A Proven Delay Bound for a Network with Aggregate Scheduling

Jean-Yves Le Boudec

EPFL- DSC Technical Report DSC2000/002

January 10, 2000

Abstract

We consider a network implementing scheduling per aggregate flow. We show that the derivation of bounds that was published in [1] is false. We give a worst case bound which is valid for any network topology, given the maximum hop count for every flow and the utilization at every link.

1 Introduction

We consider a network serving a high-priority service class in an aggregate way. All packets belonging to the high-priority class are seen at every node as a single, aggregate flow. This is a model for the expedited forwarding (EF) service class, a delay priority mechanism proposed in the context of differentiated services.

We assume that every flow is individually policed at the network boundary by one leaky bucket, so that it is made conformant to an arrival curve of the form $a(t) = rt + b$. This is the only place where flows are handled separately; after entering the network, flows are seen as aggregates.

The high priority aggregate is processed at every node according to some scheduling policy. The scheduler handles all priority traffic as one flow. We follow [1] and assume that every node can be abstracted by a service curve property, of the form $\sigma(t) = S(t-\beta)^+$ [2, 3, 4, 5]. A simplified way of thinking of the service curve property is to assume that during every period of duration t for which there is some backlog of high priority traffic at the node, the amount of service received is at least $\sigma(t)$. For example, if the node implements head of the line, non-preemptive priority queuing, then S is the rate of the outgoing link and $\beta = \text{MTU} \times S$.

In [1], the author finds a bound for the end-to-end queuing delay, and claims that it is valid for any network topology, provided that we know the maximum number of hops undergone by any flow, and the maximum intensity of high priority traffic. However, we show in this paper that the derivation of the bound in [1] does not hold, due to a subtle flaw. We then give a correct bound. For an asymptotically small utilization, our bound coincides with that in [1]. However, for large utilization factors, it becomes infinite. Explosion of bounds for aggregate scheduling is not new [6, 7]; it is also known that good bounds for aggregate scheduling may depend on complex, global conditions [8, 9]. In this working document, we do not have results about the tightness of our bound.

2 A flaw in the derivation of Theorem 1 in [1]

In [1], the degree of a node is defined as the maximum hop count for flows that use the node. The hop count for a flow at a given node is the number of hops the flow has been going through from network entry to this node. Theorem 1 gives a bound which is claimed to be valid for any network topology. However, there is a flaw, between Equations (31) and (32) on page 23. The author states that for a flow i sharing a hop of degree $h+1$, the previous node on the path of flow i must be of degree h. This is not true. The previous hop may also be of degree $h+1$, or higher, because of other flows. It is straightforward to build a network where all nodes have the same degree, for example a ring network with k nodes where all flows go exactly n hops, $n \leq k$. The proof is thus incorrect. In order to correct the formula, we need to account for the fact that the degree of a previous hop is not necessarily decreased.

Another more subtle element is that we cannot a priori assume that a finite bound exists [7], because FIFO is not a universally stable scheduling method. A formula with a correct proof is given in the next section. It also provides a minor improvement, namely, it accounts for limits on the total incoming bit rate at very node.

3 A delay bound for a general network topology

Consider a network as defined previously, and call h the maximum hop count across all flows in the network. To simplify the mathematics here we assume that the service curve guaranteed to the aggregate high priority traffic is the same at all nodes; call it $\sigma(t) = (St - \beta)^+$.

We define b_{tot} as a bound on the sum of the burst tolerances of all individual flows using any particular link. More specifically, assume that for an individual flow i, the arrival curve enforced at the network access point is $a_i(t) = r_i t + b_i$. For a link l, call $\mathcal{S}(l)$ the set of flows that constitute the aggregate high priority traffic on link l. Then we must have

$$\sum_{i \in \mathcal{S}(l)} b_i \leq b_{tot}$$

We also assume that the total average traffic intensity on any link is bounded by αS, namely we must have, for all link l

$$\sum_{i \in \mathcal{S}(l)} r_i \leq \alpha S$$

The parameter α is thus a bound on the high priority class utilization factor.

