CSCE 619-600
Networks and Distributed Processing
Spring 2017

Congestion Control
Dmitri Loguinov
Texas A&M University

April 11, 2017
Agenda

• Brief history of the Internet
• Introduction to control theory
  – Closed-loop systems
  – Examples
  – Relationship to congestion control
• Goals of congestion control
• Fairness index
**History**

- Not-so-distant history of the Internet shows why congestion control is important
- ARPAnet existed since the 1960s and eventually evolved into a global network by the early 1980s
- NCP (Network Control Protocol, 1970) was originally used in the ARPAnet
  - Host-to-host protocol encompassing both routing and transport-layer functions
- On January 1, 1983 the Internet switched to TCP/IP (see RFCs 791, 793)
- Major *congestion collapses* in 1983-87
  - Congestion collapse: 100% link utilization, zero throughput (see RFC 896)
History 2

- Congestion collapses possible when retransmissions congest the links
  - Very few new packets are being injected into the network, which results in virtually 0% goodput
  - All bandwidth used on retransmitted packets

- Causes for congestion collapses
  - Poor startup behavior of TCP
  - Inadequate retransmission timeout during congestion
  - No congestion control

- The last item is our interest here
• Read Jacobson’s excellent paper to gain insight into this problem
• What exactly is congestion?
  – Links have finite capacity
  – When the sending rates of all flows over a given link exceeds this capacity, we have congestion
• The “level” of congestion is determined both by how much the capacity is exceeded and for how long
Each router has many input and output ports
- Each has a certain fixed capacity $C_i$
- When the total rate of traffic for port $i$ exceeds $C_i$, buffer $i$ grows and eventually overflows
Since some ports receive more traffic than others, there is a possibility that the load on port $i$ exceeds $C_i$ (in the worst case by a factor of $M - 1$).

- In response to packet loss, end-flows must control their rates in a distributed fashion.
• In general, designing congestion control for distributed environments (like the Internet) is challenging
  – *Distributed* means that flows have no direct information about what other flows are doing

• Remainder of semester
  – First examine theory behind congestion control
  – Then cover classical binary feedback (AIMD and TCP)
  – Finish with recent high-speed TCP models
Control Theory Basics

• Congestion control is a framework that has roots in feedback control theory

• Control theory has many applications in the physical world
  – Controlling aircraft (autopilot), aiming missiles at targets, automobile functions (ABS, cruise-control), robots, etc.

• In general, we study closed-loop control systems
  – Closed-loop means that we get feedback about our current position, which allows us to adjust future direction (or actions) taken by the system
Control Theory Basics 2

• Closed-loop system example: a missile must hit pre-defined GPS coordinates
  – Missile continuously monitors its GPS location and adjusts for wind, air density/resistance, gravity conditions, or any other random disturbances in its flight pattern

• Open-loop control does not rely on feedback
  – The actions of the system need to be pre-programmed beforehand
  – For example, missile directives can be: keep constant acceleration equal to 3g for 2 minutes at 45 degrees to the surface and then fall vertically down
Control Theory Basics

- Recurrence equations are discrete versions of differential equations
  - Typical differential equation
    \[ \frac{dy(t)}{dt} = F(y(t), f(t), t) \]
    - where \( y(t) \) is the controlled parameter, \( f(t) \) is the feedback, and \( t \) is time
    - Feedback \( f(t) \) usually depends on \( y(t) \)
- Recurrence (difference) equation:
  \[ y(n + 1) = y(n) + F(y(n), f(n), n) \]
Control Theory Basics 4

• Example:
  – Design a cruise control module for an automobile
  – Controlled parameter $y(t)$ is the acceleration applied to the car through its gas pedal; observed feedback $f(t)$ is the speed $v(t)$ of the vehicle
  – **Goal:** make the car go exactly $v_d$ miles an hour

• Physics background
  – Acceleration $a(t)$ is the derivative of speed:
    $$a(t) = \dot{v}(t)$$
  – We can also write:
    $$v(t) = \int_0^t a(u)\,du$$
Example (continued)

- Suppose the system is ideal, which means that the applied acceleration $y(t)$ immediately translates into a change in speed:

  \[
  \dot{v}(t) = y(t) \quad \text{model of the system}
  \]

