A CALL ADMISSION PROTOCOL OF SERVICE LEVEL AGREEMENT IN ENTERPRISE NETWORK

Md Mostofa Akbar[1], Mohammed Anowarul Hassan Hassan[1],
Eric G. Manning[2,3], Gholamali C. Shoja[2]
[1]Department of CSE, Bangladesh University of Engineering and Technology, Dhaka, Bangladesh
[2]Department of CSC, University of Victoria, Victoria, BC, Canada
[3]Department of ECE, University of Victoria, Victoria, BC, Canada
Email: mostofa@cse.buet.ac.bd , tomal_cse@yahoo.com , Eric.Manning@engr.uvic.ca , gshoja@.cs.uvic.ca

ABSTRACT

The admission of multimedia streams with a particular and guaranteed level of QoS (Quality of Service) from source to destination with the goal of increasing revenue is challenging in multimedia communications over IP (Internet Protocol) networks. In this paper, we present SLA (Service Level Agreement) as an agreement between users and network operators, for the transmission of a multimedia stream with several choices of QoS level. Control of the admission of SLAs, to achieve maximum revenue while fully respecting QoS guarantees, is called SLA admission control. We present a scalable algorithm for the admission of SLAs.

KEYWORDS: SLA, QoS, Admission Control, Dynamic Threshold.

1. INTRODUCTION

Figure 1 shows a small network with five nodes and three possible paths from the source node S to the destination node D, represented by a thick line, broken line and thick broken line respectively. A user submits a service request to the network Admission Controller for some amount of bandwidth with a particular delay bound, from a source node S_i to a destination node D_i. This service request may consist of choices, such as 10 Mb/s with delay less than 20 ms at $50/hour or 20 Mb/s with delay less than 30 ms at $75/hour, corresponding to different levels of QoS. These service requests are termed as Service Label Agreement or SLA.

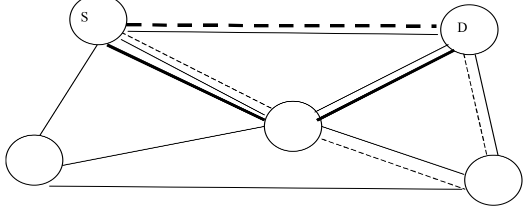

Figure 1 - A simple network with 3 SLAs.

In our paper, the proposed Admission Controller has the following contributions:

- It determines which SLAs will be admitted to provide maximum revenue in the long run.
- While admitting the SLAs, it fully respects QoS guarantees. Virtual circuits must be set up between source and destination as in ATM systems to ensure QoS..

In this paper we describe the design of an SLA Admission Controller to achieve the above goals. Related research work on admission control in IP networks will be discussed briefly, in Section 2. We present a refined definition of SLA for multimedia stream transmission in Section 3. Section 4 formulates the SLA optimization problem. The new algorithm for the SLA Admission Controller will be presented in Section 5. The complexity of SLAOpt is studied in Section 6, followed by experimental results in Section 7. Section 8 concludes the paper.

2. RELATED WORK

The current Internet is based on a best-effort datagram service model. When a packet arrives at a router, and sufficient resources (such as processing time, and outgoing link buffer space) are available, the packet is forwarded to the next router. However, if the necessary resources are not available, the incoming packet may be delayed, or even dropped. It is therefore difficult to predict, let alone guarantee, the bandwidth or latency experienced by a stream of packets under best-effort datagram service as packets may experience variable and unpredictable delays, and may arrive at the destination out of order. Hence best-effort datagram service model is not considered suitable for the Internet2 [1]. By reserving a dedicated path and bandwidth, similar solution has been developed for multimedia [2]. In this technique the requested SLAs are mapped to a Multiple Choice Multidimensional Knapsack Problem. With the objective of increasing revenue, similar works have been done using Neuro Dynamic Programming [3] and static threshold [4].

3. DEFINITION OF SLA

There are amny proposed definition of SLA [5]. In our research a Service Level Agreement SLA_i is defined as follows:

Source and Destination: Nodes of the network labelled S_i and D_i.

