QoS and queuing disciplines, traffic shaping and admission control.

Marko Luoma
Helsinki University of Technology
Laboratory of Telecommunications Technology
P.O. Box 3000
02015 HUT
Finland
email: Marko.Luoma@hut.fi

Abstract

Internet the land of hope and glory is spreading its wings all over the world. Now quality of service (QoS) is also finding the way to this area. QoS means extensive re−engineering to the Internet. It means service model creation, advanced scheduling techniques, admission control, client traffic shaping and so fort. This paper presents how traffic management aspects work in Internet and how they are bound together by means of proper service model.

1. Introduction

The Internet can be thought about in relation to its common protocols, as a physical collection of routers and circuits, as a set of shared resources, or even as an attitude about interconnecting and intercommunication. The Internet was born about 20 years ago, trying to connect together a US Defense Department network called the ARPAnet and various other radio and satellite networks. The ARPAnet was an experimental network designed to support military research − in particular, research about how to build networks that could withstand partial outages (like bomb attacks) and still function. Does this sound like today's Internet, web surfing, e−cash and Internet commerce ...

The old rule for when things are confusing is "follow the money." Well, this won't help you to understand the Internet. No one pays for "it"; there is no Internet, Inc. that collects fees from all Internet networks or users. Instead, everyone pays for their part. Well how one can then model and look what Internet really is ? The answer is no way. Well there may be a possibility to understand and actually model some parts and aspect of Internet. We can build a crude model for the way information is passed on in Internet and we can see how some aspects of implementations can change the face of this model.

Model of Internet, services provided in it and mechanisms actually implementing these services are very complex and diverse. You will see some flicks how Internet can be seen, what scheduling is, what we do with scheduling, what is the relation of scheduling to the admission control and so on. Internet cannot be engineered over a night and so cannot the traffic management mechanisms in it explained in quarter of an hour or even in a day.

2. Model of Internet

Internet is set of networks and computers connected together to form a large internetwork. This internetwork can be modeled hierarchically as on top of model a source connected through a large cloud (Internet) to a receiver. When we take closer look on this large cloud we see that it is formed from several independent clouds (ISP's). Looking closer on ISP's we see that their network on turn is formed up from several routers.

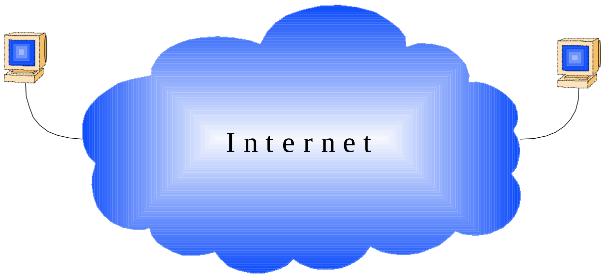

Figure 1: Internet; a calm clouded sky with full of hope and happiness

Second aspect of Internet, which makes it impossible to model, is routing. Routing makes datapaths through these different levels of hierarchy. One can see routes from place A to go place B from one not so optimal way, but still it is the way routing protocols choose it.

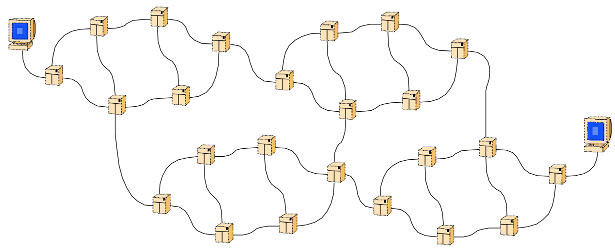

Figure 2: Internet; a mesh of routers

Third model in parallel is the model of forwarding path. Forwarding path of a router is a queue. Inside ISP's network each router form a network of queues. If you do some offline calculations ISP is modeled as a single queue just a sake of your sanity. So when reaching the top level a source is connected to a destination through a queue, which actually is a network of queues modeled as a single queue (this does not hold for real calculations, but is nice way to understand some parts of functionality).

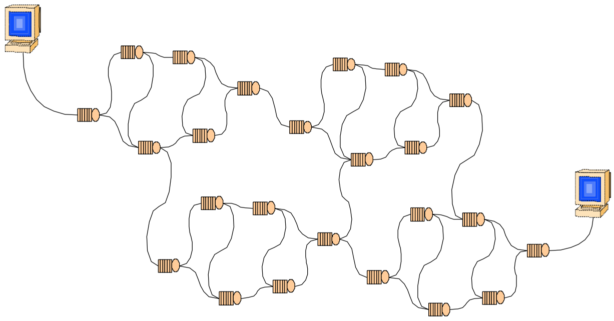
Figure 3: Internet; a network of queues

This step by step piling of the network reveals several important aspects of the Internet:

1. Internet is set of independent islands
2. Structure of each island is independent and unique.
3. There is no way to model connection through the Internet in level which would allow some sort of reasonable accuracy in performance modeling.

3. Service models for Internet

There are many ways to build service models for the Internet, some of them are more related on some technical concept and some more general. Here I will present a short view to some modeling mechanisms which may help understanding scheduling and admission control mechanisms.

