

A host selection model for a distributed bandwidth broker

Christos Bouras

Dimitris Primpas

Research Academic Computer Technology Institute, N.Kazantzaki Str., Patras University
26500 Rion, Patras, Greece

&

Department of Computer Engineering and Informatics, University of Patras, 26500 Rion,
Patras, Greece

TEL: +30 2610 {960375, 960316}

FAX: +30 2610 960358

E-MAIL: bouras@cti.gr, primpas@cti.gr

Abstract

This paper deals with a distributed bandwidth broker that we try to evaluate and therefore propose a model that will improve its operation in a backbone network. Such a distributed bandwidth broker has been implemented and tested in NS simulator. Its operation is affected very much by the network condition and the responding time is proportional to the delay of packets in the network. Therefore, the paper presents a model (heuristic) that evaluates each node and finally selects the most important node where the base bandwidth broker should be located. This model is also associated with the general model that describes the operation of the bandwidth broker, and we propose an adaptation module of the Bandwidth broker that optimizes its operation. Finally, the model is simulated in order to evaluate its performance and indicate points that can be improved.

Keywords: Bandwidth broker, distributed admission control, network provisioning

1. Introduction

A bandwidth broker [1] is a “service” that provisions a backbone network and manages the supported QoS services. Actually, it has many interfaces as it receives demands for QoS services that it manages; it processes those demands and decides if it can satisfy them [3][7]. In case that the answer is positive, the bandwidth broker configures the network devices (routers, switches etc) to provide the bandwidth guarantees. The process of each demand needs information from the network configuration as well as from the network’s policy and management as the existing SLAs etc.

Such a bandwidth broker has been implemented in NS-2 simulator [8][9], following the generic architecture and is consisted of various modules. This bandwidth broker consists of many “devices” in the network that belongs to 2 basic categories: the clients and the basic bandwidth broker server. A client operates on every node of the network and makes the communication with the base server. This client is responsible to manage the local links, provide the interface to the user and applications to make new requests and also update its local routers with the configuration modifications according to new admissions. This client also stores data regarding the adjacent nodes of the node and communicates with the base BB every time the base BB needs this information. On the other hand, the base bandwidth broker is responsible to synchronize the whole operation and inform accordingly all the clients. It

also keeps statistics for the requests and the whole operation, like the active nodes etc. So, the architecture is somewhat distributed as every “client” executes some threads locally and stores some information there, but always synchronized by the base bandwidth broker server.

The system’s operation begins when an Edge Bandwidth Broker makes a request asking prioritization for given traffic class with rate of x bps from the node the client is running to some other network node. Then, the Base Bandwidth Broker searches the routing tables to find the route from node n_0 that made the request to the other end-node n_k . Next, the Base Bandwidth Broker sends a query to the Edge Bandwidth Broker clients that run on nodes who manage the links across the routing path, asking if there is available bandwidth. If all the answers are positive, means that there is available bandwidth from node n_0 to node n_k , the Base Bandwidth Broker will send a positive answer to the source. The procedure will be completed after the Base Bandwidth Broker sends to all the Edge Bandwidth Broker clients that lay on the path n_0, n_1, \dots, n_k , messages informing them to make the bandwidth reservation on the network devices. The bandwidth broker has been implemented in NS simulator and has been evaluated through various simulation scenarios providing end to end QoS by managing the IP Premium service, which is based on DiffServ architecture. The evaluation’s results were successful and optimization points are now taken into consideration [9].

Using this distributed bandwidth broker’s model, it is obvious that it exchanges several “signaling” packets that affects the total responding time of the bandwidth broker. Therefore, the proper location of the base bandwidth broker in the backbone network is crucial. Simultaneously, there is a lot of research work on this topic, investigating bandwidth broker architectures and performance parameters [2][4][5][6]. In this point, we propose and evaluate a host selection model that tries to select the host of the base bandwidth broker in the distributed implementation that we described above.

The rest of the paper is organized as follows: Section 2 has a brief description of the proposed host selection model and the way it works. Section 3 describes the modeling of the problem and the final evaluation. Finally, section 4 describes our conclusions focusing on the improvements in the proposed model that can be done as well as the future work that we intend to do on this area.

