The Impact of Network Structure on the Stability of Greedy Protocols

Dimitrios Koukopoulos[1], Marios Mavronicolas[2],
Sotiris Nikoletseas[1], and Paul Spirakis[1,⋆]

[1] Department of Computer Engineering & Informatics, University of Patras
and Computer Technology Institute (CTI),
Riga Fereou 61, P. O. Box 1122, 261 10 Patras, Greece
`koukopou,nikole,spirakis@cti.gr`
[2] Department of Computer Science, University of Cyprus, 1678 Nicosia, Cyprus
`mavronic@ucy.ac.cy`

Abstract. A *packet-switching* network is *stable* if the number of packets in the network remains *bounded* at all times. A very natural question that arises in the context of stability and instability properties of such networks is how network structure precisely affects these properties. In this work, we embark on a systematic study of this question in the context of *Adversarial Queueing Theory,* which assumes that packets are adversarially injected into the network. We consider *size, diameter,* maximum *vertex degree,* minimum number of *disjoint paths* that cover all edges of the network, and *network subgraphs* as crucial structural parameters of the network, and we present a comprehensive collection of structural results, in the form of bounds on both stability and instability thresholds for various greedy protocols:

  – We present a novel, yet simple and natural, construction of a network parameterized by its size on which certain compositions of *universally stable,* greedy protocols are unstable for low rates. The closeness of the drop to 0.5 is proportional to the increase in size.

  – It is now natural to ask how unstable networks with *small* (constant) size be. We show that size of 22 suffices to drop the instability threshold for the FIFO protocol down to 0.704. This results is the current state-of-the-art trade-off between size and instability threshold.

  – The diameter, maximum vertex degree and minimum number of edge-disjoint paths play a significant role in an improved analysis of stability threshold for the FIFO protocol. The results of our analysis reveal that a calibration of these parameters may be a valuable asset for the design of networks with as high as possible stability threshold.

  – How much can network subgraphs that are forbidden for stability affect the instability threshold? Through improved combinatorial constructions of networks and executions, we improve the state-of-the-art instability threshold induced by certain known forbidden subgraphs on networks running a certain greedy protocol.

Our results shed more light and contribute significantly to a finer understanding of the impact of structural parameters on stability and instability properties of networks.

1   Introduction

Motivation-Framework. *Objectives.* A lot of research has been done in the field of packet-switched communication networks for the specification of their behavior. In such networks, packets arrive dynamically at the nodes and they are routed in discrete time steps across the edges. In this work, we embark on a study of the impact structural network properties have on the correctness and performance properties of networks. We study here *greedy* protocols as our test-bed. In some cases, we consider networks in which different switches can use different greedy protocols. This is motivated by the *heterogeneity* of modern large-scale networks such as the Internet.

Framework of Adversarial Queueing Theory. We focus on a basic adversarial model for packet arrival and path determination that has been recently introduced in a pioneering work by Borodin *et al.* [3]. It was developed as a robust counterpart to classical Queueing theory [4] that replaces stochastic by worst case assumptions. The underlying goal is to determine whether it is feasible to prove stability results even when packets are injected by an *adversary*. At each time step, the adversary may inject a set of packets into some nodes. For each packet, the adversary specifies a simple path that the packet must traverse; when the packet arrives to its destination, it is absorbed by the system. When more than one packets wish to cross a queue at a given time step, a *contention-resolution* protocol is employed to resolve the conflict. A crucial parameter of the adversary is its *injection rate* r, where $0 < r < 1$. Among the packets that the adversary injects in any time interval I, at most $\lceil r|I| \rceil$ can have paths that require any particular edge. We say that a packet p *requires* an edge e at time t if the edge e lies on the path from its position to its destination at time t.

Stability. *Stability* requires that the number of packets in the system remains bounded at all times. We say that a protocol P is *stable* [3] on a network $\mathcal{G}$ against an adversary $\mathcal{A}$ of rate r if there is a constant C (which may depend on $\mathcal{G}$ and $\mathcal{A}$) such that the number of packets in the system is bounded at all times by C. We say that *a protocol* P *is universally stable* [3] if it is stable against every adversary of rate less than 1 and on every network. We also say that *a network* $\mathcal{G}$ *is universally stable* [3] if every greedy protocol is stable against every adversary of rate less than 1 on $\mathcal{G}$.

Greedy Protocols. We consider six *greedy* protocols– ones that always advance a packet across a queue (but one packet at each discrete time step) whenever there resides at least one packet in the queue (see Table 1).

Network Structure. Important parameters of it are: (a) the used protocols, (b) graph parameters such as minimum degree, diameter, size, (c) forbidden subgraphs for stability and (d) the subclasses of parameterized families of networks.

Table 1. Greedy protocols considered in this paper (**US** stands for universally stable).

Protocol name	Which packet it advances:	US
Shortest-in-System (SIS)	The most recently injected packet	$\sqrt{}$
Longest-in-System (LIS)	The least recently injected packet	$\sqrt{}$
Furthest-to-Go (FTG)	The furthest packet from its destination	$\sqrt{}$
Nearest-to-Source (NTS)	The nearest packet to its origin	$\sqrt{}$
First-In-First-Out (FIFO)	The earliest arrived packet at the queue	**X**
Nearest-To-Go-Using-LIS (NTG-U-LIS)	The nearest packet to its destination or the least recently injected packet for tie-breaking	**X**

Contribution. How does the network structure precisely affect stability? In this work, we present a comprehensive collection of structural results in the form of bounds, on both stability and instability thresholds.

