Adversarial Queuing Theory

Matthew ANDREWS, Baruch AWERBUCH, Antonio FERNANDEZ, Tom LEIGHTON, Zhiyong LIU, Jon KLEINBERG:

Universal-Stability Results and Performance Bounds for Greedy Contention-Resolution Protocols, *Journal of the ACM, 2001*

Allan BORODIN, Jon KLEINBERG, Prabhakar RAGHAVAN, Mahdu SUDAN, David P. WILLIAMSON

Adversarial Queuing Theory, *Journal of the ACM, 2001*

Adversarial Queuing Theory

Studies: Stability and bounds on delay.

Questions:

Is *any* greedy protocol stable against every adversary of rate less than 1, for every network?

Is *any* greedy protocol stable with small queue size against every adversary of rate less than 1, for every network?

Does the *n*-node unidirectional ring have the property that every greedy protocol is stable against every adversary of rate less than 1?

Does *every* network have the property that every greedy protocol is stable against every adversary of rate less than 1?

Bounded Adversaries

Definition: A bounded Adversary is defined by a pair of (b,r). The requirement on the adversary is the following: of the packets that the adversary injects in any interval *I*, at most r|I|+ b can have paths that contain any one edge.

- Inspired by leaky bucket traffic shaping.
- Models packet switching.
- Meaningful for r≤1

Stability definitions

We say that a *protocol* $\mathcal{P}$ is *stable* on a network *G* against an adversary $\mathcal{A}$ if there is a constant *C* (which may depend on *G* and $\mathcal{A}$) such that, starting from an empty configuration, the number of packets in the system at all times is bounded by *C*.

We say that a *graph G* is *universally stable* if every greedy protocol is stable against every adversary of rate less than 1 on *G*.

We say that *protocol* $\mathcal{P}$ is *universally stable* if it is stable against every adversary of rate less than 1, on every network.

SIS is ∀ Stable

Lemma : Let p be a packet waiting in the queue of edge *e* at time *t* and suppose there are currently *k* - 1 other packets in the system requiring e that have priority over p. Then *p* will cross *e* within the next $(k + b)/\varepsilon$ steps.

SIS is ∀ Stable

Proof :Assume *p* does not cross *e* in the next $(k+b)/\varepsilon$ steps. Then, a distinct packet crosses *e* in each of the $(k - b)/\varepsilon$ steps. But any packet in the system during this time that has priority over *p*, and requires edge *e*, must either be one of the $k - 1$ packets existing at time *t*, or one of the (at most) $(1 - \varepsilon)(k + b)/\varepsilon + b$ packets requiring *e* that were injected during this time. Thus, at most $k - 1 + (1- \varepsilon)(k+b)/\varepsilon + b < (k+b)/\varepsilon$ packets have priority over *p* during this time, a contradiction.

SIS is ∀ Stable

We now define the numbers $k_1, k_2, k_3, \ldots$ by recurrence $k_1=b$, $k_{j+1}=(k_j+b)/\varepsilon$

Lemma: *When a packet p arrives at the queue of the jth edge* e_j *on its path there are at most* k_j - 1 *packets requiring any edge e in the path of p with priority over p*.

SIS is ∀ Stable

Proof: By induction. It holds for $j = 1$, for any edge *e*, the only packets requiring *e* that initially could have priority over *p* are the (at most) b - 1 packets injected in the same time step as *p*.

Suppose claim holds for j. *p* will arrive at the tail of e_{j+1}

In at most $(k_j+b)/\varepsilon$ steps during which time at most another $(1-\varepsilon)(k_j+b)/\varepsilon + b$ packets requiring nay edge e arrive that are younger than p. Thus at most

$$k_j - 1 + \frac{(1-\varepsilon)(k_j + b)}{\varepsilon} + b = k_{j+1} - 1$$

Packets requiring edge e are younger. Hence the claim holds.

