A Rate Threshold for Stability in Adversarial Queuing Theory
Zvi Lotker*  Boaz Patt-Shamir†  Adi Rosén‡
September 30, 2001

Abstract

We consider the setting of a communication network and a sequence of packets injected into it over time. Work in recent years concentrated on analyzing this setting when the packets are injected by an adversary, rather than by a set of random processes [6, 2, 4, 12, 10, 11, 1, 3]. Most of this work uses the model of “adversarial queuing theory” introduced by Borodin et al. [6].

In the present work we show that when the packets are injected at a rate low enough compared to the lengths of the paths in the network, then the network remains stable with any greedy protocol. Furthermore, the buffer sizes and the delays of the packets remain linear. These results refine the notion of “universal stability” [6] and partially resolve a number of open questions in the literature concerning the rates at which networks are stable [11].
1 Introduction

Recent years have seen a growing amount of work being concentrated on analyzing packet-switching networks under worst-case scenarios rather than under probabilistic assumption [6, 2, 4, 12, 10, 11, 1, 3]. Most of this work makes use of the model of “adversarial queuing theory” proposed by Borodin et al. [6]. In this setting packets are transmitted between adjacent switches over links, in discrete time steps, a prescribed number of packets over each link in any time step. New packets are injected into the network dynamically at any time. Each packet is injected into a source node and has to reach a destination node over a prescribed path. The packets travel to their destinations in a “store-and-forward” manner, being stored in buffers at intermediate switches. Two major questions arise in this setting: what are the delays incurred by the packets, and what are the sizes of the buffers used. A crucial question is the question of stability, i.e., whether the maximum buffer size grows with time (In other words, is there a finite upper bound, which is independent of time, that bounds the size of the buffers in the network).

The answers to these questions depend on the topology of the network, the injection pattern of the packets, and the contention-resolution protocol (used when more packets than the capacity of an edge attempt to cross this edge at a given time). Work in the framework of adversarial queuing theory showed that under this model for injection of packets, some networks are stable for every greedy protocol as long as the rate of injection is less than 1, while other networks do not exhibit this phenomenon. Those networks which are always stable have been coined “universally stable” networks [6], and have been fully characterized [12]: essentially the set of universally stable networks is the set of networks which are DAGs of simple cycles. From the point of view of protocols, some protocols are known to be universally stable, i.e., they are stable on any network topology (e.g., LIS, FTG), and some are known not to be always stable (e.g., FIFO, NTG) [2]. For the non-universally stable protocols it is an interesting question whether they can become unstable at arbitrary low rates, or whether there is a rate below which they become always stable. The protocols NTG, FFS, and LIFO exhibit the phenomenon of being unstable on certain setting even at arbitrary low rates [6]. As to FIFO this question is at present open.

In this work we study the interplay between the rate of injection and the stability of the system. In particular, we refine the notion of “universally stable” networks. We show that for every network and every set of paths there is a threshold $\rho$ such that if the rate of injection is below the threshold then the system is stable with any greedy protocol. We show that the value of $\rho$ is at least $1/(d+1)$, where $d$ is the length of the longest path followed by any packet.\footnote{It was recently brought to our attention that previously it was known, that the if the rate of injection is}
In a recent paper, Diaz et al. [9] showed a way to calculate for every network a bound $\rho$ such that FIFO is stable on this network as long as the rate of injection does not exceed $\rho$. When applying our results to the protocol FIFO (in which case we can also prove slightly stronger results), we much improve these bounds. Furthermore, both our proof and our way to calculate $\rho$, are much simpler than those in [9].

Since many protocols that are known to be stable can cause buffers to become exponential, an interesting question is whether there exists (deterministic distributed) protocols that maintain polynomial buffers [6, 2]. As a by product, our results show that if the rate of injection is low enough, then all greedy protocols become polynomial (in fact linear).

1.1 The Model

We model a communication network as a graph $G = (V, E)$, $|V| = n$, $|E| = m$. Each node $v \in V$ represents a communication switch, and each edge $e \in E$ represents a link between two switches. The switches store and forward packets; the packets are stored in the switches in buffers. Every switch has a buffer for each outgoing link, and stores in this buffer the packets to be sent on the corresponding link. We model here the network as a directed graph. Thus each edge $e \in E$ is a directed edge, capable of delivering packets in the direction it which it is oriented. The network is synchronous, and there is a global clock known to all switches. We number the time steps by $t \in \mathcal{N} = \{1, 2, 3, \ldots\}$. Each edge can transmit a single packet in each time step (in the direction it is oriented).