– We present an innovative *parameterized* adversarial construction for estimating *instability* thresholds in heterogeneous networks. Our parameterized approach considers that each execution (phase) of the adversarial construction consists of distinguished time periods (rounds) whose number depends on the parameterized network topology. We apply our construction in instances of a parameterized network family and prove that when the network size parameter k tends to infinity then the instability threshold for the compositions of LIS-SIS, LIS-NTS and LIS-FTG fast converges to 0.5.
– We present a general analysis showing that any network $\mathcal{G}$ has an upper bound on injection rate for FIFO stability that depends only on the minimum number of edge-disjoint paths that cover $\mathcal{G}$, the maximum in-degree, and the maximum directed path length of the network. This result improves the previous known upper bound for FIFO stability of [5] for all networks. Furthermore, for several networks our stability bound is better than the one estimated in [8] such as the network $\mathcal{U}_1$ in Figure 1.
– We demonstrate an ad-hoc FIFO network that uses only 22 queues and it is unstable for any $r \geq 0.704$. The corresponding parameterized network [8] for $r = 0.704$ needs at least 361 queues. Thus, we show that ad-hoc constructions may beat the parametric ones with respect to the network size.
– In the model of *non-simple paths* (paths do not contain repeated edges), we study two simple graphs ($\mathcal{U}_2$ and $\mathcal{U}_3$ in Figures 1, 4) that have been shown in [2] to be forbidden subgraphs for universal stability. Note that $\mathcal{U}_3$ is an extension of $\mathcal{U}_1$ (Figures 1) for $n = 0$, $m = 1$ and $d = 2$. For these graphs we show instability for lower rates than those in [2] via a different construction.

Related Work. *Adversarial Queueing Theory* was developed by Borodin *et al.* [3] as a more realistic model that replaces traditional stochastic assumptions in Queueing Theory by more robust, worst-case ones. Subsequently, adversarial queueing theory, and corresponding stability and instability issues, received a lot of interest and attention (see, e.g., [1,2,5,6,7,9]). The universal stability of SIS, LIS, NTS and FTG protocols was established by Andrews *et al.* [1]. The subfield
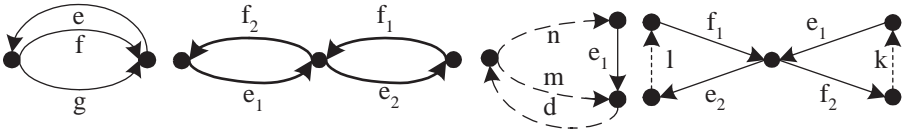
Fig. 1. Networks $\mathcal{U}_1$, $\mathcal{U}_2$ and their extensions $\Gamma(\mathcal{U}_1)$ and $\Gamma(\mathcal{U}_2)$ [2, Lemma 7]

of study of the stability properties of compositions of universally stable protocols has been opened recently by *Koukopoulos et al.* [6,7] where lower bounds of 0.683 and 0.519 on the instability threshold of the composition pairs LIS-SIS, LIS-NTS and LIS-FTG were respectively presented.

The subfield of proving stability thresholds for greedy protocols on every network was first initiated by Diaz *et al.* [5] showing an upper bound on injection rate for the stability of FIFO in networks with a finite number of queues that is based on network parameters. In an alternative work, Lotker *et al.* [8] proved that any greedy protocol can be stable in any network if the injection rate of the adversary is upper bounded by $1/(d+1)$, where d is the maximum path length that can be followed by any packet. Also, they proved that for a specific class of greedy protocols, time-priority protocols the stability threshold becomes $1/d$.

The instability of FIFO for *small-size* networks (in the model of adversarial queueing theory) was first established by Andrews *et al.* [1, Theorem 2.10] for injection rate $r \geq 0.85$. Lower bounds of 0.8357 and 0.749 on FIFO instability were presented by Diaz *et al.* [5, Theorem 3] and Koukopoulos *et al.* [6, Theorem 5.1]. An alternative approach for studying FIFO instability is based on parameterized constructions for networks with *unbounded size*. Using this approach, Lotker *et al.* [8] proved an instability threshold of $\frac{1}{2} + \epsilon$ for FIFO; the network size is a function of r that goes to infinity very fast as r goes down to 0.5.

In [2, Lemma 7], a characterization for directed network graphs (digraphs) universal stability is given when the packets follow non-simple paths (paths do not contain repeated edges). According to this characterization a digraph is universally stable if and only if it does not contain as subgraph any of the extensions of $\mathcal{U}_1$ ($\Gamma(\mathcal{U}_1)$ or $\mathcal{U}_2$ ($\Gamma(\mathcal{U}_2)$) where the parameters n, m, d, l, k represent numbers of consecutive edges with $l, k, n \geq 0$ and $m, d > 0$ (see Figure 1). These graphs have been shown to have instability thresholds of 0.84089 for a certain greedy protocol.