SIS is ∀ Stable

Theorem: The system (G, $\mathcal{A}$, SIS) is stable. Let d be the length of the longest simple directed path in G. No queue contains more than k_d packets and no packet spends more than

$$(db + \sum_{i=1}^{d} k_i) / \varepsilon$$

steps in the system

Instability and initial loads

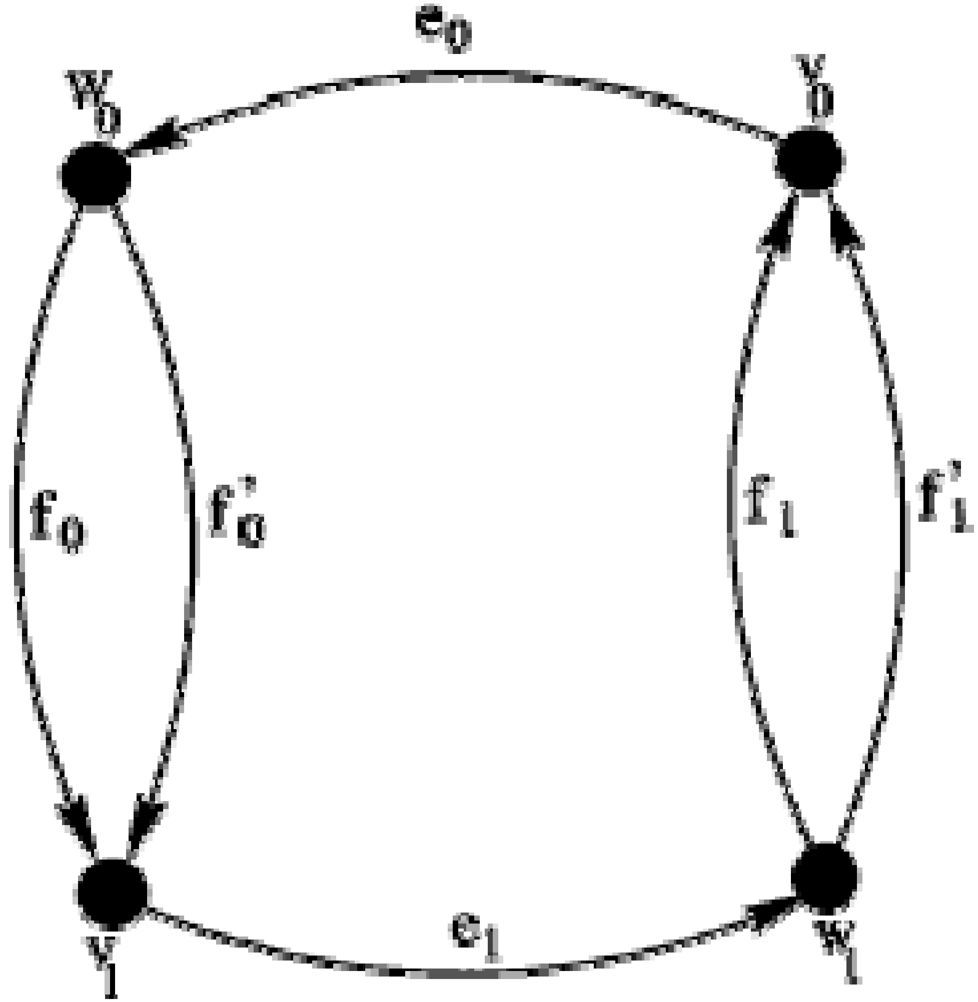
Theorem: Suppose the system (G, $\mathcal{A}$, $\mathcal{P}$) is unstable starting with some initial configuration and $\mathcal{A}$ an adversary of rate r. Then there exists a system (G', $\mathcal{A}'$, $\mathcal{P}$) that is unstable starting from an empty initial configuration where $\mathcal{A}'$ an adversary of rate r.

FIFO is unstable

Theorem: The system ($G_{basebal}$, $\mathcal{A}$, FIFO) is unstable, where $\mathcal{A}$ an adversary with rate $r \geq 0.85$

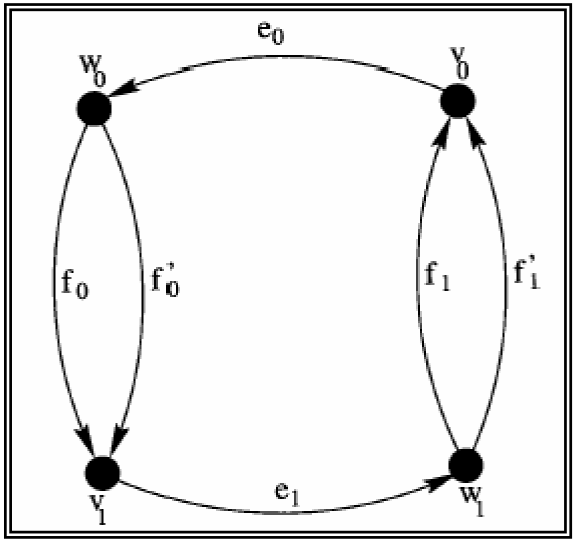
Proof: A proof by induction where the adversary will be defined in 'phases'. Initially some packets (# s_0)will be waiting for edge e_0. There will be odd and even phases and each phase will result in edge e_{i+1} having more packets in queue. than the previous.