3.1 Guaranteed Service

Traditional real−time service models have been designed on two assumptions: first, traffic sources can be well characterized by Markov chains, and second, receivers require rigid delay bound. When end system requests traditional real−time service, it must characterize its traffic so that the network can make its admission control decision. If this allocation is done with a single parameter it is called peak rate allocation but if multiplexing and some probabilistic loss and extra delay is accepted it is called probabilistic allocation.

Guaranteed service, cause it is likely to be expensive, is suited for a mission critical applications. Some examples are stock marketing and surveillance information delivery.

3.1.1 Guaranteed Service

Real−time service which provides a hard or absolute bound on the delay for every packet and offers zero packet loss is called usually guaranteed or deterministic guaranteed service.

Guaranteed service is offered for applications which require absolute delay bound but cannot characterize their traffic accurately. This type of service commitment leads to the low utilization of networking resources and therefore is very expensive for the user. Managing of guaranteed service is relatively easy because no multiplexing is done and no losses allowed so there is no probabilities related on the allocation of resources.

3.1.2 Probabilistic Service

Probabilistic service is strongly related to the guaranteed service with the exception that controlled losses are allowed. Controlled means that user admits some level of losses for which the network makes a commitment.

In probabilistic service bandwidth for a new connection is not allocated on the basis of the peak rate; rather, the allocated bandwidth is less than the peak rate of the source. As a result, the sum of all peak rates may be greater than the capacity of the output link (this is called statistical multiplexing). Statistical allocation makes economic sense when dealing with bursty sources, but it is difficult to carry out effectively. This is because of difficulties in characterizing an arrival process and lack of understanding as to how an arrival process is shaped deep in the network. This shaping is at this point harmful phenomena, it looses the controllability of source traffic to a extent that resources have to be allocated very much like in peak rate allocation.

Probabilistic service should be cheaper version of guaranteed service for a user, cause user has voluntarily allowed network to remove packets in case of congestion. Depending on the amount of the multiplexing and ratio of the losses, the utilization of the networking resources may be high.

3.2 Predictive Service

Predictive service model is common name for service models which are based on the use of traffic measurements in service allocation. Traffic is again characterized by some traffic filter but the tolerance of filter is allowed to be coarse. This coarse allocation is then verified through traffic measurements done on the aggregate traffic. Because measured information is always past information predictive service promises a more relaxed delay bound than guaranteed service.

Predictive service allows its admission control algorithm to admit more flows and attain a higher level of network utilization. This higher utilization makes possible to offer predictive service in very competitive price.

Applications which are candidates to predictive class are all interactive applications which require some form of guarantee to delivery time.

3.3 Bxxx Effort Service

Best and better effort classes are plain old Internet and some extensions of it. These extensions are local policy based networking like Differentiated Services concept, where a filter separates traffic to a relatively small amount of parallel traffic classes. These parallel classes get a share of capacity on some controlled order. Typical applications fore this service space are all applications not requiring delay bounds but may require low loss. To name some http, ftp, email and so fort are good candidates.

4. Different applications

As everything is related to everything we need to start from some point. We start from the applications which produce the traffic to the network. Traffic characteristics are very important in the admission control decision and in the overall network engineering.

Applications are the ones which produce the traffic to network. So applications with economical aspects actually dictate the service model and some of the major aspects of Internet. This is largely due to the fact that application space change very slowly leading to the necessity of adjusting network to offer services based on the requirements of present applications.

The problem with the QoS and application traffic is that in order to succeed in the QoS networking some form of admission control is needed. This admission control requires traffic profiles and parameters from applications. How one can derive this parameters is very rocky road with lots of hidden holes and slippery hills [32].

Applications can be divided into different classes from the point of view of traffic they produce, service they require and nature they are.

4.1 Interactive applications

Interactive application is typically application which requires constant attention from the user. These applications are used for human to human communication; like IPtelephony, video conversation, shared whiteboard etc. They require minimum delay through network, so that variation of the delay is also minimal. The amount of bandwidth they use is relatively small from kilobits per second (IPtelephony) to tens of kilobits per second (video conversation). Some interactive applications like distributed simulation may need a rather markeable capacity from network, but they are not likely to exist in a large scale. These applications usually use raw UDP−protocol for their communication [30].

4.2 Semi−interactive applications

Semi−interactive applications form a large group of applications which can be categorized further to the streaming applications and shell applications.

4.2.1 Streaming applications

Streaming applications are applications like RealAudio and RealVideo. They produce traffic which needs somewhat bounded delay but the variation is allowed to disperse quite freely. Correctness of information is much more relevant than the minimal transfer time. This is due to the jitter compensation they use. Jitter compensation algorithms usually calculate delay through the network and based on the arrival process of packets they either buffer more or less at the receiving end. These applications usually use UDP−protocol enhanced with RTP or some proprietary mechanism to implement rate and error control.

4.2.2 Shell applications

Shell applications perform tasks which are strongly related to the operating systems. This group consists applications and protocols like X, NFS, HTTP and Telnet. The feedback which this group gives is directly observable from user so timely transfer is important. Rate control in this group is usually performed through the TCP flow and congestion control, so these applications can adjust their operation to the changing environment.

4.3 Bulk transfer

FTP, email, nntp and other raw data transfer protocols belong to this class. These applications are used for transferring of large quantities information on the background.

