A characterization of universal stability for directed graphs in the adversarial queueing model

Carme Àlvarez
Maria Blesa
Maria Serna

Report LSI–02–4–R

A characterization of universal stability for directed graphs in the adversarial queueing model *

Carme Àlvarez Maria Blesa Maria Serna

January 24, 2002

Departament de Llenguatges i Sistemes
Universitat Politècnica Catalunya
Mòdul C6, Campus Nord
Jordi Girona Salgado 1–3
08034-Barcelona
{alvarez,mjblesa,mjserna}@lsi.upc.es

Abstract

In this paper we provide a characterization of universal stability for digraphs in the general case that the packets are not restricted to follow simple paths. Furthermore, we consider a variation of the Nearest To Go protocol and we show that stability under this queueing policy is equivalent to universal stability.

Goel [9, 10] also studied the universal stability of digraphs when only packets following simple paths are allowed, and gave a characterization in terms of three forbidden minors. We show that a graph that does not contain any of these forbidden minors is unstable, disproving this result.

1 Introduction

The stability problem has been investigated under various models of packet routing, see for example [6, 7, 4, 8, 1]. The adversarial queueing model of

*Partially supported by the IST Programme of the EU under contracts number IST-1999-14186 (ALCOM-FT) and IST-2001-33116 (FLAGS) and by the Spanish CICYT projects TIC-1999-0754-C03 (Mallba) and TIC-2000-1970-CE. The second author was also supported by pre-doctoral grant 2001FI-00659

Borodin et al. [3], was developed as a robust model of queueing theory in network traffic, and replaces stochastic by worst case inputs. Adversarial Queueing Theory considers the time evolution of a packet-routing network as a game between an adversary and a protocol. The adversary, at each time step, may inject a set of packets at some nodes. For each packet, the adversary specifies a network path that the packet must traverse. When the packet arrives to its final destination, it is absorbed by the system. If more than one packet wish to cross an edge e in the current time step, then a protocol resolves the conflict, maintaining a queue per edge. As usual, we will only consider *greedy* protocols, those that advance a packet across an edge whenever there is at least one packet waiting to use it.

A crucial parameter of the adversary is its *injection rate*. The rate of an adversary in this model, is specified by a pair (r, b) where $b \geq 1$ is a natural number and $0 < r < 1$. The adversary must obey the following rule:

> Of the packets that the adversary injects in any interval I, at most $\lceil r|I| \rceil + b$ can have paths that contain any particular edge.

The motivation for study the behavior of packet communication networks is to determine the conditions of *stability*, the fact that the number of packets in the system remains bounded, as the system dynamically evolves in time. Andrews et al. [2] showed the existence of graphs and protocols that are not universally stable. In the positive side they proved the universal stability of a directed cycle with any number of vertices.

For the case of undirected graphs, the existence of a finite set of basic undirected graphs such that a graph G is stable for every r if and only if none of these graphs is a minor of G was proved in [2]. Gamarnik [8] provides a characterization of stability in the case of undirected graphs where an edge can be crossed only in one direction in one timestep.

In this paper we consider directed graphs and provide a characterization of universal stability. We also consider the protocol NTG-LIS (see definitions later) and we show that NTG-LIS stability is equivalent to universal stability.

Goel [9, 10] also studied the universal stability of digraphs when only packets following simple paths are allowed, and claimed

> Theorem 2.8 in [10]: A digraph G is universally stable iff it does not contain any of the three forbidden minors in Figure 1.

In this paper we show that a minor of U_1', the graph U_1 in Figure 2, is unstable for the NTG-LIS protocol, thus disproving this result.

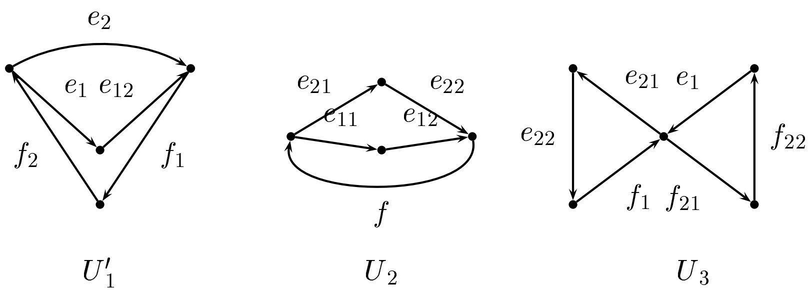

Figure 1: Goel's forbidden minors for simple-path universal stability

2 Preliminaries

All the graphs in this paper are directed, may have multiple edges but no loops. We use standard graph theory terminology. Recall that a *path* is a sequence of arcs, so that the tail of an arc is the head of the next arc. A *simple path* is a path that does not contain repeated vertices. P_k denotes a simple directed path with k edges.