2. Model’s description

We tried to approach this problem by creating a model that evaluate each node and its adjacent links and according to the weights tries to find the best node to locate the base bandwidth broker. In other words, the problem is to find the root of the graph, where the root is the most important node in the network that should host the base server in order to reduce the overall responding time. The proposed model is divided in two phases that should run sequentially.

2.1. Definition of phase 1

This model uses 6 criteria that try to evaluate (in decimal scale) the importance of each node in the network operation. In details, the criteria are:

- Access interfaces. It represents the number of sub-networks and therefore the number of the users that are connected in this node. This criterion should also take into account the importance of those access interfaces and the amount of traffic they insert into the backbone. The access interfaces are mainly focused to hierarchically federated networks.

- Node equipment. This criterion approaches the capabilities of the specific node. In particular, the grade for this arises from the evaluation of the technology of the routers and the technology and capacity of the backbone links on this router.
- Adjacent nodes. This criterion specifies the importance of the node, taking into account the number of backbone links that are connected on this router.
- Servers. In this quantity, each node is evaluated by the importance of the servers that are connected on it and run critical and famous services of the network. Except from servers, they can be GRID clusters, VoIP gateways, gatekeepers or any other machine that implies that there is strong possibility for many requests targeted in this node for the IP Premium service that the bandwidth broker manages. Therefore, this criterion evaluates the importance of the services that are located on the router's LAN.
- Routing. In this case, the node is evaluated for its importance of the node in the whole routing scheme of the network. This criterion is proportional to the network topology and means the importance of the node in the basic operation of the network (the connectivity).
- Interconnection. Finally, the last criterion is used for the condition that this node is an interconnection point with a bigger backbone network and therefore there will be requests from the adjacent bandwidth broker to handle possible QoS requests.

The evaluation of the 6 criteria should be done in the same time for all the nodes and the gradation in each one should be proportional. Some of the criteria can be considered as static (for example the routing schema in a network does not usually changes), but there are criteria that can their evaluation can change often. Such an example is the Servers or the access interfaces. Finally, the weight of each node arises as the sum of all the criteria. In case that there are 2 or more nodes with the same weight, then the model should take into account the criteria in the following order: Routing, Interconnection, Servers.

The criteria that we proposed can be extended, but the same criteria should be used for all nodes. The phase 1 is some kind elastic as it allows the network administrators to specify and adjust the criteria according to their special needs and network's conditions. But in any case, the selected criteria should be representative of the QoS service usage in order to achieve an optimal host selection.

2.2. Definition of phase 2

Next, for each node, we create the "routing" graph for this node to all the others in the network. In particular, we place each node as root and we create all the paths to all the other nodes, using the network's routing scheme. Therefore, there are N graphs (where N is the number of nodes in the network) that should be examined. Then, we define a new metric for every node, called "special-weight" of node that arises as the weight of this node (that was produced by the above criteria) multiplied with its depth in the graph. In this case, the root of each node has "special-weight" equal to 0. Next, the "special-weight" of the whole graph is the sum of the "special-weight" of all of its nodes. Finally, the problem is to find the graph that has the minimum "special-weight". We run this model for all the nodes, we create all the N graphs and calculate the "special-weight" for each one. Then, we select the graph that has the minimum calculated "special-weight" and the node that is graph's root is the node that must host the base bandwidth broker. An example of the calculation of the "special-weight" for 2 of the 6 graphs that should be examined in the network, where the weight of each node has been already calculated, is presented in Figure 1. Also, Figure 2 presents the whole algorithm for the host selection in pseudo code.