4.4 Hierarchical mapping of applications

Other way to divide applications is based on the their behavior. Behavior is inhereted from transport protocols and application aspects.

4.4.1 Elastic applications

Many application can adjust themselves to the network congestion to amount that they can be categorized to be elastic. Elastic class can be further divided into classes of based on their inherent interactivity. From the previous groups semi−interactive streaming applications can be categorized to the class of interactive burst transfer. Semi−interactive shell applications are more on the side of interactive bulk transfer class. Leaving bulk transfer class to be mapped to the asynchronous bulk transfer.

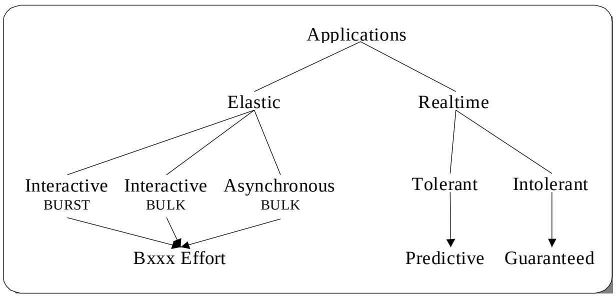

Figure 4: Application separation; behavior and related service model

4.4.2 Realtime applications

Still there are many applications which lack of the adjustment to the changing network conditions. These applications are usually some form of realtime application, either conversational or remote resource sharing. The way delay variations are compensated on receiver divides these applications into two classes: tolerant and intolerant applications. Tolerant applications usually have jitter compensation allowing delay to vary on some range. These applications will probably be dominant applications in the resource reservation space. Intolerant applications suffer from misinformation or poor perceived quality if the delay fluctuates. To operate this type of applications in packet networks is expensive, cause massive oversubscription is necessary.

5. QoS Traffic Management

QoS Traffic management in Internet means co−operative functions which make possible to offer the QoS. In short these functions are appropriate scheduling, queue management, traffic shaping and related admission control. These management functions are tightly bound together, so that they are all necessary for reasonable operation.

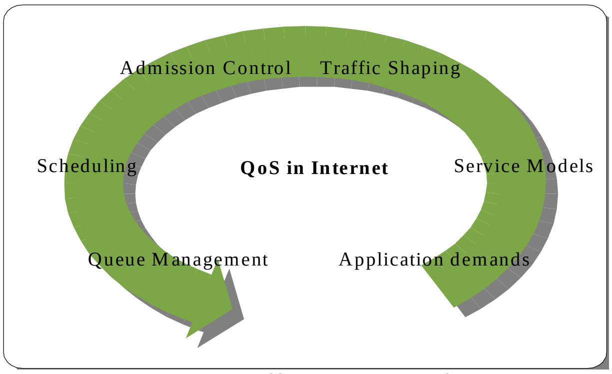

FIGURE 5: Different aspects of QoS

In following chapters all of these mechanisms will be evaluated one by one and a reference implementation is constructed. This reference implementation is based on scheduling mechanism presented by Shenker, Clark and Zhang in 1992 [22, 25] and related admission control from Jamin in 1996 [21].

6. Queue Management

Queuing in routers is organized usually in output drivers of it. Queuing may be in overall very simple or complicated mechanism. In most simplest form it is a FIFO queue serving all traffic flowing through.

The purpose of the queuing is to accommodate transient over subscription in links − called contention. In cases when this contention is more permanent phenomenon it is referred as the congestion; a situation when packets are lost due to the shortage of buffers.

Queuing is divided into two distinct parts: queues and servers. Queues are storage spaces for packets waiting to be transmitted where servers are active processors determining which packet is to be sent next. In queues exists also active processing called queue management. Queue management is responsible of processing incoming packet queuing tag (label associated each packet showing which queue it should pe put) and also removal of packets from the queues when they fill up. Most common method for this type of operation is based on Random Early Detection (RED) which removes packets not from a queue but from the incoming packet stream based on the random sampling. This randomness reduces global synchronization among dependent traffic streams [27−29].

This part of activity in routers is covered later on presentation about Random Early Detection and other co−operative queue management methods.

7. Scheduling

Scheduling is the task of allocating resources to the individual classes of traffic. This allocation means packet per packet processing of the information from queues to the link. Scheduling is always related to the type of operation we want to encourage in our network. In simplest form it can be FCFS type straight forward operation where each packet is served relative to time it arrived to the system. FCFS scheduling or as it more commonly called FIFO queuing is dominant way of operation in the Internet today. It does not allow any isolation between the traffic classes using it. Scheduling has been under extensive studies for ages, some recent studies have been concentrating on effects of the scheduling to the high speed networking. [10−14, 24−25]

Definition: A scheduler is work conserving when it is never idle if there is a packet in the queue. Otherwise it is not work conserving.