Given a digraph G, let $\mathcal{E}(G)$ denote the family of digraphs which contains G and any graph obtained by replacing at least one edge (a, b) of G for a simple path from a to b defined on a new set of vertices. The added paths are pairwise vertex-disjoint (except for a and b).

Some natural greedy protocols are the following:

FIFO First In First Out - every queue gives priority to the packet that arrives first to the queue.

LIS Longest In System - every queue gives priority to the packet that has been in the system the longest.

NTG Nearest To Go - every queue gives priority to the packet that has to cross the smallest number of edges.

NTG-LIS Works as NTG, but solving ties using the LIS protocol.

Definition 1. *A digraph G is* stable *for protocol P (the pair (G, P) is stable) if for any adversary the maximum number of packets in the system is bounded. When each packet is restricted to be a simple path we say that G (the pair (G, P)) is* simple-path stable.

It is straightforward to prove the following property.

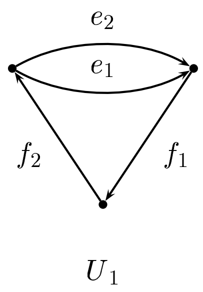

Figure 2: A non simple-path stable digraph

Lemma 1. *Let H be a subgraph of a digraph G. If the pair (H, P) is not (simple-path) stable then the pair (G, P) is not (simple-path) stable.*

Definition 2. *A digraph G is universally stable if G is stable for all greedy protocols P. When each packet is restricted to be a simple path we will say that G is* simple-path universally stable*.*

Observe that any universally stable digraph is also simple-path universally stable, but the opposite is not true.

Following, we review some basic properties that we will use later.

Lemma 2 ([10]). *If digraphs G_1 and G_2 are universally stable, then so is any graph G formed by joining them with edges that go only from G_1 to G_2.*

Corollary 1 ([10]). *All acyclic digraphs are universally stable.*

Corollary 2 ([10]). *A digraph is universally stable if and only if all of its strongly connected components are universally stable.*

Lemma 3 ([2]). *The directed cycle on any number of vertices is universally stable.*

3 Non simple-path stable digraphs

In this section we show that the characterization of the universal stability property when packets are restricted to form a simple path given in [10] is incorrect. We show that a family of subgraphs that do not contain any of Goel's forbidden minors are unstable for NTG-LIS.

Lemma 4. *The pair $(U_1, \text{NTG-LIS})$ is not simple-path stable.*

Proof. The adversary operates in five rounds. Initially, there are $\frac{s}{2}$ packets of the form $(e_1\ f_1)$ and $\frac{s}{2}$ of the form $(e_2\ f_1)$.

Round 1: for s steps, the adversary injects rs packets of the form $(f_1\ f_2)$. These injections get mixed with the initial packets at edge f_1, and are blocked there because the queueing protocol is NTG.

Round 2: for the next rs steps, the adversary injects a set of r^2s packets of the form $(f_2\ e_2)$ that are blocked at f_2 by the packets from the first round because, at that point, they are at longer distance to their destination.

Round 3: for the next r^2s steps, the adversary injects r^3s packets of the form (e_2) and r^3s of the form $(f_2\ e_1)$. All these injections are blocked by the remaining packets from the previous round because the secondary protocol is LIS.

Round 4: for the next r^3s steps, the adversary injects r^4s packets of the form (e_1) and r^4s packets of the form $(e_2\ f_1)$. A total number of r^4s simple packets will remain at e_1. The injections of the form $(e_2\ f_1)$ are blocked at e_2 by packets from the previous round because distance to their destination is longer.

Round 5: for the next r^4s steps, the adversary injects r^5s packets of the form $(e_1\ f_1)$ and r^5s packets of the form (e_2). The simple packets queued at e_1 from the previous round will block the injections introduced at this edge. The simple injections at e_2 will block the packets of the form $(e_2\ f_1)$ from the previous round.