The baseball graph



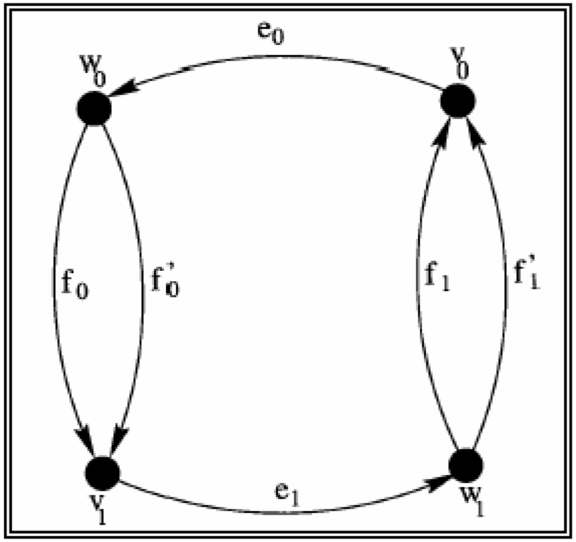
FIFO Adversary

Time interval	# packets	path	Blocked by	label	blocks
s	rs	$e_0f'_0e_1$	S	X	
rs	*r2s*	$E_0f_0e_1$	X	Y	
	r^2s	f'_0	X	W	X
r^2s	r^3s	E_1	X,Y	Z	X,Y



FIFO system evolution I

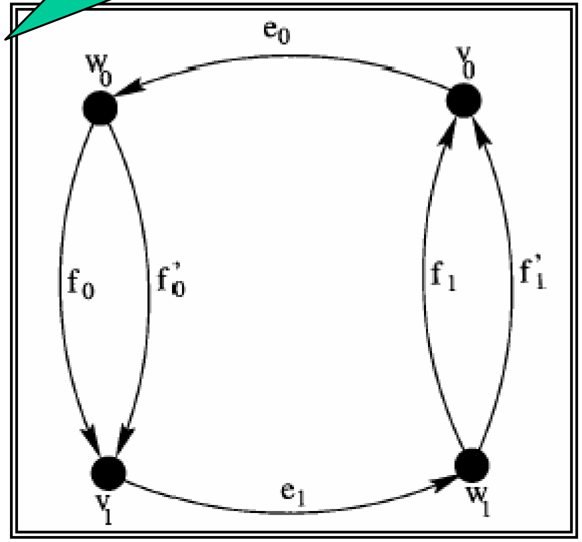
Time interval	# packets	path	Blocked by	label	blocks
s	rs	$e_0 f'_0 e_1$	S	X	



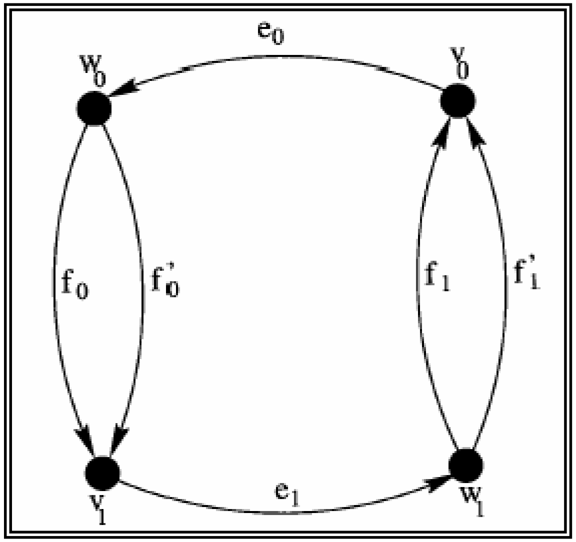
FIFO system evolution II

rs/(r+1) packets of X cross f'$_{0.}$
X packets shrinks to $r^2s/(r+1)$

Time interval	# packets	path	Blocked by	label	blocks
s	rs	$e_0 f'_0 e_1$	S	X	
rs	r^2s	$e_0 f_0 e_1$	X	Y	
	r^2s	f'_0	X	W	X



