Achieving Flow Level QoS in Cut-through Networks through Admission Control and DiffServ

Frank Olaf Sem-Jacobsen, Sven-Arne Reinemo*, Tor Skeie, and Olav Lysne
Simula Research Laboratory
P.O.Box 134, N-1325 Lysaker, Norway
frankose@ifi.uio.no, {svenar, tskeie, olavly}@simula.no

Abstract*— Cluster networks will serve as the future access networks for multimedia streaming, massive multiplayer online gaming, e-commerce, network storage etc. And for those application areas provisioning of Quality of Service (QoS) is becoming and important issue. DiffServ as specified by the IETF is foreseen to be the most prominent concept for providing predictability in the future Internet. To enable seamless interoperation with the higher level IETF concepts the QoS architecture of the lower layers should comply with the DiffServ paradigm as well. Previous work on predictability in cut-through networks has only studied** class **based QoS. In this paper we set out to achieve** flow **level QoS using flow aware admission control in combination with a flow negligent DiffServ inspired QoS mechanism. Our results show that flow level bandwidth guarantees are achievable with the use of the Link-by-Link and the Probe based schemes. In addition we are able to achieve an order of magnitude improvement in jitter and latency in individual flows.

I. Introduction

As the global Internet has evolved into a marketplace with a wealth of applications, the performance demands on the servers running these applications has grown too large to be handled by a single machine alone. This has resulted in a move from single server applications to applications running on a cluster of machines. Furthermore, new challenges have appeared, one of these are the interconnection of computers in a cluster, another is how to achieve predictable communication between machines in a cluster. This has renewed the focus on Quality of Service and resulted in several new technologies for System and Local Area Networking (SAN/LAN) [3], [4], [7], [12], [15], [21].

IETF has for several years provided the Internet community with QoS concepts and mechanisms. The best known ones are Integrated Services (IntServ) [8], Resource Reservation Protocol (RSVP) [11], and Differentiated Services (DiffServ) [5], [9]. In DiffServ QoS is realized by giving data packets differentiated treatment relative to QoS header information. In the underlying network technologies QoS has to a less extent been emphasised - the key metrics here have mainly been mean throughput and latency. To provide QoS end-to-end, possibly over heterogeneous technologies this means that the lower layers should also have support for predictable transfer including the ability to interoperate with a higher level IETF concept. This issue is being challenged by emerging SAN/LAN standards, such as InfiniBand^{TM} [4] and Gigabit Ethernet [15] providing various QoS mechanisms.

Recently we have also seen several research contributions to this field. In [10] Pelissier gives an introduction to the set of QoS mechanisms offered by IBA and the support for DiffServ over IBA. In this approach the presence of admission control is assumed. Alfaro et. al build on this scheme and present a strategy for computing the arbitration tables of IBA networks, moreover a methodology for weighting of virtual layers referring to the dual arbitrator defined by IBA [2]. The concept is evaluated through simulations assuming that only bandwidth sensitive traffic requests QoS. In [1] Alfaro et. al also include time sensitive traffic, besides calculating the worst case latency through various types of switching architectures.

Following the DiffServ philosophy no core switch should hold status information about passing-through traffic. Neither should there be any explicit signalling on a per flow basis to these components. This means that within the DiffServ framework any admission control or policing functionality would have to be implemented by boundary nodes or handled by a dedicated bandwidth broker. The core switches are assumed to perform traffic discrimination only based on service class, which is decided by a QoS tag included in the packet header - all packets carrying the same QoS tag will get equal treatment. From that viewpoint DiffServ is apparently a relative service model having difficulties giving absolute guarantees.

None of the previous debated contributions comply with the DiffServ model. In [10] Pelissier discusses interoperation between DiffServ and IBA on a traffic class and service level basis, but refer to RSVP with respect to admission control. The strategy proposed by Alfaro et. al has to recompute the IBA dual arbitrator every time that a new connection is honoured [1], [2]. Such a scheme is not associable with DiffServ. In [16] Reinemo et. al. studied the provision of QoS in cut-through networks by adhering to the DiffServ model. The problem was approached without any explicit admission control mechanism, as a pure relative model. Empirically they examined the sensitivity of different QoS properties under various load and traffic mixture conditions, hereunder assessing the effect of flow-control. This work was further studied in [20] where the concept described in [16] was extended with three different admission control mechanisms. Our contributions showed the feasibility of doing this at the *class level* (i.e. aggregated flows). One important question that we need to ask with regards to these results is “What happens

*Primary contact author.

to QoS at the *flow level*?". Even if things look good on the class level it might not look good on the flow level. The object of this paper is to study if we are able to achieve *flow level* QoS in cut-through networks by combining admission control with a class based scheme which is in compliance with the DiffServ paradigm. Specifically, we will have a QoS concept with flow aware admission control and flow negligent traffic classes. Empirically we will study the *throughput*, *latency* and *jitter* characteristics at the flow level, all in combination with three different admission control mechanisms each with a fundamentally different approach to admission control. Our first scheme assumes pre-knowledge of the network's performance behaviour without admission control and is implemented as a centralised bandwidth broker. Our second scheme is based on endpoint/egress measurements to assess the load situation, and our third scheme makes use of probe packets to assess the load situation.

