Discontinuity-induced bifurcations in TCP/RED communication algorithms

Mingjian Liu*, Alfredo Marciello†, Mario di Bernardo†, and Ljiljana Trajković*

jliu1@cs.sfu.ca, alfredomarciello@libero.it, mario.dibernardo@unina.it, ljilja@cs.sfu.ca

* Simon Fraser University, Vancouver, Canada
† University of Naples Federico II, Naples, Italy

Roadmap

- Introduction
- TCP congestion control algorithms: an overview
  - RED algorithm
- Discrete-time dynamical model of TCP Reno with RED:
  - modeling assumptions
- Bifurcation and chaos phenomena in TCP/RED
- Discontinuity-induced bifurcations
- Conclusion
- References

Motivation

- Modeling TCP Reno with RED is important to:
  - examine the interactions between TCP and RED
  - understand and predict the dynamical network behavior
  - analyze the impact of system parameters
  - investigate bifurcations and complex behavior

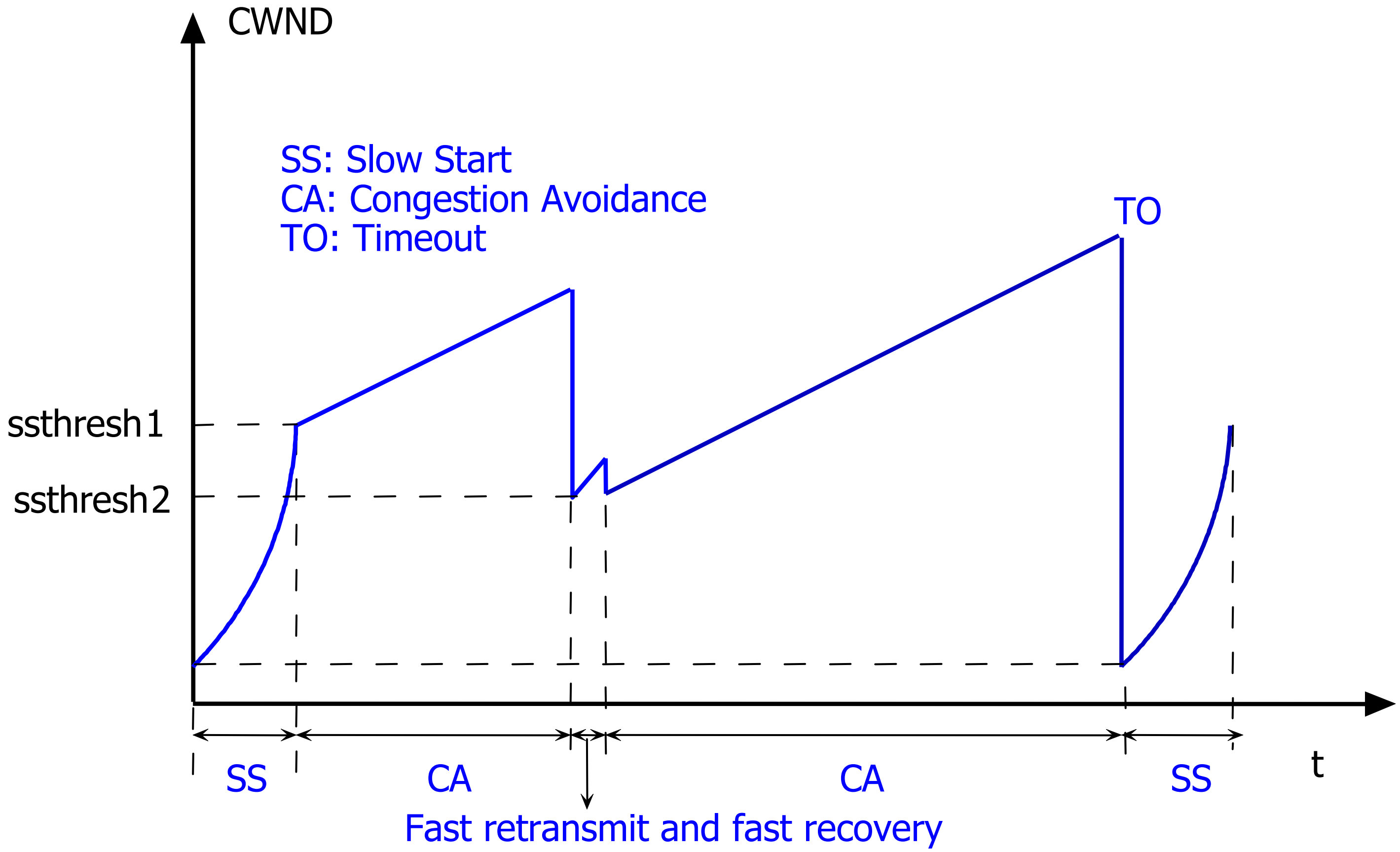
TCP: Transmission Control Protocol

RED: Random Early Detection Gateways for Congestion Avoidance

TCP

- Several flavors of TCP:
  - Tahoe: 4.3 BSD Tahoe (~ 1988)
    - slow start, congestion avoidance, and fast retransmit (RFC 793, RFC 2001)
  - Reno: 4.3 BSD Reno (~ 1990)
    - slow start, congestion avoidance, fast retransmit, and fast recovery (RFC 2001, RFC 2581)
  - NewReno (~ 1996)
    - new fast recovery algorithm (RFC 2582)
  - SACK (~ 1996, RFC 2018)

TCP Reno



CWND
SS: Slow Start
CA: Congestion Avoidance
TO: Timeout
TO
ssthresh1
ssthresh2
SS
CA
CA
SS
t
Fast retransmit and fast recovery

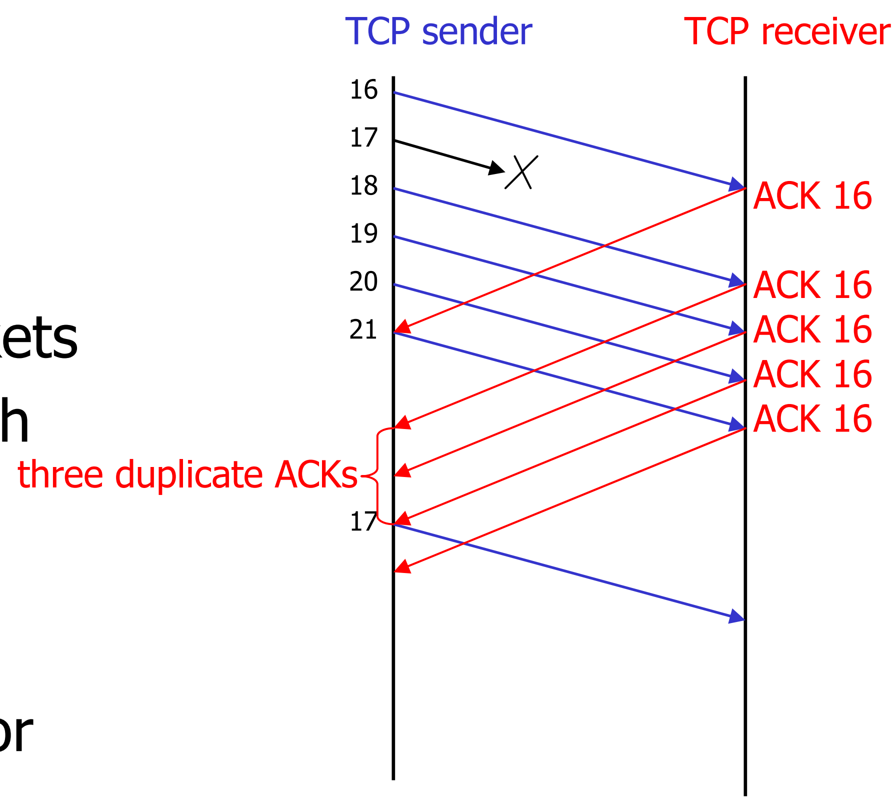
TCP Reno: slow start and congestion avoidance