New packets are injected into the network at any time step. Each packet $p$ is injected into an arbitrary source node $s_p$, and has to reach an arbitrary destination node $d_p$. To reach $d_p$ from $s_p$, the packet has to follow a prescribed path $\pi_p = (s_p = v_0, v_1, \ldots, v_k = d_p)$.

Each time step, in each node, is divided into two sub-steps: in the first sub-step packets are sent from the buffers in which they are stored (on the edge that corresponds to each buffer). At most one packet is sent per edge. In the second sub-step each nodes receives the packets sent to it over its incoming edges, and the packets injected into it from the outside. Those packets that reach their destination are absorbed. The other packets are placed in the buffers that correspond to the next edge on their path.

The injection of packets into the network is controlled by an adversary. We use the model of Adversarial Queuing Theory introduced by Borodin et al. [6].

**Definition 1:** We say that the adversary injecting packets is an $(w, r)$ adversary, for some $r \leq 1$ and some integer $w > 1$, if the following holds: for any time $t \in \mathcal{N}$, let $\mathcal{I}^t$ be the set of packets injected during the $w$ time steps from $t$ to $t + w - 1$, inclusive. Let $\Pi^t$ be the set bounded by $1/m$, where $m$ is the number of links in the network, then the system remains stable with any greedy protocol [5].
of paths that the packets in $I'$ have to follow: $I' = \{\pi_p : p \in I\}$. Then, every edge $e \in E$ is used by the paths of $I'$ at most $|rw|$ times.

A protocol is a set of $n$ algorithms, one per each switch and run locally in it, that control the sending of the packets from the buffers on the corresponding outgoing links. In particular each algorithm uses only information available in its specific switch and decides if a packet is to be sent over an outgoing link and which packet it will be. A greedy protocol is a protocol that always sends a packet over an outgoing edge $e$ from node $v$, if there is at least one packet in $v$ that has $e$ on its path. We consider here only greedy protocols.

### 1.2 Our Results

Given any network, we denoted by $d$ the length of the longest path followed by any packet. We show that as long as the rate of injection is at most $1/(d + 1)$, then the system remains stable with any greedy protocol. In addition we show that not only the system is stable, but also that all greedy protocols become polynomial, and in fact linear. Thus our results provide insight also into the behavior of universally stable protocols, showing that they become linear when the rate is low enough, on any network. We show that any packet stays in any one buffer at most $|wr|$ time units, if the sequence of packets is injected by a $(w, r)$ adversary. This proves on the one hand that all buffer sizes are bounded from above by $|wr|$ at any time, and on the other hand that any packet is delivered within $\ell \cdot |wr|$ time steps, where $\ell$ is the length of the path that it has to follow. An open question raised in [2] is the question whether any of FTG, NTS or SIS (which are all universally stable protocol which can have exponential buffers) become polynomial below some rate. Our results answer this question in the affirmative, albeit only giving a threshold that is a function of the network and the set of paths.

Results in [6] show that the protocol FTG (and in fact also LIFO and NTS) are unstable for arbitrary low rates. The proofs there use a network and a set of paths such that to show that FTG is unstable for rate $r$, packets with paths of length $16/r$ are used. In view of these results, our bounds on the value $\rho$, in terms of $d$, are optimal up to a small constant factor.

For specific greedy protocols (such that FIFO and LIS) we can slightly improve our results and show that the network becomes stable already when the rate is at most $1/d$ rather than $1/(d + 1)$.

We note that given a $(w, r)$ adversary, buffers of size $|wr|$ are necessary, as the adversary can inject at the tail of an edge $e$, in one time step, $|wr|$ packets that require edge $e$. Thus our bound on the size of any buffer is optimal.

Gamarnik [11] posed the question of which networks are always stable for injection rates under a given value $r$. Our results give a partial answer to this question, by showing for
every $r$ a set of networks that exhibit this phenomenon. The question of fully characterizing the set of such networks for every $r$ remains however open.

## 2 Upper Bounds on Buffer Sizes

In this section we prove that any network is stable with any greedy protocol in the face of a sequence of packets injected by an adversary with parameters $(w, r)$, for rate $r \leq 1/(d + 1)$. The parameter $d$ denotes here the length (in edges) of the longest path followed by any packet in the sequence. In particular we prove below that any packets stays in any one queue no more that $\lfloor wr \rfloor$ time steps.

