Information Processing Letters 109 (2008) 151–154

Contents lists available at ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

Stability bounds in networks with dynamic link capacities ☆

V. Cholvi

Departamento de Lenguajes y Sistemas Informáticos, Universitat Jaume I, Campus de Riu Sec, 12071 Castellón, Spain

ARTICLE INFO

Article history:
Received 6 June 2008
Received in revised form 5 September 2008
Available online 24 September 2008
Communicated by C. Scheideler

Keywords:
Packet switched networks
Network stability
Interconnection networks
Adversarial queueing theory
Greedy scheduling protocols
Performance evaluation

ABSTRACT

We address the problem of stability in networks where the link capacities can change dynamically. We show that every network running a *greedy* scheduling policy is universally stable at any injection rate $r < 1/(Cd)$, where d is the largest number of links crossed by any packet and C is the maximum link capacity. We also show that *system-wide time priority* scheduling policies are universally stable at any injection rate $r < 1/(C(d-1))$.

1. Introduction

In order to characterize the performance of a network, one crucial issue is that of *stability*, which has become a major topic of study in the last decade. Roughly speaking, a communication network system is said to be stable if the number of packets waiting to be delivered (backlog) is finitely bounded at any one time. The importance of such an issue is obvious, since if one cannot guarantee stability, then one cannot hope to be able to ensure deterministic guarantees for most of the network performance metrics.

In the last few years, much of the analysis of worst-case behavior of connectionless networks and scheduling policies has been performed using *adversarial* models [1,2]. These models consider the time evolution of a packet-routing network as a game between a malicious adversary that has the power to perform a number of actions (such as injecting packets at particular nodes, choosing their destination, routing them, etc.) and the underlying system. Such an adversary, based on the knowledge of behavior of the system, can devise the scenario that maximizes the "stress" on the system. Consequently, it provides us with a valuable tool with which to analyze the network in a worst-case scenario.

Being consistent with the standard use of the term, we say that a network is *stable* under a given scheduling policy if for any bounded adversary the backlog at any node (i.e., the number of packets "in transit") is bounded (by a value that does not depend on the time). Perhaps the most natural question regarding stability of given protocol is to unveil whether or not it is a stable policy with every network. The answer to this question may depend on the rate at which packets are injected into the network, the capacity of the links, which is the rate at which a link forwards outgoing packets, and the scheduling policy that is used to resolve the conflict when more than one packet wants to cross a given link in a single time step.

In [1,2] it was shown that while some scheduling disciplines like Farthest-to-Go, Longest-in-System, Nearest-to-Source and Shortest-in-System are *universally stable* (we say that a scheduling policy is universally stable if all networks are stable under it), other scheduling policies like FIFO, LIFO, Nearest-to-Go and Farthest-from-Source are not. Therefore, the issue of providing stability conditions for scheduling policies that are not universally stable has received a lot of attention.

A way to approach stability consists in providing thresholds on the rate at which packets are injected into

☆ This work was partially supported by the Spanish MEC under Grant No. TIN2008-03687 and by Bancaixa under Grant No. P1-1B2007-44.

E-mail address: vcholvi@uji.es.

0020-0190/$ – see front matter
doi:10.1016/j.ipl.2008.09.020

the network based on some network parameters. By taking into account information on the largest number of links that can be crossed by any packet in the network (denoted d), Lotker et al. [3] proved that any *work-conserving* scheduling policy (also known as *greedy*: those that always schedule packets if there is anything in the queue) can be stable in any network if the injection rate of the adversary is upper bounded by $1/(d+1)$. They also proved that for *system-wide time priority* scheduling policies (i.e., policies under which, a packet arriving at a queue at time t has priority over any other packet that is injected after time t), the stability threshold becomes $1/d$ (for instance, FIFO is a system-wide time priority scheduling policy). Such bounds were respectively reduced to $1/d$ and $1/(d-1)$ by Echagüe et al. [4], who also showed that the bound obtained for system-wide time priority scheduling policies is tight. However, these results assumed that all network links are identical. By considering that link capacities may vary over time, Koukopoulos et al. [5] proved that any greedy scheduling policy can be stable in any network if the injection rate of the adversary is upper bounded by $1/(C(d+1))$, where C is the maximum link capacity. In the case of system-wide time priority scheduling policies, this bound was set up to $1/(Cd)$.