- Slow start:
  - $cwnd = IW$ (1 or 2 packets)
  - when $cwnd < ssthresh$
    $cwnd = cwnd + 1$ for each received ACK
- Congestion avoidance:
  - when $cwnd > ssthresh$
    $cwnd = cwnd + 1/cwnd$ for each ACK

cwnd : congestion window size
IW : initial window size
ssthresh : slow start threshold
ACK : acknowledgement
RTT : round trip time

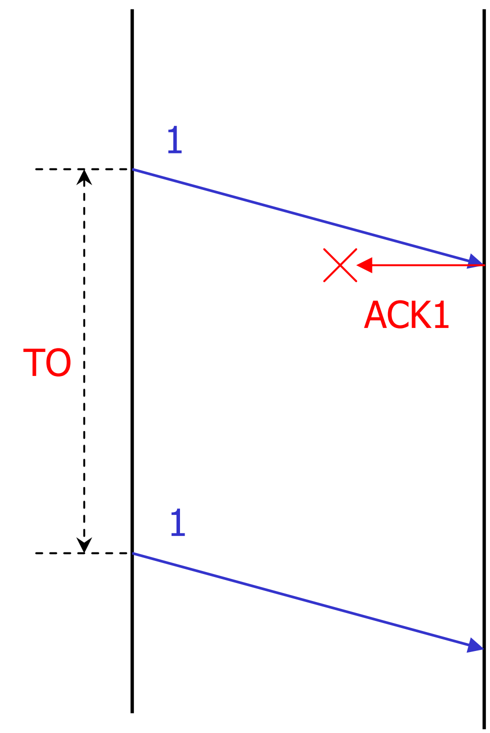
TCP Reno:
fast retransmit and fast recovery

- three duplicate *ACKs* are received
- retransmit the packet
- *ssthresh* = *cwnd*/2,
  cwnd = *ssthresh* + 3 packets
- *cwnd* = *cwnd* + 1, for each additional duplicate ACK
- transmit the new data, if *cwnd* allows
- *cwnd* = *ssthresh*, if ACK for new data is received



TCP Reno: timeout

- TCP maintains a retransmission timer
- The duration of the timer is called retransmission timeout
- Timeout occurs when the ACK for the delivered data is not received before the retransmission timer expires
- TCP sender retransmits the lost packet
- *ssthresh* = *cwnd/2*
  cwnd = 1 or 2 packets



AQM: Active Queue Management

- AQM (RFC 2309):
  - reduces bursty packet drops in routers
  - provides lower-delay interactive service
  - avoids the “lock-out” problem
  - reacts to the incipient congestion before buffers overflow
- AQM algorithms:
  - RED (RFC 2309)
  - ARED, CHOKe, BLUE, …

RED

- Random Early Detection Gateways for Congestion Avoidance
  - Proposed by S. Floyd and V. Jacobson, LBN, 1993:

    S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE/ACM Trans. Networking*, vol. 1, no. 4, pp. 397–413, Aug. 1993.
- Main concept:
  - drop packets before the queue becomes full

RED variables and parameters

- Main variables and parameters:
  - average queue size: $\bar{q}_{k+1}$
  - instantaneous queue size: q_{k+1}
  - drop probability: p_{k+1}
  - queue weight: w_q
  - maximum drop probability: $p_{\max}$
  - queue thresholds: $q_{\min}$ and $q_{\max}$

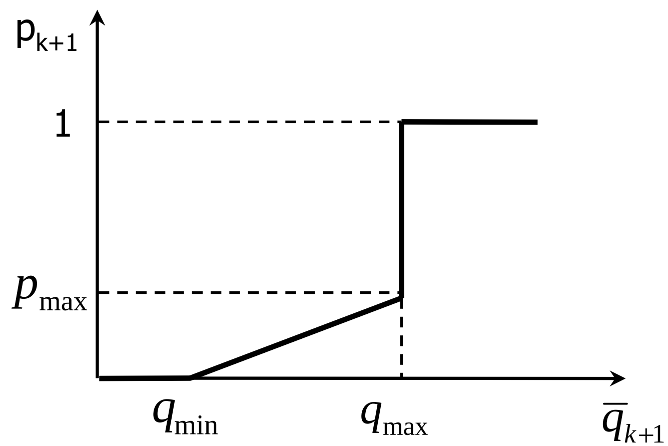
RED algorithm

Calculate:

- average queue size for each packet arrival

$$\overline{q}_{k+1} = (1 - w_q) \cdot \overline{q}_k + w_q \cdot q_{k+1}$$

- drop probability



RED algorithm: drop probability

- if $(q_{\min} < \bar{q}_{k+1} < q_{\max})$

$$p_{k+1} = \frac{\bar{q}_{k+1} - q_{\min}}{q_{\max} - q_{\min}} p_{\max}$$

- else if $(\bar{q}_{k+1} \geq q_{\max})$

$$p_{k+1} = 1$$

- else $(\bar{q}_{k+1} \leq q_{\min})$

$$p_{k+1} = 0$$

- mark or drop the arriving packet with probability p_{k+1}

Modeling methodology

- Categories of TCP models:
  - averaged and discrete-time models
  - short-lived and long-lived TCP connections
- TCP/RED model:
  - discrete-time model with a long-lived connection
- State variables:
  - window size (TCP)
  - average queue size (RED)

TCP/RED model

- Key properties of the proposed TCP/RED model:
  - slow start, congestion avoidance, fast retransmit, and fast recovery (simplified)
  - Timeout:

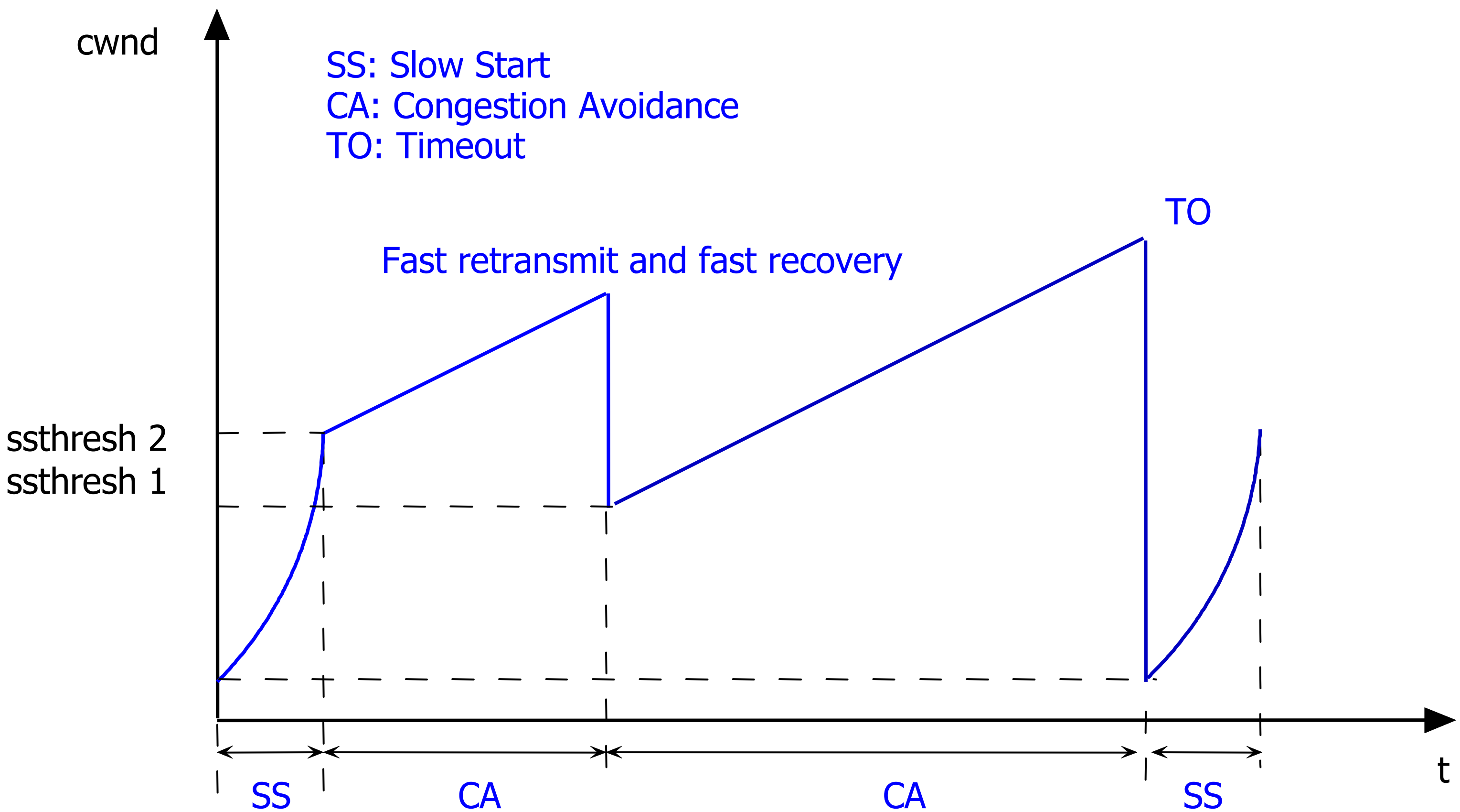
    J. Padhye, V. Firoiu, and D. F. Towsley, "Modeling TCP Reno performance: a simple model and its empirical validation," *IEEE/ACM Trans. Networking*, vol. 8, no. 2, pp. 133–145, Apr. 2000.
  - Captures the basic RED algorithm

Assumptions

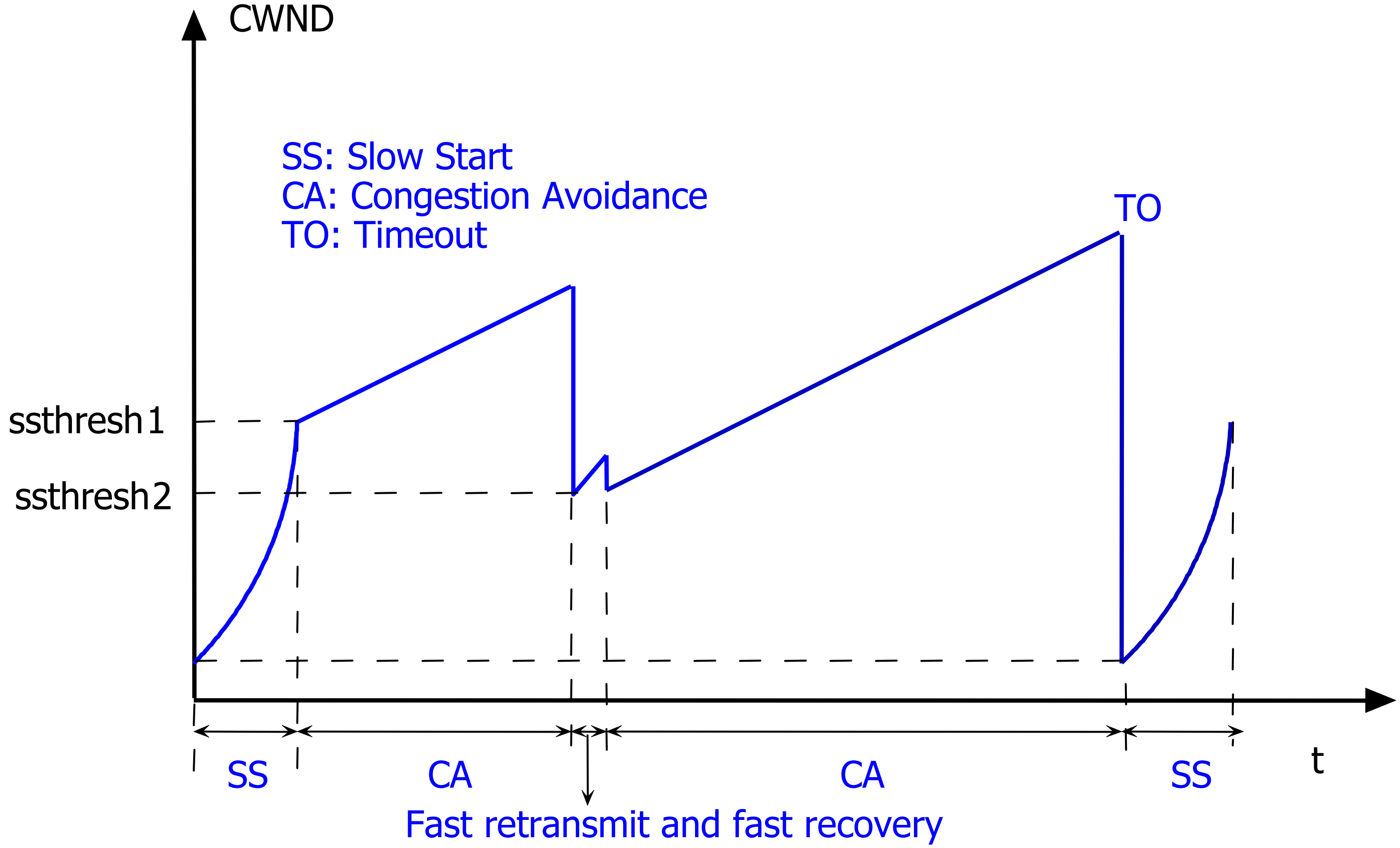
- long-lived TCP connection
- constant propagation delay between the source and the destination
- constant packet size
- ACK packets are never lost
- timeout occurs only due to packet loss
- the system is sampled at the end of every RTT interval

TCP/RED model simplifications

- Simplified fast recovery



TCP Reno: fast recovery



TCP/RED model simplifications

- TO = 5 RTT

  V. Firoiu and M. Borden, “A study of active queue management for congestion control,” in *Proc. of IEEE INFOCOM 2000*, vol. 3, pp. 1435–1444, Tel-Aviv, Israel, Mar. 2000.

