A note on the use of topology extensions for provoking instability in communication networks

Blesa, M.J.

Research Report LSI-04-16-R

Departament de Llenguatges i Sistemes Informàtics

UNIVERSITAT POLITÈCNICA DE CATALUNYA

A note on the use of topology extensions for provoking instability in communication networks*

Maria J. Blesa

Universitat Politècnica de Catalunya
Dept. Llenguatges i Sistemes Informàtics
Campus Nord, E-08034 Barcelona, Spain
mjblesa@lsi.upc.es

Abstract. We study some aspects of the instability of the *last-in-first-out* (LIFO) scheduling protocol in under-loaded packet-switched networks under the Adversarial Queueing Theory model, which allows to consider worse-case scenarios in the study of network traffic. Using a typical strategy in the literature, we improve a known lower bound for the instability of LIFO. The strategy for obtaining such an improvement consist in extending the topology of the original network for which the previous lower bound was shown. Moreover, we show that every additional application of this technique leads to a further (although every time smaller) improvement of the lower bound for instability. However, the number of improvements that this strategy can achieve is limited. Not even when extending the topology infinitely is possible to obtain a lower bound for the instability of LIFO lower than 0.61804.

1 Introduction

Communication protocols have existed for thousands of years in human environments. Since computers have started communicating with each other over electronic networks, the number of different protocols has exploded. In a communication network, the queueing *protocol* (or scheduling policy) determines the order in which the packets requiring to cross a link are scheduled to be forwarded. Most protocols aim to move information across a network in an efficient and reliable manner. This often requires congestion and flow control, error detection and correction, and handshaking to coordinate the information transfer.

Appropriate models to study networking systems implementing specific communication protocols are needed. Those models could help us on understanding better the dynamics of nowadays' communication networks, and therefore on detecting and overcoming the conditions leading to undesirable negative effects, as well as helping on their further prevention. One of those undesirable negative effects is the *instability* of the networking system, i.e., the lack of stability. Stability, in the context of communication networks, refers to the fact that the amount of packets in the network, i.e., the *network traffic*, remains always bounded. That bound is a constant that might depend on system parameters, e.g. the diameter of the network, but not on time. Indirectly, stability is also referring to the limited size of the buffers that hosts use to store in-route packets and, also in many cases, implies the on-time delivery of packets to their destination.

Stability is studied in relation to the three main components forming a synchronous communication system $(\mathcal{N}, \mathcal{A}, \mathcal{P})$: the network $\mathcal{N}$, the traffic pattern defined by $\mathcal{A}$, and the protocol $\mathcal{P}$. *Networks* are usually modelled by directed graphs in which nodes represent the hosts and edges represent the links between those hosts. The *traffic pattern* controls where and how packets join the system and defines their trajectory. The problem of deciding stability and detecting sufficient conditions for it to hold (or not to hold) have been investigated under various models of packet routing. Some models make probabilistic assumptions on the traffic pattern, while others perform

* Work partially supported by the FET Programme of the EU under contract number IST-2001-33116 (FLAGS), and by the Spanish CICYT projects TIC-2001-4917-E and TIC2002-04498-C05-03 (TRACER). M.J. Blesa is also supported by the Catalan Research Council of the Generalitat de Catalunya under grant 2001FI-00659.

worse-case analysis by replacing more traditional stochastic arrival assumptions by worst-case inputs in the traffic pattern. These latter are closer to the traditional analysis of algorithms and to real network configurations.

The Adversarial Queueing Theory Model (AQT) proposed by Borodin et al. [6], which is a robust model of queueing theory in network traffic, can be considered as the pioneering work in studying stability via worst-case analysis. Recent research on stability has mainly considered the AQT model since it can describe the behaviour of both connection-less and short-term connection networks, as well as connection-oriented networks. This includes the Internet and ATM networks. A considerable amount of results is available nowadays (see, e.g., [6,4,7,17,18,9,2,13,1,5,3,14,15,8,16,19]).

The AQT model considers the time evolution of a packet-routing network as a game between an adversary and a queueing protocol. The system is considered to be *synchronous*. At each time step the adversary may inject a set of packets to some of the nodes. For each packet, the adversary specifies the route that it must traverse (static routing), after which the packet will be absorbed. If more than one packet wishes to cross an edge e at the same time step, then the queueing policy chooses one of these packets to send across e. The remaining packets wait in the queue. This game then advances to the next time step. The goal of the adversary is to try to prevent the protocol from guaranteeing load and delay bounds. On the contrary, the main goal of the model is to study conditions for stability of the network under different protocols.

In order not to trivially overload the system and in order to be able to guarantee delay bounds, it is necessary to restrict the traffic arriving to the network. The constraints on the traffic pattern must ensure that, over long periods of time, the maximum traffic injected in a link is roughly the amount of traffic that the link can forward. Two parameters (r, b) constraint an adversary in the AQT model, where $b \geq 0$ is *the burstiness* (i.e., the maximum excess of packets that can be injected in one step requiring any particular link) and $0 < r < 1$ is the *injection rate* (i.e., the frequency at which the adversary introduces packets into the network). Let $N_e(I)$ be the number of packets injected by the adversary in a time interval I that have paths requiring a particular edge e. The adversary must obey the following (leaky-bucket) constraint,

$$N_e(I) \leq \lceil r|I| \rceil + b. \qquad (1)$$

In this paper, we study some aspects on the instability of the *last-in-first-out* protocol in packet-switched communication networks under the Adversarial Queueing Theory model.