2   The Model

The adversarial queueing model considers a communication network that is modelled by a directed graph $\mathcal{G} = (V, E)$, where $|V| = n, |E| = m$. Each node $u \in V$ represents a communication switch, and each edge $e \in E$ represents a link between two switches. In each node, there is a buffer (queue) associated with each outgoing link. Buffers store packets that are injected into the network with a

route, which is a simple directed path in $\mathcal{G}$. When a packet is injected, it is placed in the buffer of the first link on its route.

In order to formalize the behavior of a network under the adversarial queueing model, we use the notions of *system* and *system configuration*. A triple of the form $\langle \mathcal{G}, \mathcal{A}, \mathsf{P} \rangle$ where $\mathcal{G}$ is a network, $\mathcal{A}$ is an adversary and P is the used protocol on the network queues is called a system. The execution of the system proceeds in global time steps numbered $0, 1, \ldots$. Each time-step is divided into two sub-steps. In the first sub-step, one packet is sent from each non-empty buffer over its corresponding link. In the second sub-step, packets are received by the nodes at the other end of the links; they are absorbed (eliminated) if that node is their destination, and otherwise they are placed in the buffer of the next link on their respective routes. In addition, new packets are injected in the second sub-step. Furthermore, the configuration C^t of a system $\langle \mathcal{G}, \mathcal{A}, \mathsf{P} \rangle$ in every time step t is a collection of sets $\{S_e^t : e \epsilon \mathcal{G}\}$, such that S_e^t is the set of packets waiting in the queue of the edge e at the end of step t. The time evolution of the system is a sequence of such configurations $C^1, C^2, \ldots$ such that the load restriction imposed by the Adversarial Queueing Theory [3] is satisfied.

In the adversarial constructions we study here for proving instability, we assume that there is a sufficiently large number of packets s_0 in the initial system configuration. This will imply instability results for networks with an *empty* initial configuration, as established by Andrews *et al.* [1, Lemma 2.9]. Also, for simplicity, and in a way similar to that in [1], we omit floors and ceilings and sometimes count time steps and packets roughly. This only results to loosing small additive constants while we gain in clarity.

3   Stability in Heterogeneous Networks

Before proceeding to the adversary constructions we give two basic definitions.

Definition 3.1 *We denote X_i the set of packets that are injected into the system in the ith round of a phase. These packet sets are characterized as "investing" flows because they will remain in the system till the beginning of the next phase.*

Definition 3.2 *We denote $S_{i,j}$ the jth set of packets the adversary injects into the system in the ith round of a phase. These packet sets are characterized as "short intermediate flows" because they are injected on judiciously chosen paths of the network for blocking investing flows.*

A Parameterized Network Family. We provide here a parameterized family of heterogeneous networks $\mathcal{N}_k$. The motivation that led us to such a parameterization in the network topology is *two-fold*: (a) the existence of many parallel queues in the network allows the adversary to *simultaneously* inject several short intermediate flows that block the investing flows in the system, without violating the rule of the restricted adversarial model, (b) such a parameterized network topology construction, enables *a parameterized analysis* of the system configuration evolution into distinguished rounds whose number depends on the parameterized network topology. In LIS-FTG composition, the parameterization, besides
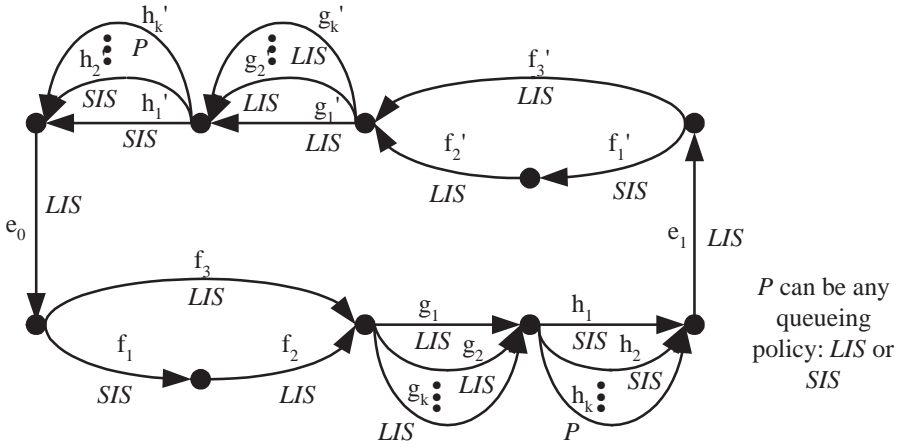
Fig. 2. A network $\mathcal{N}_k$ that uses LIS-SIS protocols

the parallel edges, includes additional chains of queues for the exploitation of FTG in blocking investing flows.

A Parameterized Adversarial Construction. In order for our adversarial construction to work, we split the time into phases. In each phase we study the evolution of the system configuration by considering distinguished time rounds. For each phase, we inductively show that the number of packets in the system increases. Applying repeatedly this inductive argument we show instability.