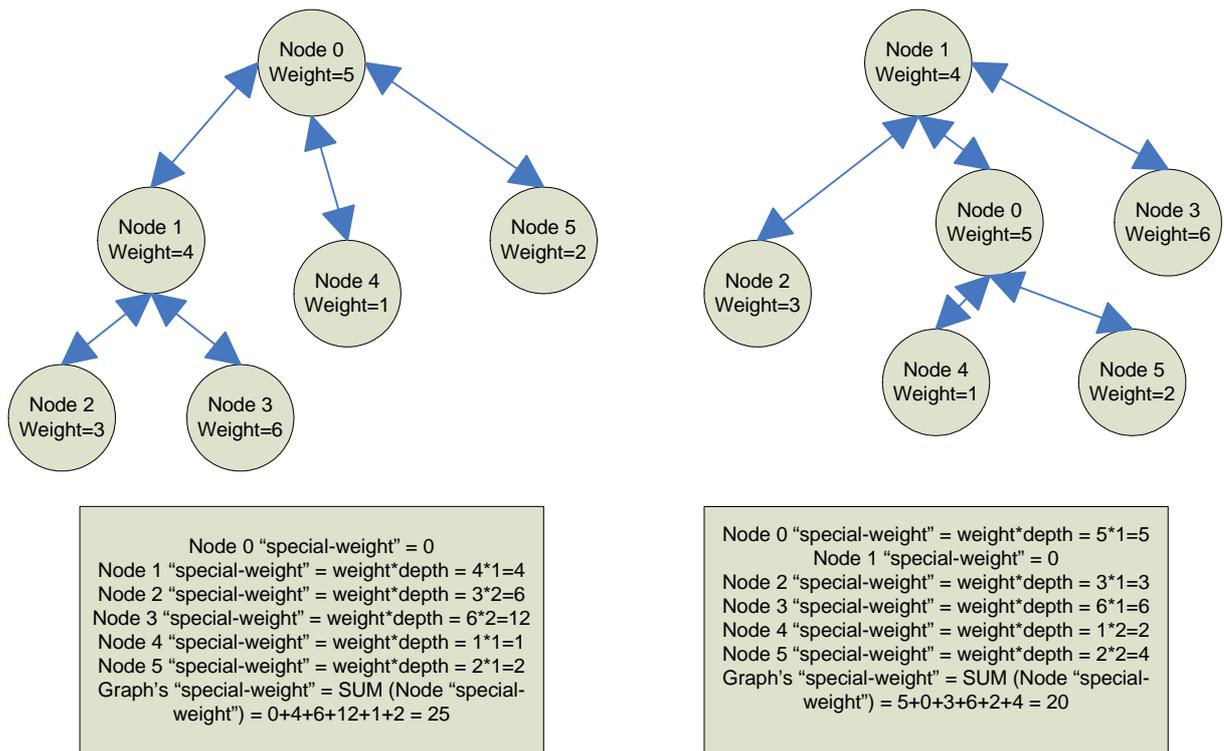


Figure 1: An example of the calculation of the "weight-depth" in a network

In addition, the algorithm at phase 2 assumes that the entire network's backbone links has the same layer 2 technology and capacity. This problem affects the output of the model if the assumption is not correct in some domains. In this case, it can be overcome by adding a new metric at phase 2 of the host selection model. In particular, the node's "special-depth" should be calculated by the multiply of the node's weight, the depth and the quantity Z , where $Z = 1/\text{transmission_rate}$ instead of node's weight and depth only.

```

Get all the nodes of the network
For each node {
    Get number of users connected there through sub-networks
    am store it
    Get node equipment (hardware, model etc) and store it
    Get number of adjacent nodes and store it
    Get number of servers and store it
    Evaluate the routing and store it (in a scale from 1 to
10)
    Evaluate the interconnection with other backbone
networks and store it (in a scale from 1 to 10)
}
Search the stored information for all the nodes for criteria
1 to 4, evaluate them analogical in scale 1 to 10 and store
the evaluation.
For each node {
    Sum the evaluation of all the criteria
    Store the result as node's weight
}
For each node {

```

```

    Place him as root and construct the graph with all the
    other nodes using the routing scheme of the network (how
    traffic is routed for root to every other node)
    For each node in this graph {
        Calculate weight*depth in the graph
        Store the result as "special-weight"
    }
    Calculate the sum of the "special-weight" of all nodes
    Store the result as graph's "special-weight"
}
Sort the array with the stored graph's "special-result".
Get the graph with minimum "special-weight" and find its
root.
Locate the base bandwidth broker on this root.