In this paper, we show that work-conserving scheduling policies are stable when the injection rate is lower than $1/(dC)$. Furthermore, we also show that any system-wide time priority scheduling policy is stable when the injection rate is lower than $1/(C(d-1))$. In this latter case, we show that the injection rate is optimal, in the sense that it is possible to find a network topology and an adversary such that the resulting system becomes unstable for an injection rate of $1/(C(d-1))$.

The rest of the paper is organized as follows. In Section 2 we introduce our adversarial model. In Sections 3 and 4 we provide, respectively, the stability conditions for work-conserving and system-wide time priority packet scheduling policies. Finally, in Section 5 we present our conclusions.

2. The model

We use a generalized version of the adversary model proposed by Andrews et al. [1] (which is commonly used in the literature), which has been adjusted to reflect the fact that link capacities may vary arbitrarily.

Under this model, time is seen as discrete and the time evolution of a packet-switching network is seen as a game between a bounded adversary and a queue policy. At each time step the adversary injects a set of packets into some of the nodes in the network. The adversary is free to choose both the source and the destination node for any injected packet. Furthermore, it also specifies the sequence of links (the route) that each individual packet must traverse. Its only restriction (hence the term “bounded”) is that it cannot fully load any link. Packets are transmitted between adjacent nodes so that, at every step, the scheduling policy decides which packets have to cross each link. A packet will be absorbed after traversing its route. The network is modeled as a set of nodes interconnected by directed point-to-point links. Each node contains a queue for each outgoing link and uses it to store there the packets to be sent along the corresponding link. Edges can have different integer capacities in the interval $[1, C]$ with $C > 1$, which may or may not vary over time. We denote as $C_e(t)$ the capacity (service rate) of edge e at time step t.

The adversary is defined by a pair of parameters (b, r), where $b \geqslant 1$ is a natural number and r is $0 < r < 1$. The parameter b (usually called *burstiness*) models the short bursts of packets we can inject into the network. The parameter r (called the *load rate*) models the long-term rate at which packets can be injected into the network. Let us denote by $A_e(x)$ the total number of packets that the adversary injects during any time interval of length x that traverses the edge e. The adversary must satisfy, for every edge e and every sequence T_x of x consecutive time steps,

$$A_e(x) \leqslant b + r \sum_{t \in T_x} C_e(t).$$

We emphasize that this is a connectionless model; however, all our stability results are also applicable to connection-oriented networks [6], since they are more restrictive.

3. Stability conditions for work-conserving scheduling policies

In this section we obtain an expression of the longest time a packet can wait in the queue of any link with any work-conserving scheduling policy. This expression is then used to derive an upper bound on the injection rate r to guarantee stability.

The following theorem provides a bound on the injection rate that guarantees network stability under any work-conserving scheduling policy.

Theorem 3.1. *Any network in which all queues use a work-conserving packet scheduling policy and packets are injected by a (b, r) adversary is stable if $r < 1/(Cd)$, where d is the largest number of links that can be crossed by any packet in the network and C denotes the maximum link capacity. Furthermore, the worst-case end-to-end delay D is bounded above by $D \leqslant d(b+2)/(1-Crd)$.*

Proof. The proof has two parts. First, we prove that if $r < \frac{1}{Cd}$ then the maximum time interval any packet takes to cross any link is bounded, which implies stability. Second, we prove that, if the first part is true, then D is also bounded above by $d(b+1)/(1-Crd)$.

