Dynamics of Network Resource Management

Ibrahim Matta
Computer Science Department
Boston University

The Internet is …

HUGE

- So HUGE that no one really knows how big
- But, rough estimates:
  - Users ~ 1.8B in 2009 (source: eTForecasts)
  - Web sites > 182M active in Oct 2008 (source: netcraft)
  - Web pages ~ 150B (source: Internet archive)

The Internet is …

DYNAMIC

- Users log in and out
- New services get added
- Routing policies change
- Denial-of-Service (DoS) attacks
-

Motivation

- How to manage such a huge and highly dynamic structure like the Internet?
- How can we build Future networks?
  - Can't build and hope they work
  - Understand the steady-state and dynamics of what we are building
- Need methodologies
  - Optimization Theory
  - Control Theory
  -

Focus

- Congestion Control
- Adopt techniques from
  - Optimization Theory
  - Control Theory
- With emphasis on "Modeling"
- Prices
  - Congestion Prices
  - Exogenous Prices
    - non-load related, e.g. random wireless losses

An Optimization Theoretical Framework

- Utilities and Prices
- Kelly's Framework
- Fairness Criteria
- Discussion

Utility

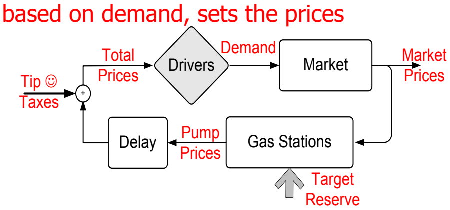
- Life … involves daily decisions
- Gas Prices are affecting these decisions
  - Drivers will observe prices, decide
    - Walk
    - Bike
    - Stay home
    - Take the subway
    - Drive

    (Walk, Bike, Stay home: Can still go to the movies ☺)

    (Drive: If it is raining)
- Utility
  - How much driving means to me compared to other things in life?
  - Unknown to the gas stations
- Each driver has his/her own utility

A slightly bigger Gas Picture

- Drivers, observe the gas price and drive the total demand
- Market (OPEC + Government + Oil companies), based on demand, sets the prices

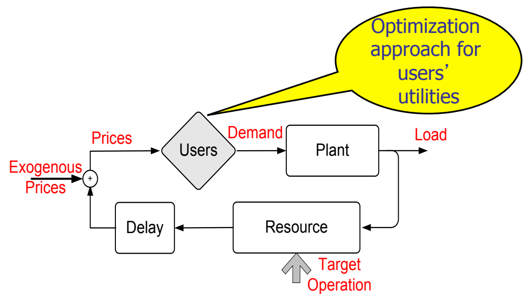


- System is in equilibrium if demand is balanced with supply

From Oil/Gas → Data Networks

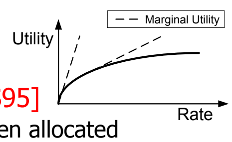
- Users drive the demand on the network
  - Have different Utilities
    - Download music, play games, make phone calls, deny service,…
- Network, observes the demand, sets prices
  - Price as real money
    - Smart Market [MV95], Paris metro [O97]
  - Price as a congestion measure
    - Queuing Delay, packet loss or marking, additional resources to be allocated
- What is the goal of Network Design? [S95]
  - Make users happy
  - Maximize the sum of Utilities for all users

From Oil/Gas ➡ Data Networks



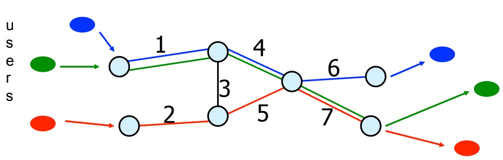
Network users' Utilities

- Users have different utilities, however
  - Higher the rate, the better
  - Decreasing marginal utility



- Formally: Elastic traffic [S95]
  - User r has utility $U_r(x_r)$ when allocated $x_r > 0$ rate
  - $U_r(x_r)$ is an increasing function, strictly concave function of x_r
  - $U'_r(x_r)$ goes to ∞ as x_r goes to 0
  - $U'_r(x_r)$ goes to 0 as x_r goes to ∞