```

Figure 2: The host selection model in pseudo-code

3. Evaluation of the model

This model is a first step in order to evaluate and optimize the operation of a bandwidth broker. It can not be a very strict model, as the requests for a QoS service are not always in proportion with the above criteria. This can be explained due to the fact that there is always the parameter of the usage of the service and also the requests are not normally distributed in the time. Therefore, this model tries to provide a generic idea of the expected most important node in the network, taken into account the topology, the routing and the specific characteristics of the network.

3.1. Bandwidth broker's operation modeling

In order to approach the overall performance evaluation of the bandwidth broker we should study each component separately and notice those that can be a bottleneck. The bandwidth broker consists of the user interface module, the data storage module, the admission control and the network configuration. According to the distributed nature of the bandwidth broker, it is obvious that the most time consuming module is the admission control, with second the network configuration. Taken into account that the network configuration is done by the edge clients when a flooding message from the base server have informed them, we can conclude that the module that affects the whole operation is the admission control.

The admission control, as has already been described, operates in all the edge clients while the base bandwidth broker has the coordination and synchronization. Therefore, many "signaling" packets are exchanged that are the "bottleneck" for an efficient and fast operation. So, the basic problem is to minimize the exchanged "signaling packet" during the whole operation. This minimization is achieved by selecting the best node to host the base bandwidth broker. In order to do this, we should evaluate each node about the expected requests that can be targeted to it. This evaluation is achieved at the first phase of the host selection model. Next, we try to select the best node according to the "weight-depth" of the node and the whole graph. This second phase has been proposed based initially in a heuristic base, but can also be acknowledged with mathematical calculations and modeling.

In every request to the bandwidth broker, the element that is the “bottleneck” is the node at the routing path that is in the longest depth from the base bandwidth broker. This is explained due to the fact that the “queries” for checking the available bandwidth and the information about the final decision are all transmitted simultaneously and not sequentially.

So, in a network where the base bandwidth broker is located in node i , the whole operation is described by the following equation:

$$X_i = \sum_{k=1}^N (\sum_{y=1}^N (W_{y,k}(\max_depth_{y-k,i}))) \quad (I)$$

Where $W_{y,k}$ characterizes the number of requests between the hosts y and k during network’s operation. Finally, $\max_depth_{y-k, i}$ characterizes the maximum depth of the nodes that belongs to the routing path between nodes y and k . In order to adjust the network and maximize the operation of the bandwidth broker, it is necessary to select the appropriate node $i \in \{1, 2, \dots, N\}$ where the quantity X is the minimum. Therefore:

$$\text{Min}(X_i) = \text{Min} (\sum_{k=1}^N (\sum_{y=1}^N (W_{y,k} (\max_depth_{y-k,i})))) \quad (II)$$

The heuristic model is really close to the analytical as it is described by the following equation

$$\text{Min}(Z_i) = \text{Min} (\sum_{k=1}^N (W_k \text{ depth}_{k, i})) \quad (III)$$

In particular, (II) arises from (III) with the assumption that $\sum_{y=1}^N (W_{y,k} \times (\max_depth_{y-k,i}))$ is described by $W_k \times \text{depth}_{k, i}$. This condition could be approached if the unit W_k is modeled appropriately and represents the mean value of the requests from all the nodes of the network to this node k . and also the requests that target to node k are made analogically by all the other network’s nodes. The latter assumptions are the same with the assumption of heuristic model and they depend on the human parameter (usage of the service from end users).