The last-in-first-out protocol (LIFO) schedules queued packets according to a *local criterion* in which the highest priority is given to the packet that has arrived the latest in the queue. This locality property makes the LIFO protocol easy to be implemented. Under the AQT model, the fact that some systems considering LIFO might not be stable was already shown in the earliest literature [4]. Furthermore, LIFO can be make unstable at arbitrarily low rates [6]. The stability of LIFO is upper-bounded by the recent result in [8], which shows that every greedy protocol is stable when the injection rate is smaller than $1/d$, where d denotes the diameter of the network.[1]

The LIFO protocol is gaining popularity in the last years due to the recent discovery of the significant quality improvement on the performance of interactive real-time services, such as IP telephony and IP teleconferencing (and, in general, any voice and video transmission), when the network is congested [21,11,12]. LIFO has been shown to be an excellent overload strategy for time-constrained services since it provides fast packet forwarding and low latency. The advantage of being simple, has made it an attractive candidate to be used in new equipments using IP and supporting modern networking applications.[2] New TCP-friendly protocols considering the LIFO protocol have been recently proposed with the aim of improving those features of TCP that are not suitable for continuous media applications, like re-transmission, packetization and scheduling [20].

[1] The diameter of a graph is the maximum length of shortest paths between two vertices in the graph.

[2] This scheme is patented by the Swiss Federal Institute of Technology (EPFL).

Contributions and organisation. In spite of these new interests on LIFO for modern networking applications, there are still a few results to be found in the literature concerning stability issues in networks which schedule packets according to this protocol. This lack is specially emphasised when comparing to the amount of results available in the literature for other protocols, e.g., for the FIFO protocol. In this work, we aim to contribute on reducing this lack concerning the LIFO protocol. Hopefully, these contributions will also give some ideas on the relative power of extending the topology of a network. Topology extension has been exhaustively used in the literature as a strategy for decreasing existing lower bounds on instability of other protocols (see, e.g., [7,17,14] for FIFO). We show that under some protocols (like it is the case for LIFO), this strategy improves the lower bound down to a minimum, but not further.

The paper is organised as follows. In Section 2 we describe the structure of the networks used in this work and how their topology can be extended. In Section 3, we improve a known lower bound for the instability of LIFO via topology extensions and show that every additional application of this technique leads to a further (although every time smaller) improvement of the lower bound. However, the number of improvements is limited, and it is not possible to obtain a lower bound for the instability of LIFO lower than 0.61804. Finally, Section 4 contains some of the conclusions we can extract from this work and further open questions related to it.

2 Preliminaries: Topology extensions of inverted symmetric networks

Stability is studied in relation to the three main components forming a synchronous communication system $(\mathcal{N}, \mathcal{A}, \mathcal{P})$: the network $\mathcal{N}$, the traffic pattern defined by $\mathcal{A}$, and the protocol $\mathcal{P}$. Concerning the protocol point of view, the first issue to solve consists on determining, for a given protocol $\mathcal{P}$, if any system which schedules packets according to $\mathcal{P}$ is stable, i.e., if any system $(\mathcal{N}, \mathcal{A}, \mathcal{P})$ is stable for all $\mathcal{N}$ and for all $\mathcal{A}$. To prove that this is not always the case, and thus that $\mathcal{P}$ can produce instability in some systems, one has to find the appropiate $\mathcal{N}$ and $\mathcal{A}$ that make the traffic load to grow unboundedly. For this aim, the tendence in the literature is to look for networks $\mathcal{N}$ with some *inverted symmetry* in its topology (see, e.g., [6,4,7,17]). A topology with inverted symmetry makes it easier both, to design an appropiate adversary $\mathcal{A}$ which behaves regularly, and also to proof via an inductive reasoning that the traffic it produces brings the system to instability.

Once it is known that a given protocol $\mathcal{P}$ can bring a certain system $(\mathcal{N}, \mathcal{A}, \mathcal{P})$ to instability, the main interest from the protocol point of view lies on placing exactly (and make as close as possible) the frontier for its stability and its instability. For this aim, lower bounds for the instability of $\mathcal{P}$ are needed. These lower bounds apply to the parameters controlling the possible traffic patterns, i.e., the injection rate r and the burstiness b in the AQT model. Since large bursts do not cause instability [10], the interest lies on determining (and further improving) the minimum injection rate r that implies the instability of systems scheduling the traffic according to $\mathcal{P}$. For this aim, given an unstable system $(\mathcal{N}, \mathcal{A}, \mathcal{P})$, the strategy used in the literature consisted on *extending the topology* of the network $\mathcal{N}$ and adapting appropiately the traffic pattern defined by $\mathcal{A}$, such that the new system is also unstable but for a lower injection rate (see [7,17,14] for FIFO).

In this work, as it is usual in the literature, we will deal with networks whose topology have some inverted symmetry and we will also extend their topology in order to obtain new networks, which keep the inverted symmetry, but whose new topology is more interesting for our goals. The aim is to formalise and quantify the topology extension strategy and its influence on the lower bounds for instability that can be obtained.

2.1 Inverted symmetry

Figure 1(a) represents the general simplest form of a network with inverted symmetry in its topology. A network with this property has two distiguished connected (half) parts. Every node v in one of the parts has one (and only one) corresponding node v' in the other part. Every edge $e = (u, v)$ in one of the parts has its corresponding edge $e' = (u', v')$ in the other part, in which u' is the node corresponding to u, and v' is the node corresponding to v, respectively. Let us distinguish an 'input' vertex v_{in} and an 'output' vertex v_{out} in one of the parts. The vertex v_{in} is such that no edge in the part of the network to which it belongs to incides on it, but some edge has source in it. Analogously, no edge has source in v_{out} but it is the target of some edge in the part it belongs to. Let v'_{in} and v'_{out} denote the corresponding vertices in the symmetric part. Then, in a network with inverted symmetry, there exist only two *bridge-edges*, $e = (v_{out}, v'_{in})$ and $e' = (v'_{out}, v_{in})$ connecting the two symmetric parts of the network. In instability proofs, the bridge-edges play a special role in the inductive reasoning, since their queues are the ones that unboundedly grow over time. The number of packets requiring edge e are counted, for example, at odd phases. Then, the packets queued at the end of that phase will contribute to the next phase, in which we will focus on counting the packets requiring edge e'. This behaviour alternates between odd and even phases.