Our results are important in two ways. Firstly, they are important as a means to achieve QoS in cut-through networks. Secondly, they are important to bridge QoS between the global Internet and a local cluster. If IETF standards such as DiffServ or IntServ are applied for some applications on the Internet we need ways to represent these QoS attributes on our cluster to be able to serve the application request according to their specified QoS.

The rest of this paper is organised as follows. In section II we give a description of our QoS architecture, in section III our three admission control mechanisms are described, and in section IV our simulation scenario is described. We discuss our performance results in section V , and in section VI we finish off with some concluding remarks.

II. QoS Architecture

THE architecture used in our simulations is inspired by IBA link layer technology [4] and is a flit based virtual cut-through (VCT) switch. The overall design is based on the canonical router architecture described in [6].

In VCT the routing decision is made as soon as the header of the packet is received and if the necessary resources are available the rest of the packet is forwarded directly to the destination link [14]. If the necessary resources are busy the packet is buffered in the switch. In addition we use flow control on all links so all data is organised as flow control digits (flits) at the lowest level.

The switch core consists of a crossbar where each link and VL has dedicated access to the crossbar. Each link supports one or more virtual lanes (VL), where each VL has its own buffer resources which consist of an input buffer large enough to hold a packet and an output buffer large enough to hold two flits to increase performance. Output link arbitration is done in a round robin fashion.

To achieve QoS our switch architecture support QoS mechanisms similar to the ones found in the IBA architecture. IBA supports three mechanisms for QoS which are mapping of service level (SL) to VL, weighting of VLs and prioritising VLs as either low priority (LP) or high priority (HP). A more detailed description of these QoS aspects can be found in [16].

The routing used is a newly introduced routing algorithm called *Layered shortest path routing* (LASH) [13]. LASH is a minimal deterministic routing algorithm for irregular networks which only relies on the support of virtual layers. There is no need for any other functionality in the switch, so LASH fits well with our simple approach to QoS. An in-depth description of LASH is found in [13].

III. Admission Control

IN this section we propose three different admission control (AC) mechanisms that we carefully evaluate in section V.

A. Link by Link Based Admission Control

In the Link-by-Link (LL) approach a bandwidth broker (B) knows the load on every link in the network and will consult the availability of bandwidth on every link between source and destination before accepting or rejecting a flow. This solution assumes that both topology and routing information about the network is available.

For the AC decision we adopt the *simple sum* approach as presented in [19]. This algorithm states that a flow may be admitted if its peak rate p plus the peak rate of the already admitted flows s is less than the link bandwidth bw. Thus the requested flow will be admitted if $p+s < bw$ [19]. We deduce the effective bandwidth from the measurements obtained in [16]. Since we are dealing with service levels where each SL has different bandwidth requirements it is natural to introduce some sort of differentiation into the equation. We achieve this by dividing the link bandwidth into portions relative to the traffic load of the SLs, and include only the bandwidth available to a specific service level bw_{sl}. Giving $p+s_{sl} < bw_{sl}$, where $bw_{sl} = bw_{link} * \frac{load_{sl}}{load_{total}}$. And s_{sl} is the sum of the admitted peak rates for service level SL and bw_{link} is the effective link bandwidth.

B. Egress Based Admission Control

The Egress Based (EB) approach is a fully distributed AC scheme where the egress nodes are responsible for conducting the provisions. Basically, we adopt the Internet AC concept presented by Cetikaya and Knightly in [17]. This method does not assume any pre-knowledge of the network behaviour as was the case with the previous solution. Also different from the previous approach is the use of a delay bound as the primary AC parameter. For clarity we give a brief outline of the algorithm below, a more detailed description can be found in [18].