FIFO system evolution III



Time interval	# packets	path	Blocked by	label	blocks
s	$r^2s/(r+1)$	$e_0 f'_0 e_1$	S	X	
rs	r^2s	$e_0 f_0 e_1$	X	Y	
	0	f'_0	X	W	X
r^2s	r^3s	e_1	X,Y	Z	X,Y

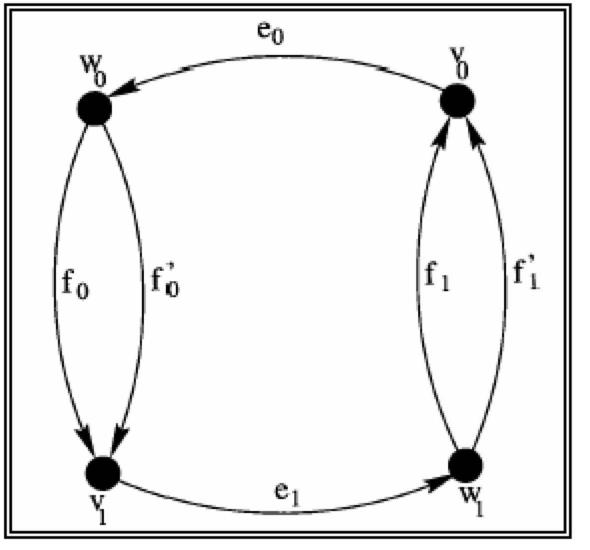
r^2s packets cross e_1 so the queue of e_1 contains $r^3s + r^2s/(r+1)$ packets.

FIFO system evolution: The Invariant

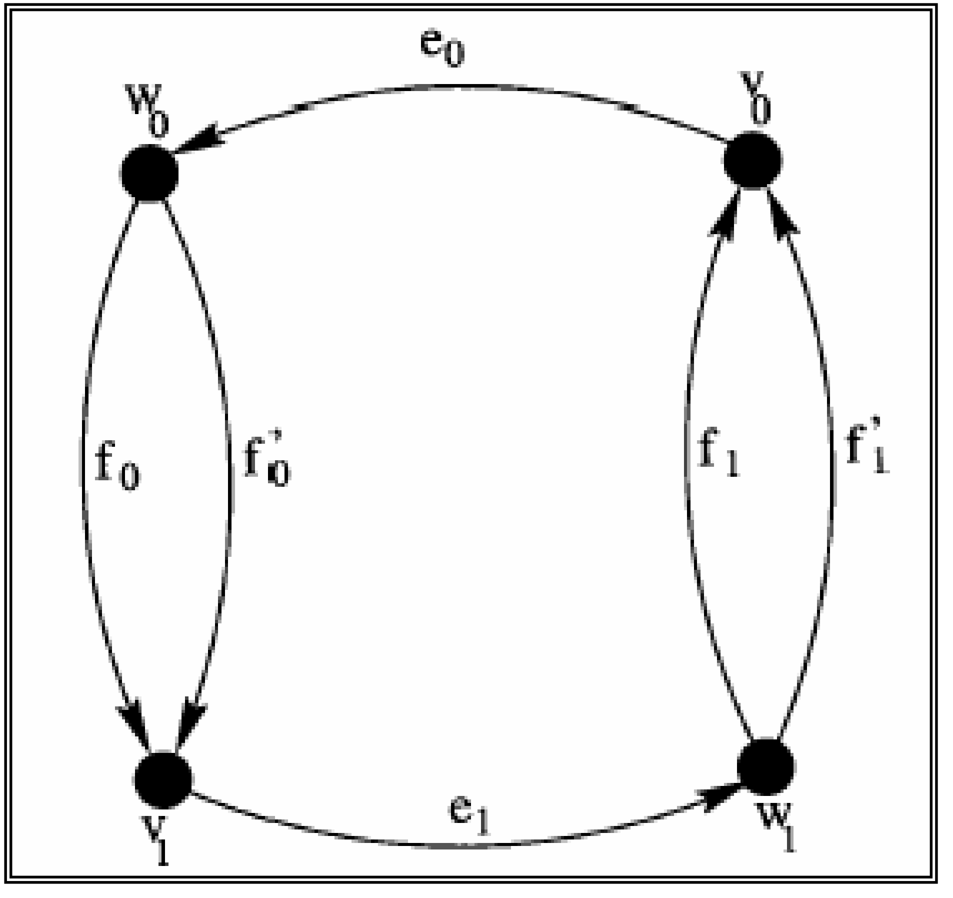
For $r \geq 0.85$

$r^3 s + r^2 s/(r + 1) > s$, hence the invariant holds.