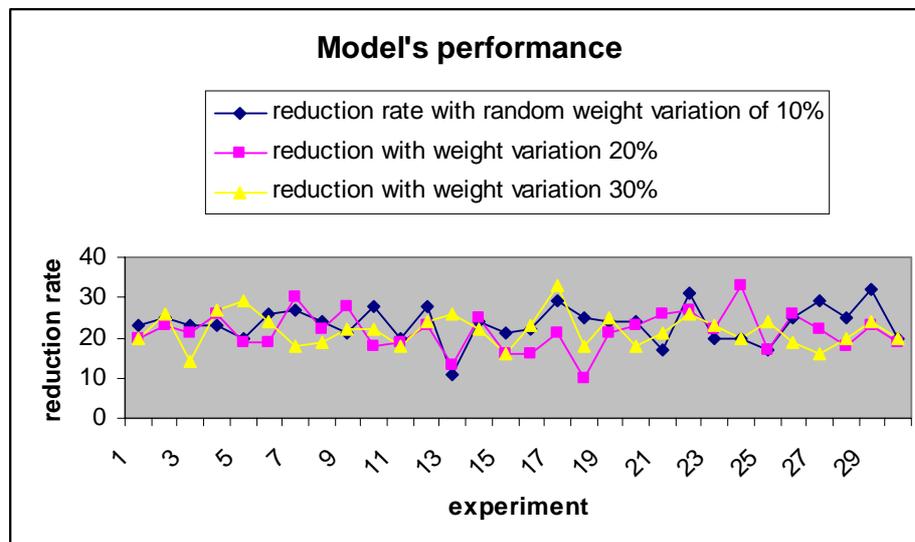


Figure 3 the reduction in packets/link for the 2 topologies

3.2.Model’s simulation

The next step was to simulate the above model through a java program and try to measure its accuracy. Therefore, we loaded some data (from Figure 1) and we tried to identify how elastic the model is according to the first phase’s criteria. The idea of the simulation was to create some network topologies and use the model to find the 2 best locations for the base

bandwidth broker. For the simulation, we used the example that Figure 1 shows (these 2 topologies with the relevant weight of every node).

Next, the program makes random requests between nodes according to their weights as they have been expressed in phase 1 (and in Figure 1). The random requests were 100 and they were repeated in 30 iterations. Also, the program takes into account some possible variation of the calculated weights; therefore we used random variations of maximum 10%, 20% and 30% respectively. Next the program calculates the total number of messages/link that are used by the bandwidth broker in this time period and also the number of links in order to approach the final execution time. Finally, Figure 3 presents the overall reduction rate that the model can achieve for the 3 cases (variation of initial weight of 10%,20% and 30%).

According to the results, the model gives a reasonable result when the approximation that phase's 1 criteria succeed is realistic even with a variation of 30% on each link. Of course it is reasonable that nodes in bigger depth in host selection graph have lower tolerance in weight's variation than the other. Also, Figure 4 compares the reduction rate that is achieved if the bandwidth broker uses the node that is selected by the model and its first candidate (the node in the second place from model's results). Therefore, the measured execution time (with the assumption that is affected only by the packets transmission in the links) has a reduction between 8.2 to 26% in the selected host than its major candidate. The points that have negative values means that in this cases the major candidate has better performance that the selected node by the model. Due to the fact that we measured some negative points, we repeated the above simulation tests (50 iterations of 100 random requests) for a number of times to measure the percentage of host selection success, as it is determined by the execution time (it means we measured for every request if the selected host or the alternative gives better performance). The result show that the success rate (in comparison with its major candidate host) was between 90 and 96%. Therefore, the output that the model produces is reasonable and seems to outbalance in most cases.