The method is entirely measurement based and relies on that the sending nodes timestamp all packets enabling the egress nodes to make two types of measurements. First, by dividing time into timeslots of length τ and counting the number of arriving packets, the egress nodes can deduce the arrival rate of packets in a specific timeslot. By computing the maximum arrival rate for increasingly longer time intervals we get a peak rate arrival envelope $R(t)$, where $t = 0, ..., T$ timeslots, as described in [17]. Second, by comparing the originating timestamp relative to the arrival time, the egress node can calculate the transfer time of a packet. Having this information available the egress node can furthermore derive the time needed by the infrastructure to service k following packets; i.e. a consecutive stream of packets where the next packet in the service class enters the infrastructure before the previous packet has departed the egress node. By doing this for larger and larger k sequences of packets within a measuring interval of length $T\tau$ and subsequently inverting this function we achieve the service envelope $S(t)$, giving the amount of packets processed by the network in a given time interval t. Now repeating this M times, the mean $\overline{R}(t)$ and the variance $\sigma^2(t)$ of $R(t)$, and the mean $\overline{S}(t)$ and variance $\Psi^2(t)$ of $S(t)$ may be calculated. If a flow request has a peak rate P and a delay bound D it may be accepted if the peak rate P plus the measured arrival rate $R(t)$ is less than the service rate allowing for the delay $D, S(t + D)$.

The EB scheme will admit as much traffic as it can without breaking the delay bound. The key instrument of the scheme is the given delay bound for the different flows, and the efficiency of the algorithm is linked to its ability to limit the service levels to operate within the delay bounds.

C. Probe Based Admission Control

As an alternative to passively monitoring the network activity in the egress nodes of the network, as was the case with the EB scheme, it is possible for the end nodes in the network to take a more active role in the AC decision. This can be done by actively sending probe packets through the network from source to destination and monitor the arrival of the probes at the egress of the network [25], [26]. If the size and rate of the probe packets is designed correctly they should give the egress node the opportunity to calculate how the new flow will be treated by the network. Several probing schemes have been proposed in the literature, some of which are described in [25], [26]. In [25] Bianchi et.al. propose a probing scheme where the load is inferred by measuring the jitter for the probe packets. They require that the probe packets are forwarded through the network with the lowest priority of all packets. This ensures that the probe packets will be unable to steal bandwidth from the already existing traffic in the network while additionally giving worst-case measurements of the network jitter and thus guaranteeing that the traffic, when admitted, will get at least the service of the probe packets. When applying this in our simulation scenario it is natural to let the probe traffic be forwarded on one of the low priority SLs with a relatively low weight, possibly equal to 1. The AC decision is based on $ttime_{max} - ttime_{min} < j$ and $p = 0$. Where $ttime$ is the transmission time and p is the number of packets rejected. For each probe packet received the receiver registers the packet's transmission time, e.g. the time the packet spends in the network. When an adequate amount of probe packets have been sent and received the receiver calculates the jitter by subtracting the minimum packet transmission time from the maximum packet transmission time. This value is compared to the jitter requirements embedded in the probe packets and an admission decision is sent back to the sender. If the perceived jitter was less than the requirement the flow is accepted, otherwise the flow is rejected. Additionally if any of the probe packets are rejected by the sender due to the limited size of the send queue buffer the flow is also rejected.

D. Target for Admission Control

The main findings for the work in [16] are that (i) throughput differentiation can be achieved by weighting of VLs and by classifying the VLs as either low or high priority, (ii) the balance between VL weighting and VL load is not crucial when the network is operating below the saturation level. In general this sets the target for the AC, since as long as we can ensure that the load of the various service classes is below saturation level we can also guarantee that each of these classes get the bandwidth they request. The target for admission control is thus the point where the amount of accepted traffic is starting to become less than the traffic offered. The effective bandwidth at this point will be used as a steering vehicle by the LL method. The success of this method is documented in [20].

Another main finding in [16] is that although the latency characteristics below saturation were fairly good, significant jitter was observed. We challenged this problem in [20] with the EB method, unfortunately the EB scheme was unable to give bandwidth guarantees and the latency results was slightly worse than the LL scheme at the class level. We have included the EB scheme for comparison only, and added the PB scheme in order to improve the latency and jitter performance.