At the end of the fifth round, there are r^5s packets of the form $(e_2\ f_1)$ and r^5s packets of the form $(e_1\ f_1)$. The adversary described above only uses packets describing simple paths and makes network U_1 unstable when $2r^5s > s$, i.e. at injection rate $r > 0.87055$. □

Note that U_1 is a minor of U_1', but U_1 does not contain any of the Goel's forbidden minors.

In all the figures dashed arcs represent a path and the associated label its length, and a continuous line represents an arc and its label the arc's name.

Lemma 5. *Any digraph in $\mathcal{E}(U_1)$ is not simple-path stable for* NTG-LIS.

Proof. Let G be the graph in $\mathcal{E}(U_1)$ described at Figure 3. There are two vertices a and b with two paths p_n and p_m from a to b and a path p_d from b to a, with $d \geq 2$. We assume that $n \geq m$. We call e and f the first and

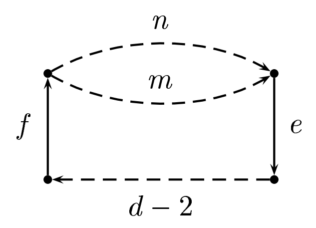

Figure 3: A graph in $\mathcal{E}\,(U_1)$

last edges in p_d, respectively. By including a whole path (e.g. p_d) in the description of the form of a packet, we mean that the path to be followed by the packet must include all this path.

The adversary operates in five rounds. Initially, there are $\frac{s}{2}$ packets of the form $(p_n\ e)$ and $\frac{s}{2}$ of the form $(p_m\ e)$.

Round 1: for s steps, the adversary injects rs packets of the form (p_d). These injections get mixed with the initial packets at edge e. Except the first rm injections, the rest of them are blocked there because the queueing protocol is NTG (m is the time in which a first packet arrives to e). We consider s big enough to guarantee that a continous flow arrives to edge e after the arrival of the first packet.

Round 2: for the next $rs - rm$ steps, the adversary injects a set of $r^2s - r^2m$ packets of the form $(f\ p_m)$. These packets will be blocked at f by the packets from the first round because they will be at longer distance to their destination, but notice that rd of them will no be blocked (d is the time needed for the first packet in e to arrive to f).

Round 3: for the next $r^2s - r^2m - rd$ steps, the adversary injects $r^3s - r^3m - r^2d$ packets of the form (p_m) and $r^3s - r^3m - r^2d$ packets of the form $(f\ p_n)$. The first injeccions are blocked because secondary protocol is LIS. The later injeccions are blocked because $n \geq m$ (or also because of the LIS protocol if $n = m$).

Round 4: for the next $r^3s - r^3m - r^2d$ steps, the adversary injects $r^4s - r^4m - r^3d$ packets of the form (p_n), which are blocked due to the LIS effect. Also $r^4s - r^4m - r^3d$ packets of the form $(p_m\ e)$ are injected, which are blocked at the first edge of p_m by packets from the previous round because distance to their destination is longer.

Round 5: for the next $r^4s - r^4m - r^3d$ steps, the adversary injects $r^5s - r^5m - r^4d$ packets of the form $(p_n\ e)$, that will be all blocked by the packets

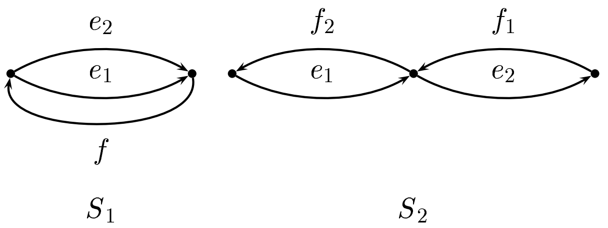

Figure 4: Some no universally stable digraphs

of the form (p_n) from the previous round. The adversary also introduces $r^5s - r^5m - r^4d$ injeccions of the form (p_m), that will block the packets of the form $(p_m\ e)$ queued since the previous round.

At the end of the fifth round, there are $r^5s - r^5m - r^4d$ packets of the form $(p_n\ e)$ and $r^5s - r^5m - r^4d$ packets of the form $(p_m\ e)$. The adversary described above only uses packets describing simple paths and makes the graph G described above unstable when $2(r^5s - r^5m - r^4d) > s$.

Note that $C = 2(m+d) > 2(r^5m+r^4d)$ is a constant quantity and then, for a big enough s, an injection rate r can be found such that $2r^5s - C > s$ holds.