**Lemma 1:** For any network, if the sequence of packets is injected by an $(w, r)$ adversary, with $r \leq 1/(d + 1)$, and the schedule is a greedy schedule, then no packet stays in the same buffer more than $\lfloor wr \rfloor$ time steps (i.e., if a packet arrives to a certain buffer in time step $t$, it will leave this buffer by the end of time step $t + \lfloor wr \rfloor$).

**Proof:** We prove that any packet that arrives to any buffer at time step $t$, leaves this buffer by time step $t + \lfloor wr \rfloor$. The proof is by induction on $t$.

We prove the base of the induction for any $t \leq dwr + 1$. Let $p$ be a packet that arrives to the buffer at the tail of edge $e$ at time $t \leq dwr + 1$. Assume towards a contradiction that $p$ is at the same buffer at the end of time step $t + \lfloor wr \rfloor$. This means that for each of the $\lfloor wr \rfloor$ time steps in $[t + 1, t + \lfloor wr \rfloor]$ some other packet was sent over edge $e$ (since we consider a greedy protocol). I.e., we identify $\lfloor wr \rfloor + 1$ packets that require edge $e$ and are injected into the system by the end of time step $t + \lfloor wr \rfloor - 1$ (these are the packet $p$ itself, and the $\lfloor wr \rfloor$ packets that were sent over $e$). Since $t \leq dwr + 1$, we have $t + \lfloor wr \rfloor - 1 \leq (d + 1)wr$. By the definition of the adversary the number of packets that require $e$ and are injected by the end of any time step $t' \leq (d + 1)wr$ is at most $\lfloor (d + 1)r \rfloor \lfloor wr \rfloor$. Since we assume $r \leq 1/(d + 1)$ this is at most $\lfloor wr \rfloor$. A contradiction to the fact that we identified $\lfloor wr \rfloor + 1$ packets.

We now prove the claim for any $t > dwr + 1$. This is done based on the induction hypothesis that for any packet that arrives at some buffer at time $t' < t$, this packet leaves the buffer by time step $t' + \lfloor wr \rfloor$.

Let $p$ be a packet that arrives to the buffer at the tail of edge $e$ at some time step $t$. Consider any packet that requires edge $e$ and was injected by time step $t - d\lfloor wr \rfloor$. Using the induction hypothesis we know that such packet left the buffer into which it was injected by time step $t - d\lfloor wr \rfloor + \lfloor wr \rfloor$ and then it left the next buffer by time step $t - d\lfloor wr \rfloor + 2\lfloor wr \rfloor$, etc. I.e., it arrived to its destination by time step $t - d\lfloor wr \rfloor + d\lfloor wr \rfloor = t$ (since the length of its path is at most $d$, and all its “arrival times” are earlier than $t$, so the induction hypothesis holds). Now assume towards a contradiction that packet $p$ is still at the tail of edge $e$ at
the end of time step \( t + \lfloor wr \rfloor \). That is, there are \( \lfloor wr \rfloor \) other packets that crossed edge \( e \) in \([t + 1, t + \lfloor wr \rfloor]\). As before this identifies \( \lfloor wr \rfloor + 1 \) distinct packets that require edge \( e \), are present in the network at the end of time step \( t \) or later, and are injected by time step \( t + \lfloor wr \rfloor - 1 \). However we know that any packet injected by time step \( t - d\lfloor wr \rfloor \) already left the network by the end of time step \( t \). Therefore those \( \lfloor wr \rfloor + 1 \) packets must have been injected in \([t - d\lfloor wr \rfloor + 1, t + \lfloor wr \rfloor - 1]\). There are \( \lfloor wr \rfloor (d + 1) - 1 \) time steps in this interval, therefore the number of packets that require \( e \) and can be injected during this interval in bounded by \( \lfloor (d + 1)r \rfloor \lfloor wr \rfloor \). Since \( r \leq 1/(d + 1) \) this is at most \( \lfloor wr \rfloor \), a contradiction. \( \square \)

For protocols where a packet arriving at a certain buffer at time \( t \) has priority over any packet injected after time \( t \), we can relax the condition that \( r \leq 1/(d + 1) \) to be \( r \leq 1/d \). Note that among such protocols are the protocols FIFO and LIS.

We define the following notion.

**Definition 2:** A time priority protocol is a greedy protocol under which a packet arriving at a buffer at time \( t \), has priority over any other packet that is injected after time \( t \).