In what follows, we denote as d_p the number of links that a packet p has to cross (note that $d = \max_p\{d_p\}$). We denote by $C_i^p(t)$ the service rate at time step t of the ith queue for packet p. We also denote by a_i^p and f_i^p the time instants that packet p respectively arrives at and departs from its ith queue, where $1 \leqslant i \leqslant d_p$.

Remark 1. Note that we do not assume, a priori, whether the scenario formed by the network topology, the scheduling policy, and the adversary, is stable or not. Thus, if it is unstable, the time packet p takes to cross its ith queue could be infinite (i.e., $f_i^p = \infty$).

If packet p crosses its ith link in time step f_i^p, it will arrive at its $(i+1)$st queue at time step $a_{i+1}^p = f_i^p$. Finally, we denote by Q_i^p the time packet p takes to cross its ith link, i.e., $Q_i^p = f_i^p - a_i^p$. Let $Q = \max_{p,1\leqslant i\leqslant d_p} Q_i^p$.

Remark 2. Note that if $f_i^p = \infty$ (for some packet p) then $Q = \infty$. However, we base our proof of finding, under which conditions, $Q < \infty$ (which will imply that, consequently, $f_i^p < \infty$).

Part (1): We base our proof on finding the conditions that make $Q < \infty$. Let p be a packet that attains the maximum Q (i.e., $Q_i^p = Q$) at its ith queue. Let t_B be the oldest time step in which the full capacity of the edge associated with the ith queue has been used since time $a_i^p + 1$ (i.e., there are no packet in the queue any more from earlier time steps). Hence, we have that the interval $(t_B, f_i^p]$ is a busy period for the ith queue (i.e., during that interval the ith queue is non-empty).

Define ϕ_i^p as the set formed by all packets served by the ith queue during the interval $(t_B, f_i^p]$ and let p^* be the oldest packet in ϕ_i^p (i.e., $\forall p' \in \phi_i^p$ $(a_1^{p'} \geqslant a_1^{p^*})$). Hence, by definition of p^*, all packets in ϕ_i^p must have been injected during the interval $[a_1^{p^*}, f_i^p]$ (remember that packets are injected instantaneously at their ingress nodes).

Based on the above mentioned scenario and on the definition of the adversarial model, $\sum_{t=a_i^p+1}^{f_i^p-1} C_i^p(t)$ will be bounded by the maximum number of packets injected (by the adversary) during the interval $[a_1^{p^*}, f_i^p - 1]$ minus the packets served (by the ith queue) during the interval (t_B, a_i^p).

$$\begin{aligned}
\sum_{t=a_i^p+1}^{f_i^p-1} C_i^p(t) &\leqslant r \sum_{t=a_1^{p^*}}^{f_i^p-1} C_i^p(t) + b - \sum_{t=t_B}^{a_i^p} C_i^p(t) \\
&= r \sum_{t=a_1^{p^*}}^{t_B-1} C_i^p(t) + r \sum_{t=t_B}^{a_i^p} C_i^p(t) \\
&\quad + r \sum_{t=a_i^p+1}^{f_i^p-1} C_i^p(t) + b - \sum_{t=t_B}^{a_i^p} C_i^p(t) \\
&\qquad \text{since } a_1^{p^*} < t_B \\
&\leqslant r \sum_{t=a_1^{p^*}}^{t_B-1} C_i^p(t) + r \sum_{t=a_i^p+1}^{f_i^p-1} C_i^p(t) + b \\
&\qquad \text{since } r \leqslant 1.
\end{aligned}$$

Taking into account that $(f_i^p - 1) - (a_i^p + 1) \leqslant \sum_{t=a_i^p+1}^{f_i^p-1} C_i^p(t)$, we have that

$$\begin{aligned}
&(f_i^p - 1) - (a_i^p + 1) \\
&\quad \leqslant r(t_B - a_1^{p^*})C + r(f_i^p - a_i^p)C + b \\
&\quad = r(t_B - a_j^{p^*} + a_j^{p^*} - a_1^{p^*})C + r(f_i^p - a_i^p)C + b \\
&\quad = r(t_B - a_j^{p^*}) + r(a_j^{p^*} - a_1^{p^*})C + r(f_i^p - a_i^p)C + b \\
&\quad \leqslant r(a_j^{p^*} - a_1^{p^*})C + r(f_i^p - a_i^p)C + b \\
&\qquad \text{since } a_j^{p^*} \geqslant t_B.
\end{aligned}$$