Network Model



- Consider a network of J resources
- Consider R the set of all possible routes
- Associate a route r with each user
- Define a 0-1 routing matrix A s.t.
  - a_{jr} = 1 if resource j is on route r
  - a_{jr} = 0 otherwise

$$\begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \end{pmatrix}$$

An Optimization Problem [K97]

SYSTEM(U, A, C):

$$\max \sum_{r \in R} U_r(x_r)$$

subject to

$$Ax \leqslant C$$

over

$$x \geqslant 0.$$

C: Capacity vector
A: routing matrix
x: rates allocated

- A (unique) solution exists
- However, utilities are unknown to the network

Introducing prices …

- Break the problem into:
  - R different problems, a problem for each user
  - 1 Network problem
- Prices act as a mediator between the network and the users
  - Prices can be used to measure utilities
  - Users choose an amount to pay for the service
  - Network, based on the load, charges a price

User Maximization Problem

- Let user r, pays w_r per unit time, to receive x_r proportional to w_r

$$\lambda_r = \frac{w_r}{x_r} \quad \left(\frac{\$/t}{b/t} = \$/b\right)$$

- λ_r is the charge per unit flow

$USER_r(U_r; \lambda_r)$:

$$\max U_r\left(\frac{w_r}{\lambda_r}\right) - w_r$$

over

$$w_r \geqslant 0.$$

Network Optimization Problem

- Let the network knows the vector *W*
- Then the Network Maximization problem:

NETWORK(A, C; w):

$$max \sum_{r \epsilon R} f(x_r, w_r)$$

subject to

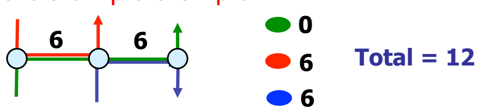
$$Ax \leqslant C$$

over

$$x \geqslant 0.$$

Network Optimization Problem

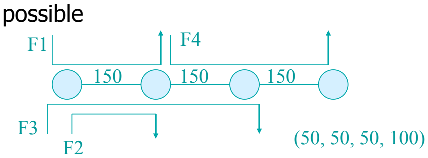
- A Greedy network choice $max \sum_{r \epsilon R} x_r w_r$
- Indeed, for w_{r}=1, maximizes overall throughput
- But, lacks traditional fairness concepts
- Here is a simple example:



Fairness criterion depends on the function that the network is optimizing for

Max-Min Fairness

- Fair
  - all sources get an equal share on every link *provided* they can use it
- Efficient
  - each link is utilized to the maximum load possible



Fairness criterion (1/3)

- Max-min Fairness
  - No rate can increase, no matter how large, while decreasing another rate that is less than it, no matter how small
  - Absolute priority to small-rate users
- X is proportionally fair if [K97]:
  - Feasible $x \geqslant 0$ and $Ax \leqslant C$
  - For any other feasible vector x^*, the aggregate of proportional changes is zero or negative:

$$\sum_{r \in R} \frac{x_r^* - x_r}{x_r} \leqslant 0.$$

Fairness criterion (2/3)

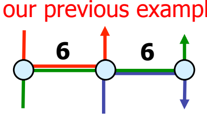
- X is weighted proportional fair if

$$\sum_{r \in R} w_r \frac{x_r^* - x_r}{x_r} \leqslant 0.$$

  - A flow of w=2, is treated like 2 flows of w=1
- Network would choose one of these
  - $max \min_{r \epsilon R} x_r$ ⇒ Max-min Fairness
  - $\max \sum_{r \in R} \log x_r$ ⇒ Rates are proportionally fair
  - $\max \sum_{r \in R} w_r \log x_r$ ⇒ Rates are weighted proportionally fair

Fairness criterion (3/3)

- In our previous example



- Maximizing total throughput
  - 0 6 6
- Proportional allocation (w_r=1)
  - 2 4 4
- Max min allocation
  - 3 3 3