- RED: parameter count is not used

if $(q_{\min} < \bar{q} < q_{\max})$

$$p_b = p_{\max} \times \frac{\bar{q} - q_{\min}}{q_{\max} - q_{\min}}$$

$$p_a = \frac{p_b}{1 - count \times p_b}$$

$$\xrightarrow{p_a = p_b}$$

if $(q_{\min} < \bar{q} < q_{\max})$

$$p_a = p_{\max} \times \frac{\bar{q} - q_{\min}}{q_{\max} - q_{\min}}$$

TCP/RED model

- S-model: a discrete nonlinear dynamical model of TCP Reno with RED
- Two state variables:
  - window size
  - average queue size
- The proposed TCP/RED model is:
  - simple and intuitively derived
  - able to capture detailed dynamical behavior of TCP/RED systems
  - has been verified via ns-2 simulations

TCP/RED model: state variable and parameters

- q_{k+1}: instantaneous queue size in round k+1
- $\overline{q}_{k+1}$: average queue size in round k+1
- W_{k+1}: current TCP window size in round k+1
- w_q: queue weight in RED
- p_k: drop probability in round k
- RTT_{k+1}: round-trip time at k+1
- C: capacity of the link between the two routers
- M: packet size
- d: round-trip propagation delay
- ssthesh: slow start threshold
- rwnd: receiver's advertised window size

TCP/RED model: no loss

- drop probability: $p_k W_k < 0.5$
- window size:

$$W_{k+1} = \begin{cases} \min(2W_k, ssthresh) & if\ W_k < ssthresh \\ \min(W_k + 1, rwnd) & if\ W_k \geq ssthresh \end{cases}$$

- where:
  - W_{k+1}: window size in round k+1
  - ssthesh: slow start threshold
  - rwnd: receiver's advertised window size

TCP/RED model: no loss

- current queue size:

$$q_{k+1} = q_k + W_{k+1} - C \cdot \frac{RTT_{k+1}}{M}$$
$$= q_k + W_{k+1} - \frac{C}{M}(d + \frac{q_k M}{C})$$
$$= W_{k+1} - \frac{C \cdot d}{M}$$

- where:
  - RTT_{k+1}: round-trip time at k+1
  - C: capacity of the link between the two routers
  - M: packet size
  - d: round-trip propagation delay

TCP/RED model: no loss

- average queue size:

$$\bar{q}_{k+1} = (1 - w_q) \cdot \bar{q}_k + w_q \cdot \max(W_{k+1} - \frac{C \cdot d}{M}, 0)$$

- hence:

$$\bar{q}_{k+1} = (1 - w_q)^{W_{k+1}} \bar{q}_k + (1 - (1 - w_q)^{W_{k+1}}) \cdot \max(W_{k+1} - \frac{C \cdot d}{M}, 0)$$

S-TCP/RED model: no packet loss

- drop probability: $p_k W_k < 0.5$

- window size:

$$W_{k+1} = \begin{cases} \min(2W_k, ssthresh) & if\ W_k < ssthresh \\ \min(W_k + 1, rwnd) & if\ W_k \geq ssthresh \end{cases}$$

- average queue size:

$$\overline{q}_{k+1} = (1-w_q)^{W_{k+1}} \overline{q}_k + (1-(1-w_q)^{W_{k+1}}) \cdot \max(W_{k+1} - \frac{C \cdot d}{M}, 0)$$

S-TCP/RED model: one packet loss

- drop probability: $0.5 \leq p_k W_k < 1.5$
- window size: $W_{k+1} = \frac{1}{2} W_k$
- average queue size:

$$\overline{q}_{k+1} = (1 - w_q)^{W_{k+1}} \overline{q}_k + (1 - (1 - w_q)^{W_{k+1}}) \cdot \max(W_{k+1} - \frac{C \cdot d}{M}, 0)$$

TCP/RED model: two packet losses

- drop probability: $p_k W_k \geq 1.5$
- window size: $W_{k+1} = 0$
- average queue size: $\overline{q}_{k+1} = \overline{q}_k$

RED: default parameters

- RED parameters:

  S. Floyd, "RED: Discussions of Setting Parameters," Nov. 1997: http://www.icir.org/floyd/REDparameters.txt

Queue weight (w_q)	0.002
Maximum drop probability (p_{max})	0.1
Minimum queue threshold (q_{min})	5 (packets)
Maximum queue threshold (q_{max})	15 (packets)
Packet size (M)	4,000 (bytes)

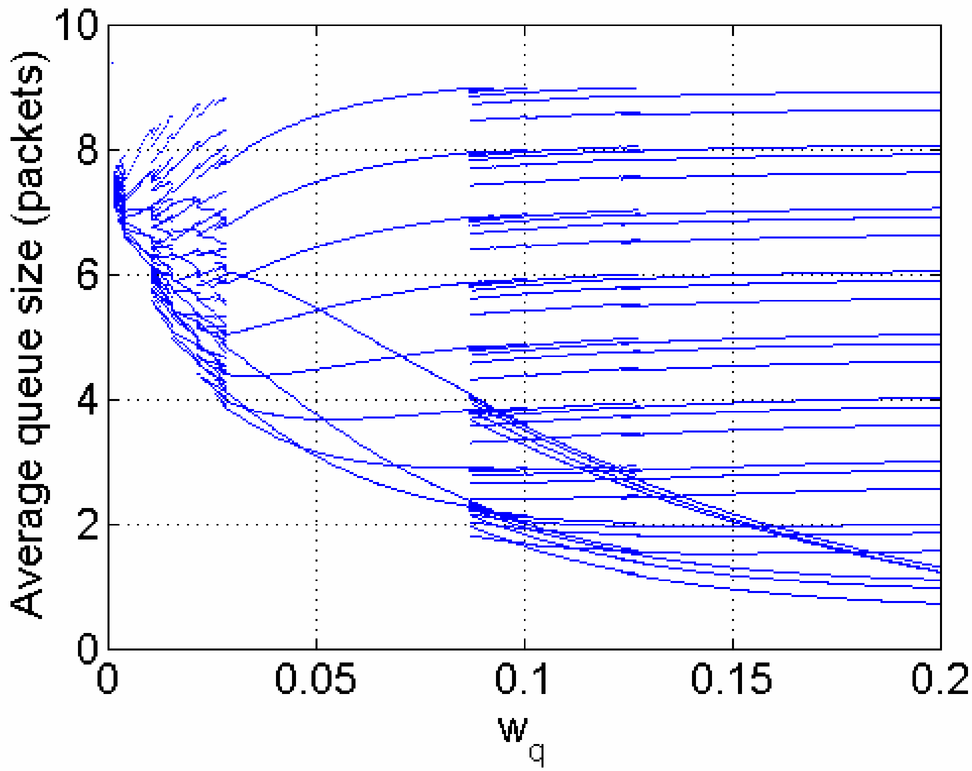
TCP/RED: bifurcation and chaos

- Bifurcation diagrams for various values of the system parameters:
  - queue weight: w_q
  - maximum drop probability: p_{max}
  - queue thresholds: q_{min} and q_{max} ($q_{max}/q_{min} = 3$)
  - round-trip propagation delay: d