At the end of the phase we have more packets in the symmetric edge



FIFO is unstable!

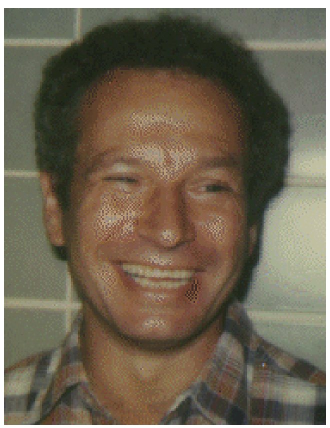


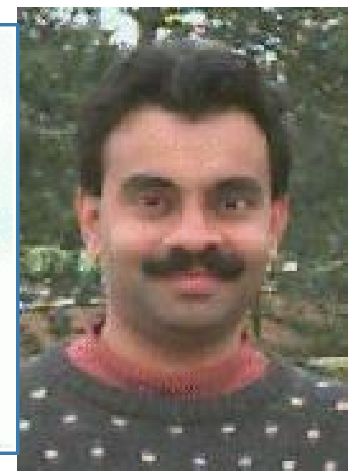
Universal Stability

Protocol	Universal stability?
FIFO	No
LIFO	No
NTG (Nearest to Go)	No
FFS (Furthest from Source)	No
FTG (Farthest to Go)	Yes
NTS (Nearest to Source)	Yes
SIS (Shortest in System)	Yes
LIS (Longest in System)	Yes

Characterizing network stability

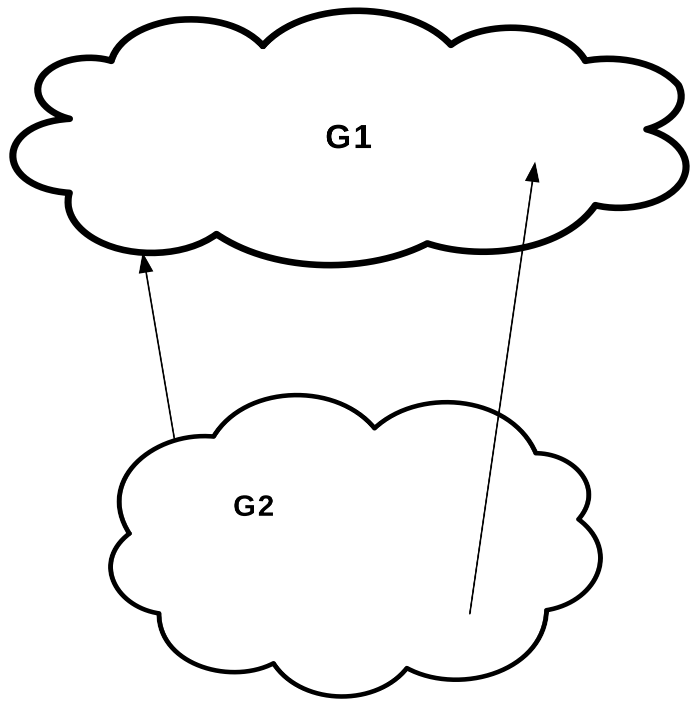
- Allan Borodin, Jon Kleinberg, Prabhakar Raghavan, Madhu Sudan, David P. Williamson: Adversarial Queuing Theory, Journal of the ACM 2001
- **Theorem**: Every DAG is universally stable!




- Theorem: The directed cycle is universally stable.

- **Lemma**: If the directed graphs G_1, G_2 are universally stable then so is any directed graph G formed by joining them with edges that only go from G_1 to G_2



- **Proof**: The adversary has rate 1-ε and burst size w. Since G_1, is universally stable any packets originating in G_1 get out in T_1 steps. Some of them may enter G_2 . Let T_2 be the size of a time window for G_2 . New packets in G_2 will have entered in $T_1 + T_2$. The # of paths needing to cross some edge in G_2 will be at most $(T_1 + T_2 + w)(1-\varepsilon)$.
- For $0< \varepsilon' < \varepsilon$

$$T_2(1-\varepsilon') \geq (T_1 + T_2 + w)(1-\varepsilon) \Leftrightarrow$$

$$T_2 \geq (T_1 + w)(1-\varepsilon)/(\varepsilon' - \varepsilon)$$

- This implies that the packets in G_2 could have been introduced by a $(T_1 + w)(1-\varepsilon)/(\varepsilon- \varepsilon')$, $1- \varepsilon'$ adversary. From G_2 universal stability the packets from G_1 will not cause a problem.