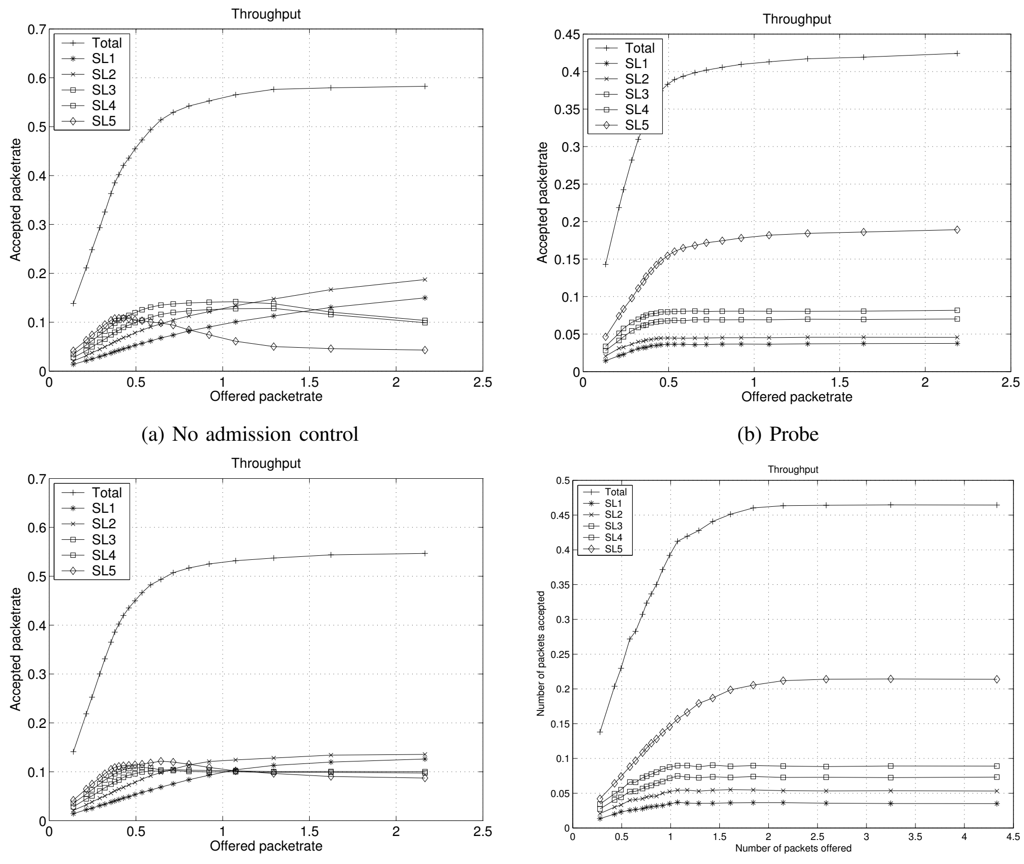
IV. Simulation Scenarios

FOR all simulations we have used a flit level simulator developed in house at Simula Research Laboratory.

We have performed simulations on a network with 32 switches, where each switch is connected to 5 end nodes and the maximum number of links per switch is 10 in addition to the end nodes. We have randomly generated 16 irregular topologies and we have run measurements on these topologies at increasing load. We use LASH as our routing algorithm and random pairs as our traffic pattern [13], [27]. In the random pairs scheme each source sends to *one* destination and no destination receives from more that *one* source. The link speed is one flit per cycle, the flit size is one byte and the packet size is 32 bytes for all packets.

The five different end nodes send traffic on one of five different service levels. One service level for each node (Table I), SL 1 and 2 are considered to be of the expedited forwarding (EF) class in DiffServ terminology,SL 3 and 4 are considered to be of the assured forwarding (AF) class. SL 5 is considered as best effort (BE) traffic and from that viewpoint is not a subject for AC.

For the LL scheme all simulations where run with a target load deduced from our measurements in [16]. In the first part of the simulation the send rate is steadily increased by adding more and more flows until admission is denied by the AC scheme. When this happens the current rate is not changed, but the node will try to add more flows for a fixed number of times before it gives up. For the EB scheme the send rate is increased in the same way, but the AC decision is primarily



(a) No admission control (b) Probe

(c) Egress (d) Link by link

Fig. 1. Average class throughput

SERVICE LEVELS				
SL	**DS[1]**	**Load %**	**BW[2]**	**Pri**
1	EF	10	4	high
2	EF	15	6	high
3	AF	20	8	low
4	AF	25	10	low
5	BE	30	1	low

TABLE I

THE FIVE SERVICES LEVELS USED IN SIMULATION.

based on measured latency as described in section III-B. For the PB scheme the AC decision is based on the measured jitter for the probe packets as described in section III-C.

All simulations use traffic that is modelled to display self-similar behaviour according to [24]. Analyses of real-life network traffic traces have shown that the packet arrival rate in a network is not totally independent of the arrival of any other packet such as in a Poisson process. Instead the arrival rate display a degree of self similarity where the traffic is repeated on smaller and larger time scales in accordance with fractal theory [22], [23]. In [24] Willinger et.al. show that an aggregation of Pareto distributed on/off sources are within the necessary mathematical criterion to produce self-similar traffic. This is the approach we have adopted in this paper.

[1]The DiffServ equivalent service class.

[2]The maximum number of flits allow to transmit when scheduled.

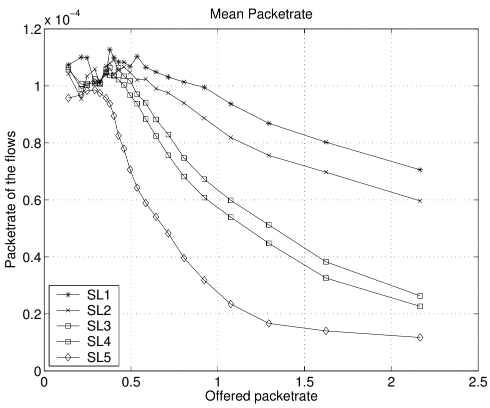
V. Performance Results

We will now discuss our results with regards to throughput, latency and jitter, all in that order.