General Parameterized Utility [MW00]

$$U(x) = \frac{x^{1-\alpha}}{1-\alpha}$$

- $\alpha \to 0$ **(linear utility)**
- $\alpha \to 1$ **(log utility)**
- $\alpha \to \infty$ **(min utility)**

Kelly [K97,K99,KMT98]

NETWORK(A, C; w):

$$\max \sum_{r \in R} w_r \log x_r$$

subject to

$$Ax \leqslant C$$

over

$$x \geqslant 0.$$

$$L(x, z; \mu) = \sum_{r \in R} w_r \log x_r + \mu^T (C - Ax)$$

$$\frac{\partial L}{\partial x_r} = \frac{w_r}{x_r} - \sum_{j \in r} \mu_j,$$

$$x_r = \frac{w_r}{\Sigma_{j \in r} \mu_j}$$

- Proof outline: Theory of constrained convex optimization and using Lagrange multipliers
  - μ_j = cost incurred or shadow price of additional capacity (λ's in earlier slides)
- A solution exists
  - X = weighted proportionally fair
  - Solves Network, User and System for log utility functions

Discussion

- Just to recap
  - Interested in maximizing the aggregate utilities
  - Network wouldn't know the utilities
  - Broke the problem into users and one network problem
  - So, we introduced the vector *W* as a mediator
  - Shown that a solution exists
  - Fairness criterion depends on the network maximization function

Discussion

- But, we need to address few issues:
  - Network does not know *W*
    - Network implicitly determines *W* from the user's behavior along its path, which is chosen by the network on behalf of the user
    - Or, Network puts an implicit weighting for relative utilities of different users
  - No central controller to know *W* and allocate rates

Look into individual controllers for the users and for the resources

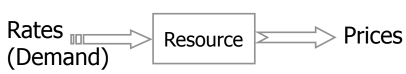
Network Dynamics & Control Theory Preliminaries

- System Modeling and Feedback Control
- TCP
- AQM
- TCP + RED

Control Problem

- The basic control problem: Control the output (results) for a given input

- Examples:



Questions to ask

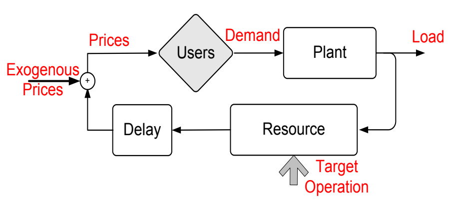
- Steady state
  - What is the long range value of the output?
  - How far is it from the reference value?
- Transient Response
  - How does the system react to perturbations?
- Stability
  - Is this system stable?
- Stability Margins
  - How far is the system from being unstable?

Open-loop Control

- There is no feedback
  - Controlled directly by an input signal
- Simple
- Example: Microwave
  - Food will be heated for the duration specified
- Not as common as closed-loop control

Feedback (Closed-loop) Control

- Feedback control is more interesting …
- Multiple controllers may be present in the same control loop

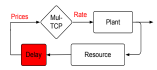


Feedback (Closed-loop) Control

- Feedback control makes it possible to control well even if
  - We don’t know everything
  - We make errors in estimation/modeling
  - Things change
- Flow/congestion control example:
  - No need to EXACTLY know
    - Number of users
    - Connections’ arrival rate
    - Resource’s service rate
  - Continually measure & correct

Feedback (Closed-loop) Control

- Feedback delay is usually associated with feedback control



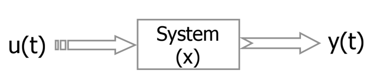
- Feedback delay: Time taken from the generation of a control signal until the process reacts to it and this reaction takes effect at the resource and effect is observed by the user/controller
- **Feedback delay can compromise stability!!**
  - The process may be reacting to some past condition that is no longer true

System Models

- Deterministic vs. Stochastic
  - Are stochastic effects (noise, uncertainties) taken into account?
- Time-invariant vs. Time-varying
  - Do system parameters change over time?
- Continuous-time vs. Discrete-time
  - Is time divided into discrete-time steps?
- Linear vs. Non-linear
  - Do dynamic equations contain non-linear terms?