Theorem 3.1 *Let $r > 0.5$. There is a network $\mathcal{N}_k$ where k is a parameter linear to the number of network queues and an adversary $\mathcal{A}$ of rate r, such that the system $\langle \mathcal{N}_k, \mathcal{A}, \mathsf{Pr} \rangle$ is unstable if Pr is a composition of LIS protocol with any protocol of a) SIS, b) NTS and c) FTG.*

Sketch of proof. Part a) This proof is based on the preservation of all the investing flows injected during a phase into the system. We consider an instance of the parameterized network family (network $\mathcal{N}_k$ in Figure 2). All the queues use the LIS protocol except the queues $f_1, f_1', h_1, \ldots, h_{k-1}, h_1', \ldots, h_{k-1}'$ that use the SIS protocol. Moreover, the edges h_k, h_k' use protocol P, that can be any of LIS or SIS because there is no packet conflict in these queues.

Inductive Hypothesis: At the beginning of phase j, there are s_j packets that are queued in $g_1', h_1', \ldots, h_{k-1}'$ requiring to traverse the edges e_0, f_1, f_2, g_1, h_1.

Induction Step: At the beginning of phase $j+1$ there will be more than s_j packets (s_{j+1} packets) that will be queued in $g_1, h_1, \ldots, h_{k-1}$ requiring to traverse the edges $e_1, f_1', f_2', g_1', h_1'$.

We construct an adversary $\mathcal{A}$ such that the induction step holds. Proving that the induction step holds, we ensure that the inductive hypothesis will hold at the beginning of phase $j+1$ for the symmetric edges with an increased value of s_j, $s_{j+1} > s_j$. From the inductive hypothesis, initially, there are s_j packets (called $S - flow$) in the queues $g_1', h_1', \ldots, h_{k-1}'$ requiring to traverse the edges

e_0, f_1, f_2, g_1, h_1. In order to prove that the induction step works it is assumed that there is a large enough number s_j of packets in the initial system configuration. Phase j consists of $l = k + 1$ rounds with $l \geq 3$, that is $k \geq 2$. The sequence of injections is as follows:

Round 1: It lasts s_j steps. The adversary injects a set X_1 of $|X_1| = rs_j$ packets in e_0 wanting to traverse the edges $e_0, f_3, g_1, h_1, e_1, f_1', f_2', g_1', h_1'$ and a set $S_{1,1}$ of $|S_{1,1}| = rs_j$ packets in f_1 that require to traverse only the edge f_1.

Evolution of the system configuration. X_1 packets are blocked by the $S - flow$ packets in e_0. $S - flow$ packets are delayed by $S_{1,1}$ packets in f_1. Thus, at the end of this round a set Y of $|Y| = rs_j$ packets of S remain in queue f_1.

Round 2: It lasts rs_j steps. The adversary injects a set X_2 of $|X_2| = r^2 s_j$ packets in e_0 requiring to traverse the edges $e_0, f_3, g_1, h_2, e_1, f_1', f_2', g_1', h_1'$. Also, it injects a set $S_{2,1}$ of $|S_{2,1}| = r^2 s_j$ packets in f_2 wanting to traverse f_2, g_2, h_1.

Evolution of the system configuration. X_1 and $S_{2,1}$ packets are blocked in g_1 and f_2 correspondingly by Y packets. X_2 packets are blocked in e_0 by X_1 packets.

Since the number of rounds depends on the network topology (i.e. $l = k+1$), we next analyze an intermediate round t, $3 \leq t < l$.

Round t (intermediate round): It lasts $r^{t-1}s_j$ steps. The adversary injects $t - 1$ short intermediate flows $S_{t,1}, \ldots, S_{t,t-1}$ of $|S_{t,1}| = \ldots = |S_{t,t-1}| = r^t s_j$ packets. Flow $S_{t,j}$ with $1 \leq j \leq t - 1$ and $j \neq 2$ is injected in g_{j+1} wanting to traverse the edges g_{j+1}, h_j. Flow $S_{t,2}$ is injected in queue f_2 wanting to traverse the edges f_2, g_3, h_2. In addition, an investing flow X_t of $|X_t| = r^t s_j$ packets is injected in f_3 wanting to traverse the edges $f_3, g_1, h_t, e_1, f_1', f_2', g_1', h_1'$.

Evolution of the system configuration. $S_{t,j}$ packets with $3 \leq j \leq t-1$ are blocked in g_{j+1} by $S_{t-1,j-1}$ packets, while flows $S_{t,1}$ and $S_{t,2}$ are blocked in g_2 and f_2 correspondingly by $S_{t-1,1}$ packets. X_t packets are blocked in g_1 by X_{t-1} packets that were injected in round $t - 1$. At the end of round t, there is a number of $t - 3$ different cases for the queues where $X_1, \ldots, X_{t-1}$ are queued depending on their position at the beginning of the round and the injection rate r. In case i $(1 \leq i \leq t - 3)$ we have: at the beginning of round t, a portion or all X_i packets along with $X_{i+1}, \ldots, X_{t-1}$ are queued in g_1, while the rest X_i packets are queued in h_i and all the $X_1, \ldots, X_{i-1}$ packets are queued in $h_1, \ldots, h_{i-1}$ correspondingly. Then, at the end of round t, two cases can happen. In both cases, $X_1, \ldots, X_{i-1}$ packets are queued in $h_1, \ldots, h_{i-1}$ correspondingly. In the first one, a portion or all X_i packets in g_1 are queued with the rest X_i packets in h_i and $X_{i+1}, \ldots, X_{t-1}$ packets remain in g_1. In the second one, all the X_i packets in g_1 are queued with the rest X_i packets in h_i, a portion of X_{i+1} packets is queued in h_{i+1}, while the rest X_{i+1} packets along with $X_{i+2}, \ldots, X_{t-1}$ remain in g_1. Note that, in all possible system configurations at the end of round t, *the investing flows $X_1, \ldots, X_t$ remain into the system.*

Round l: It lasts $r^{l-1}s_j$ steps. The adversary injects an investing flow X_l of $|X_l| = r^l s_j$ packets in f_3 wanting to traverse the edges $f_3, g_1, h_l, e_1, f_1', f_2', g_1', h_1'$.