2.2 Budgets

Let us call a *budget*, the structure composed by four vertices, v_i, w_i, v'_i and w'_i, and the fast edges $f_i = (w_i, v_1)$ and $f'_i = (w'_i, v'_i)$ (see Figure 1(b)). Moreover, a budget also includes two non-fast edges $b_i = (v_i, w)$ and $b'_i = (v'_i, w')$, which have source in the vertices v_i and v'_i of the budget, respectively. The target vertices w and w' of these edges will usually be the vertices v_{i+1} and v'_{i+1} of the next adjacent budget. Given a fast edge and a non-fast edge leading to the same node, if one packet crosses each of them in the same time step then, the packet crossing the fast edge will arrive first than the packet crossing the normal edge. Typically, when designing an adversary to produce instability in networks with such a structure, injections are done in the fast edges at alternating phases, i.e., in f_i (at the i-th round of odd phases) and in f'_i (at the i-th round of even phases). Since the protocol is LIFO, they will be blocked respectively in the queues of b_i and b'_i when at the same time packets arrive to v_i and v'_i via normal edges. In the forthcoming, we will denote the i-th budget in a network topology as $B_i = \{\{v_i, w_i, v'_i, w'_i\}, \{f_i = (w_i, v_i), f'_i = (w'_i, v'_i), b_i = (v_i, *), b'_i = (v'_i, *)\}\}$.

2.3 Bottlenecks

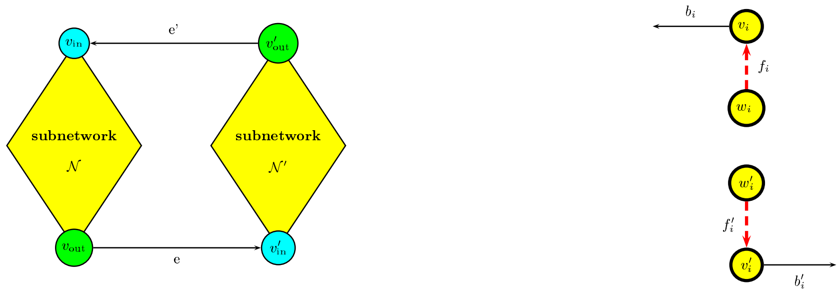
Let us call a *bottleneck*, the structure composed by three vertices, u, v_1 and w, and the edges $g_0 = (u, v_1)$, $g'_0 = (u, v_1)$ and $g = (v_1, w)$ between them (see Figure 3, center). Since we consider edges with unitary forwarding capacity (per time step), a bottleneck permits to accumulate traffic in the unique edge outgoing from v_1 as soon as packets arrive simultaneously through the edges inciding in it. Bottlenecks have been used in previous works to improve existing lower bounds for the instability of some other protocols, e.g. in [7,17,14] for FIFO.

2.4 Network topology extensions

Two types of extensions are performed depending on how the topology (and the symmetry) is extended, horizontally or vertically.[3] By introducing budgets the topology is extended horizontally and, by introducing bottlenecks, the topology is extended vertically.

The first budget $B_0 = \{\{v_0, w_0, v'_0, w'_0\}, \{f_0 = (w_0, v_0), f'_0 = (w'_0, v'_0), b_0 = (v_0, *), b'_0 = (v'_0, *)\}\}$ is introduced into a network with inverted symmetry by making the b_0 and b'_0 edges incide, respectively, in the 'input' vertices v_{in} and v'_{in} of the two symmetric parts of the network (i.e., $b_0 = (v_0, v_{\text{in}})$ and $b'_0 = (v_0, v'_{\text{in}})$), and by making the bridge edges incide now not in the other symmetric part of the network, but in the budget (i.e., $e = (v_{\text{out}}, v'_0)$ and $e' = (v_{\text{out}}, v_0)$). Instead, a new budget can be added to the i-th budget of a network by making the edges b_i of the i-th budget to incide into the node v_{i+1} of the $(i+1)$-th budget and, symmetrically, the edge b'_i into node v'_{i+1} (see Figures 2(a) and 2(b)).

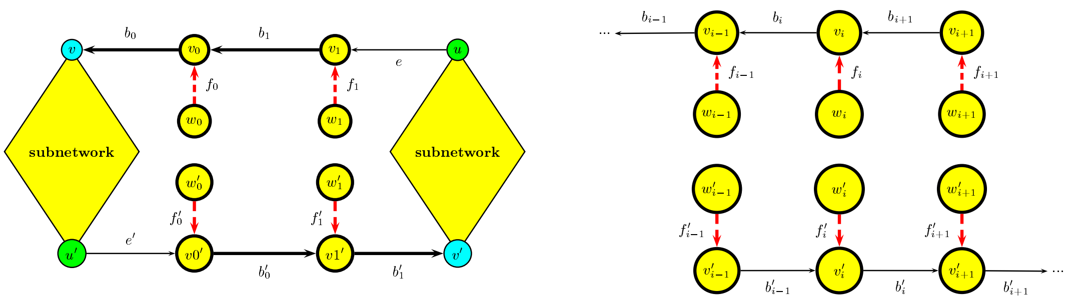
[3] The names *vertical* and *horizontal* are chosen just to improve the legibility of the paper and to make the graphical representations more intuitive.