Also, like in [1], define

$$\Delta = \frac{\beta}{S}$$

In [1], the parameter b_{tot} is not used. Instead, it is assumed there that all flows have the same arrival curve $a_i(t) = r_{min} t + b_{max}$, which is less general. We can map our result to the notation in [1] by letting

$$b_{tot} = \alpha S \frac{b_{min}}{r_{max}}$$

Finally, we assume that the peak rate of all incoming traffic at any link is bounded by some constant C. The model in [1] corresponds to $C = +\infty$. For a router with large internal speed and buffering only at the output, C is the sum of the bit rates of all incoming links. The delay bound is better for a smaller C.

Theorem 3.1. *If* $\alpha < \frac{C}{(C-S)(h-1)+S}$ *then a bound on the end-to-end delay for high priority traffic is*

$$D = \frac{h}{1 - u\alpha(h-1)} \left(\Delta + u \frac{b_{tot}}{S} \right)$$

with $u = \frac{C-S}{C-\alpha S}$.

The proof is given in appendix. We now compare our bound to Theorem 1 in [1]. The bound given there can be re-written as

$$\bar{D} = h(1+\alpha)^{h-1} \left(\Delta + \frac{b_{tot}}{S} \right)$$

α	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10	0.11	0.12
$\bar{D}$	3.58	6.90	10.80	15.34	20.59	26.65	33.60	41.53	50.55	60.77	72.33	85.34
D	3.60	7.04	11.33	16.83	24.13	34.29	49.39	74.19	122.50	257.74	2827.42	$+\infty$

Table 1: The bound $\bar{D}$ in [1] (whose proof is incorrect) versus the bound D in this paper (a correct bound), for Example 4 of [1]. $h = 10$, $b_i = 100$B for all flows, $r_i = 32$kb/s for all flows, $S = 155$Mb/s and $C = +\infty$.

In the case where $C = +\infty$, the condition on α is $\alpha < \frac{1}{h-1}$, we have $u = 1$ and our bound becomes

$$D = \frac{h}{1 - \alpha(h-1)} \left(\Delta + \frac{b_{tot}}{S} \right)$$

so that the two bounds differ only by the first term. A limited development up to the first order shows that both D and $\bar{D}$ can be written as

$$h(1 + (h-1)\alpha) \left(\Delta + \frac{b_{tot}}{S} \right) + o(\alpha)$$

where $o(\alpha)$ is some function of α that tends to 0 faster than α when α tends to 0, and all other parameters in the theorem are kept constant. Thus, when $C = +\infty$, the two bounds coincide for asymptotically small utilization factors.

Table 3 compares our bound for Example 4 of [1].

4 Discussion

Our bound explodes when α tends to $\frac{C}{(C-S)(h-1)+S}$, which is very unfortunate. It is not clear whether the delay does become unbounded for general topologies or whether it is an artifact of our bound. Explosion of bounds for aggregate scheduling is not new, and has been encountered for example in [6, 7].

Note that if C is close to S, then $\frac{C}{(C-S)(h-1)+S}$ is close to 1. This is the case for a high speed add-drop ring network, where the bit rate is much higher for the ring than for the input links. In that case, u tends to 0 and

$$D = h(\Delta + u\frac{b_{tot}}{S}) + o(u)$$

with $u = \frac{C-S}{C-\alpha S}$; this confirms that the explosion does not take place in that case.

It is also known that, in practice, worst case delays for FIFO networks are usually much less than predicted by simple bounding techniques. For example, in [8, 9], it is assumed that individual flows are shaped at network entry in such a way that the spacing between packets is at least equal to the route interference number (RIN) of the flow. The RIN is defined as the number of occurrences of a flow joining the path of some other flow. Then the end-to-end delay is bounded by the time to transmit a number of packets equal to the RIN. This usually results in much smaller bounds than the bounds here, albeit at the expense of more restrictions on the routes. If this is confirmed, this would indicate that a technology like MPLS is more suited for providing the guaranteed service than expedited forwarding.

A Proof of our bound

The proof consists in showing separately that (1) if a finite bound exists, then the formula in the Theorem is true and (2) that a finite bound exists.