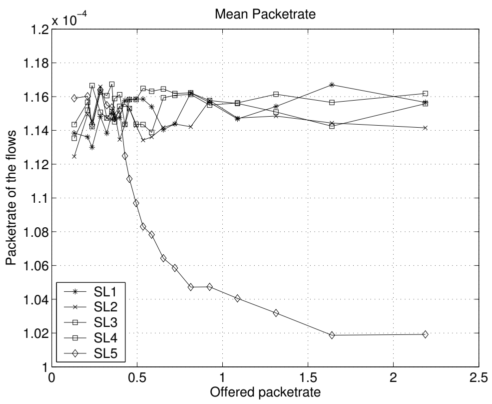
A. Throughput

As was presented in section IV our traffic is divided into five different classes. The throughput results for these classes are presented in figure 1, while throughput results for flows are presented in figure 2. Figure 1(a) shows the throughput achieved without any form of admission control (NoAC). We observe that we are unable to give all service classes the requested bandwidth as we enter saturation. Furthermore, the high priority (HP) classes preempts the low priority (LP) classes, i.e. the bandwidth differentiation is no longer according to the percentages in Table I. This behaviour is reflected at the flow level in figure 2(a) where we see that the throughput per flow is decreased as the number of flows is increased. With this in mind we will evaluate each of our proposed AC schemes.

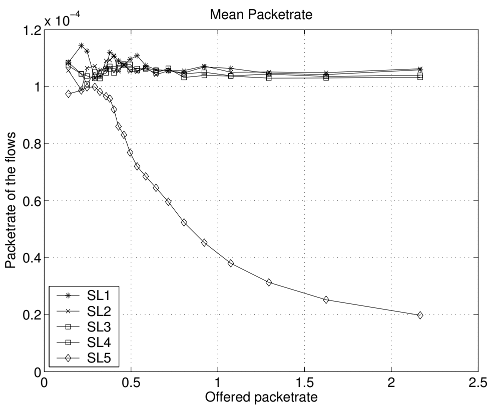
Our first candidate is the probe based (PB) scheme where jitter is the primary AC parameter. From figure 1(b) we observe that this scheme performs very well. The admission control decision is very precise about when to accept and reject traffic and we see bandwidth differentiation that is relative to the actual requests. In other words we are able to give bandwidth guarantees with this scheme. Looking at flow level



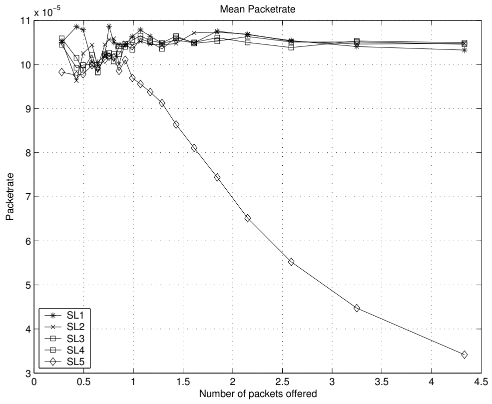
(a) No admission control



(b) Probe



(c) Egress



(d) Link by link

Fig. 2. Average flow packetrate

throughput we see from figure 2(b) that all flows get the requested bandwidth at the cost of less bandwidth for best effort traffic in SL5.

The next candidate is the EB scheme using latency as the primary AC parameter. It is apparent from figure 1(c) that the EB is unable to give bandwidth guarantees at the class level. The load is allowed to increase beyond the saturation point and admits too much traffic. In addition the differentiation between SLs deteriorates as the load increases. The poor performance[3] of the EB scheme can be ascribed to the use of delay as the primary AC parameter, this results in the bandwidth requirements being ignored and that the number of flows accepted in each class is not differentiated in relation to the actual requests.