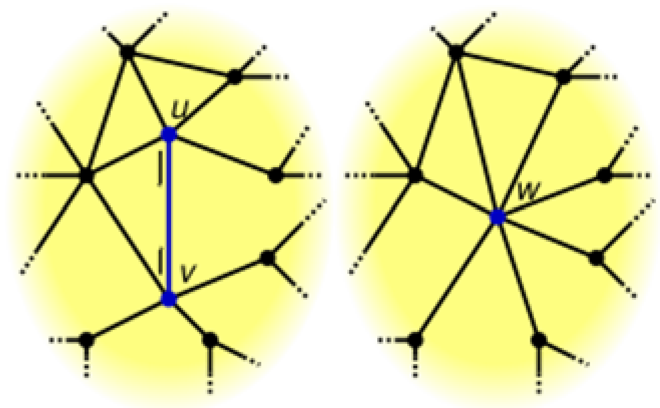
- **Corollary**: A directed graph is universally stable iff

all its strongly connected components are universally stable.

- **Lemma** : A directed cycle is universally stable. [Andrews et al.]

Definition:

- edge contraction is an operation which removes an edge from a graph while simultaneously merging together the two vertices it used to connect.



- A graph H is a minor G if it can be obtained from a subgraph of *G* by zero or more applications of edge contraction.

Definition:

- A set of graphs $\mathcal{G}$ is said to be *minor-closed* if whenever G $\in \mathcal{G}$ every minor of G is also in $\mathcal{G}$.

Theorem [Robertson Seymour]:

Any minor-closed set of graphs is defined by the exclusion of a finite set of graphs as minors.

Theorem :

If G is universally stable, and H is a minor of G, then H is universally stable.

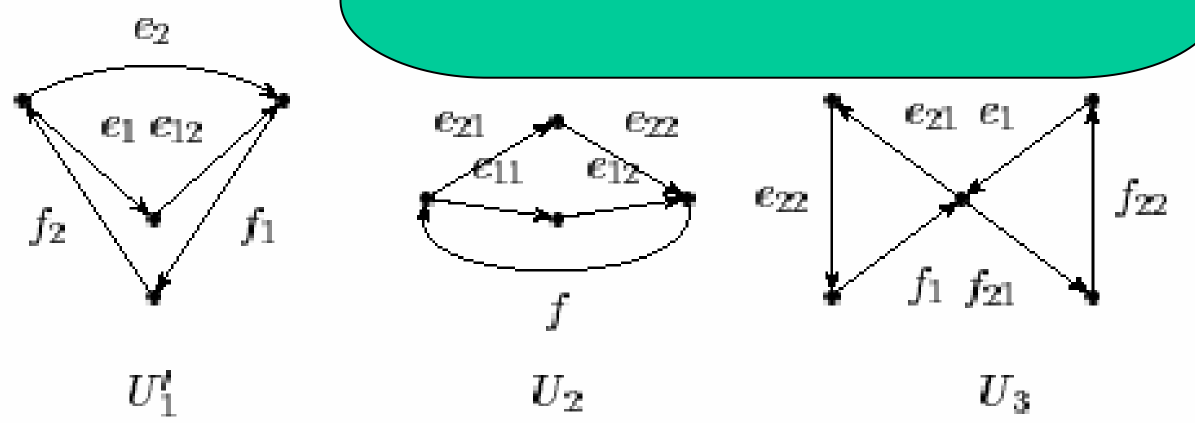
Corollary:

There exists a polynomial time algorithm to decide universal stability (test whether a given graph H is a minor of G.)