Since $f_i^p - a_i^p = Q_i^p = Q$ and $a_j^{p^*} - a_1^{p^*} \leqslant (j-1)Q$, we have that

$$Q - 2 \leqslant rQ(d-1)C + rQC + b,$$

$$Q \leqslant rQCd + b + 2.$$

Then, if $r < 1/(dC)$ we obtain that $Q < \infty$.

Combining the two cases, we have that if $r < 1/(dC)$ then $Q < \infty$.

Part (2): The proof uses the time-stopping method. Consider a packet p that traverses a path with d_p hops, where $d_p \leqslant d$. For any time $t > 0$, consider the virtual system made of the original network, where all sources are stopped at time t. This network satisfies the assumption of part (1), since there are only a finite number of packets at the input. Call $D'(t)$ the worst-case end-to-end delay of packet p for the virtual network indexed by t. From the above derivation we see that $D'(t) \leqslant d_p Q \leqslant d(b+2)/(1-rdC)$ for all t. Letting t tend to $+\infty$ shows that the worst-case delay remains bounded above by $d(b+2)/(1-rdC)$. □

4. Stability conditions for system-wide time priority scheduling policies

The next theorem provides a bound on the injection rate that guarantees network stability under any system-wide time priority scheduling policy.

Theorem 4.1. *Any network in which all queues use a system-wide time priority packet scheduling policy and packets are injected by a (b,r) adversary is stable if $r < 1/(C(d-1))$, where d is the largest number of links that can be crossed by any packet in the network and C denotes the maximum link capacity. Furthermore, the worst-case end-to-end delay D is bounded above by $D \leqslant d(b+2)/(1-rC(d-1))$.*

Proof. The proof here is similar to the proof of Theorem 3.1.

Part (1): The main difference is that now, $\sum_{t=a_i^p+1}^{f_i^p-1} C_i^p(t)$ will be bounded by the maximum number of packets injected during $[a_1^{p^*}, a_i^p - 1]$ (instead of $[a_1^{p^*}, f_i^p - 1]$, since the policy is system-wide time priority) minus the packets served during the busy period interval (t_B, a_i^p).

$$\begin{aligned}
\sum_{t=a_i^p+1}^{f_i^p-1} C_i^p(t) &\leqslant r \sum_{t=a_1^{p^*}}^{a_i^p-1} C_i^p(t) + b - \sum_{t=t_B}^{a_i^p} C_i^p(t) \\
&= r \sum_{t=a_1^{p^*}}^{t_B-1} C_i^p(t) + r \sum_{t=t_B}^{a_i^p-1} C_i^p(t) + b - \sum_{t=t_B}^{a_i^p} C_i^p(t) \\
&\qquad \text{since } a_1^{p^*} < t_B \\
&\leqslant r \sum_{t=a_1^{p^*}}^{t_B-1} C_i^p(t) + b \quad \text{since } r \leqslant 1.
\end{aligned}$$

Taking into account that $(f_i^p - 1) - (a_i^p + 1) \leqslant \sum_{t=a_i^p+1}^{f_i^p-1} C_i^p(t)$, we have that

$$\begin{aligned}
&\bigl(f_i^p - 1\bigr) - \bigl(a_i^p + 1\bigr)\\
&\quad \leqslant r\bigl(t_B - a_1^{p^*}\bigr)C + b\\
&\quad = r\bigl(t_B - a_j^{p^*} + a_j^{p^*} - a_1^{p^*}\bigr)C + b\\
&\quad = r\bigl(t_B - a_j^{p^*}\bigr) + r\bigl(a_j^{p^*} - a_1^{p^*}\bigr)C + b\\
&\quad \leqslant r\bigl(a_j^{p^*} - a_1^{p^*}\bigr)C + b \quad \text{since } a_j^{p^*} \geqslant t_B.
\end{aligned}$$