Delay Bound: The network link propagation delays plus the switching delays between Source and Destination nodes must be bounded above by d_{ij} for the jth level of QoS for SLA_i.

Bandwidth Requirement: QoS level j of SLA_i requires bandwidth B_{ij} between S_i and D_i.

Duration: Duration of the multimedia session to be active.

Rate: For each level j of QoS, there must be a time rate of revenue.

QoS levels: Users are allowed to submit different options for desired QoS levels and bid prices.

4. SLA OPTIMIZATION PROBLEM

4.1 ASSUMPTIONS:

We need to make the following assumptions about the network :

1. The switching delays are ignored. The total delay of a path is the sum of hops from source to destination.
2. Call set up and teardown times are negligible compared to the duration of the SLA.
3. Bandwidth for a SLA is constant during the lifetime.

4.2 FORMULATION OF THE SLA OPTIMIZATION PROBLEM:

We consider here n SLA requests $SLA_1, SLA_2,.. SLA_n$ to offer to the SLA Admission Controller. Each SLA_i has different QoS levels for bandwidths and revenue from S_i to D_i. The required bandwidth and rate offered by the user for the *jth* QoS level are b_{ij} and r_{ij} respectively. Let there be i_n paths $P_{i1}, P_{i2}, \ldots, P_{in}$ from S_i to D_i. A path P_{ik} may satisfy a QoS level Q_{ij} if the associated delay bound d_{ij} is greater than the delay on path P_{ik}, defined by $Delay(P_{ik})$. Selector variable λ_{ijk} denotes whether the *kth* feasible path of the *ith* SLA satisfies its *jth* QoS level:

$$\lambda_{ijk} = \begin{cases} 1, \text{iff } Delay(P_{ik}) \le d_{ij} \\ 0 \end{cases}$$

The *j*th QoS level of SLA_i has at best K options to select a particular path and a particular QoS level. Here, K is the maximum number of paths selected using the K shortest paths algorithm. Each of these options represents a QoS level.

In this work, only gross revenue is examined for simplicity. The associated gross revenue can be expressed as $u_{ijk} = r_{ij} \times \lambda_{ijk}$.

If the *jth* QoS level of SLA_i is served by path P_{ik} then the resource (bandwidth) consumed from link *m* of the network is:

$$R_{ijkm} = b_{ij} \times \gamma_{ik}^{m}.$$

Here,

$$\gamma_{ik}^{m} = \begin{cases} 1, \text{iff path } P_{ik} \text{ uses the } m\text{th link of the} \\ \quad \text{network} \\ 0 \end{cases}$$

Here the objective function is to maximize $\sum_i \sum_j \sum_k \delta_{ijk} u_{ijk}$. Here δ_{ijk} is the selection variable for selecting the *jth* QoS level using path P_{ik}.

Hence $\sum_j \sum_k \delta_{ijk} = 1$ and $\delta_{ijk} = \{0,1\}$.

The resource constraints are $\sum_i \sum_j \sum_k \delta_{ijk} R_{ijkm} \le B_m$, m = 1, 2, 3,…… N, where B_m is the total available capacity on the *m*th link.

5. PROPOSED ADMISSIONCONTROLLER

The arriving SLA requests are batched for an interval of time called an epoch. At the end of each epoch, the SLA Admission Controller processes the batched SLAs, admitting some and rejecting the rest. the details of computational technique of the SLA Admission Controller is described here:

5.1 DYNAMIC THRESHOLD CONTROLLER

The threshold controller proposed by Mokhtar [4] is used for admitting a bunch of SLAs. The algorithm is static and not suitable for a SLA admission controller where SLAs are allowed to get in and out dynamically. They used predefined static Threshold for an Admission Controller which does not change with the increase or decrease of load in the system. We have extended the concept of threshold controller for our proposed system. We propose a method of calculating the thresholds for a particular SLA of an *epoch* dynamically according to its contribution to the total revenue and bandwidth ratio. It works in the following three steps:

Step 1 finds the total revenue and total required bandwidth of all the QoS levels of the SLAs in the system.