Queue weight (w_q)	0.002
Maximum drop probability (p_{max})	0.1
Minimum queue threshold (q_{min})	5 (packets)
Maximum queue threshold (q_{max})	15 (packets)
Packet size (M)	4,000 (bytes)

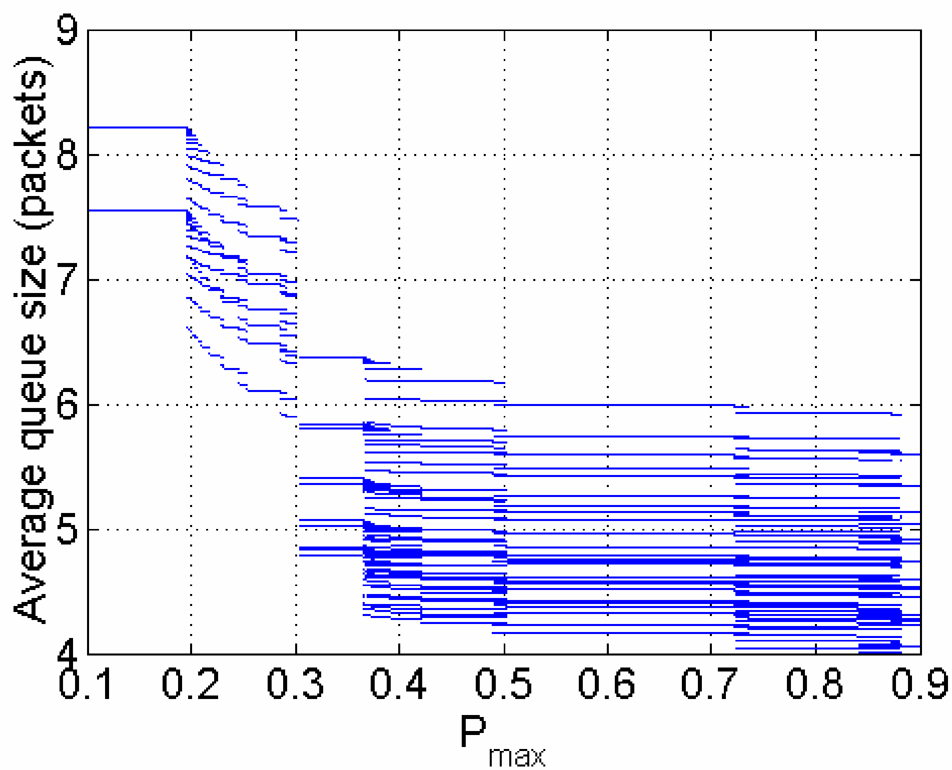
Average queue size vs. w_q

- $p_{max} = 0.1$, $q_{min} = 5$, $q_{max} = 15$, and sstresh = 80



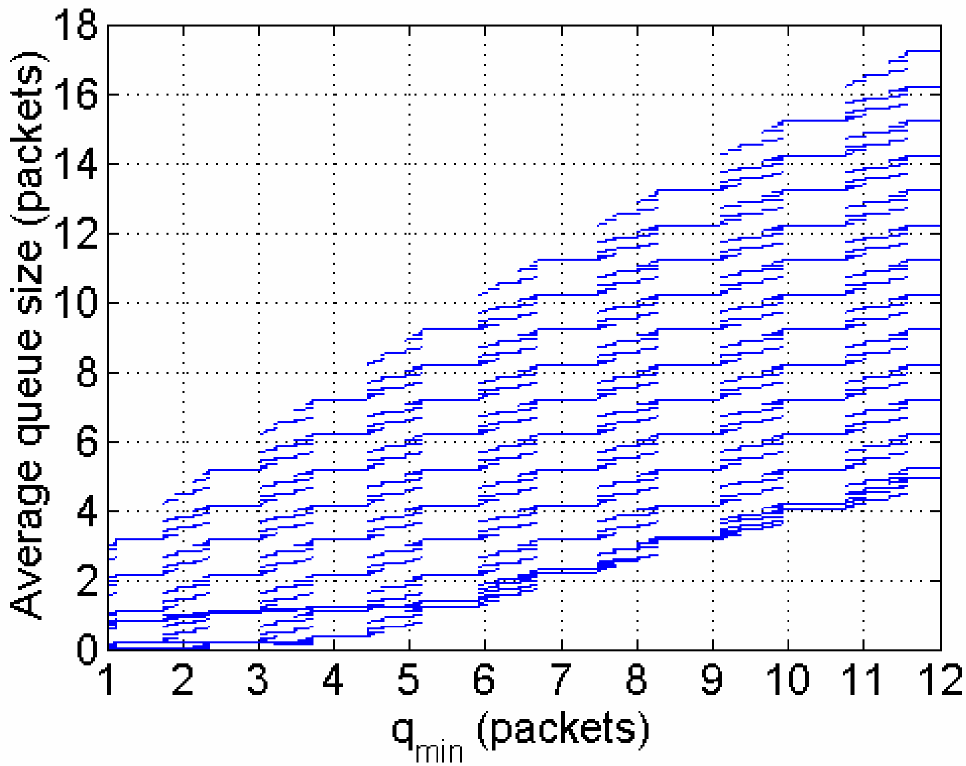
Average queue size vs. p_{max}

- $w_q = 0.01$, $q_{min} = 5$, $q_{max} = 15$, and ssthresh = 20



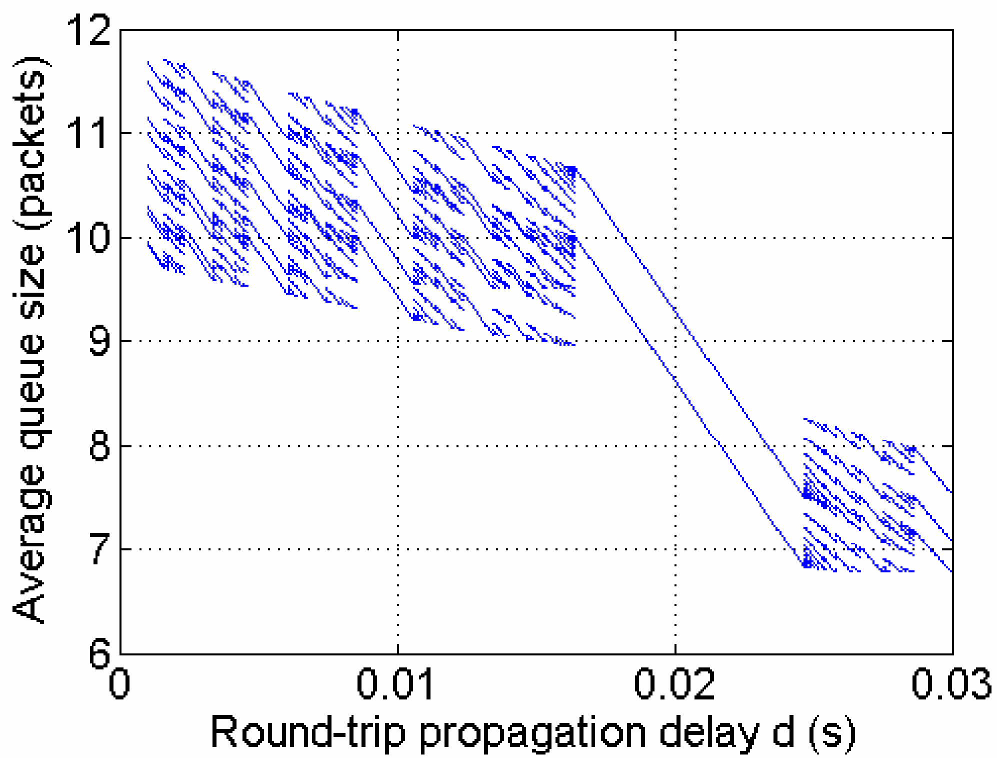
Average queue size vs. q_{min}/q_{max}

- $w_q = 0.01$, $p_{max} = 0.1$, $q_{max} = 3\ q_{min}$, and ssthresh = 20



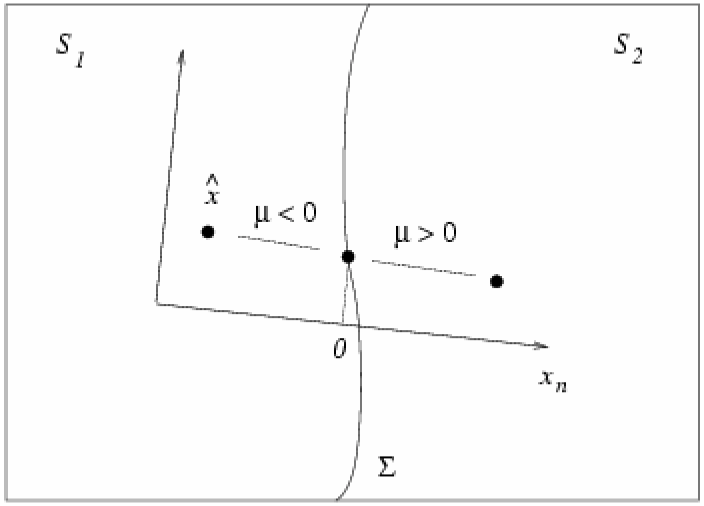
Average queue size vs. d

- $w_q = 0.01$, $p_{max} = 0.1$, $q_{min} = 5$, $q_{max} = 15$, and ssthresh = 20



An analytical explanation