Theorem[Goel]

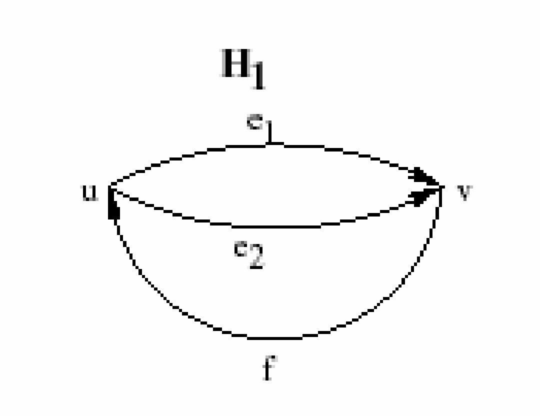
- : A directed graph is ... contain any of ... list.

for digraphs without parallel edges. [Alvarez, C., Blesa, M., Serna, M]



- Ring is characterized.
- What about other simple graphs?
- Begin with adversaries without simple paths.

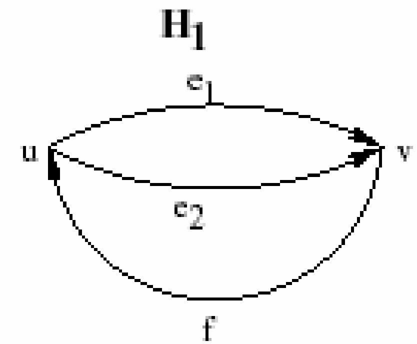
- **Lemma** : H_1 is not universally stable.



- Since we are talking about network stability the adversary is allowed to define the queuing discipline.
- The proof proceeds by an inductive definition of an adversary. Each phase of the adversary has 4 periods.
- Precondition: s packets (of some set S) are waiting for edge f.

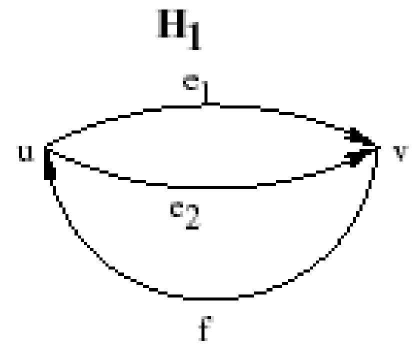
The adversary definition; system evolution

Time intv.	# packets	path	Blocked by	label	blocks
s	λs	fe_2	S	D	
λs	$\lambda^2 s$	e_2	D	A	
λs	$\lambda^2 s$	fe_1	D	A’	
$\lambda^2 s$	$\lambda^3 s$	e_2	A	B	
$\lambda^2 s$	$\lambda^3 s$	$e_1 f$	A’	B’	
$\lambda^3 s$	$\lambda^4 s$	$e_2 f$	B	C	
$\lambda^3 s$	$\lambda^4 s$	e_1		C’	B’



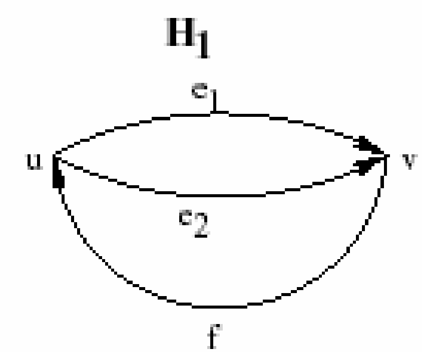
The adversary definition; system evolution

Time intv.	# packets	path	Blocked by	label	blocks
s	λs	fe_2	S	D	



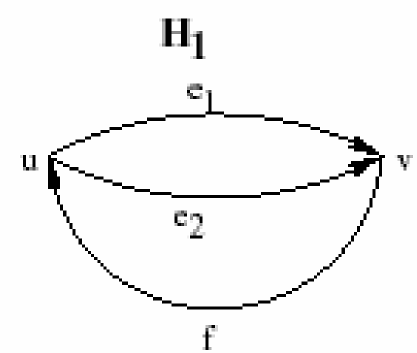
The adversary definition; system evolution

Time intv.	# packets	path	Blocked by	label	blocks
s	λs	fe_2	S	D	
λs	$\lambda^2 s$	e_2	D	A	
λs	$\lambda^2 s$	fe_1	D	A'	



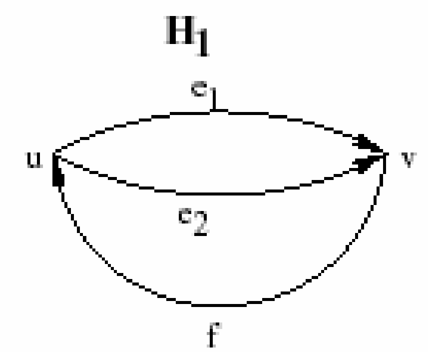
The adversary definition; system evolution