$$total\_revenue = \sum_i \sum_j r_{ij}$$

$$total\_bandwidth = \sum_i \sum_j b_{ij}$$

In **Step 2**, the respective contribution ratio, C_{ij} of each *QoS* of each *SLA* is measured using the following equation

$$C_{ij} = \frac{r_{ij} / total\_revenue}{b_{ij} / total\_bandwidth}$$

And it is scaled from 0 to 100 from the following equation to get the scaled contribution, SC_{ij}

$$SC_{ij} = \frac{C_{ij} - \min c}{\max c - \min c}$$

Where min *c*= min(C_{ij}) and max *c*= max(C_{ij}) for all *i* and *j*.

Finally, in **Step 3**, the respective thresholds, t_{ij} are calculated by virtue of *Squashing Function*

$y=(\alpha+\beta\times SC_{ij})$,where α and β are scaling factors. (We assume α=10 and β=5 in our experiment)

and

$t_{ij} = B - B \times \frac{1}{1+\exp(-y)}$, Where *B*=min (B_m), m=1,2,….M [M= Total No. of links]

Here, *Sigmoid* (*Squashing) Function* [6] is used to let the *SLAs* contributing little revenue be admitted to some extent.

5.2 ADMISSION CONTROL

We have followed the algorithm to generate the *batched_sla_list* and their properties as proposed by Akbar [2]. Instead of *Heuristic approach*, we have applied *Dynamic Threshold Controller* to do admission control.

The pseudo-code for the admission control algorithm using dynamic threshold (after generating *SLAs*) is as follows:

```
//batched_sla_list is the list of SLAs batched in an
//epoch
// QoS_list is the list of alternative QoS of an SLA.
for i←1 to size(batched_sla_list) do
    for j←1 to size(batched_sla_list[i].QoS_list) do
        path_list ← Calcultae_K_shortestpaths
        (batched_sla_list[i]. source,batched_sla_list
        [i]. destinatiomn)
        //Here Dijkstra's algorithm [7] is used for
        //finding shortest paths
        for k←1 to size(path_list[ ]) do
            Calculate_remaining_Bandwidth()
            //It Calculates the minimum of the
            //remaining bandwidth of the links from
            //source to destination of the current
            //path after admitting the SLA
    if(remaining_bandwidth>batched_sla_list[i].
    QoS_list[j].threshold then
        admit ( batched_sla_list[i] . QoS_list[j]))
        check for next batched_sla_list[i]
                endif
            endfor
        endfor
    endfor
```

6. COMPUTATIONAL COMPLEXITY

As referred by Akbar [2], the complexity of the total computation for path calculation of *n* SLAs is $O(nN\log N)+O(KnN)$. To calculate the thresholds, we have to only compute the sum of revenues and required bandwidths for all *SLAs,* whose complexity is $O(n)$.

So total complexity is: $O(nN\log N)+O(KnN)+O(n)$.

7. EXPERIMENTAL RESULTS

In our experiment, we simulated on a network with 25 nodes and 41 links, where the capacity of each link is 100,000 units.

We have compared the results obtained by our protocol with SLAOpt [2] and Greedy Method. According to Greedy Method, SLAs are admitted (*First come, First served*) as long as there is enough bandwidth for them without considering their revenue and required bandwidth. The experimental results are given in the figures below:

Initial batch size	Dynamic Threshold Controller	SLAOpt	Greedy Appoach
150	62	4140	16
300	15	10703	0
450	31	21187	16
600	47	46531	15
750	47	78937	16
900	63	127703	15
1050	78	137968	15
1200	78	202500	16
1350	94	211516	16
1500	109	248797	15

Table 1: Time required (in ms) by Dynamic Threshold Controller compared with SLAOpt [2] and Greedy method.