- Nonsmooth systems may exhibit discontinuity-induced bifurcations: a class of bifurcations unique to their nonsmooth nature
- These phenomena occur when a fixed point, cycle, or aperiodic attractor interacts nontrivially with one of the phase space boundaries where the system is discontinuous



Discontinuity-induced bifurcations: classification

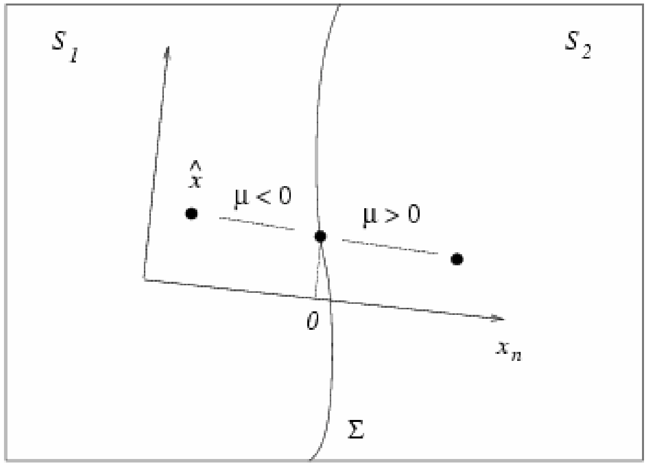
- Standard:
  - SN (smooth saddle-node)
  - PD (smooth period-doubling)
- C-bifurcations or DIBs
  - PWS maps: border collisions of fixed points
  - PWS flows: discontinuous bifurcations of equilibriums
  - Grazing bifurcations of periodic orbits
  - Sliding bifurcations

Border collisions in PWS maps

- Consider a map of the form:

$$x_{k+1} = \begin{cases} F_1(x_k, p), & H(x_k) < 0 \\ F_2(x_k, p), & H(x_k) > 0 \end{cases}$$

- A fixed point is undergoing a border-collision bifurcation at *p=0* if:

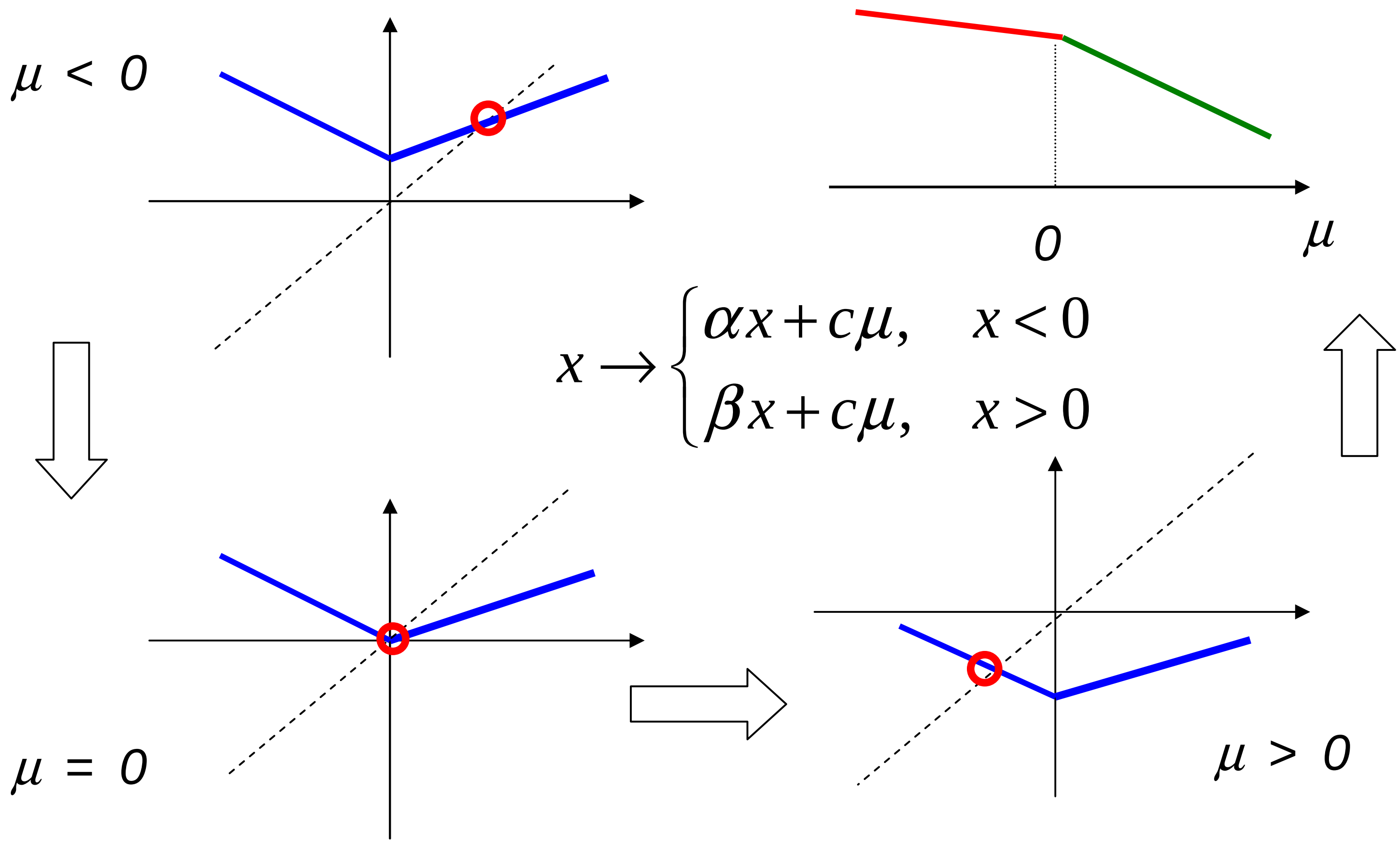


- $\mu \in (-\varepsilon, 0) \Rightarrow x^* \in S_1$
- $\mu \in (0, \varepsilon) \Rightarrow x^* \in S_2$
- $\mu = 0 \Rightarrow x^* \in \Sigma$
- $DF_1 \neq DF_2$ on Σ

Classifying border collisions

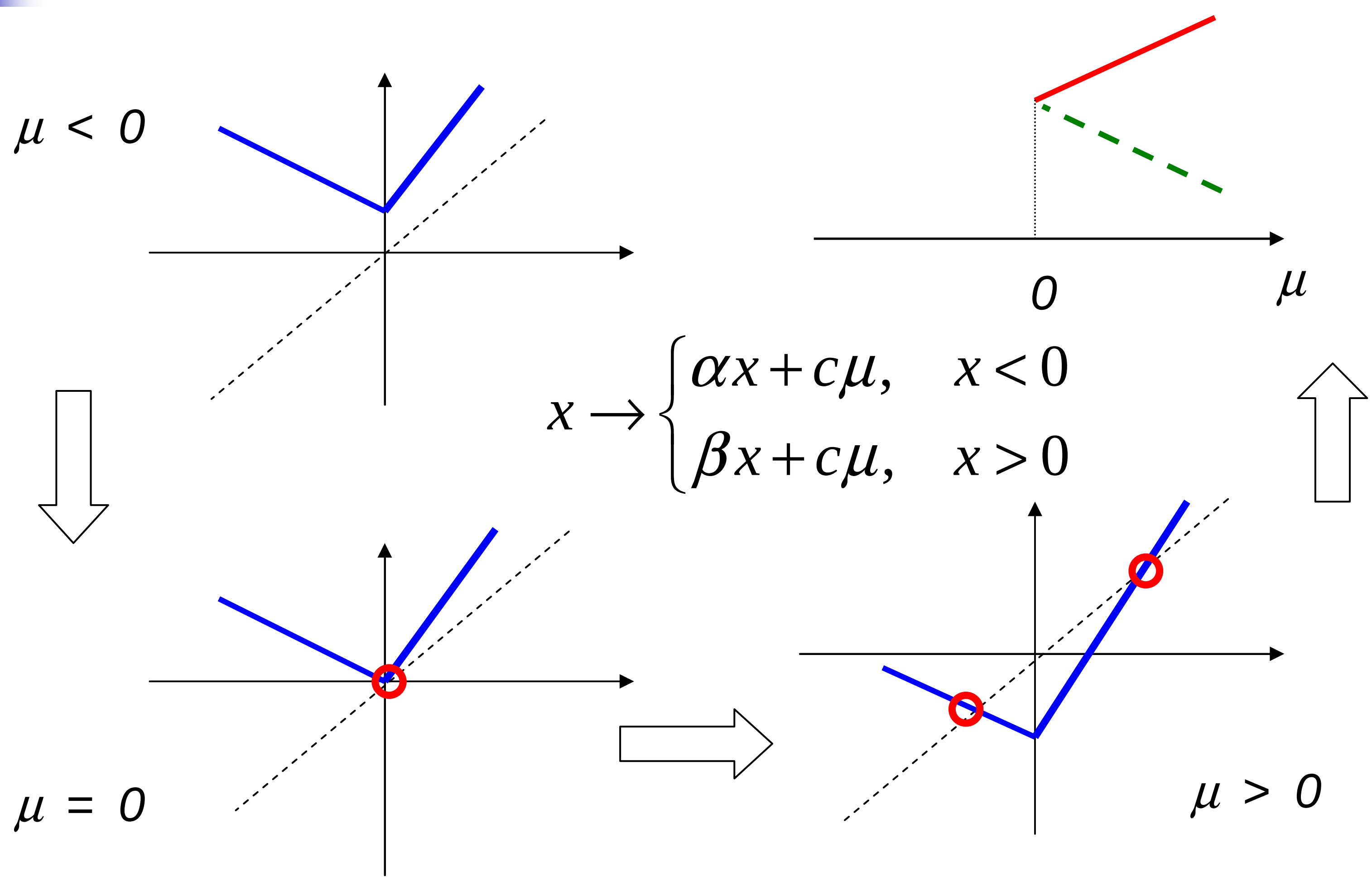
- Several scenarios are possible when a border-collision occurs
- They can be classified by observing the map eigenvalues on both sides of the boundary
- The phenomenon can be illustrated by a very simple 1D map where the eigenvalues are the slopes of the map on both sides of the boundary

Persistence



$$x \rightarrow \begin{cases} \alpha x + c\mu, & x < 0 \\ \beta x + c\mu, & x > 0 \end{cases}$$