(a) Every node w in $\mathcal{N}$ has one (and only one) corresponding node w' in $\mathcal{N}'$. Every edge $e = (w_1, w_2)$ in $\mathcal{N}$ has its corresponding edge $e' = (w'_1, w'_2)$ in $\mathcal{N}'$, in which w'_1 is the node corresponding to w_1, and w'_2 is the node corresponding to w_2, respectively. The bridge-edges $e = (v_{\text{out}}, v'_{\text{in}})$ and $e' = (v'_{\text{out}}, v_{\text{in}})$ connect the two parts.

(b) i-th budget that we use to extend network $\mathcal{N}_f$. Edges f_i and f'_i are fast; b_i and b'_i are normal.

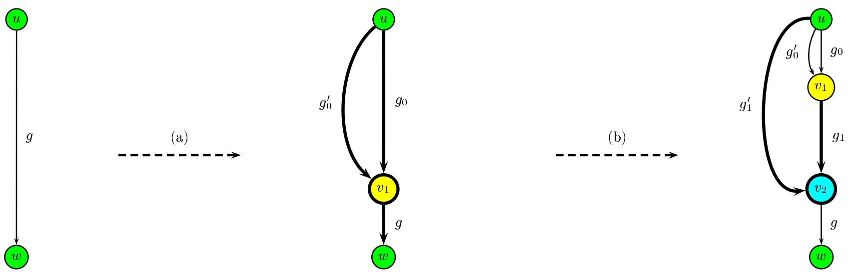
Fig. 1. Schema of an inverted symmetric network (left) and structure of a budget (right)



(a) Two budgets embedded into an inverted symmetric network.

(b) Composition of budgets.

Fig. 2. Manipulation of budgets



(a) Introduction of a bottleneck into an edge. Edge g is extended by introducing a new node v_1 and the edges $g_0 = (u, v_1)$ and $g = (v_1, w)$. A new edge g'_0 is introduced between the initial source node u and the new node v_1.

(b) Introduction of the second bottleneck. The path (u, v_1, w) is extended to (u, v_1, v_2, w) by introducing a new node v_2 between v_1 and w. A new edge g'_1 is added between the initial source node u and the new node v_2.

Fig. 3. Manipulation of bottlenecks

A first bottleneck is introduced into an edge $g = (u, w)$ by adding a new node v_1 between the endpoints, and the corresponding edges $g_0 = (u, v_1)$ and $g = (v_1, w)$. Observe that edge g has been transformed into the path (u, v_1, w). Another edge, $g'_0 = (u, v_1)$ is also introduced between the source node u and the new node v_1. Figure 3(a) represents this process of inserting the first bottleneck. Observe that edge $g = (v_1, w)$ will accumulate a packet when packets arrive simultaneously from g_0 and g'_0. Then, a second bottleneck can be added to this new structure by introducing a new node v_2 between v_1 and w, and the corresponding edges to transform the former path (u, v_1, w) into (u, v_1, v_2, w). A new edge g'_1 is again added between the source node u and the new node v_2 (see Figure 3(b)). Further bottlenecks can be added to this new structure by applying the same idea.

We would like to refer quantitatively to the extensions that can be applied to the topology of a network. Let $\mathcal{N}$ be a network with inverted symmetry; then a network $\mathcal{N}'$ is the (n, m)-th extension of $\mathcal{N}$, where $n, m \geq 0$, if $\mathcal{N}'$ is obtained by extending n times the topology of $\mathcal{N}$ horizontally (i.e., by introducing n budgets) and m times vertically (i.e., by introducing m bottlenecks). We will use this notation to express and quantify the accumulating power of a traffic pattern (defined by an adversarial strategy) over a network, which is the (n, m)-th extension of another given network $\mathcal{N}$, in terms of the quantity of extensions applied.

3 Topology extension to improve bounds for the instability of LIFO

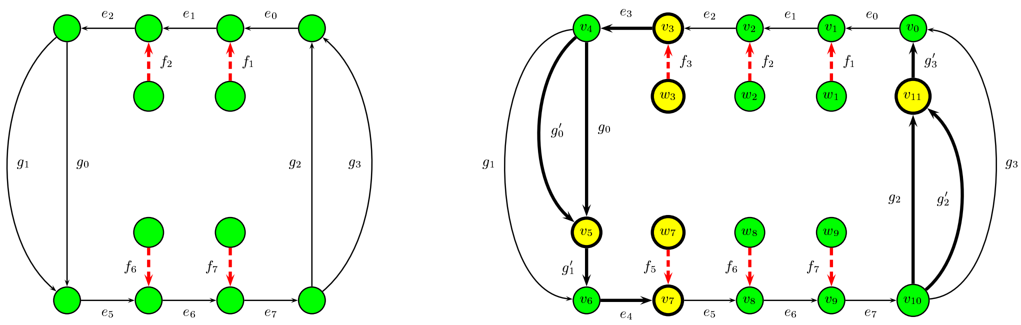
The most common way used in the literature to prove that a protocol $\mathcal{P}$ is not stable consists in finding a network $\mathcal{N}$, which schedules packets according to $\mathcal{P}$, and to find an adversary $\mathcal{A}$ able to bring the system $(\mathcal{N}, \mathcal{A}, \mathcal{P})$ to bad scenarios in which the quantity of packets in the system grows unboundedly. Usually, the proofs focus in some (from a topological point of view) interesting edge(s) of the network and show that the size of its queue, i.e., the number of packets requiring that edge, increases infinitely. In these proofs, the time is usually organized in phases, each of which consists of rounds. A round can be one or more discrete steps of time. Then instability is usually shown via an inductive reasoning on the number of phases that the adversary plays.