Evolution of the system configuration. X_l packets are blocked in g_1 by X_{l-1} that was injected in the system at round $l-1$. $X_1, \ldots, X_{l-1}$ packets are blocked in the system by flows $S_{l-1,1}, \ldots, S_{l-1,l-2}$. Therefore, at the end of round l, the number

of packets that are queued in $g_1, h_1, \ldots, h_{l-2} = h_{k-1}$ requiring to traverse the edges $e_1, f'_1, f'_2, g'_1, h'_1$ is $s_{j+1} = |X_1| + \ldots + |X_l|$.

In order to have instability, we must have $s_{j+1} > s_j$. This holds for $r^{k+2} - 2r + 1 < 0$. This argument can be repeated for an infinite number of phases ensuring the instability of the system $\langle N_k, \mathcal{A}, \mathsf{LIS} - \mathsf{SIS} \rangle$. Also, $k \to \infty \implies r^{k+2} \to 0$, because $0 < r < 1$. Thus, for instability it suffices $-2r + 1 < 0$, i.e. $r > 0.5$.

Parts b, c) The adversarial constructions that are used in *Parts a, b, c.* One difference is the use of NTS and FTG protocols in the networks of *Parts b, c* correspondingly where SIS is used in the network of *Part a*. The topology of the used networks in *Parts a, b, c* is similar. The only difference is the use of additional paths in the network of *Part b* that start at queues that use FTG. These paths have sufficient lengths, such that the short intermediate packet flows have priority over the investing packet flows when they conflict in queues that use FTG. On the other hand, the injection of short intermediate flows in *Part b* with the same paths as in *Part a* is enough to guarantee their priority over investing flows when they conflict in queues that use NTS. As in *Part a*, it is proved that for any $r > 0.5$ the system $\langle \mathcal{N}_k, \mathcal{A}, \mathsf{Pr} \rangle$ is unstable if Pr is a composition of LIS NTS or FTG. □

Notice that our method converges very fast to 0.5 for small values of the parameter k that depends on the network size. This can be shown easily if in the inequality $r^{k+2} - 2r + 1 < 0$ the parameters r, k are replaced by appropriate values. Therefore, for $k = 7$ the instability threshold is 0.501 and the number of network queues is 36 in the case of LIS-SIS and LIS-NTS (given by $8 + 4k$), while it is 102 in the case of LIS-FTG (given by $14 + 4k + 10(k - 1)$).

4   Structural Conditions for FIFO Stability

We denote "old" a packet that was injected in previous time periods than the current one. The earliest time step in a time period, at which all the old packets in the system have been served is denoted by M. Denote $j(\mathcal{G})$ the minimum number of edge-disjoint paths that cover the network $\mathcal{G}$. Thus, consider the disjoint paths $\Pi_1, \Pi_2, \ldots, \Pi_{j(\mathcal{G})}$. The number of packets in the path Π_j, where $1 \leq j \leq j(\mathcal{G})$, at time step M will be denoted as $s(\Pi_j)$. Furthermore, the maximum in-degree, and the maximum directed path length of the network $\mathcal{G}$ will be denoted as $\alpha(\mathcal{G})$ and $d(\mathcal{G})$ respectively. Notice that if $\alpha(\mathcal{G}) = 1$ then we have a tree or a ring, that is known to be universally stable [1], so we assume $\alpha(\mathcal{G}) > 1$. We show:

Theorem 4.1 *Let* $r_\mathcal{G}^2 \Sigma_{i=0}^{d(\mathcal{G})-1} (\alpha(\mathcal{G}) + r)^i = \frac{1}{j(\mathcal{G})}$. *Then for any network* $\mathcal{G}$, *and any adversary with* $r \leq r_\mathcal{G}$ *the system* $\langle \mathcal{G}, \mathcal{A}, \mathsf{FIFO} \rangle$ *is stable.*

Proof. Let us denote the queues of $\mathcal{G}$ as $Q_1, Q_2, \ldots, Q_m$ and their loads at time $t \geq 0$ as $q_1(t), q_2(t), \ldots, q_m(t)$. Let $P(0) = \Sigma q_i(0)$ be the initial load. We will construct an infinite sequence of consecutive distinguished time periods, t_i, at which $P(t_i) \leq P(0)$ thus keeping the network stable. The fact that we are using a FIFO protocol implies that after a certain time all the old packets will leave the system. We will compute a bound to this time.