- Then how do we adjust acceleration based on the sampled speed $v(t)$?

  \[
  \dot{y}(t) = F(v(t)) \quad \text{controller}
  \]

- In this case, the feedback is vehicle speed $f(t) = v(t)$ and function $F(.)$ can be:

  \[
  F(v(t)) = -\alpha(v(t) - v_d) = -\alpha e(t) \quad \text{error term}
  \]
Control Theory Basics 6

• Example (continued)
  − Is the previous system of equations stable?

\[
\begin{align*}
\dot{v}(t) &= y(t) \\
\dot{y}(t) &= -\alpha(v(t) - v_d)
\end{align*}
\]

• A more fundamental issue arises when the system is not perfectly known
  − Suppose the wind, various hills, and friction reduce the applied acceleration by some unknown constant \( a_f \):

\[
\begin{align*}
\dot{v}(t) &= y(t) - a_f \\
\dot{y}(t) &= -\alpha(v(t) - v_d)
\end{align*}
\]
Control Theory Basics 7

- Another controller

\[
\begin{align*}
v(n) &= v(n-1) + y(n-1) \\
y(n) &= -\alpha(v(n-1) - v_d)
\end{align*}
\]

- Where does the system evolve under the above control actions? Does it reach the desired speed? Does it stabilize or does it oscillate?

\[\alpha = 1\]
\[\alpha = 0.8\]
\[\alpha = 0.4\]
\[\alpha = 0.1\]
Control Theory Basics 8

• If the equation depends on history beyond current time $n$, it is called delayed
  – Often abbreviated DDE (Delayed Differential Equation):
    \[
    \frac{dy(t)}{dt} = F(y(t), y(t - D), f(t), t), \quad D > 0
    \]

• DDEs are significantly more complex than ordinary differential equations (ODEs)
Control Theory Basics

• In the discrete case, we have a similar form:

\[ y(n + 1) = y(n) + F(y(n), y(n - D), f(n), n) \]

• Finally, when the feedback itself is delayed, we have these two versions:

\[ \frac{dy(t)}{dt} = F(y(t), y(t - D), f(t - D), t) \]

\[ y(n + 1) = y(n) + F(y(n), y(n - D), f(n - D), n) \]

• Congestion control falls into this category
  – Feedback is provided by network routers (can be packet loss, explicit feedback, or something else) that is delayed by propagation and queuing delays of the path
Control Theory Basics 10

• **Definition**: when $F(.)$ does not explicitly depend on time $t$ (step $n$), the equation is called **homogeneous**
  - This is the case in most congestion control, so we often omit $t$ (or $n$) from the general form

• **Definition**: if function $F(.)$ is linear in variables $y$ and $f$, the control equation is said to be **linear**

• **Examples**
  - Non-homogenous, but linear: $\dot{y}(t) = y + t^2$
  - Homogeneous, non-linear, delayed
    
    $$\dot{y}(t) = -y^3(t - D) + C$$
    
  - And an equivalent recurrence:
    $$y(n + 1) = y(n) - y^3(n - D) + C$$
Control Theory Basics 11

• Assume $D\rightarrow$ is the **forward** delay from sender to router and $D\leftarrow$ is the **backward** delay
  
  – Feedback $f(n)$ depends on rate $y(n - D\rightarrow)$ seen by the router and new rate $y(n)$ depends on $f(n - D\leftarrow)$

<table>
<thead>
<tr>
<th>y(n)</th>
<th>Controlled system: at time $n$ takes input $y(n - D\rightarrow)$</th>
</tr>
</thead>
</table>

$$f(n) = F_1(y(n - D\rightarrow))$$

$$y(n) = F_2(f(n - D\leftarrow))$$

**delayed feedback** $f(n - D\leftarrow)$

$$y(n) = F_2(F_1(y(n - D\rightarrow - D\leftarrow)))$$
Example: one flow with $D^{\rightarrow} = 3$, $D^{\leftarrow} = 2$

This explains why $y(n)$ is a function of its own value RTT time units ago, i.e., $y(n - D^{\rightarrow} - D^{\leftarrow})$