System Modeling

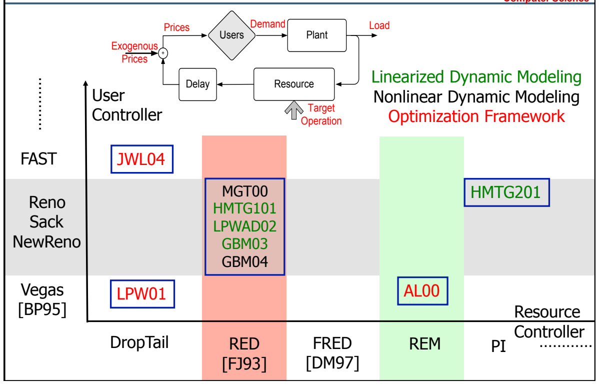
- Characterize the relationships among system variables as a function of time



$$\dot{x} = f(x, u)$$
$$y = h(x, u)$$

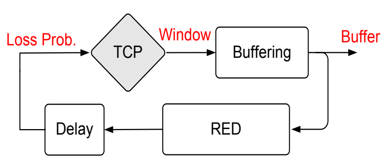
In general, f and h are nonlinear functions

Instantiations

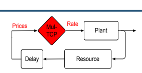


TCP & RED

- One of the instantiations that received a lot of attention
  - Neither TCP nor RED [FJ93] was introduced from a control theoretic framework



TCP Modeling [K99]



- Think about an aggregate of m TCP flows, MulTCP [CO98]
- Congestion window changes:

cwnd	: Congestion Window
T	: Round-trip time
x	: Throughput
p	: Loss probability
m	: Number of flows

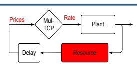
$$\frac{m}{\mathtt{cwnd}}(1-p) - \frac{\mathtt{cwnd}}{2m}\,p$$

$$\frac{\left(\frac{m}{\mathtt{cwnd}}(1-p) - \frac{\mathtt{cwnd}}{2m}\,p\right)/T}{T/\mathtt{cwnd}} \quad x = \frac{cwnd}{T}$$

Rate of ACKs

$$\frac{d}{dt}x_r(t) = \frac{m_r}{T_r^2} - \left(\frac{m_r}{T_r^2} + \frac{x_r(t)^2}{2m_r}\right)p_r(t)$$

TCP Modeling [K99]



- Depending on the total traffic passing through a resource, a congestion signal is generated with probability:

$$\mu_j(t) = p_j\left(\sum_{s:j\in s} x_s(t)\right)$$

$$\frac{d}{dt}x_r(t) = \frac{m_r}{T_r^2} - \left(\frac{m_r}{T_r^2} + \frac{x_r(t)^2}{2m_r}\right)p_r(t)$$

$$p_r(t) = \sum_{j\in r}\mu_j(t)$$

TCP-Reno Utility Function

$$\frac{d}{dt}x_r(t) = \frac{m_r}{T_r^2} - \left(\frac{m_r}{T_r^2} + \frac{x_r(t)^2}{2m_r}\right)p_r(t)$$

- For m=1 and small p, we have:

$$\frac{d}{dt}x(t) = \frac{1}{T^2} - \frac{x(t)^2}{2}p$$

$$\frac{d}{dt}x(t) = \frac{x(t)^2}{2}\left(\frac{2}{T^2\,x(t)^2} - p\right)$$

$$\dot{U}(x) = \frac{2}{T^2\,x(t)^2}, \quad U(x) = \frac{-2}{T^2\,x}$$

Min potential delay allocation

E2E Congestion Avoidance TCP Vegas

- End-to-end, dynamic window, implicit
- Expected throughput = transmission_window_size/propagation_delay
- Numerator: known
- Denominator: measure *smallest* RTT
- Also know *actual* throughput, measure it every RTT
- Difference = how much to reduce/increase rate
- New Congestion Avoidance Algorithm
  - (expected - actual)* RTT packets in bottleneck buffer
  - adjust sending rate *linearly* if this is too large or too small
- Generally loses to TCP Reno!