Let's now, consider the worst case of an old packet being last in a queue Q_j at time 0 and targeted with the largest simple path in the network. Rename the queues in this simple path as $Q_j \equiv Q_{j_0}, \ldots = Q_{j_{d(\mathcal{G})-1}}$. Note that at time $M_1 = q_{j_0}$ all packets of this queue will have been served. Thus these packets have passed to the next queues in the path. Moreover, they can be delayed by at most rM_1 new injections. Furthermore, the size of any Q_{j_i} is bounded above by $(\alpha(\mathcal{G}) + r)M_1$. We repeat the same procedure, each time considering the last queue in the path that still contains old packets. After $d(\mathcal{G}) - 2$ additional steps $(M_2, M_3, \ldots, M_{d(\mathcal{G})-1})$ all the old packets would disappear or being in $Q_{j_{d(\mathcal{G})}}$. Define $P(t) = \max_{i=0}^m \{q_i(t)\}$. Working in the previous way, an absolute bound for the delay of the last old packet in Q_j is $M = M_1 + \ldots + M_{d(\mathcal{G})-1}$, where for every $0 < i < d(\mathcal{G})$, we have $M_1 \leq q(\Sigma_{j<i} M_j)$, with $M_0 = 0$. Moreover, during a period of $q(t)$ steps starting at time t, we have $P(t + P(t)) \leq (\alpha(\mathcal{G}) + r)P(t)$. Solving the recurrence, the total time is $M \leq \Sigma_{i=0}^{d(\mathcal{G})-1}(\alpha(\mathcal{G}) + r)^i P(0)$

At time step M all the old packets have been absorbed and only the injected packets in the time period $[0 \ldots M]$ will remain in the system. Because $j(\mathcal{G})$ is the minimum number of edge-disjoint paths in the network, during this period in the worst case at most $j(\mathcal{G})rM$ packets will be injected in the network. Therefore the total number of packets in the network at time step M is at most $P(M) \leq j(\mathcal{G})rM$. At time step M, $s(\Pi_j)$ packets exist in each disjoint path Π_j from the definitions. Note that the minimum number of packets in a disjoint path Π_j at time step M $(min\{s(\Pi_j)\})$ is significantly bigger comparing to the number of network edges. This allows us to assume that when a disjoint path Π_j has $s(\Pi_j)$ packets, then in each time step of a time period of $s(\Pi_j)$ time steps, r packets arrive into the path and one packet leaves it.

Assume now $s = \min \{s(\Pi_j)\}$. The change of the number of packets in the disjoint path Π_j in absolute values, Δ_{Π_j}, at $M + s$ time step will be $\Delta_{\Pi_j} = \Sigma_0^{\min\{s(\Pi_j)\}}|r-1| = |r-1|\min\{s(\Pi_j)\} \leq |r-1|s(\Pi_j)$. Thus, the total change of the system configuration will be $\Sigma_{\Pi_j}\Delta_{\Pi_j} \leq \Sigma_{\Pi_j}|r-1|s(\Pi_j) = |r-1|\Sigma_{\Pi_j}s(\Pi_j)$. But, $P(M) = \Sigma_{\Pi_j}s(\Pi_j)$. Thus, $\Sigma_{\Pi_j}\Delta_{\Pi_j} \leq |r-1|P(M)$ is at most the change of the system configuration for a time period with $s = min\{s(\Pi_j)\}$ steps. Consider now, the consecutive time intervals with duration: $s, rs, r^2 s, \ldots, r^k s$, where k is such that $r^k s \geq 1$ and $r^{k+1}s < 1$. The same argument as in the case of s time steps can be used for $r^i s$ time steps. For each of these time intervals the change of the system configuration will be at most $r^i(r - 1)P(M)$. Let t_1 be the time at which $r^k s$ finishes. The packets in network $\mathcal{G}$ at time t_1 are all new. Thus, the number of packets in the system at time t_1 is at most $P(t_1) \leq P(M) + (r - 1)P(M) + r(r - 1)P(M) + \ldots + r^{k-1}(r - 1)P(M) = r^k P(M)$.

For stability, we need $P(t_1) \leq P(0)$. Thus, we must choose an r such that $r^k P(M) \leq P(0)$. But, $P(M) \leq j(\mathcal{G})rM \leq j(\mathcal{G})r\Sigma_{i=0}^{d(\mathcal{G})-1}(\alpha(\mathcal{G}) + r)^i P(0)$. Thus, $r^k j(\mathcal{G})r\Sigma_{i=0}^{d(\mathcal{G})-1}(\alpha(\mathcal{G})+r)^i P(0) \leq P(0)$. For $k = 1$ this equation takes its smallest value $r^2\Sigma_{i=0}^{d(\mathcal{G})-1}(\alpha(\mathcal{G}) + r)^i \leq \frac{1}{j(\mathcal{G})}$. This is equivalent to find in the real interval $(0, 1)$, the root $r_{\mathcal{G}}$ of the polynomial $-r^2 j(\mathcal{G})(\alpha(\mathcal{G})+r)^{d(\mathcal{G})}+r^2 j(\mathcal{G})+\alpha(\mathcal{G})+r-1$. By the Bolzano Theorem, this polynomial has a root $r_{\mathcal{G}}$ in $(0, 1)$. $\square$
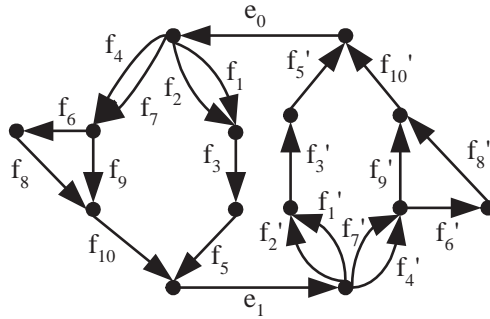
Fig. 3. The Network $\mathcal{G}$