Part 1: We assume that a finite bound exists. Call D' the worst case delay across all nodes, and h the maximum hop count for any flow in the network. Consider a buffer at some link l. An arrival curve for the aggregate traffic is

$$a(t) = \min\left(Ct, \sum_{j \in \mathcal{S}(l)} a_j(t + (h-1)D')\right)$$

The former part in the formula is because the total incoming bit rate is limited by C; the latter is because any flow reaching that node has undergone a delay bounded by $(h-1)D'$. Thus

$$a(t) \leq a'(t) = \min\left(Ct, \alpha St + b'\right)$$

with $b' = b_{tot} + \alpha S(h-1)D'$.

A bound on the delay at our node is given by the horizontal deviation between the arrival curve $a'(t)$ and the service curve $\sigma(t) = (St - \beta)^+$ [2, 3, 4, 5]. The alert reader will enjoy doing the computation and will find that a bound on the delay is $\Delta + u\frac{b'}{S}$. Since D' is the worst case delay, we must have $D' \leq \Delta + u\frac{b'}{S}$ (to see why, simply consider a node l where the worst case delay D' is attained). Thus, we must have

$$D' \leq \Delta + \frac{ub_{tot}}{S} + u\alpha(h-1)D' \tag{1}$$

Define function g by $g(D')$ = the right hand-side in Equation (1), so that we can rewrite the equation as: $D' \leq g(D')$. The condition

$$\alpha < \frac{C}{(C-S)(h-1)+S}$$

means that $u\alpha(h-1) < 1$; a simple inspection of g shows that there is a unique fix-point D_1, namely, a value such that $g(D_1) = D_1$. Furthermore, if D' is finite, then $D' \leq g(D')$ implies that $D' \leq D_1$. Thus D_1 is a bound for the delay at any hop. The end-to-end delay is thus bounded by hD_1, which, after some algebra, provides the required formula.

Part 2: We now prove that a finite bound exists. We use the time-stopping method in [10]. For any time $\tau > 0$, consider the virtual system made of the original network, where all sources are stopped at time τ. This network satisfies the assumptions of part 1, since there is only a finite number of bits at the input. Call $D'(\tau)$ the worst case delay across all nodes for the virtual network indexed by τ. From the above derivation we see that $D'(\tau) \leq D_1$ for all τ. Letting τ tend to $+\infty$ shows that the worst case delay at any node remains bounded by D_1.

□

References

[1] Anna Charny. Delay bounds in a network with aggregate scheduling, 1998. Cisco, available from `ftp://ftpeng.cisco.com/ftp/acharny/aggregate_delay.ps`.

[2] R.L. Cruz. Quality of service guarantees in virtual circuit switched networks. *IEEE JSAC*, pages 1048–1056, August 1995.

[3] J.-Y. Le Boudec. Application of network calculus to guaranteed service networks. *IEEE Transactions on Information Theory*, 44:1087–1096, May 1998.

[4] C.S. Chang. On deterministic traffic regulation and service guarantee: A systematic approach by filtering. *IEEE Transactions on Information Theory*, 44:1096–1107, August 1998.

[5] R. Agrawal, R. L. Cruz, C. Okino, and R. Rajan. Performance bounds for flow control protocols. *IEEE/ACM Transactions on Networking (7) 3*, pages 310–323, June 1999.

[6] Parekh A. K. and Gallager R. G. A generalized processor sharing approach to flow control in integrated services networks: The multiple node case. *IEEE/ACM Trans. Networking, vol 2-2*, pages 137–150, April 1994.

[7] M. Andrews, B. Awerbuch, A. Fernandez, J. Kleinberg, T. Leighton, and Z. Liu. Universal stability results for greedy contention resolution protocols. In *37th Annual IEEE Symposium on Foundations of Computer Science (FOCS'96), Burlington VT*, October 1996.

[8] JY Le Boudec and G. Hebuterne. Comment on a deterministic approach to the end-to-end analysis of packet flows in connection oriented network. *IEEE/ACM Transactions on Networking*, February 2000.

[9] I. Chlamtac, A. Faragó, H. Zhang, and A. Fumagalli. A deterministic approach to the end-to-end analysis of packet flows in connection oriented networks. *IEEE/ACM transactions on networking*, (6)4:422–431, 08 1998.

[10] C.S. Chang. *A filtering theory for Communication Networks.* Springer Verlag, 1999.