Given a protocol $\mathcal{P}$, the strategy used in the literature for consecutively improving the best known lower bound for the instability of a protocol $\mathcal{P}$ consisted in extending the topology of the original network $\mathcal{N}'$ of the communication system $(\mathcal{N}', \mathcal{A}, \mathcal{P})$ in which that lower bound produced instability and adapting property the traffic pattern $\mathcal{A}$ (see, e.g., [7,17,14] for FIFO). For some time, the best lower bound for the instability of LIFO was known to be $1/\sqrt{2}$ and it was obtained in the network $\mathcal{N}'$ depicted in Figure 4(a).

Based on this idea of extending the topology, we propose a new network that improves the lower bound for the instability of LIFO from $1/\sqrt{2}$ to 0.658. The new network $\mathcal{N}$ we propose is the $(1, 1)$-th extension of the network $\mathcal{N}'$ used in [4] (see Figure 4(b)). The extension of the topology is done as explained in Section 2.4. Although this lower bound improvement is good news, it has no further relevance anymore from the protocol point of view, since nowadays we know that LIFO can be made unstable at arbitrarily low rates [6]. The focus of this section is therefore rather on the technique than on the improvement of the lower bound. Our aim is to exemplify how a known lower bound for the instability of LIFO can be decreased by topology extension. This will be the starting point to show that this technique only achieves a limited number of improvements and that, once this limit is reached, more clever adversaries or new alternative techniques should be found to go on improving lower bounds for instability.

Theorem 1. *Let $r > 0.658$. There is a network $\mathcal{N}$ with a non-empty initial configuration and an adversary $\mathcal{A}$ of rate r such that the system $(\mathcal{N}, \mathcal{A}, \text{LIFO})$ is not stable.*

Proof. We consider the network $\mathcal{N}$ in Figure 4(b). Edges denoted by f_1, f_2, f_3, f_5, f_6 and f_7 are fast; the rest are normal. Given a fast edge and a normal edge leading to the same node, if one packet crosses each of them in the same time step then, the packet crossing the fast edge will arrive first than the packet crossing the normal edge. Any injection made on a normal edge is supposed to arrive later than any packet already queued in the edge at the beginning of the time step.



(a) Network $\mathcal{N}'$ used in [4] to show instability for LIFO at any rate $r > 1/\sqrt{2}$. Edges f_1, f_2, f_6 and f_7 are fast edges, the rest are normal.

(b) Network $\mathcal{N}$ that we use to show instability for LIFO at any rate $r > 0.658$. Edges f_1, f_2, f_3, f_5, f_6 and f_7 are fast, the rest are normal. $\mathcal{N}$ is the $(1,1)$-th extension of $\mathcal{N}'$ in (a).

Fig. 4. (Inverted symmetric) networks which are not stable for LIFO.

The inductive hypothesis [4] is the following: *At the beginning of phase j, there are s packets in the queues at e_7 and g'_3 that want to traverse edges e_0, e_1, e_2, e_3, g_0, g'_1, while at the beginning of phase $j+1$, packets are queued at edges e_3 and g'_1, and want to traverse edges e_4, e_5, e_6, e_7, g_2, g'_3.* The inductive hypothesis holds alternatively for each symmetric side of the topology. From the inductive hypothesis, initially, there are s packets (called S-flow) in the queues of edges e_7 and g'_3 requiring to traverse edges e_0, e_1, e_2, e_3, g_0, g'_1. The adversary $\mathcal{A}$ plays injections in three rounds.

Round 1: for s steps, the adversary injects a set G of rs packets that want to traverse edges f_1, e_1, e_2, e_3, g_0, g'_1, e_4, e_5, e_6, e_7, g_2, g'_3. Since f_1 is a fast edge, any packet coming out from v_0 arrives later to v_1 than any packet coming out at the same time from w_1. After the s steps, the whole set G of injections will be blocked at the queue of e_1 because the queueing protocol is LIFO. Packets of the S-flow will follow their way.

Round 2: for the next rs steps, the adversary injects a set R of r^2s packets that want to traverse edges f_2, e_2, e_3, g'_0, g'_1, e_4, e_5, e_6, e_7, g_2, g'_3. Since f_2 is a fast edge, the set R will be blocked at the queue of edge e_2 by the packets of G queued at e_1. Packets of G will be also blocked by a set of r^2s single-edge injections on g_0. Then, at the end of this sub-phase, there will be r^2s packets of G at the queue of g_0.

Round 3: for the next r^2s steps, the adversary injects a set V of r^3s packets that want to traverse edges f_3, e_3, g_1, e_4, e_5, e_6, e_7, g_2, g'_3. Since f_3 is a fast edge, the packets of the set V will be all blocked at the queue of e_3 by the packets from R queued at e_2. Packets from the set G (at the edge g_0) and from the set R (at edge g'_0) will meet in edge g'_1, where the adversary performs r^3s single injections. At the end of the round, $r^2s + r^3s$ packets will remain queued in g'_1.

This is the end of phase j, and there is a total amount of $r^2s + 2r^3s$ packets queued at e_3 and g'_1 that want to traverse edges e_4, e_5, e_6, e_7, g_2 and g'_3, so the inductive hypothesis for phase $j+1$ holds. The adversary $\mathcal{A}$ described above makes the network $\mathcal{N}$ not to be stable for LIFO when $r^2s + 2r^3s > s$, i.e. at injection rate $r > 0.658$. □

[4] Instability proofs are usually based on induction. The goal is to demonstrate that the number of packets in the system increases from phase to phase (and, by applying the inductive hypothesis, they can increase infinitely). The configuration of the system at the end of each phase must be the analogous to the configuration at the beginning (in terms of the type of packets and their location), but with an increased number of packets. For the sake of simplicity, in our proofs we only reproduce the inductive phase and sometimes we omit some additive constants in our analysis. Those omissions, however, do not change the final result.

