enough bandwidth is found on a given link, a second admission control is applied. This admission control will check if there is any possible bandwidth preemption from other streams that traverse the link and that can satisfy the user's minimum bandwidth requirements. This minimum bandwidth requirements will be compared to the link's available bandwidth, $Bavailable(L_j)$, added with the link L_j's maximum preemptable bandwidth, $Max\_prep(L_j)$. This condition can be expressed as follows:

$$\forall L_j \in link\_set(R_n)\ [Bmin(R_n) \leq Bunused\_prep(L_j)] \qquad (2)$$

Where $Bunused\_prep(L_j) = Bavailable(L_j) + Max\_prep(L_j)$

If the second admission condition succeeds, the minimum flow cost problem [5] will be used to calculate the amount of preemptable bandwidth that each stream has to release in order to: (1) satisfy the user's minimum requirements; and (2) minimize the total amount of the quality loss (see sub-section 3.4 for the problem formulation). If the second admission control does not hold, it means that the path cannot offer the user's minimum bandwidth requirements even when using preemption. In this case, the search is aborted on this specific candidate path.

3.3 Quality and Priority Requirement

In *Sakate et al.* [I], they allow the user to separate the bandwidth quality in such way that each unit of bandwidth corresponds to a quality value. When a unit of the range denoted *bw* is preempted from a stream St, , independently of any link, the corresponding value, represented by the function $D_i(bw)$ below, will indicate the amount of quality value that has been lost.

The following function specifies different bandwidth range with their associated values related to the stream St_i:

$$D_i(bw) = \begin{cases} d_1 & (p_0 \leq bw < p_1) \\ d_2 & (p_1 \leq bw < p_2) \\ \vdots & \\ d_{z-1} & (p_{z-1} \leq bw \leq p_z) \end{cases}$$

Where $p_0 = Bmin(St_i)$, $p_z = Bmax(St_i)$ and

$\forall$ s $(0 \leq s \leq z\text{-}1)$ $[d_{s-1} \geq d_s \geq 0]$

In addition to this quality value associated to different bandwidth range, another parameter can be added to let the user associate a priority to each stream. We denote this value, associated to a given stream St,, as W_i. For instance, for a stream St_i to have a higher priority than another stream St_{i+1}, the priority W_i associated to the stream St_i should have a higher value to the priority W_{i+1} associated to the stream St_{i+1}.

3.4 Problem Formulation

The second admission control expressed in (2) specifies the necessity of looking for preemptable bandwidth that can be released by other streams that traverse the link L_j to satisfy new user R_n's minimum bandwidth requirements, $Bmin(R_n)$. Each stream St_{ij}, $1 \leq i \leq m\text{-}1$ that traverses the link L_j on this path is characterized by its minimum bandwidth $Bmin(St_{ij})$, its maximum bandwidth $Bmin(St_{ij})$ and its current used bandwidth $Bcur(St_{ij})$.

If only the second admission condition succeeds on the link L_j, then we need to preempt some bandwidth from the streams St,, $1 \leq i \leq m\text{-}1$ that traverse this link in order to allocate it to the new user (this set is denoted $Stream(L_j)$). The problem is then to decide for δ_{ij}, $1 \leq i \leq m\text{-}1$ that: (1) satisfies the new user's minimum bandwidth requirements; and (2) minimizes the total loss of prioritized quality value. The problem is then formulated as follows:

Restriction:

$$\forall L_j \in link\_set(R_n), \forall St_{ij} \subset Stream(L_j), \quad 1 \leq i \leq m-1$$

$$\delta_{ij} \leq Bcur(St_{ij}) - B\min(St_{ij}) \qquad (3)$$

$$B\min(R_n) \leq Bunused(L_j) + \sum_{1 \leq i \leq m-1} \delta_{ij} \qquad (4)$$

Objective function:

$$Min \sum_{1 \leq i \leq m-1} \int_{Bcur(St_{ij})-\delta_{ij}}^{Bcur(St_{ij})} Q_i(bw)\, dbw \qquad (5)$$

$$Where \quad Q_i(bw) = W_i * D_i(bw)$$

$Q_i(bw)$ is called a *prioritized quality value function* where both the priority requirements, W_i, and the quality requirements, $D_i(bw)$, are considered.

Equation (3) specifies that on the link L, , a certain amount of bandwidth δ_{ij} is available for preemption from the stream St_{ij}, $1 \leq i \leq m\text{-}1$. This amount should satisfy equation (4), which expresses the new user R_n's minimum bandwidth requirements. If equation (4) holds, we have to decide from which streams to preempt the missing bandwidth. For this purpose, an objective function, equation (5) is used to minimize the total loss of prioritized quality value among the streams.

The expression $\int_{Bcur(St_{ij})-\delta_{ij}}^{Bcur(St_{ij})} Qi\ (bw)\ dbw$ in equation (5) represents the loss of prioritized quality values when bandwidth δ_{ij} is preempted from the stream St_{ij}.

3.5 Path Selection Optimization Criteria

To select a path from those that can satisfy the user quality requirements, some criteria should be considered. These criteria should be: 1) The number of links/hops to the tree; 2) The number of streams that are going to be preempted; and 3) The amount of preemptive bandwidth per link and per path.

Since our protocol uses message passing to coordinate the bandwidth preemption/allocation when performing parallel path search, this data can be collected easily.

4. CONCLUSION

This protocol, which uses the above-described algorithm, permits bandwidth preemption in order to allow users to join an existing multicast tree. The distributed aspect of the protocol is realized in that every link on a feasible path will participate in finding the necessary bandwidth for the new connection request.

The novelty of this proposed protocol compared to the one proposed by *Sakate* [I] is that: 1) We may have more than one on-tree node to permit a choice of the best path that lets the new user graft to the multicast tree; 2) Instead of considering the network as a whole, the algorithm is applied on a hop-by-hop basis along the candidate path(s); 3) *Sakate* minimum cost problem involves all the streams of the network; it is a centralized approach. There can be a case where the intersection of a given stream with the new stream gives a disconnected path. In that case, another algorithm, which is solved by using linear programming, should bc used [I]. In our case, this problem does not occur because we apply the *DBPA* algorithm on a hop-by-hop basis where the minimum cost problem involves only the streams that traverse a specific link; the streams do not need to a form a connected path.

5. REFERENCES

[1] H. Sakate, H. Yamaguch, K. Yasumoto, "Resource Management for Quality of Service Guarantees in Multi-party Multimedia Application", the IEEE 1998 Int. Conf. on Network Protocols (ICNP'98), pp. 189-196, 1998.

[2] M. Faloutsos, A. Banerjea, and R. Pankaj, "QoSMIC: Quality of Service Sensitive Multicast Internet protoCol", SIGCOMM '98, pp. 144- 153, September 1998.

[3] S. Chen, K. Nahrstcdt and Y. Shavitt, "QoS-aware Multicast Routing Protocol", to appear in IEEE Journal on Special Areas in Communication, 2000.

[4] Cisco Systems Inc., "Multicast Source Discovery Protocol", Cisco IOS IP and IP Routing Configuration Guide, Release 12.1, 1999. http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/120newft/120t/120t7/msdp.htm.

[5] R. K. Ahuja, T. L. Magnanti, J. B. Orlin, "Networks Flows, Theory, Algorithm, and Application", Prentice hall 1993.