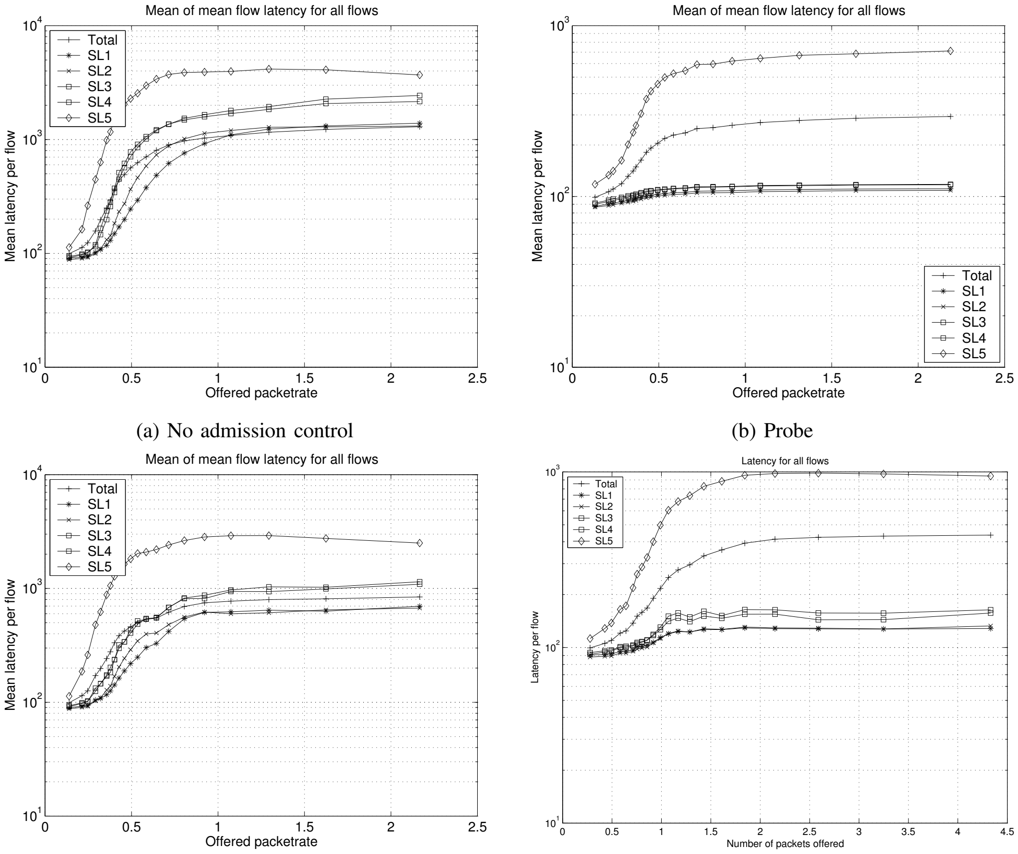
Our final candidate is the LL scheme which use bandwidth as the primary AC parameter. The LL scheme presented in figure 1(d) shows several improvements compared to the mediocre performance of the EB method. It is in fact as good as the PB approach as we achieve a bandwidth differentiation which is relative to the actual requests, meaning that we can give bandwidth guarantees. Moreover, we are able to utilise the network resource well since we go close to the saturation point without passing it. The good performance of the LL can be attributed to the fact that it knows the load of every link in the network and is able to make the AC decision based on the load along the actual source/destination path. At the flow level the LL approach is as good as the PB scheme. It is able to give all flows the requested bandwidth at the cost of less bandwidth available to best effort traffic in SL5.

[3]As a sidenote, the performance of the EB scheme is worse when using self-similair compared to the use of a Poisson process as was the case in [20].

B. Latency

Now we will study the ability to guarantee latency at the flow level. Typically we want to have low latency for flows in the high priority SLs, while we accept higher latency values for the low priority SLs. For the best effort traffic in SL5 we do not try to meet any latency requirements.

Figure 3(a) shows the average per flow latency for increasing load values without admission control. Comparing this with the results from PB in figure 3(b) shows that there is an improvement in latency of about one order of magnitude for HP flows. For the EB scheme in figure 3(c) the results are similar, but the improved latency is not as low as is the case for the PB method. On the other hand the differentiation between flows in different SLs is better in this scheme. HP SLs stabilize at 700 and LP SLs stabilize at 1000 cycles. For the PB scheme both HP and LP SLs stabilize at 100. Still, the EB scheme is unable to achieve latency as low as the other schemes even if its using latency as the primary AC parameter. The reason for this is probably the unpredictable latency characteristics in cut-through networks as observed in



(a) No admission control

(b) Probe

(c) Egress

(d) Link by link

Fig. 3. Average flow latency

[16]. Finally the LL scheme is able to get an improvement in latency on par with the PB scheme. From figure 3(d) we see an improvement in latency of more than one order of magnitude. It is also able to differentiate better between flows in different SLs than both the PB and EB scheme. The good performance of the LL scheme can be ascribed to its detailed knowledge about the network.

C. Jitter

Our third QoS attribute is the variation in latency, also referred to as jitter. We want our jitter to be as low as possible and to better evaluate this we have plotted the maximum observed jitter for all our AC schemes in figure 4. The plots contain the maximum per flow jitter observed throughout the simulation run.