3.1 More improvements are possible, but not many

Let us denote as $Q(\mathcal{N}, n, m)$ the number of packets that the (n, m)-th extension of a network $\mathcal{N}$ can accumulate at each inductive phase. For obtaining the lower bound of $1/\sqrt{2}$ for the instability of LIFO in [4], the traffic pattern had to assure a cumulative excess of $Q(\mathcal{N}', 0, 0) = 2r^2 s$ extra packets at each phase of the induction (over network $\mathcal{N}'$ in Figure 4(a)). Observe that the network $\mathcal{N}$ that we have used in the previous section is the $(1, 1)$-th extension of network $\mathcal{N}'$ and that, using an analogous traffic pattern, the system described in the proof of Theorem 1 accumulates $Q(\mathcal{N}', 1, 1) = r^2 s + 2r^3 s$ extra packets per phase. This increase on the cumulative power represents a decrease on the lower bound for the instability of LIFO from $1/\sqrt{2}$ to 0.658.

It is easy to verify that, when considering a network $\mathcal{N}^*$ which is the (n, n)-extension of $\mathcal{N}'$, for any $n > 1$, an analogous adversarial strategy applied over $\mathcal{N}^*$ would be able to accumulate a traffic of $Q(\mathcal{N}', n, n)$ extra packets in each phase of the induction, where

$$Q(\mathcal{N}', n, n) = r^2 s + r^3 s + \ldots + 2r^{(n+2)} s \;=\; r^{(n+2)} s + \sum_{i=0}^{n} r^{(i+2)} s, \tag{2}$$

Then, the following relation can be then established between the (n, n)-th extension and the $(n-1, n-1)$-th extension,

$$\begin{cases} Q(\mathcal{N}', 0, 0) = 2r^2 s \\ Q(\mathcal{N}', 1, 1) = r^2 s + 2r^3 s \\ Q(\mathcal{N}', n, n) = Q(\mathcal{N}', n-1, n-1) + \left(r^{(n+1)} s \cdot (2r-1)\right) \quad \text{, for any } n > 1. \end{cases}$$

Observe that the factor $(2r-1)$ in the general term determines the behaviour of the recurrence. Depending on which is the value of the injection ratio r, we can say whether further extensions of the topology would allow to accumulate more, the same, or less.[5] The injection rate $r = 0.5$ is a critical value that defines the following behaviour for $n > 1$,

$$\begin{array}{l} \cdot\, r > 0.5 \Rightarrow 2r - 1 > 1 \Rightarrow Q(\mathcal{N}', n, n) \geq Q(\mathcal{N}', n-1, n-1) \\ \cdot\, r = 0.5 \Rightarrow 2r - 1 = 0 \Rightarrow Q(\mathcal{N}', n, n) = Q(\mathcal{N}', n-1, n-1) \\ \cdot\, r < 0.5 \Rightarrow 2r - 1 < 1 \Rightarrow Q(\mathcal{N}', n, n) < Q(\mathcal{N}', n-1, n-1)\,. \end{array}$$

Thus, additional extensions do not always allow to accumulate more and they do not necessarily represent an improvement on the lower bound for the instability of LIFO. For every (n, n)-th extension, instability appears when the number of packets that can be accumulated at the end of every inductive phase is greater than the number of packets at the beginning of it, i.e., when

$$Q(\mathcal{N}', n, n) \;=\; Q(\mathcal{N}', n-1, n-1) + r^{(n+1)} s \cdot (2r-1) \;=\; \frac{s\left(r^2 - 2r^{(n+3)} + r^{(n+2)}\right)}{1-r} > s\,, \tag{3}$$

or equivalently, when

$$\frac{Q(\mathcal{N}', n, n)}{s} = \frac{\left(r^2 - 2r^{(n+3)} + r^{(n+2)}\right)}{1-r} > 1. \tag{4}$$

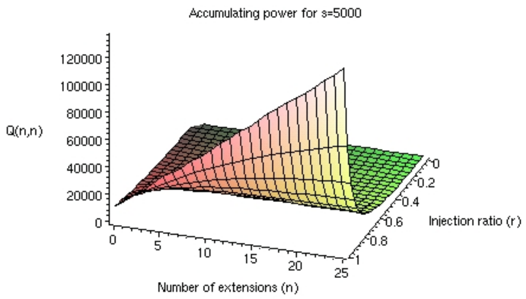
The first injection ratio r for which this expression holds defines a new (and better) lower bound for the instability of the LIFO protocol. This occurs, for example, for any $r > 0.630$ with the $(3, 3)$-th extension, for any $r > 0.620$ with the $(7, 7)$-th extension and, for any $r > 0.6184$ with the $(10, 10)$-th extension. There is an asymptotic frontier for the expression in (4) concerning the (n, n)-th extension when n tends to infinite. Since

$$\lim_{n\to\infty} \frac{\left(r^2 - 2r^{(n+3)} + r^{(n+2)}\right)}{1-r} = \frac{r^2}{1-r}$$

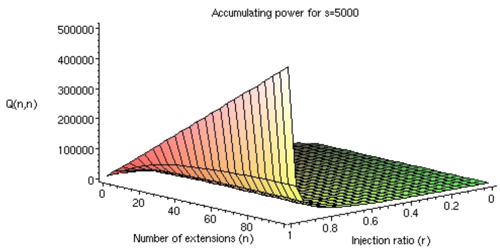
becomes greater than one when $r > 0.61804$, this implies that, even if we would extend the original network $\mathcal{N}'$ infinitely and apply an analogous adversarial strategy over it, the lower bound for the instability of LIFO would never be better than 0.61804. [6]

[5] Figure 5 depicts this behaviour graphically for a fixed initial configuration.