4 Equivalence between NTG-LIS and universal stability

In this section we give a characterization of universal stability, in the general case, furthermore we show that stability for the NTG-LIS protocol is equivalent to universal stability.

Lemma 6. *The pair $(S_1, \text{NTG-LIS})$ is not stable.*

Proof. At the beginning there are s packets that want to traverse edge f, half of them are of the form $(e_1\ f)$ and half of them are of the form $(e_2\ f)$. The adversary $\mathcal{A}$ will play injections in four rounds:

Round 1: for s steps, the adversary injects rs packets of the form $(f\ e_2)$. These injections get mixed with the initial packets at edge f, and are blocked there because the queueing protocol is NTG.

Round 2: for the next rs steps, the adversary injects a set of r^2s packets of the form $(f\ e_1)$ and r^2s packets of the form (e_2). Injections $(f\ e_1)$ are

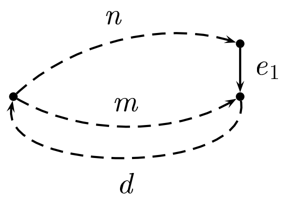

Figure 5: A graph in $\mathcal{E}(S_1)$

blocked by the packets $(f\ e_2)$ from the first round because they are at the same distance to their destination and the secondary protocol that we apply is LIS. The r^2s injections of the form (e_2) are also blocked.

Round 3: for the next r^2s steps, the adversary injects r^3s packets of the form (e_2) and r^3s of the form $(e_1\ f)$. The simple injections on e_2 will be blocked by the packets at e_2 from the previous round. The $(e_1\ f)$ injections get mixed with the packets of the form $(f\ e_1)$ from the previous round at edge f where the injections get blocked because their distance to destination is longer.

Round 4: for the next r^3s steps, the adversary injects r^4s packets of the form $(e_2\ f)$ and r^4s packets of the form (e_1). The simple injections at e_1 block the packets $(e_1\ f)$ from the previous round because their distance to destination is shorter. The $(e_2\ f)$ injections are blocked by the (e_2) packets from the previous round.

At the end there are $2r^4s$ packets queued at e_1 and e_2 that want to traverse edge f, r^4s are of the form $(e_1\ f)$ and r^4s are of the form $(e_2\ f)$. If cycles are allowed, the adversary $\mathcal{A}$ described above makes the network U_1 unstable when $2r^4s > s$, i.e. at injection rate $r > 0.84089$. □

Lemma 7. *Any digraph in $\mathcal{E}(S_1)$ is not stable for* NTG-LIS.

Proof. Let G be the graph in $\mathcal{E}(\mathcal{S}_1)$ described at Figure 5. This graph is formed from $\mathcal{S}_1$ by extending the edges by paths. Let us denote by p_n, p_m and p_d the path replacing the edges. Let us assume that $n \geq m - 1$.

The adversary operates in four rounds. Initially, there are s packets wanting to traverse p_d and they are distributed in the following way: x packets of the form $(p_n\ e_1\ p_d)$ and y packets of the form $(p_m\ p_d)$.

Round 1: for s steps, the adversary injects rs packets of the form $(p_d\ p_m)$. These injections are blocked by the initial packets because the queueing protocol is NTG, but rm of them are lost. We consider a big enough s to guarantee that a continous flow arrives to the first edge of p_d after the arrival of the first packet.

Round 2: for the next $rs - rm$ steps, the adversary injects a set of $r^2s - r^2m$ packets of the form $(p_d\ p_n\ e_1)$ and $r^2s - r^2m$ packets of the form (p_m). These injections are blocked by the remaining packets from the previous round because the queueing protocol is NTG (or because of the LIS protocol if $n = m - 1$). Note that rd injections of the form (p_m) will be lost.

Round 3: for the next $r^2s - r^2m$ steps, the adversary injects $r^3s - r^3m$ packets of the form (p_m) that will be all blocked by packets from the previous round. The adversary also injects $r^3s - r^3m$ packets of the form $(p_n\ e_1\ p_d)$ that will collapse with packets from the previous round. These injections will be blocked there because their distance to destination is longer at that point, but rd of them will be lost. At the end of this round there are $r^3s - r^3m - r^2d$ packets of the form (p_m) and $r^3s - r^3m - rd$ packets of the form $(p_n\ e_1\ p_d)$.