TCP-Vegas Utility Function

- At steady state:

$$x = \frac{\alpha D}{T_q}$$

$$T_q = \frac{b}{C},\ \dot{T}_q(t) = \frac{\dot{b}(t)}{C} = \frac{1}{C}(y(t) - C), T_q = \text{price}$$

$$\dot{U} = \frac{\alpha D}{x},\ \ U = \alpha D \log(x)$$

WPF allocation

RED Modeling

- Buffer evolution

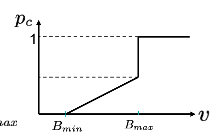
$$\dot{b}(t) = \sum x(t) - C$$

- RED averaging

$$\dot{v}(t) = -\beta C(v(t) - b(t))$$

- RED marking

$$p_c(t) = \begin{cases} 0 & v(t) \le B_{min} \\ \sigma(v(t) - \varsigma) & B_{min} < v(t) < B_{max} \\ 1 & v(t) \ge B_{max} \end{cases}$$



RED Pricing Function

- Assume linear function of instantaneous queue length: p(t) = K q(t)

$$\dot{p}(t) = K\,\dot{q}(t)$$

$$\dot{q}(t) = y(t) - C$$

$$\dot{p}(t) = K\left(y(t) - C\right)$$

- p = Lagrangian multiplier (price)

Nonlinear Models

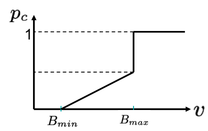
Computer Science

- Sources of nonlinearity
  - Nonlinear components
    - Example: Rate Controlled MulTCP

$$\frac{d}{dt} x_r(t) = \frac{m_r}{T_r^2} - \left(\frac{m_r}{T_r^2} + \frac{x_r(t)^2}{2m_r} \right) p_r(t)$$

  - Different operating regions
    - Example: RED

$$p_c(t) = \begin{cases} 0 & v(t) \leq B_{min} \\ \sigma(v(t) - \varsigma) & B_{min} < v(t) < B_{max} \\ 1 & v(t) \geq B_{max} \end{cases}$$



- Hard Nonlinearities
- Soft Nonlinearities

Nonlinear Models

Computer Science

- Nonlinear control theory deals directly with nonlinear differential equations
  - Stability: Lyapunov functions
  - Transient Response: Numerical solutions
- Sometimes it gets very complicated
- Linearization: Process of transforming a nonlinear set of equations into a linear set of equations around *a single point of operation*

Linearization

Computer Science

$$\dot{x} = f(x, u)$$
$$y = h(x, u)$$

Linearization around x_0

$$\dot{x} = Ax + Bu$$
$$y = Cx + Du$$

- Concerned with local stability
- Assumes a single operating point
  - Studies perturbations around this point
- Expands the nonlinear DE into Taylor series, then ignores high-order terms

Linear Models

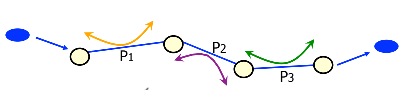
- Once we have a Linear Model
  - ➔ Apply classical (first-course) control theory
- See Control Theory Primer slides & notes

Linear vs. Nonlinear

- Linear Control
  - Rely on "small range of operation" assumption
  - Simple to use
  - Has a unique equilibrium point (if stable)
  - Satisfies the superposition property
- Nonlinear Control
  - Wide range of operation
  - Could be more complex to use
  - Multiple equilibrium points may exist
  - Most control systems are nonlinear

Nonlinear Model of Sources' and Network's Adaptations

- Kelly's optimization framework
  - Maximize users' utilities subject to the network's capacity constraints [K99]



$$\frac{d}{dt}x_r(t) = \kappa\left(w_r - x_r(t)\sum_{l\in r} p_l\Big(\sum_{l\in s} x_s(t)\Big)\right)$$