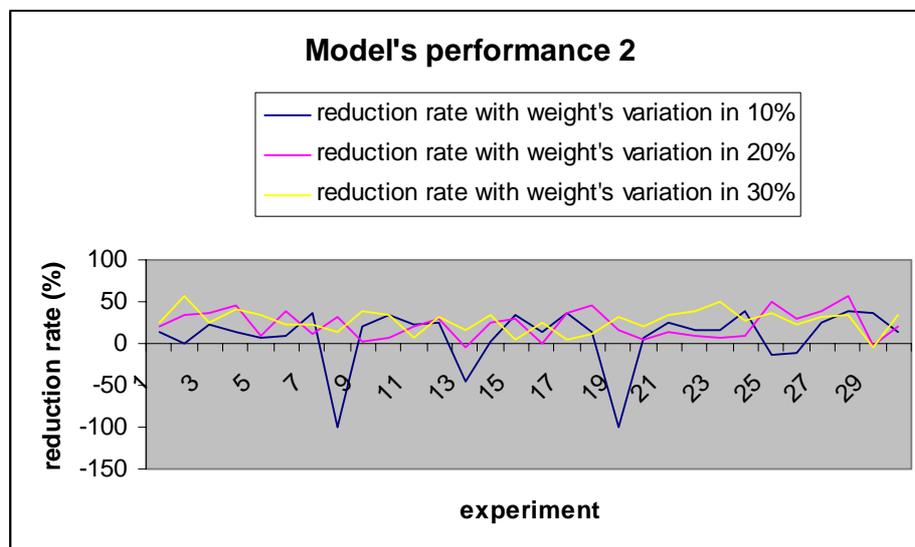


Figure 4 the reduction in execution time for the 2 topologies

4. Conclusion – Future Work

This paper presents a model that evaluates each node and selects the most important node where the base bandwidth broker should be located, taken into account its distributed nature. The model was simulated in order to evaluate its performance and finally showed very good performance as well as a reasonable compliance on phase's 1 criteria.

The major problem is such models comprise when the network's status changes often and therefore the model should be executed periodically in order to optimize the performance. A very interesting solution and idea is to adapt the operation of the bandwidth broker periodically. In particular, each node of the network has a bandwidth broker client and a specific one runs as a base bandwidth broker agent. The above host selection model can be configured to run periodically (in intervals that should depend on network changes) and in case of resulting a different node as the most capable to host the base bandwidth broker agent, the bandwidth broker moves to a temporary "adaptation" mode. In this mode the bandwidth broker does not process any request and buffers them. Then, the client on the new host becomes base bandwidth broker server and replicates the stored information from old base bandwidth broker. Next, the old base bandwidth broker shuts down and starts running only as client. Also the new host broadcasts a message to the clients, introducing itself as the new base bandwidth broker and the status of the bandwidth broker changes again into normal and starts processing the buffered requests. This idea overcomes the problem of sudden or often changes of phase's 1 criteria that evaluate every node and optimize the whole operation.

Finally, we already have plans for future work that mainly focus on the implementation of the adaptation mode that was described above in current implementation in NS-2. Also we plan to study possible improvements in the admission control schema such as an optimization algorithm that will extend the admission control algorithm. Also, we have implemented a second version of the bandwidth broker that operates centrally and we plan to make some comparison tests [10]. Additionally, we are working on an extension of the existing bandwidth broker; studying a protocol for inter domain communication with adjacent bandwidth brokers.

5. References

- [1] RFC 2905 "AAA Authorization Application Examples", J. Vollbrecht, P. Calhoun, S. Farrell, L. Gommans, G. Gross, B. de Bruijn, C. de Laat, M. Holdrege, D. Spence, August 2000
- [2] Manzoor Hashmani and Mikio Yoshida "ENICOM's Bandwidth Broker", Saint 2001 Workshops, pp 213-220, Jan 8-12, 2001, San Diego, USA
- [3] Przemyslaw Jaskola, Krzysztof Malinowski "Two methods of optimal Bandwidth allocation in TCP/IP networks with QoS differentiation", 2004 International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS' 04), San Jose, California, USA, July 25 - 29 2004
- [4] QBone Bandwidth Broker Architecture, <http://qbone.internet2.edu/bb/bboutline2.html>
- [5] Techniques for End-to-End Quality-of-Service control in multi-domain IP Networks, <http://homes.dico.unimi.it/~pagae/NPTLab/camp.html>
- [6] Active Resource Management (ARM) For The Differentiated Services Environment, http://www.caip.rutgers.edu/TASSL/Projects/Adaptive_QoS/ananth/bandwidth.html
- [7] C. Bouras, K. Stamos, "An Adaptive Admission Control Algorithm for Bandwidth Brokers", 3rd IEEE International Symposium on Network Computing and Applications (NCA04), Cambridge, MA, USA, August 30 - September 1 2004, pp. 243 - 250

- [8] C. Bouras, D. Primpas, K. Stamos, N. Stathis, "Design and implementation of a Bandwidth Broker in a simulation environment", Interworking 2004, Ottawa, Canada, November 29 – December 1
- [9] C. Bouras, D. Primpas, "An admission control and deployment optimization algorithm for an implemented distributed Bandwidth Broker in a simulation environment", 4th International Conference on Networking – ICN 2005, Reunion Island, France, April 17 - 21 2005, (to appear)
- [10] <http://ouranos.ceid.upatras.gr/DiffServ-NS/default.htm>