Round 4: for the next $r^3s - r^3m - r^2d$ steps, the adversary injects $r^4s - r^4m - r^3d$ packets of the form $(p_n\ e_1)$ and $r^4s - r^4m - r^3d$ packets of the form $(p_m\ p_d)$. The injections on p_n will maintain blocked some packets of the previous round and, the injections on p_m will be blocked by the remaining packets from the previous round.

At the end of the forth round, there are $r^4s - r^4m + r^3d - r^2d$ packets of the form $(p_n\ e_1\ p_d)$ and $r^4s - r^4m - r^3d$ packets of the form $(p_m\ p_d)$. The adversary described above uses packets describing cycles and makes the graph G described above unstable when $2r^4s - 2r^4m - r^2d > s$. Note that $C = 2r^4m + r^2d > 2m + d$ and for an big enough s, an injection rate r can be found such that $2r^4s - C > s$ holds. □

Lemma 8. *The pair* $(S_2, \text{NTG-LIS})$ *is not stable.*

Proof. At the beginning there are s packets requiring to traverse edge f_2. Half of them are of the form $(e_1\ f_2)$ and half of the form $(e_2\ f_1\ f_2)$. Then the adversary will play injections in four rounds.

Round 1: for s steps, the adversary injects rs packets of the form $(f_2\ e_1)$. These injections get mixed with the initial packets at edge f_2, and are blocked there because the queueing protocol is NTG.

Round 2: for the next rs steps, the adversary injects a set of r^2s packets of the form $(f_2\ e_1\ e_2)$ that are blocked by the remaining packets at f_2 from the previous round because the protocol is NTG.

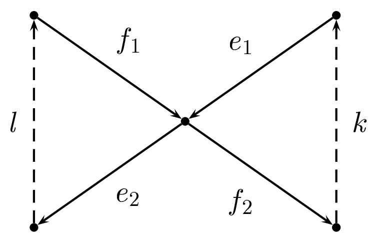

Figure 6: A graph in $\mathcal{E}(S_2)$

Round 3: for the next r^2s steps, the adversary injects r^3s packets of the form (e_2) and r^3s of the form $(e_1\ f_2)$. A total number of r^3s packets of the from (e_2) will remain at e_2. Injections of the form $(e_1\ f_2)$ will meet packets from the previous round at edge e_1 and, because the secondary protocol is LIS, they will be all blocked at this edge.

Round 4: for the next r^3s steps, the adversary injects r^4s packets of the form $(e_2\ f_1\ f_2)$ and r^4s packets of the form (e_1). The simple injections at e_1 block the packets $(e_1\ f_2)$ from the previous round because their distance to destination is shorter. The $(e_2\ f_1\ f_2)$ injections are blocked by the (e_2) packets from the previous round.

At the end of the forth round there are r^4s packets of the form $(e_2\ f_1\ f_2)$ and r^4s packets of the form $(e_1\ f_2)$. If cycles are allowed, the adversary described above makes the network U_2 unstable when $2r^4s > s$, i.e. at injection rate $r > 0.84089$. □

Lemma 9. *Any digraph in $\mathcal{E}(S_2)$ is not stable for* NTG-LIS.

Proof. Let G be a graph in $\mathcal{E}(S_2)$ described in Figure 6. This graph is formed from $\mathcal{S}_2$ by extending its cycles with paths. We call p_k and p_l the additional edges in the two sides, note that k, l or both may be zero.

The adversary operates in four rounds. Initially, there are $\frac{s}{2}$ packets of the form $(e_1\ f_2)$ and $\frac{s}{2}$ packets of the form $(e_2\ p_l\ f_1\ f_2)$.

Round 1: for s steps, the adversary injects rs packets of the form $(f_2\ p_k\ e_1)$. These injections are blocked by the initial packets because the queueing protocol is NTG. Note that rl of such new packets are lost.

Round 2: for the next $rs - rl$ steps, the adversary injects a set of $r^2s - r^2l$ packets of the form $(f_2\ p_k\ e_1\ e_2)$. These injections are blocked by the remaining packets from the previous round because the queueing protocol is NTG.

Round 3: for the next $r^2s - r^2l$ steps, the adversary injects $r^3s - r^3l$ packets of the form (e_2) and $r^3s - r^3l$ packets of the form $(e_1\ f_2)$. These injections will be blocked by packets from the previous round when collapsing at edges e_2 and e_1 respectively, but rk of them will be lost.