[6] Observe that this bound is already almost obtained with the $(10, 10)$-th extension.



(a) First 25 extensions for every $0 < r < 1$.



(b) First 100 extensions for every $0 < r < 1$.

Fig. 5. Quantity of packets accumulated (per inductive phase) by the (n, n)-th extension of $\mathcal{N}'$ with a fixed initial configuration of $s = 5000$ packets.

4 Conclusions and open questions

Under the Adversarial Queueing model, we have studied the instability of the *last-in-first-out* scheduling protocol (LIFO) in under-loaded packet-switched networks whose topologies have inverted symmetry. Topologies with these properties were used previously in the literature to show instability of some systems when using specific protocols to forward the packets between nodes. Indirectly, this also provides a lower bound on the instability of the protocol.

We have proposed a new network that improves the lower bound for the instability of LIFO in [4] from $1/\sqrt{2}$ to 0.658. The new network $\mathcal{N}$ we propose is the $(1, 1)$-th extension of the network $\mathcal{N}'$ used in [4] (see Figure 4(b)). The technique of topology extension was already used in the literature to improve the existing lower bounds for other protocols (e.g., for FIFO [7,17,14]), but never applied for LIFO before. Although this lower bound improvement is good news, it has no further relevance anymore from the protocol point of view, since nowadays we know that LIFO can be made unstable at arbitrarily low rates [6]. The focus of this result is therefore rather on the technique and its power than on the improvement of the lower bound.

We have also shown that this technique would provide further improvements in additional applications. Given a network which is the (n, n)-th extension of $\mathcal{N}'$, instability would hold as soon as $Q(\mathcal{N}', n, n) > s$, where $Q(\mathcal{N}', n, n)$ denotes the quantity of packets than can be accumulated at each phase of the inductive reasoning (see Equations 2 and 3). The first injection ratio for which this holds defines a lower bound for the instability of the protocol. Each (n, n)-th extension provides an improvement on the lower bound with respect to the lower bound obtained by the previous $(n-1, n-1)$-th extension. However, we show that this technique only achieves a limited number of improvements. Not even when extending the topology of the network infinitely it is possible to obtain a better lower bound for the instability of LIFO better than 0.61804. Once this limit is reached, more clever adversaries or new alternative techniques should be found to go on improving lower bounds for instability.

According to the adversarial strategies used in this work, it seemed reasonable to extend always the topology of the network the same number of times in both directions (vertical and horizontal), i.e, to deal with (n, n)-th extensions. However, another interesting topic for future research is to study what happens when dealing with (m, n)-th extensions for the cases in which $n \neq m$.

References

1. C. Àlvarez, M. Blesa, J. Díaz, A. Fernández, and M. Serna. Adversarial models for priority-based networks. In B. Rovan and P. Vojtas, editors, *28th International Symposium on Mathematical Foundations of Computer Science* (MFCS'03), volume 2747 of *Lecture Notes in Computer Science*, pages 142–151, Bratislava, Slovakia, August 25-29 2003. Springer-Verlag.