**Lemma 2:** For any network, if the sequence of packets is injected by an \((w, r)\) adversary, with \( r \leq 1/d \), and the protocol is a time priority protocol, then no packet stays in the same buffer more than \( \lfloor wr \rfloor \) time steps (i.e., if a packets arrives to a certain buffer in time step \( t \), it will leave this buffer by the end of time step \( t + \lfloor wr \rfloor \)).

We repeat the proof of the previous lemma with one change applied at two places: in the present case, when assuming towards a contraction that packet \( p \) is still in the same buffer at the end of time step \( t + \lfloor wr \rfloor \), and identifying the packets that cause this delay, we know that those packets must have been injected no later than time step \( t \) (rather than time \( t + \lfloor wr \rfloor - 1 \)). This is because packets injected after time step \( t \) will not delay packet \( p \) if the protocol is a time priority protocol. This allows us to prove the lemma with the relaxed condition that \( r \leq 1/d \).

For completeness we give below the full proof.

**Proof:** We prove that any packet that arrives to any buffer at time step \( t \), leaves this buffer by time step \( t + \lfloor wr \rfloor \). The proof is by induction on \( t \).

We prove the base of the induction for any \( t \leq dwr + 1 \). Let \( p \) be a packet that arrives to the buffer at the tail of edge \( e \) at time \( t \leq dwr \). Assume towards a contradiction that \( p \) is at the same buffer at the end of time step \( t + \lfloor wr \rfloor \). This means that during the \( \lfloor wr \rfloor \) time steps in \([t + 1, t + \lfloor wr \rfloor]\) some other packet was sent over edge \( e \) (since we consider a greedy protocol). I.e., we identify \( \lfloor wr \rfloor + 1 \) packets that require edge \( e \) (These packets are the packet \( p \) itself, and the \( \lfloor wr \rfloor \) packets that were sent over \( e \)). These packets must have been injected into the system by the end of time step \( t \); any packet injected after \( t \) will not
delay $p$ according to a time priority protocol. Now, By the definition of the adversary the number of packets that require $e$ and are injected by the end of any time step $t \equiv dwr$ is at most $[dr][wr]$. Since we assume $r \leq 1/d$ this is at most $[wr]$. A contradiction to the fact that we identified $[wr] + 1$ packets.

We now prove the claim for any $t > dwr$. This is done based on the induction hypothesis that for any packet that arrives at some buffer at time $t' < t$, this packet leaves this buffer by time step $t' + [wr]$.

Let $p$ be a packet that arrives at the buffer at the tail of edge $e$ at some time step $t$. Consider any packet that requires edge $e$ and was injected by time step $t - d[wr]$. Using the induction hypothesis we know that such packet left the buffer into which it was injected by time step $t - d[wr] + [wr]$, left the next buffer by time step $t - d[wr] + 2[wr]$, etc. I.e., it arrived to its destination by time step $t - d[wr] + d[wr] = t$ (since the length of its path is at most $d$, and all its “arrival times” are earlier than $t$, so the induction hypothesis holds). Now assume towards a contradiction that packet $p$ is still at the tail of edge $e$ at the end of time step $t + [wr]$. That is, there are $[wr]$ other packets that crossed edge $e$ in $[t + 1, t + [wr]]$. As before this identifies $[wr] + 1$ distinct packets that require edge $e$, and are injected by the end of time step $t$; any packet injected after $t$ will not delay $p$ since the protocol is a time priority protocol. However we know that any packet injected by time step $t - d[wr]$ already left the network by the end of time step $t$. Therefore those $[wr] + 1$ packets must have been injected in $[t - d[wr] + 1, t]$. There are $[wr]$ $d$ time steps in this interval, therefore the number of packets that require $e$ and can be injected during this interval in bounded by $[dr][wr]$. Since $r \leq 1/d$ this is at most $[wr]$, a contradiction.

We note that similar results can be proved for the case where the adversary is allowed to initiate the system with an arbitrary set of packets in the buffers. Let $S > 0$ be the maximum, over the edges $e \in E$, of the number of packets requiring $e$, present in the network when it starts to operate. Then the system remains stable as long as the rate of injection $r$ is less than (rather than at most) $1/(d+1)$ (or $1/d$ for time priority networks). The bound on the delay of any packet in any specific buffer becomes however $(S + w) \cdot r \cdot (\frac{1}{r} + \frac{1}{\varepsilon})$, for every $0 < \varepsilon < 1/(d+1) - r$ (or $0 < \varepsilon < 1/d - r$ for time priority protocols).

References