The NoAC results show that there is substantial increase in jitter for all flows even at very low load. With the introduction of the PB scheme jitter is reduced by one order of magnitude for high priority flows (figure 4(b)). For low priority flows the reduction is slightly less. The primary AC parameter for this method is jitter and thus it performs well. Still, even if we are able to reduce jitter significantly our guarantees are coarse since jittering is still in the order of 300 to 600 cycles.

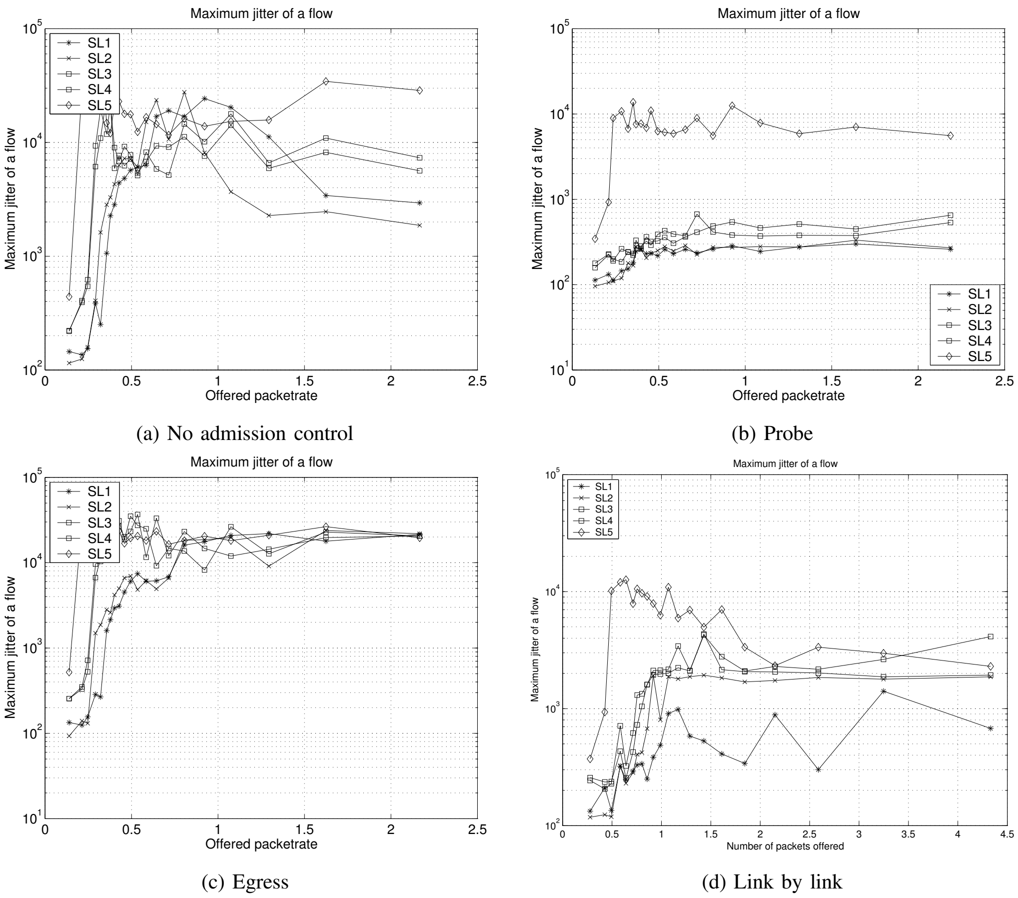
The EB scheme performs worse with a jitter of 12000 cycles regardless of SLs. This is understandable as the EB method focus on latency instead of jitter.

Moving on to our last candidate, the LL mechanism in figure 4(d), we see an improvement in overall jitter. But the improvement is worse than what is the case for both the EB and PB scheme. Furthermore, the jitter in flows from SL1 and SL4 shows a large amount of variation compared to both EB and PB approach. This is probably caused by the fact that this scheme ignores latency and jitter properties and only concentrates on throughput when making the AC decisions.

VI. Conclusion

In this paper we set out to achieve flow level QoS with regards to throughput, latency and jitter. Towards this goal we have evaluated three different admission control schemes for virtual cut-through networks. One is a probe based scheme using jitter as the primary AC parameter, another is a measurements based approach using lantecy as the primary AC parameter and yet another is a centralised bandwidth broker approach using pre-knowledge of the network link load without admission control as the primary AC parameter.

Our contributions are as follows. First, flow level bandwidth guarantees are achievable with the use of the Link-by-Link and the Probe based schemes, while the Egress Based method is unable to achieve good guarantees. Second, improved per flow latency and jitter properties are achievable with both the Probe and Egress based methods, but strict guarantees are hard to give since jitter is still high. Overall, the Probe based scheme gives us the best performance with regards to throughput,



(a) No admission control

(b) Probe

(c) Egress

(d) Link by link

Fig. 4. Maximum flow jitter

latency and jitter. The final conclusion is that we are able to achieve flow level QoS with a combination of DiffServ and admission control in cut-through networks.