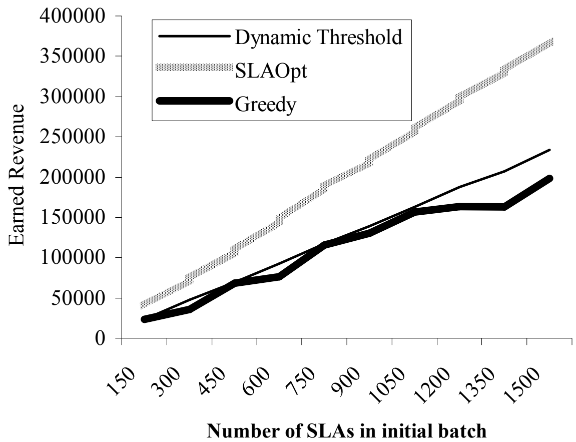

Figure.2 Earned Revenue by Dynamic Threshold Controller compared with SLAOpt [2] and Greedy Approach.

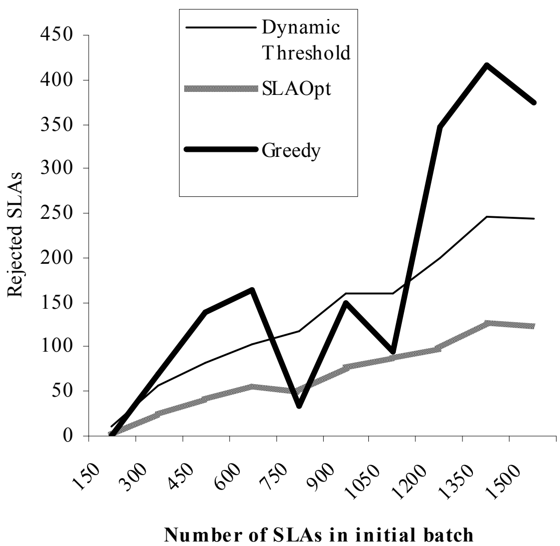

Figure.3 Rejected SLAs by Dynamic Threshold Controller compared with SLAOpt [2]and Greedy Approach

7.1 ANALYSIS OF THE RESULT

If we look at the time requirement in Table 1 we find that time requirement of the new approach is far less than the time requirement of SLAOpt. But the earned revenue of the new approach not as bad as the ad hoc greedy approach. The figure showing the number of rejected SLAs show the same trend. Thus the new approach seems to be a very good solution in terms of both scalability and earned revenue and suitable for real time admission control in networks which deal with a large number of SLAs.

8. CONCLUSION

In this paper, we introduced a new Call Admission Protocol. This simulation can be easily implemented for use in the Bandwidth Broker for a network connected by ATM switches and fiber optic links. This approach obviously increases the revenue with the cost of penalizing 1%-5% of the total requests taking the decision in a few milliseconds. The beauty of this technique of admission controller is that it justifies both the needs of any commercial system i.e., very good response time and contribution in terms of money. Distributed version of the new admission controller for more scalability and application of this technique for multimedia server admission controller could be a good topic for future research work.

9. REFERENCES

[1] B. Teitelbaum. Future Priorities for Internet2 QoS -- Internet2 QoS, WG: Papers, October 2, 2001, http://www.internet2.edu/qos/wg/documents.shtml.

[2] Md Mostofa Akbar, G. Manning, Robert K. Watson, Gholamali C. Shoja, Shahadat Khan,Kin F. Li. "Optimal Admission Control and QoS Adaptation of Service Level Agreements in Enterprie Networks". SCI 2002, Orlando, FL July 14-18, 2002.

[3] Peter Marbach, Oliver Mihatsch, John L. Tsitsiklis. "Call Admission Control and routing in Integrated Services Networks Using Neuro-Dynamic Programming", IEEE Journal on Selected Areas in Communications, February 2000, Volume 18, Number 2.

[4] Mokhtar A. Aboelaze. "A Call Admission Protocol for Cellular Networks that Supports Differentiated fairness", Proceedings of the IEEE Semiannual Vehicular Technology Conference IEEEVTC, Milan, Italy May 17-19, 2004.

[5] Allot Communications. "QoS and Service Level Agreement Management Solutions & Case Studies", http://www.allot.com/.

[6] Tom M. Mitchell. "Machine Learning",1997,pp96-97.

[7] E. Dijkstra. A note on two problems in connexion with graphs, Numerische Mathematik, 1959, pp. 269-271.