Work conserving: General processor sharing (GPS), weighted fair queuing (WFQ), virtual clock (VC), delay earliest due date (Delay−EDD), weighted round robin (WRR), deficit round robin (DRR)

Non work conserving: Hierarchical round robin (HRR), Jitter−EDD, Stop−and−Go

Definition: A scheduler is sorted priority scheduler when it uses an artificial parameter, global (virtual) time associated to each link, to calculate a time stamp

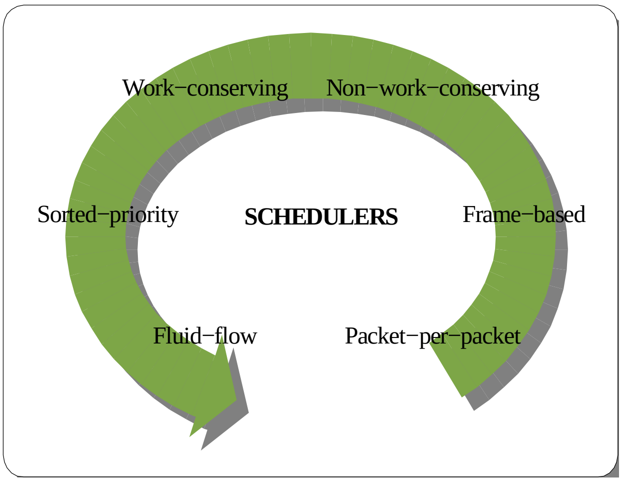

Figure 6: Different types of schedulers

used in determining which packet is to be sent next, for each packet.

Definition: A scheduler is frame based when it divides time into frames of constant or variable size and serves relative (allocated) amount from all queues during one frame.

Sorted priority: VC, WFQ, Delay−EDD

Frame based: (Constant) HRR, Stop−and−Go (variable) WRR, DRR

7.1 Characteristics of good scheduling mechanism

1. Isolation between traffic components (connections, flows, classes ...) eq bandwidth guarantees. Scheduling algorithm must be able to isolate traffic from possibly misbehaving other traffic streams. This is necessary even when there is policing at the edge of the network due to there will likely be an accumulation of burstiness on the path through the network. (In real world different traffic streams are not completely independent and therefore laws of random processes are not valid).

2. Delay. Amount of end to end delay in the forwarding path which is related to the scheduling is significant. Therefore it should be minimized.

3. Fairness. Link capacity should be divided fairly. Notion of fairness may although differ quite much (you pay what you get, everybody gets equal, excess capacity is divided equally (proportionally ...).

4. Complexity. Good scheduling algorithm is easy to implement. It has low overhead and is even in some cases possible to implement in hardware.

5. Utilization. Scheduling algorithm must always utilize bandwidth efficiently.
6. Required state information. Good scheduling algorithm requires minimum state information. More state information scheduler needs more delay it usually produces and less high utilization it can reach in the network.
7. Protocol overhead. As explained earlier many queuing and scheduling algorithms need extra protocol overhead in order to accommodate packets in the queues and in the scheduling to put virtual finishing times and other priority information.

7.2 Link sharing models for Internet

Link sharing is the actual implementation of the scheduling combined with the queue management. The requirement of link sharing is the ability to share bandwidth on a link between multiple organizations, multiple protocol families and multiple traffic types. In this case we are not interested in individual flows which pass through the system rather we are interested in the aggregate behavior of the traffic. [13, 27]

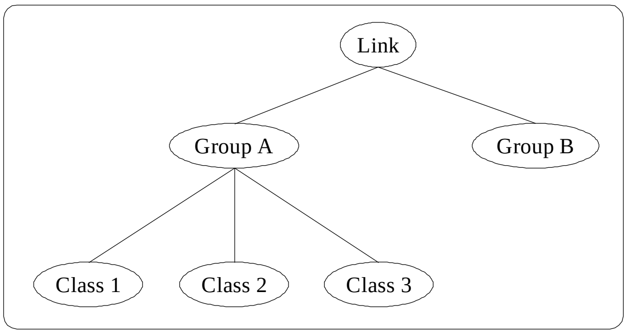

Figure 7: Link sharing concept

In hierarchical link sharing the goal is to offer tools which make sharing of a link or several links economically and functionally justifiable. The problem is nowadays with the different types of traffic: how to offer different types of applications forwarding treatment they need and not so much on the protocol families. In majority of cases the protocol is IP and shared links with different organizations, they come up only very few times.

7.2.1 CBQ scheme

In class based queuing (CBQ) scheme capacity is divided based on the link sharing concept. CBQ scheme uses general scheduler which is active during time when there is no congestion on any of the leaf classes. Then each of classes in link share structure is receiving resources they need and no pathological queuing occurs.

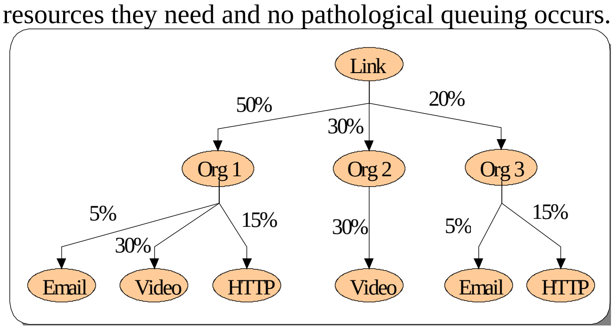

FIGURE 8: CBQ link share structure under no congestion

If and when some of the classes becomes unhappy a link share scheduler is activated. Link share scheduler is responsible on offering isolation between traffic classes so that allocated requirements for the individual classes can be met. Link sharing guidelines for contending situations vary based on service one is offering. Ancestor only link sharing offers to the unhappy class possibility to increase capacity as long as there is a capacity left from its parents. Top level link sharing modifies this approach by adding an parameter which reflects how many steps this 'borrowing' of capacity can go. In link sharing structure where there is many levels a borrowing of bandwidth may be limited to some top level after which all branches of possibly different organizations have to be satisfied.