Round 4: for the next $r^3s - r^3l - rk$ steps, the adversary injects $r^4s - r^4l - r^2k$ packets of the form (e_1) that will block packets from the previous round at e_1 because they are closer to their destination. The adversary also injects $r^4s - r^4l - r^2k$ packets of the form $(e_2\ p_l\ f_1\ f_2)$ that will be blocked by the (e_2) remaining packets from the previous round.

At the end of the forth round, there are $r^4s - r^4l - r^2k$ packets of the form $(e_2\ p_l\ f_1\ f_2)$ and $r^4s - r^4l - r^2k$ packets of the form $(e_1\ f_2)$.

The adversary described above uses packets describing cycles and makes the graph G described above unstable when $2(r^4s - r^4l - r^2k) > s$. Note that $C = 2(l + k) > 2(r^4l + r^2k)$, and for a big enough s, an injection rate r can be found such that $2r^4s - C > s$ holds.

□

Observe that all the subdigraphs of S_1 and S_2 are trivially universally stable we obtain the following characterization.

Theorem 1. *A digraph is universally stable if and only if it does not contain as subgraphs any of the digraphs in $\mathcal{E}(S_1) \cup \mathcal{E}(S_2)$.*

Proof. Observe that all the strongly connected components of a digraph G must either be a simple cycle or consists of at least two simple cycles which either share an edge or a vertex.

If all the strongly connected componnents of G are simple cycles, G does not have any subgraph in $\mathcal{E}(S_1) \cup \mathcal{E}(S_2)$. Furthermore, by Lemma 3 and Corollary 2, G is universally stable.

Otherwise, G contains a subgraph in $\mathcal{E}(S_1) \cup \mathcal{E}(S_2)$, and by lemmas 7 and 9, G is not universally stable. □

As all the digraphs in $\mathcal{E}(S_1) \cup \mathcal{E}(S_2)$ are unstable for the NTG-LIS protocol, we obtain our last result.

Corollary 3. *A digraph is universally stable if and only if it is stable for* NTG-LIS.

According to the proof of Theorem 1 the following result is immediate

Theorem 2. *A digraph G is universally stable if and only if all its strongly connected components are simple cycles.*

Therefore as the strongly connected components of a digraph can be computed in time linear in the number of arcs and vertices [5]. Therefore, testing for universally stability (or NTG-LIS stability) can be done in polynomial time.

5 Conclusions

We have shown that testing universal stability is equivalent to testing NTG-LIS stability.

As many efforts have been devoted to the study of the FIFO protocol we point out that, to the best of our knowledge, it is not known whether universal stability is equivalent to FIFO stability. In particular, it is not known whether the pair (U_1,FIFO) is stable.

Moreover, we would like to remark that we have to distinguish between simple-path stability and stability (in the general case). Note that, the pair (U_1,NTG-LIS) is not stable while it is easy to see that U_1 is simple-path universally stable.

References

[1] M. Andrews. Inestability of fifo in session-oriented networks. In *11th. ACM-SIAM Symposium on Discrete Algorithms*, pages 440–447, 2000.

[2] 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*, 48(1):39–69, January 2001.

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

[4] M. Bramson. Inestability of fifo queuing networks. *Annals of Applied Probability*, 4:414–4311, 1994.

[5] T. Cormen, C. Leisersoon, and R. Rivest. *Introduction to algorithms.* Second edition. The MIT Press and McGraw-Hill, 2001.

[6] R. Cruz. A calculus for network delay. Part I: Network elements in isolation. *IEEE Transactions on Information Theory*, 37:114–131, 1991.

[7] R. Cruz. A calculus for network delay. Part II: Network analysis. *IEEE Transactions on Information Theory*, 37:132–141, 1991.

[8] D. Gamarnik. Stability of adversarial queues via fluid models. In *39th. IEEE Symposium on Foundations of Computer Science*, pages 60–70, 1998.

[9] A. Goel. Stability of networks and protocols in the adversarial queueing model for packet routing. In *ACM-SIAM Symposium on Discrete Algorithms*, pages 911–912, 1999.

[10] A. Goel. Stability of networks and protocols in the adversarial queueing model for packet routing. *Networks*, 37(4):219–224, 2001.