In order to compare this bound with the bound of [5] we make the following analysis: Let m be the number of queues in any network $\mathcal{G}$. We get by definition of the minimum number of edge-disjoint paths $j(\mathcal{G})$ that $\frac{1}{j(\mathcal{G})} \geq \frac{1}{m}$ (1). The bound of [5] is the biggest r satisfying $\frac{2-r}{1-r} rm \Sigma_{i=0}^{d(\mathcal{G})-1} (\alpha(\mathcal{G})+r)^i \leq 1$ (2). Let us call it r_1. Our new bound is the longest value of r satisfying $r^2 j(\mathcal{G}) \Sigma_{i=0}^{d(\mathcal{G})-1} (\alpha(\mathcal{G})+r)^i \leq 1$ (3). Let us call it r_2. Thus, r_1 satisfies equation (2) as an equality. The same holds for r_2 and equation (3). Let $f(r) = \Sigma_{i=0}^{d(\mathcal{G})-1} (\alpha(\mathcal{G})+r)^i$. Note that $f(r)$ is monotone and increasing with $r > 0$. But, $\frac{2-r_1}{1-r_1} r_1 m f(r_1) = 1 \implies \frac{1}{2m} \leq r_1 f(r_1) \leq \frac{1}{m}$ (4) because $\forall r \in (0,1)$ it holds that $1 \leq \frac{2-r}{1-r} \leq 2$. Since $r_2 \leq 1$, it holds $r_2 f(r_2) > r_2^2 f(r_2^2) = \frac{1}{j(\mathcal{G})}$. Thus, from (1), (4) we take $r_2^2 f(r_2^2) > r_1 f(r_1) \implies r_2^2 > r_1$ and this implies $r_2 > r_1$ since $r_1 \in (0,1)$.

Lemma 4.2 *In all networks $\mathcal{G}$, we have $\sqrt{r_1} < r_2$ (thus, $r_2 > r_1$).*

The upper bound $1/d(\mathcal{G})$ was obtained in [8] for FIFO stability. From this and the previous lemma we conclude to the following theorem,

Theorem 4.3 *Let $r^* = \max\{r_2, \frac{1}{d(\mathcal{G})}\}$. Then for every $\mathcal{G}$, and any adversary with $r \leq r^*$ the system $\langle \mathcal{G}, \mathcal{A}, \mathsf{FIFO} \rangle$ is stable.*

To illustrate the strength and applicability of our analytical techniques towards the threshold of $1/d(\mathcal{G})$ for FIFO stability in [8], we apply them to a simple network with three queues (network $\mathcal{U}_1$ in Figure 1). The upper bound for this is $1/3$ in [8], while in our case is:

Corollary 4.4 *Let $0 < r < 0.339$. Then, for network $\mathcal{U}_1$ and any adversary $\mathcal{A}$ with injection rate r the system $\langle \mathcal{U}_1, \mathcal{A}, \mathsf{FIFO} \rangle$ is stable.*

5   Instability of Small-Size FIFO Networks

Theorem 5.1 *Let $r \geq 0.704$. There is a network G and an adversary $\mathcal{A}$ of rate r such that the system $\langle G, \mathcal{A}, \mathsf{FIFO} \rangle$ is unstable.*

Sketch of proof. The main ideas that are hidden behind the adversarial construction we use to prove this theorem are the following: (i) We split the time
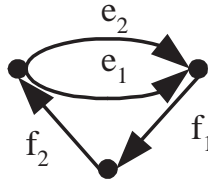
Fig. 4. Network U_3

into phases. In each phase we study the evolution of the system configuration by considering corresponding time rounds. For each phase, we inductively show that the number of packets in the system increases. This inductive argument can be applied repeatedly, thus showing instability. (ii) We use an inductive hypothesis with two parts. The first part specifies the position of the initial packets at the beginning of a phase (at the beginning of phase j, s_j packets are queued in s $e_1, f_3', f_4', f_5', f_6', f_8'$ requiring to traverse the edges e_0, f_1, f_3, f_5) and that their number is smaller than the number of packets in the corresponding subset of queues that will serve as initial packets at the beginning of the next phase (at the beginning of phase $j + 1$, s_{j+1} packets will be queued in $e_0, f_3, f_5, f_4, f_6, f_8$ requiring to traverse the edges e_1, f_1', f_3', f_5'). This part of inductive hypothesis holds for $r \geq 0.704$. The second part guarantees that the initial packets in each phase will traverse their path as a continuous flow. This holds for $r \geq 0.609$. Clearly, $r \geq \max\{0.704, 0.609\} = 0.704$ suffices for instability of the network $\mathcal{G}$. (iii) We achieve further delay of packets initially residing in the system by exploiting multiple "parallel" paths of the network topology. (iv) In order to create instability, we heavily exploit the *fair mixing* property of FIFO stating that if two packet sets arrive at the same queue simultaneously will mix according to the initial proportions of their sizes.   □

6   Unstable Subgraphs

Consider the networks $\mathcal{U}_2$ and $\mathcal{U}_3$ (see Figures 1, 4) that use NTG-U-LIS protocol.