Time intv.	# packets	path	Blocked by	label	blocks
λs	$\lambda^2 s$	e_2	D	A	
λs	$\lambda^2 s$	fe_1	D	A’	
$\lambda^2 s$	$\lambda^3 s$	e_2	A	B	
$\lambda^2 s$	$\lambda^3 s$	$e_1 f$	A’	B’	

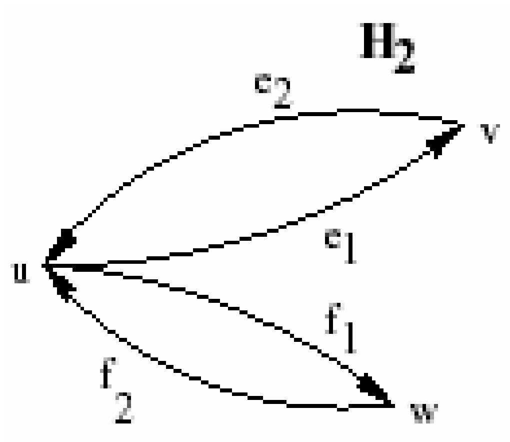


The adversary definition; system evolution

Time intv.	# packets	path	Blocked by	label	blocks
$\lambda^2 s$	$\lambda^3 s$	e_2	A	B	
$\lambda^2 s$	$\lambda^3 s$	$e_1 f$	A’	B’	
$\lambda^3 s$	$\lambda^4 s$	$e_2 f$	B	C	
$\lambda^3 s$	$\lambda^4 s$	e_1		C’	B’

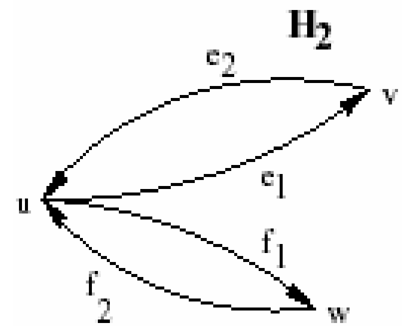


- **Lemma** : H_2 is not universally stable. [Goel]



The adversary definition; system evolution

Time intv.	# packets	path	Blocked by	label	blocks
s	λs	$e_1e_2f_1e_2$	S	A	
λs	$\lambda^2 s$	e_2	A	B	
λs	$\lambda^2 s$	f_2	A	B’	
$\lambda^2 s$	$\lambda^3 s$	$e_2\ e_1$	B	C	
$\lambda^2 s$	$\lambda^3 s$	f_2	B’	C’	
$\lambda^3 s$	$\lambda^4 s$	e_2	C	D	B
$\lambda^3 s$	$\lambda^4 s$	e_1	C’	D’	



- **Lemma** : Any graph obtained by replacing edges of H_2 or H_1 by disjoint directed paths is not universally stable.

- Any strongly connected digraph must either be a cycle or it must consist of at least two cycles which either share an edge or a vertex.

- H_2 and H_1 show that in both cases they are unstable.

Departament de Llenguatges i Sistemes Informűtics, Universitat Politθcnica de Catalunya, Technical Report.

Report Number: LSI-02-4-R
Title: A characterization of universal stability for directed graphs in the adversarial queueing model
Author(s): Alvarez, C., Blesa, M., Serna, M

- **Theorem**: A digraph is universally stable if all its strongly connected components are simple cycles.

Bounds on Delay && Queues buffers

Protocol	Queue size	Delay
Farthest-to-Go (FTG)	$O(bm^{d-1}/\varepsilon)$	$O(bm^{d-1}/\varepsilon)$
Nearest-to-Source (NTS)	$O(bm^{d-1}/\varepsilon)$	$O(bm^{d-1}/\varepsilon)$
Shortest-in-System (SIS)	$O(b/\varepsilon^{d})$	$O(db/\varepsilon^{d})$
Longest-in-System (LIS)	$O(b/\varepsilon^{d})$	$O(b/\varepsilon^{d})$

Open Questions

- *Deterministic distributed* queuing protocol with polynomial bounded e2e delays.
- Do we have stability with small rates? (Closed for FIFO, LIFO, NTG, FFS)
- Adaptive routing and adversarial Queuing Theory.