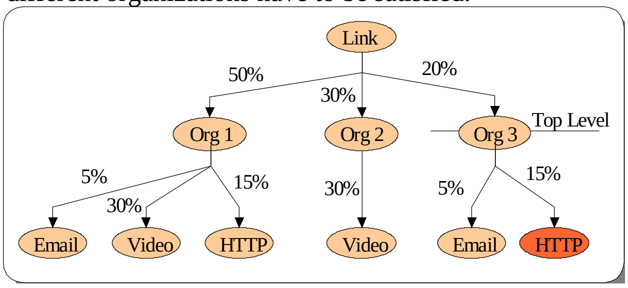

FIGURE 9: CBQ link share structure under congestion; horizontal line represents TopLevel parameter

In CBQ the general scheduler within a priority class is freely chosen. Implementations of CBQ use weighted round robin (WRR) and packet by packet round robin (PRR) based on their relatively low complexity in computation. As seen operation of CBQ link share scheduler is heuristically simple but algorithms implementing it are not so straight forward.

7.2.2 CSZ Scheme

In the CSZ scheme [13], link is not divided to multiple organizations.. The goal of the scheme is to offer reasonable segregation an isolation to different traffic classes. In CSZ guaranteed service is provided by the WFQ scheduling [12]. WFQ assigns a share of link capacity to each active flow; the admission control criterion is merely that the sum of the previously assigned bandwidths plus the bandwidth requested by the prospective flow does not exceed link capacity.

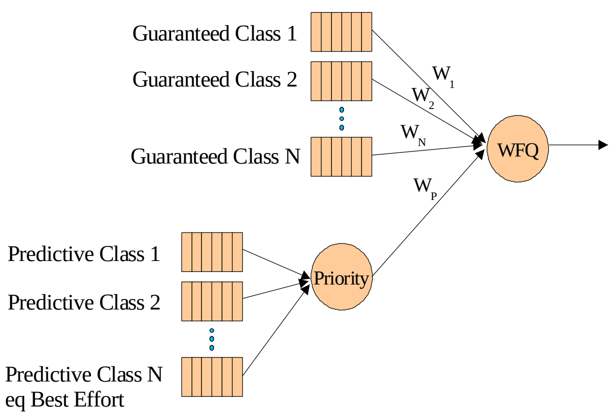

FIGURE 10: SCZ scheduler with proper service categories

The scheduling discipline for predictive service is a priority queue, the scheduler attempts to minimize the maximal (min/max.) delays actually experienced in each class, but does not guarantee an absolute maximum delay bound. Because of the min/max. scheduler, for the same amount of bandwidth reserved, the predictive service users will see lower delay than the guaranteed service users (No context switching delays between queues and time window slicing). Under the CSZ model, a switch can support multiple levels of predictive service, each with its own delay bound. General criteria is that the delay bounds of different levels of predictive service should be on the order of magnitude apart.

8. Admission Control

Connection admission control or admission control is the element which tries to preserve controlled operation in the network by a priori allocation of resources. Resources in the network (meaning bandwidth, processing and buffering) are divided to each connection based on their requests. A variety of different admission schemes have been proposed in the literature [1−5, 7−9, 15−20, 22−23, 31]. Some of these schemes require an explicit traffic model, some only require traffic parameters such as the peak and average rates and some rely on the measurements taken out of the network.

The general reason why this type of resource separation and allocation is required is based on the notion that multimedia applications require bounded delay packet delivery. The ability of bounded delay services to achieve high utilization and also meet their service commitments depends crucially on their admission control algorithm. Conversely, the ability of an admission control algorithm to increase network utilization is ultimately constrained by the service commitments the network makes.

8.1 Stochastic admission control

Stochastic admission control is based on a priori characterization of connections and calculations based on these a priori values. These admission control mechanisms use conservative approach to the allocations of resources, since they don't have any knowledge of how network is actually loaded rather how it should be loaded if all of the connections use resources exactly the way they have claimed.

Stochastic admission control algorithms have been studied for the ATM and Internet for a long time and numerous papers and books have written about those.

In general they can be divided into classes based on the approach they take to parameter calculation, load and loss estimation.

- Effective bandwidth based, a single nominal value which satisfies connections QoS demand is calculated based on the traffic parameters and some worst case traffic profile.
- Fluid flow approximation based, A fluid− flow model characterizes traffic as a Markov modulated continuous stream of bits with peak and mean rates. An effective bandwidth of the connection is calculated based on this approximation.
- Gaussian approximation based, similar as before but in the calculation of capacities a gaussian distribution of the aggregate rate is expected. This leads to taking variance also to the consideration. Overall admission is based on the probability that rate distributions tail passes link capacity only allowed portion of the time.
- Large deviation approximation based, resource saturation is used as the criteria instead of QoS parameters of connections. In large deviation algorithms the focal point of the resource calculation is shifted to the region of resource saturation.
- Effective variance based, effect of burstiness is taken as key criteria in admission control, more burstiness less predictable is the behavior of aggregate traffic stream and therefore less multiplexing is allowed in the admission (lower utilization target).
- Convolution based, rate distribution of connections is deduced from the connection parameters. Overall admission is based on the aggregate rate distribution which is calculated through the convolution of each individual connection.