Additive Increase — Multiplicative Decrease

Steady-state and stability

- Steady state
  - Set the derivatives to 0, we get the steady-state point(s)
  - We have a single equilibrium point here

$$\frac{d}{dt}x_r(t) = \kappa\left(w_r - x_r(t)\sum_{l\in r} p_l\left(\sum_{l\in s} x_s(t)\right)\right) \Rightarrow x_r = \frac{w_r}{\sum_{j\in r}\mu_j}.$$

- Stability
  - Provided through a Lyapunov function

$$\frac{d}{dy}C_j(y) = p_j(y).$$

$$\mathcal{U}(x) = \sum_{r\in R} w_r \log x_r - \sum_{j\in J} C_j\left(\sum_{s:j\in s} x_s\right)$$

Lyapunov

- Scalar function, strictly convergent
- Finding a function guarantees stability
- Not finding a function, doesn't say anything
- Art to find one

$$\frac{\partial}{\partial x_r}\mathcal{U}(x) = \frac{w_r}{x_r} - \sum_{j\in r} p_j\left(\sum_{s:j\in s} x_s\right)$$

$$\frac{d}{dt}\mathcal{U}(x(t)) = \sum_{r\in R}\frac{\partial\mathcal{U}}{\partial x_r}\cdot\frac{d}{dt}x_r(t)$$

$$= \sum_{r\in R}\frac{\kappa_r}{x_r(t)}\left(w_r - x_r(t)\sum_{j\in r} p_j\left(\sum_{s:j\in s} x_s(t)\right)\right)^2 > 0$$

(except at steady-state)

Difficult road ahead…

- Coming up with Lyapunov functions, even for simple models, is not easy
- As we move towards
  - More sophisticated models
    - Feedback delay
    - Different regions/aspects of TCP
      - Timeouts
      - Slow-start
      - Self-clocking
  - Challenging environments
      - High bandwidth-delay product networks
      - Effect of exogenous losses (e.g., wireless)
  - Accounting for different AQM at the resources
  - Interference processes as in DoS attacks
- It gets harder very quickly

Linear Models

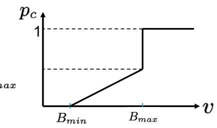
Computer Science

- Many sources of nonlinearity
  - Nonlinear components
    - Example:

$$\frac{d}{dt}x_r(t) = \frac{m_r}{T_r^2} - \left(\frac{m_r}{T_r^2} + \frac{x_r(t)^2}{2m_r}\right)p_r(t)$$

  - Different operating regions
    - Example: RED

$$p_c(t) = \begin{cases} 0 & v(t) \le B_{min} \\ \sigma(v(t) - \varsigma) & B_{min} < v(t) < B_{max} \\ 1 & v(t) \ge B_{max} \end{cases}$$



  - Need to study every point/region separately

Linearization

Computer Science

- Concerned with local stability
- Assumes a single operating point
  - Studies perturbations around this point
- Expands the nonlinear DE into Taylor series, then ignores high-order terms
  - Example: aggregating all sources and assuming one resource in $\frac{d}{dt}x_r(t) = \kappa\left(w_r - x_r(t)\sum_{l\in r} p_l(\sum_{l\in s} x_s(t))\right)$

$$\frac{d}{dt}x(t) = \kappa(w - x(t)p(x(t)) = f(x(t))$$

$$\frac{d}{dt}f(t) = -\kappa(p + p'x)f(t), \quad f(t) = x(t) - x$$

$$\frac{d}{dx(t)}f(t)|_{x(t)=x}$$

Control Theoretic Analysis

Computer Science

- Linearized Model

$$\frac{d}{dt}f(t) = -\kappa(p + p'x)f(t), \quad f(t) = x(t) - x$$

- Taking the Laplace Transform

$$sF(s) - f(0) = -\kappa(p + p'x)\,F(s)$$

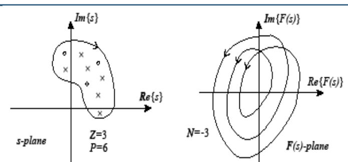
$$F(s) = \frac{f(0)}{s + \kappa(p + p'x)}$$

- Stable if $s = -\kappa(p + p'x) < 0$ (overdamped)
- For impulse perturbation, steady-state error = $\mathrm{Lim}_{s\to 0}$ sF(s) = zero

How about feedback delay?