2. C. Àlvarez, M. Blesa, and M. Serna. Universal stability of undirected graphs in the adversarial queueing model. In *14th ACM Symposium on Parallel Algorithms and Architectures* (SPAA'02), pages 183–197, Winnipeg, Canada, August 2002. ACM Press New York, USA.
3. C. Àlvarez, M. Blesa, and M. Serna. A characterization of universal stability in the adversarial queuing model. *SIAM Journal on Computing*, 2003. Accepted for publication.
4. M. Andrews, B. Awerbuch, A. Fernández, J. Kleinberg, T. Leighton, and Z. Liu. Universal stability results for greedy contention–resolution protocols. *Journal of the ACM*, 48(1):39–69, 2001.
5. R. Bhattacharjee and A. Goel. Instability of FIFO at arbitrarily low rates in the adversarial queueing model. In *44th Annual IEEE Symposium on Foundations of Computer Science* (FOCS'03), pages 160–167, Cambridge, USA, October 11-14 2003. IEEE Computer Society Press.
6. A. Borodin, J. Kleinberg, P. Raghavan, M. Sudan, and D. Williamson. Adversarial queueing theory. *Journal of the ACM*, 48(1):13–38, 2001.
7. J. Díaz, D. Koukopoulos, S. Nikoletseas, M. Serna, P. Spirakis, and D. Thilikós. Stability and non-Stability of the FIFO Protocol. In *13th annual ACM Symposium on Parallel Algorithms and Architectures* (SPAA'01), pages 48–52, Crete Island, Greece, July 2001.
8. J. Echagüe, V. Cholvi, and A. Fernández. Universal stability results for low rate adversaries in packet switched networks. *IEEE Communication Letters*, 7(12), December 2003.
9. A. Goel. Stability of networks and protocols in the adversarial queueing model for packet routing. *Networks*, 37(4):219–224, 2001.
10. B. Hajek. Large bursts don't cause instability. *IEEE Transactions on Automatic Control*, 45:116–118, January 2000.
11. M. Hamdi. LCFS queuing for real-time audio-visual services in packet networks. In *International Workshop on Audio-Visual Services over Packet Neworks* (AVSPN'97), Aberdeen, September 1997.
12. M. Hamdi, R. Noro, and J.-P. Hubaux. *Fresh Packet First* scheduling for voice traffic in congested networks. Technical Report SSC034, Computer Science Department, Swiss Federal Institute of Technology (EPFL), December 1997.
13. D. Koukopoulos, M. Mavronicolas, S. Nikoletseas, and P. Spirakis. On the stability of compositions of universally stable, greedy contention-resolution protocols. In D. Malkhi, editor, *16th International Symposium on Distributed Computing* (DISC'02), volume 2508 of *Lecture Notes in Computer Science*, pages 88–102, Toulouse, France, October 28-30 2002. Springer.
14. D. Koukopoulos, M. Mavronicolas, S. Nikoletseas, and P. Spirakis. The impact of network structure on the stability of greedy protocols. In R. Petreschi, G. Persiano, and R. Silvestri, editors, *5th International Conference on Algorithms and Complexity* (CIAC'03), volume 2653 of *Lecture Notes in Computer Science*, pages 251–263, Rome, Italy, May 28-30 2003. Springer.
15. D. Koukopoulos, M. Mavronicolas, and P. Spirakis. FIFO is unstable at arbitrarily low rates. *Electronic Colloquium on Computational Complexity*, 10(16), 2003.
16. D. Koukopoulos, M. Mavronicolas, and P. Spirakis. Instability of networks with quasi-static link capacities. In J. F. Sibeyn, editor, *10th Internaltional Colloquium on Structural Information Complexity* (SIROCCO'03), volume 17 of *Proceedings in Informatics*, pages 179–194, Umeå, Sweden, June 18-20 2003. Carleton Scientific.
17. D. Koukopoulos, S. Nikoletseas, and P. Spirakis. Stability behavior of FIFO protocol in the adversarial queueing model. In A. K. Y. Manolopoulos, S. Evripidou, editor, *8th Panhellenic Conference on Informatics* (PCI'2001), volume 2563 of *Lecture Notes in Computer Science*, Nicosia, Cyprus, November 8-10 2001. Springer-Verlag.
18. D. Koukopoulos, S. Nikoletseas, and P. Spirakis. Stability issues in heterogeneous and FIFO networks under the adversarial queueing model. In *8th International Conference on High Performance Computing* (HIPC'01), volume 2228 of *Lecture Notes in Computer Science*, pages 3–14, Hyderabad, India, December, 17-20 2001. Springer-Verlag Heidelberg.
19. Z. Lotker, B. Patt-Shamir, and A. Rosén. New stability results for adversarial queuing. *SIAM Journal on Computing*, 33(2):286–303, 2004.
20. L. Yongxiang, K. Srijith, L. Jacob, and A. Ananda. TCP-CM: a transport protocol for TCP-friendly transmission of continuous media. In *21st International Performance, Computing, and Communications Conference* (IPCCC'02), pages 83–91, Phoenix, USA, April 3-5 2002.
21. C. Yoon and C. Un. Performance of personal portable radio telephone systems with and without guard channels. *IEEE Journal on Selected Areas in Communications*, 11(6):911–917, 1993.

Departament de Llenguatges i Sistemes Informàtics
Universitat Politècnica de Catalunya

Research Reports - 2004

- LSI-04-1-R : *Automatic Generation of Polynomial Loop Invariants: Algebraic Foundations*, Rodríguez, E. and Kapur, D.
- LSI-04-2-R : *Comparison of Methods to Predict Ozone Concentration* , Orozco, J.
- LSI-04-3-R : *Towards the definition of a taxonomy for the cots product´s market* , Ayala, Claudia P.
- LSI-04-4-R : *Modelling Coalition Formation over Time for Iterative Coalition Games*, Mérida-Campos, C. and Willmott, S.
- LSI-04-5-R : *Illegal Agents? Creating Wholly Independent Autonomous Entities in Online Worlds*, Willmott, S.
- LSI-04-6-R : *An Analysis Pattern for Electronic Marketplaces*, Queralt, A. and Teniente, E.
- LSI-04-7-R : *Exploring Dopamine-Mediated Reward Processing through the Analysis of EEG-Measured Gamma-Band Brain Oscillations*, Vellido, A. and El-Deredy, W.
- LSI-04-8-R : *Studying Embedded Human EEG Dynamics Using Generative Topographic Mapping*, Vellido, A. and El-Deredy, W. and Lisboa, P.J.G.
- LSI-04-9-R : *Similarity and Dissimilarity Concepts in Machine Learning*, Orozco, J.
- LSI-04-10-R : *A Framework for the Definition of Metrics for Actor-Dependency Models*, Quer, C. and Grau, G. and Franch, X.
- LSI-04-11-R : *QM: A Tool for Building Software Quality Models*, Carvallo, J.P. and Franch, X. and Grau, G. and Quer, C.
- LSI-04-12-R : *COSTUME: A Method for Building Quality Models for Composite COTS-based Software Systems*, Carvallo, J.P. and Franch, X. and Grau, G. and Quer, C.
- LSI-04-13-R : *Enabling Collaboration in Virtual Reality Navigators*, Theoktisto, V. and Fairén, M. and Navazo, I.
- LSI-04-14-R : *DesCOTS: A Software System for Selecting COTS Components*, Carvallo, J.P. and Franch, X. and Grau, G. and Quer, C.
- LSI-04-15-R : *Evaluation and symmetrisation of alignments obtained with the Giza++ software*, Lambert, P. and Castell, N.
- LSI-04-16-R : *A note on the use of topology extensions for provoking instability in communication networks*, Blesa, M.J.

Hardcopies of reports can be ordered from:

Núria Sanchez

Departament de Llenguatges i Sistemes Informàtics
Universitat Politècnica de Catalunya
Campus Nord, Mòdul C6
Jordi Girona Salgado, 1-3
03034 Barcelona,Spain
nurias@lsi.upc.es

See also the Departament WWW pages, http://www.lsi.upc.es/