Theorem 6.1 *Let $r \geq 0.794$. There is a network $\mathcal{U}_i$ and an adversary $\mathcal{A}$ of rate r, such that the system $\langle \mathcal{U}_i, \mathcal{A}, \mathsf{NTG - U - LIS} \rangle$ is unstable where $\mathcal{U}_i$ is the network a) $\mathcal{U}_2$, b) $\mathcal{U}_3$.*

Sketch of proof. We assume that there is a large enough number of packets s_j in the initial system configuration. Furthermore, we consider that the time is split into phases, each one of which consists of three distinguished time rounds with durations s_j, rs_j and $r^2 s_j$ for a phase j respectively. The proof is based on induction on the number of phases. In *Part a* the *inductive argument* states that if at the beginning of a phase j, there are s_j packets in the queues e_1, e_2 (the same queues in *Part b*) requiring to traverse the edges e_1, f_2 and e_2, f_1, f_2 correspondingly (the edge f_1 in *Part b*), then at the beginning of phase $j + 1$ there will be more than s_j packets in the queues e_1, e_2 (the same queues in *Part*

b) requiring to traverse the edges e_1, f_2 and e_2, f_1, f_2 correspondingly (the edge f_1 in *Part b*). In *Parts a,b* the basic idea behind the adversarial construction is the injection of two types of packet sets during a phase. One packet type is used for the reproduction of the inductive argument. The adversary tries to keep as many of these packets can in the queues e_1, e_2 of the network at the end of each phase. The packets of the other type are injected on judiciously chosen paths to keep as many of the packets of the other type in the system. In *Part a* at the end of round 3, there is a remaining portion of packets $(r^3 s_j)$ in e_1 (e_2 in *Part b*) that require to traverse the edges e_1, f_2 (e_2, f_1 in *Part b*) and $r^3 s_j$ packets in e_2 (e_1 in *Part b*) requiring to traverse the edges e_2, f_1, f_2 (e_1, f_1 in *Part b*). Therefore, the number of packets in e_1, e_2 requiring to traverse the edges e_1, f_2 and e_2, f_1, f_2 (the edge f_1 in *Part b*) is $s_{j+1} = 2r^3 s_j$. For instability it suffices $s_{j+1} > s_j$, i.e. $r \geq 0.794$. This argument can be repeated for an infinite number of phases showing that the number of packets in the system increases forever.   □

7   Conclusions

Note that the technique used to get an instability threshold of 0.5 for certain compositions of universally stable protocols might produce lower bounds, if one finds a network allowing more investing flows to stay in the network. The technique that gets the upper bound on FIFO stability is based on a fundamental FIFO property, namely that in any FIFO network, old packets *exit the network* after some bounded time (by their size and the network structure). We feel that a refinement of such a technique may answer the fundamental open question of whether FIFO is stable below a certain *fixed* rate.

References

1. M. Andrews, B. Awerbuch, A. Fernandez, J. Kleinberg, T. Leighton, and Z. Liu, "Universal Stability Results for Greedy Contention-Resolution Protocols," *Journal of the ACM,* Vol. 48, No. 1, pp. 39–69, January 2001.

2. C. Alvarez, M. Blesa, and M. Serna, "Universal stability of undirected graphs in the adversarial queueing model," *Proceedings of the 14th Annual ACM Symposium on Parallel Algorithms and Architectures,* pp. 183–197, August 2002.

3. A. Borodin, J. Kleinberg, P. Raghavan, M. Sudan and D. Williamson, "Adversarial Queueing Theory," *Journal of the ACM,* Vol. 48, No. 1, pp. 13–38, January 2001.

4. H. Chen and D. D. Yao, *Fundamentals of Queueing Networks,* Springer, 2000.

5. J. Diaz, D. Koukopoulos, S. Nikoletseas, M. Serna, P. Spirakis and D. Thilikos, "Stability and Non-Stability of the FIFO Protocol," *Proceedings of the 13th Annual ACM Symposium on Parallel Algorithms and Architectures,* pp. 48–52, 2001.

6. D. Koukopoulos, M. Mavronicolas, S. Nikoletseas and P. Spirakis, "On the Stability of Compositions of Universally Stable, Greedy, Contention-Resolution Protocols," *Proceedings of the 16th International Symposium on DIStributed Computing,* LNCS 2508, pp. 88–102, October 2002.

7. D. Koukopoulos, S. Nikoletseas, and P. Spirakis, "Stability Issues in Heterogeneous and FIFO Networks under the Adversarial Queueing Model," Invited Keynote Address, *Proceedings of the 8th International Conference on High Performance Computing 2001,* pp. 3–14, December 2001.
8. Z. Lotker, B. Patt-Shamir and A. Rosen, "New Stability Results for Adversarial Queuing," *Proceedings of the 14th Annual ACM Symposium on Parallel Algorithms and Architectures,* pp. 192–199, August 2002.
9. P. Tsaparas, *Stability in Adversarial Queueing Theory,* M.Sc. Thesis, Computer Sc. Dept., Univ. of Toronto, 1997.