8.2 Measurement based admission control

There are several proposals for the measurement based admission control to various different service models [1−9, 23, 31, 43, 45]. Common to these techniques is that they try to approach the problem of admission control through some level of dynamics. Dynamics is achieved through the measurements which are used for some parameter estimation or model estimation.

8.2.1 Model estimation

Model estimation is based on the notion that measuring interarrival distribution gives better view than single parameter estimation. Model estimation is used either to calculate exact delays and losses through the convolution of the measured distribution and assumed worst case distribution of the new connection request [45], to update gaussian distribution mean and variance values through the measurements and using worst case assumption in the connection request [43] or estimating same parameters in the rate envelope [43].

Rate envelope is interesting concept from the fact that it uses very light calculation mechanism. In rate envelope transmission rate of the aggregated traffic is measured in different length windows introducing temporal behavior of the aggregate traffic. This rate envelope is used with the new conection source parameters to estimate possible losses.

8.2.2 Parameter estimation

Parameter estimation mechanisms are some what similar to the gaussian distribution approximation. Parameter on the other hand can be also the delay.

Estimate of the load can be done in, at least, two different ways, using time window mechanism or by using exponentially weighted moving average filter.

The time window measurement process uses 2 parameters, T and S. T is the measurement window and S is the sampling period. So that single window is divided into multiple periods. During a sampling period an average load is computed. This average load is simply the sum of bytes in packets receiving service from the class measured divided by the length of the sampling period.

The load estimate is updated based on following conditions:

- At the end of every measurement window estimate of the load is set to the highest average load computed for any sampling period during the previous window.
- If a newly computed average load for a given sampling period is larger than the current value of the load estimate, estimate is set to the newly computed average.
- Whenever a new flow is admitted, the measurement estimate is immediately increased by the token bucket rate of the newly admitted flow.

Delay estimate of a class is produced by examining queuing delay. Estimate is measured in windows lasting T time units. Each packet during this period is examined and estimate of delay is updated.

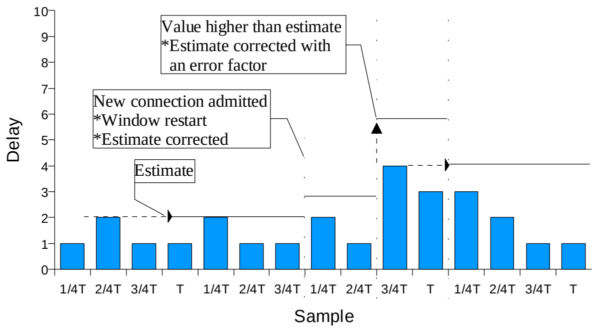

FIGURE 11: Computation ofdelay estimate for a class

Delay estimate is updated:

- At the end of period T to the maximum value noticed in previous period
- When a new connection is admitted, to the value of previous estimate plus token bucket delay of a new connection.
- When a sample higher than estimate occurs, to the value of an error factor times the sample value.

8.3 JSCZ admission control

In this chapter an admission control algorithm for out example environment is presented. Our example environment has so far service model presented in figure 4 and scheduling system presented in figure 10.

8.3.1 Predictive service

Following inequalities must hold for the requesting connection α at the predictive service class *k* in order to be admitted in service.

$$\nu\mu > r_k^{\alpha} + \hat{\nu}_G + \sum_{i=1}^{N} \hat{\nu}_i \qquad (1)$$

Stating that overall target link utilization is not allowed to be exceeded by requesting connection.

$$D_k > \hat{D}_k \cdot \frac{b_k^{\alpha}}{\mu - \hat{\nu}_G - \sum_{i=1}^{k-1} \hat{\nu}_i} \qquad (2)$$

Stating that delay bounds on same priority level are not allowed to be violated by requesting connection.

$$D_j > \hat{D}_j \cdot \frac{\mu - \hat{\nu}_G - \sum_{i=1}^{j-1} \hat{\nu}_i}{\mu - \hat{\nu}_G - \sum_{i=1}^{j-1} \hat{\nu}_i - r_k^{\alpha}} + \frac{b_k^{\alpha}}{\mu - \hat{\nu}_G - \sum_{i=1}^{k-1} \hat{\nu}_i - r_k^{\alpha}} \quad k < j \leq K \qquad (3)$$

Stating that new connection is not allowed to cause any delay violation to the connections in lower priority levels.