Since $f_i^p - a_i^p = Q_i^p = Q$ and $a_j^{p^*} - a_1^{p^*} \leqslant (j-1)Q$, we have that

$$Q - 2 \leqslant rQ(d-1)C + b,$$

$$Q \leqslant rQC(d-1) + b + 2.$$

On performing some algebra we found that if $r < 1/(C(d-1))$, we obtain that $Q < \infty$.

Part (2): The same reasoning as in part (2) of Theorem 3.1 is used here. Consider a packet p that traverses a path with d_p hops, where $d_p \leqslant d$. For any time $t > 0$, consider the virtual system made of the original network, where all sources are stopped at time t. This network satisfies the assumption of part (1), since there are only a finite number of packets at the input. Call $D'(t)$ the worst-case end-to-end delay of packet p for the virtual network indexed by t. From the above derivation we see that

$$D'(t) \leqslant d_p Q \leqslant d\frac{b+2}{1 - r(d-1)C} \quad \text{for all } t.$$

Letting t tend to $+\infty$ shows that the worst-case delay remains bounded above by $d(b+2)/(1 - r(d-1)C)$. □

At this point, we note that the above mentioned injection rate is optimal in the sense that it is possible to find a network topology and an adversary such that the resulting system becomes unstable for $r = 1/(C(d-1))$, with $C \geqslant 1$. Indeed, as it has been pointed out in [7], there exists a network topology with arbitrary fixed link capacities $c \geqslant 1$ and an adversary such that the resulting system becomes unstable for $r = 1/(c(d-1))$. Therefore, we only have to consider the same network with $c = C$ and the same adversary to create instability for $r = 1/(C(d-1))$.

5. Conclusions

In this work, we have given a bound of $r < 1/(Cd)$ on the injection rate below which all work-conserving scheduling policies are stable in any network. We have also obtained a slightly better bound of $r < 1/(C(d-1))$ for system-wide time priority scheduling policies. As it has been shown here, at least in the case of system-wide time priority packet schedulers, we need to consider some additional parameters to improve the above mentioned bound (if this is possible). Therefore, an interesting issue is to approach stability by considering some network parameters other than d and C.

Acknowledgement

The author wishes to thank Christian Scheideler for several constructive suggestions and corrections.

References

[1] M. Andrews, B. Awerbuch, A. Fernández, T. Leighton, Z. Liu, J. Kleinberg, Universal-stability results and performance bounds for greedy contention-resolution protocols, Journal of the ACM 48 (1) (2001) 39–69.

[2] A. Borodin, J. Kleinberg, P. Raghavan, M. Sudan, D. Williamson, Adversarial queueing theory, Journal of the ACM 48 (1) (2001) 13–38.

[3] Z. Lotker, B. Patt-Shamir, A. Rosén, New stability results for adversarial queuing, SIAM Journal on Computing 33 (2) (2004) 286–303.

[4] J. Echagüe, V. Cholvi, A. Fernández, Universal stability results for low rate adversaries in packet switched networks, IEEE Communication Letters 7 (12) (2003) 578–580.

[5] D. Koukopoulos, M. Mavronicolas, P. Spirakis, Performance and stability bounds for dynamic networks, Journal of Parallel and Distributed Computing 67 (4) (2007) 386–399.

[6] R. Cruz, A calculus for network delay. Parts I and II, IEEE Transaction on Information Theory 37 (1) (1991) 114–131 and 131–141.

[7] A. Charny, J.-Y. Le Boudec, Delay bounds in a network with aggregate scheduling, in: First International Workshop on Quality of Future Internet Services, Berlin, Germany, 2000.