Non-smooth saddle-node



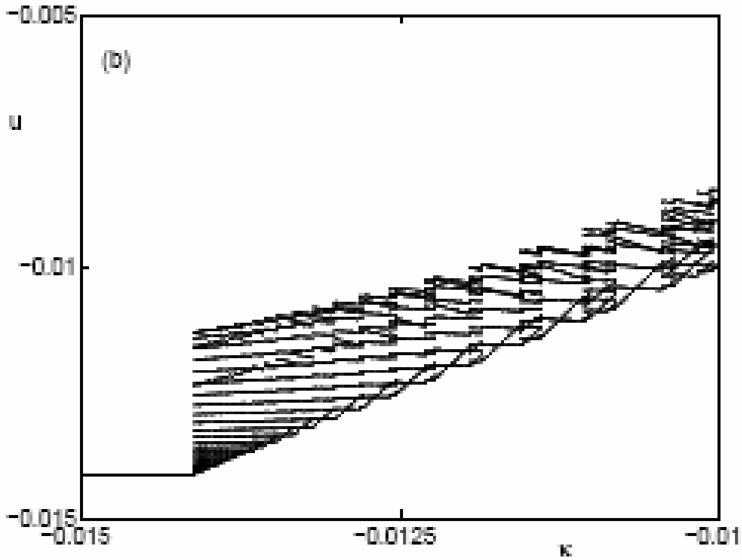
$$x \to \begin{cases} \alpha x + c\mu, & x < 0 \\ \beta x + c\mu, & x > 0 \end{cases}$$

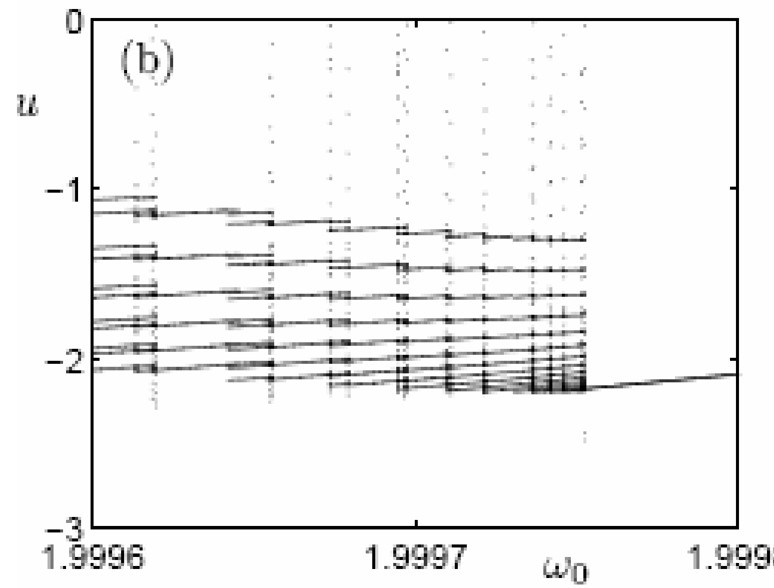
Border-collisions in the TCP/RED model

- The analysis has focussed mostly on continuous maps
- Recently proposed: further bifurcations are possible when the map is piecewise with a gap
- Complete classification method is available only for the one-dimensional case
- The TCP/RED case is a 2D map with a gap: its dynamics resemble closely those observed in very different systems: the impact oscillator considered by Budd and Piiroinen, 2006
- They might be explained in terms of border-collision bifurcations of 2D discontinuous maps

Numerical evidence

Cascades of corner-impact bifurcations in a forced impact oscillator show a striking resemblance to the phenomena detected in the TCP/RED model. They were explained in terms of border-collisions of local maps with a gap.





C. J. Budd and P. Piiroinen, "Corner bifurcations in nonsmoothly forced impact oscillators," to appear in *Physica D*, 2005.

Conclusions

- We consider a discrete-time two-dimensional model for TCP Reno with RED that includes:
  - slow start, congestion avoidance, fast retransmit, timeout, elements of fast recovery, and RED
- It captures the main features of the dynamical behavior of TCP/RED communication algorithms
- The model was used to study bifurcations and chaos in TPC/RED systems with a single connection
- Bifurcations diagrams were characterized by period-adding cascades and devil staircases
- The observed behavior can be explained in terms of a novel class of piecewise-smooth maps with a gap

References: TCP/RED and the model

[1] V. Jacobson, “Congestion avoidance and control,” *ACM Computer Communication Review*, vol. 18, no. 4, pp. 314–329, Aug. 1988.

[2] M. Allman, V. Paxson, and W. Steven, “TCP congestion control,” *IETF Request for Comments* , *RFC 2581*, Apr. 1999.

[3] S. Floyd and V. Jacobson, “Random early detection gateways for congestion avoidance,” *IEEE/ACM Trans. Networking*, vol. 1, no. 4, pp. 397–413, Aug. 1993.

[4] *V. Firoiu and M. Borden, “A study of active queue management for congestion control,” in Proc. IEEE INFOCOM 2000, Tel-Aviv, Israel, Mar. 2000, vol. 3, pp. 1435–1444.*

[5] J. Padhye, V. Firoiu, and D. F. Towsley, “Modeling TCP Reno performance: a sample model and its empirical validation,” *IEEE/ACM Trans. Networking*, vol. 8, no. 2, pp. 133-145, Apr. 2000.

[6] Khalifa and Lj. Trajkovic, “An overview and comparison of analytical TCP models,” in *Proc. IEEE Int. Symp. Circuits and Systems*, Vancouver, BC, Canada, May 2004, vol. V, pp. 469-472.

[7] M. Liu, H. Zhang, and Lj. Trajkovic, “Stroboscopic model and bifurcations in TCP/RED," in *Proc. IEEE Int. Symp. Circuits and Systems*, Kobe, Japan, May 2005, pp. 2060-2063.

[8] H. Zhang, M. Liu, V. Vukadinovic, and Lj. Trajkovic, “Modeling TCP/RED: a dynamical approach,” *Complex Dynamics in Communication Networks*, Springer Verlag, Series: Understanding Complex Systems, 2005, pp. 251-278.

References: bifurcations

[9] C. J. Budd and P. Piiroinen, “Corner bifurcations in nonsmoothly forced impact oscillators,” to appear in *Physica D*, 2005.

[10] P. Jain and S. Banerjee, “Border-collision bifurcations in one-dimensional discontinuous maps,” *Int. Journal Bifurcation and Chaos*, vol. 13. pp. 3341--3351, 2003.

[11] S. J. Hogan, L. Higham, and T. C. L. Griffin, “Dynamics of a piecewise linear map with a gap," to appear in *Proc. Royal Society*, London, 2005.

[12] P. Ranjan and E. H. Abed, “Bifurcation analysis of TCP-RED dynamics,” in *Proc. ACC*, Anchorage, AK, USA, May 2002, pp. 2443--2448.

[13] P. Ranjan, E. H. Abed, and R. J. La, “Nonlinear instabilities in TCP- RED,” in *Proc. IEEE INFOCOM 2002*, New York, NY, USA, June 2002, vol. 1, pp. 249–258.

[14] R. J. La, “Instability of a tandem network and its propagation under RED,” *IEEE Trans. Automatic Control*, vol. 49, no. 6, pp. 1006-1011, June 2004.

[15] P. Ranjan, E. H. Abed, and R. J. La, "Nonlinear instabilities in TCP-RED,” *IEEE/ACM Trans. on Networking*, vol. 12, no. 6, pp. 1079–1092, Dec. 2004.

In case you wish to read more