8.3.2 Guaranteed service

Following inequalities must hold for admitting new connection to the guaranteed service class.

$$\nu\mu > r_G^{\alpha} + \nu_G \qquad (1)$$

Stating that aggregate utilization in the guaranteed service class is not allowed to exceed link utilization target for that class.

$$D_j > \hat{D}_j \cdot \frac{\mu - \hat{\nu}_G - \sum_{i=1}^{j-1} \hat{\nu}_i}{\mu - \hat{\nu}_G - \sum_{i=1}^{j-1} \hat{\nu}_i - r_G^{\alpha}} \quad 1 \leqslant j \leqslant K \qquad (2)$$

Stating that no delay bounds is allowed to be violated in any of predictive service classes if their bandwidth is reduced by the amount of the new guaranteed service class connection.

9. Traffic shaping

Traffic shaping is mechanism where burstiness of the source is artificially altered to the lower values. This lowering of burstiness may happen inside the end system or it may be done in the access node of the network.

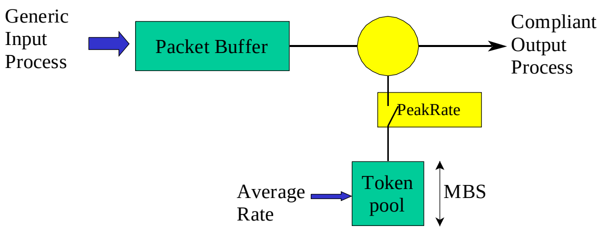

Figure 12: Generic traffic shaper

Why traffic should be shaped ? What can we achieve through shaping ? Answers for these questions are not obvious but if one studies the effects of bursts for a queue the effect becomes more obvious.

What traffic shaping actually does is that it delays or removes packets in a burst so that they form a constant flow (or bursts of widely spaced packets) or it alters the environment so that traffic becomes self smoothed. This flow while it passes the network goes through several queues which in many cases are partly congested. In a congested queue a mechanisms like random early detection (RED) and other active queue management methods choose connections which seem to be responsible ones for creating the congestion. How this is done is actually by monitoring occupancy of queue related to different flows (fair buffer allocations) or by random pick (RED) which usually hits the one most utilizing buffer space. [6, 21, 23]

So we see that if we space our packets as far as we can from the previous ones we are likely to have lower values of packet loss. This is one side of the picture. The other side of the picture is the delay. End to end delay of shaped traffic is usually lower than the other which is not shaped. This is due to the fact that many scheduling algorithms slice time for the different classes or connections so that traffic becomes somewhat self smoothed. During these occasion packets may stay in queues for an extensive long periods of time. Other component also having similar attitude to burstiness is queue management algorithm which usually favors flows behaving nicely eq not having too high burstiness.

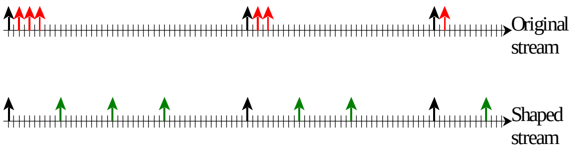

Figure 13Effect of shaping

How shaping then shows to the user ?

We saw earlier the effect of losses and delays. But if we don’t care losses and delays why should we then shape our traffic. At this point laws of the economics come to the picture. Operator having admission control based network can offer connections with better price for a connection using traffic shaping. Follow the money plays role at this point.

Because many important applications use TCP to guarantee delivery, there is a window of opportunity to use TCP’s windowing in traffic shaping. What TCP’s windowing does is just aggregating traffic into the burst. This is just the opposite what we are trying to do and what we can do about it is to rewrite the TCP windowing header information as traffic goes by. This leads to constant low window sending with relatively small bursts.

Other options are to use buffers is end systems or to change the way scheduling is done in the end system, meaning that network connection is scheduled open for application only for a fraction of time eq for a packet time.

Conclusions

As you have seen everything relates to everything. One can look a small detail of scheduling or admission control out of the concept of whole Internet, but this leads to path of investigating problem which has no relevance to the overall performance. Internet as decentralized it is operates with great interaction among different elements of the traffic control.

You have seen that application requirements push Internet to implement service model which offers three to four different classes of service, for which different characteristics are important. In order to satisfy the requirements of some of those classes some sort of admission control has to be implemented; trend is pushing to dynamic world and specially measurement based direction. Each admission control method relates heavily to the scheduling done in the network. Many of the scheduling mechanisms need traffic to be smoothed in order to provide delay guarantees.

Over all this presentation has opened some thoughts and ideas which you hopefully see interesting. There is lot of things to do in this area and definitely many mountains to conger.