REFERENCES

[1] F. J. Alfaro, J. L. Sanchez, J. Duato, and C. R. Das. A strategy to compute the InfiniBand arbitration tables. In *Proceedings of IPDPS*, April 2002.

[2] F. J. Alfaro, J. L. Sanchez, and J. Duato. A strategy to manage time sensitive traffic in InfiniBand. In *Proceedings of Workshop on Communication Architecture for Clusters (CAC)*, April 2002.

[3] N. J. Boden, D. Cohen, R. E. Felderman, A. E. Kulawik, C. L. Seitz, J. N. Seizovic, and W. K. Su. Myrinet – a gigabit-per-second LAN. IEEE MICRO, 1995.

[4] InfiniBand Trade Association. Infiniband architecture specification.

[5] Differentiated Services. RFC 2475.

[6] J. Duato, S. Yalamanchili, and L. Ni, *Interconnection Networks an engineering approach*, IEEE Computer Society, 1997.

[7] R. W. Horst. Tnet: A reliable SAN. *IEEE Micro*, 15(1):37–45, 1995.

[8] Integrated Services. RFC 1633.

[9] K. Kilkki. Differentiated services for the Internet. *Macmillian Tech. Publishing*, 1999.

[10] J. Pelissier. Providing quality of service over InfiniBand^{TM} architecture fabrics. In *Proceedings of Hot Interconnects X*, 2000.

[11] ReSource ReserVation Protocol. RFC 2205.

[12] M. D. Schroder et.al., "Autonet: a high-speed, self-configuring local area network using point-to-point links," SRC Research Report 59, Digital Equipment Corporation, 1990.

[13] T. Skeie, O. Lysne, and I. Theiss. Layered shortest path (LASH) routing in irregular system area networks. In *Proceedings of CAC*, 2002.

[14] P. Kermani and L. Kleinrock, "Virtual Cut-through: A New Computer Communication Switching Technique," in *Computer Networks*, no. 4, vol. 3, 1979.

[15] R. Seifert, *Gigabit Ethernet,* Addison Wesley Pub Co., 1998.

[16] S. A. Reinemo and T. Skeie and O. Lysne, "Applying the DiffServ Model in Cut-through Networks," in *Proceedings of the 2003 International Conference on PDPTA*, 2003.

[17] C. Cetikaya and E. W. Knightly, "Egress admission control," in *INFOCOM (3)*, pages 1471-1480, 2000.

[18] J. Schlembach and A. Skoe and P. Yuan and E. Knightly, "Design and Implementation of Scalable Admission Control," in *QoS-IP*, pages 1-15, 2001.

[19] S. Jamin and S. J. Shenker and P. B. Danzig, "Comparison of Measurement-Based Admission Control Algorithms for Controlled-Load Service," in *INFOCOM (3)*, pages 973-980, 1997.

[20] S. A. Reinemo and Frank Olaf Sem-Jacobsen and T. Skeie and O. Lysne, "Admission Control for DiffServ based Quality of Service in Cut-through Networks," in *Proceedings of the 10th International Conference on High Performance Computing, 2003.*

[21] C.Bell et.al., "An Evaluation of Current High-Performance Networks," in Proceedings of the IPDPS*, 2003.*

[22] W. E. Leland et.al., "On the self-similar nature of Ethernet traffic," in Proceedings of the ACM SIGCOMM*, 1993.*

[23] V. Paxson and S. Floyd, "Wide area traffic: The failure of Poisson modeling," in IEEE/ACM Trans. on Networking*, 3(3):226-244, 1995.*

[24] W. Willinger et.al. "Self-similarity through high-variability: statistical analysis of Ethernet LAN traffic at the source level," in IEEE/ACM Trans. on Networking*, 5(1):71-86, 1997.*

[25] G. Bianchi and F. Borgonovo and A. Capone and L. Fratta and C. Petrioli "Endpoint ad-mission control with delay variation measurements for qos in ip networks," in ACM SIGCOMM Computer Communications Review*, 32(2):61-69, 2002.*

[26] Lee Breslau and Edward W. Knightly and Scott Shenker and Ion Stoica and Hui Zhang "End-point admission control: architectural issues and performance," in Proceedings of the ACM SIGCOMM*, 2000.*

[27] Tor Skeie, Olav Lysne, Jose Flich, Pedro Lopez, Antonio Robles and Josè Duato LASH-TOR: A Generic Transition-Oriented Routing Algorithm Submitted to ICPADS 2004.