References

1. Casetti, C., Kurose, J., and Towsley, D., An Adaptive Algorithm for Measurement−based Admission Control in Integrated Services Packet Networks, Technical Report TR 96−76. This is the tech report version of a paper that appeared in the October 1996 Workshop for High−Speed Networks.
2. Floyd, S., Comments on Measurement−based Admissions Control for Controlled−Load Services. July 1996. submitted to Computer Communications Review.
3. Gibbens, R., and Kelly, F., Measurement−based connection admission control, 15th International Teletraffic Congress Proceedings, June 1997.
4. Jamin, S., Shenker, S., and Danzig, P., Comparison of Measurement−based Admission Control Algorithms for Controlled−Load Service, Infocom 97.
5. Jamin, S., Danzig, P., Shenker, S., and Zhang, L., A Measurement−based Admission Control Algorithm for Integrated Services Packet Networks, IEEE/ACM Transactions on Networking, Dec. 1996.
6. Knightly, E and Rossaro, P: Improving QoS through Traffic Smoothing, In Proceedings of IFIP IWQoS’96, Paris, France.
7. Tse, D. and Grossglauser, M: "Measurement−Based Call Admission Control: Analysis and Simulations," Proc. IEEE INFOCOM, Kobe, Japan, April 1997.
8. Grossglauser M and Tse, D: "A Framework for Robust Measurement−Based Admission Control," Proc. ACM SIGCOMM, Cannes, France, August 1997.
9. Jamin, S., Danzig, P., Shenker, S. and Zhang, L:. A measurement−based admission control algorithm for integrated services packet networks. In SIGCOMM Symposium on Communications Architectures and Protocols, Cambridge, Massachusetts, September 1995.
10. Clark, D., Shenker, S. and Zhang, L: Supporting real−time applications in an integrated services packet network: architecture and mechanism. In SIGCOMM Symposium on Communications Architectures and Protocols, pages 14−26, Baltimore, Maryland, August 1992. ACM. Computer Communication Review, Volume 22, Number 4.
11. Schulzrinne, H., Kurose, J. and Towsley, D: An evaluation of scheduling mechanisms for providing

best−effort, real−time communication in wide−area networks. In Proceedings of the Conference on Computer Communications (IEEE Infocom), Toronto, Canada, June 1994.

12. A. Demers, S. Keshav, and S.J. Shenker. Analysis and Simulation of a Fair Queueing Algorithm". Proc. of ACM SIGCOMM '89, Sept. 1989.
13. S.J. Shenker, D.D. Clark, and L. Zhang. A Scheduling Service Model and a Scheduling Architecture for an Integrated Services Packet Network. URL ftp://parcftp.parc.xerox.com/pub/net−research/archfin.ps, 1993.
14. S. Floyd and V. Jacobson. Link−Sharing and Resource Management Models for Packet Networks". ACM/IEEE Transactions on Networking, 3(4):365−386, Aug. 1995.
15. H. Saito and K. Shiomoto. Dynamic Call Admission Control in ATM Networks". IEEE Journal of Selected Areas in Communication, 9(7):982−989, Sept. 1991.
16. H. Saito. Call Admission Control in an ATM Network Using Upper Bound of Cell Loss Probability". IEEE Transactions on Communications, 40(9):1512−1521, Sept. 1992.
17. G. Kesidis, J. Walrand, and C−S. Chang. Effective Bandwidths for Multiclass Markov Fluids and Other ATM Sources". ACM/IEEE Trans− actions on Networking, 1(4):424−428, Aug. 1993.
18. R. Guerin, H. Ahmadi, and M. Naghshineh. Equivalent Capacity and Its Application to Bandwidth Allocation in High−Speed Networks". IEEE Journal of Selected Areas in Communication, 9(7):968−981, Sept. 1991.
19. Z. Dziong and L.G. Mason. Fair−Efficient Call Admission Control Policies for Broadband Networks: A Game Theoretic Framework". ACM/IEEE Transactions on Networking, 4(1):123−136, Feb. 1996.
20. Harry G. Perros and Khaled M. Elsayed: Call Admission Control Schemes: A Review, IEEE Communications Magazine, November 1996.
21. G. de Veciana and J. Walrand: Traffic Shaping for ATM Networks: Asymptotic Analysis and Simulations, UCB/ERL Technical Report M92/135.
22. Saranka Sari: Connection Admission Control in an ATM Node, Master's thesis, Helsinki 1996.
23. Kalevi Kilkki: Traffic characterization and Connection Admission Control in ATM Networks, PhD thesis, Espoo 1994.
24. D. Stiliadis amd A. Varma: Latency−Rate Servers: A General Model for Analysis of Traffic Scheduling Algorithms, UCSC−CRL−95−38, July 1995.
25. D. Stiliadis amd A. Varma: Design and Analysis of Frame−based Fair Queueing: A New Traffic Scheduling Algorithm for Packet Switched Networks, SIGMETRICS'96.
26. S. Floyd and M. Speer: Experimental Results for Class−Based Queueing, Draft paper, November 1998.
27. Floyd, S., and Jacobson, V., Random Early Detection gateways for Congestion Avoidance V.1 N.4, August 1993, p. 397−413.
28. Parris M, Jeffays K and Smith F: Lightweight Active Router−Queue Management for Multimedia Networking, In proceedings of SPIE Multimedia Computing and Networking '99, Volume 3020, January 1999.
29. B. Suter, T.V. Lakshman, D. Stiliadis, A. Choudhury, "Efficient Active Queue Management for Internet Routers," Proceedings of Interop'98
30. Luoma, Marko., Peuhkuri, Markus., Yletyinen, Tomi. Quality of service for IP voice services. Voice, Video & Data Communications '98.
31. J. Qiu, "Measurement−based admission control in integrated−services networks", Master Thesis, May, 1998.
32. M Luoma & M Ilvesmäki: Simplified Management of ATM traffic, SPIE Voice, Video & Data Communications '97, November 1997.
33. Saito H: Teletraffic technoogies in ATM Networks, Artech House 1994.