- What if the system has feedback delay T ?
- Use Nyquist stability criterion …

Cauchy' s Principle



- Z: number of zeros of F(s)
- P: number of poles of F(s)
- N: number of encirclements of origin
- For G(s)H(s), and contour around right-hand s-plane,
  - N: encirclements around -1
  - P: number of unstable poles of GH
  - Z: number of unstable zeros of F = closed-loop poles
  - If P=0, and N=0, then Z=0 and system is stable

Nyquist Test

- What if the system has feedback delay T ?
- If the plot of the open-loop G(jω)H(jω) does not encircle the point -1 as ω is varied from -inf to +inf, then the system is stable
- *The number of unstable closed-loop poles (Z) is equal to the number of unstable open-loop poles (P) plus the number of encirclements (N) of the point (-1, j0) of the Nyquist plot of GH, that is: Z = P + N*

Nyquist Test

Computer Science

- What if the system has feedback delay T ?
- If the plot of the open-loop G(jω)H(jω) does not encircle the point -1 as ω is varied from -inf to +inf, then the system is stable
- Thus, we need to study the behavior of:

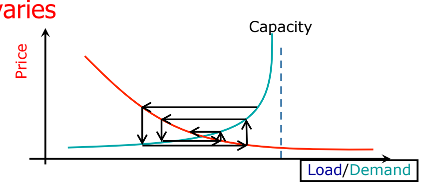
  as ω is varied $\frac{e^{-j\omega T}}{j\omega}$
- Sufficient condition for stability:

$$\kappa T(p + p'x) < \pi / 2$$

Routing is also a dynamical system!

Computer Science

- Link price functions reflect prices fed back to routing as the load on the links varies



- Convergence and stability can be proved using Lyapunov functions

Lyapunov for Routing

Computer Science

- Need to show that mapping function is contractive, i.e., range of function reduces
- Consider an adaptive routing system over two paths, with "N" total traffic, and fraction α being re-routed based on path prices
- Find necessary condition for stability
- Show it is also sufficient

References (1/2)

Computer Science

- [AL00] S. Athuraliya and S. Low Optimization Flow Control: II Implementation 2000
- [ALLY01] S. Athuraliya, S. Low, V. Li and Q. Yin REM: Active Queue Management IEEE Networks 2001
- [BB95] I-TCP: A. Bakre and B. Badrinath Indirect TCP for Mobile Hosts ICDCS 1995
- [BK01] J. Byers and G. Kwon STAIR: Practical AIMD Multirate Multicast Congestion Control NGC 2001
- [BM02] D. Barman and I. Matta Effectiveness of Loss Labeling in Improving TCP Performance in Wired/Wireless Networks ICNP 2002
- [BP95] L. Brakmo and L. Peterson TCP Vegas: End to End Congestion Avoidance on a Global Internet JSAC 1995
- [BSK95] H. Balakrishnan, S. Seshan, and R. Katz Improving Reliable Transport and Handoff Performance in Cellular Wireless Networks ACM Wireless Networks 1995
- [BV99] S. Biaz and N. Vaidya Distinguishing Congestion Losses from Wireless Losses using Inter-Arrival Times at the Receiver ASSET 1999
- [C098] J. Crowcroft and P. Oechslin Differentiated end-to-end Internet services using weighted proportionally fair sharing TCP ACM CCR 1998
- [DKS90] A. Demers, S. Keshav and S. Shenker Analysis and Simulation of a Fair Queuing Algorithm. Internetworking: Research and Experience 1990
- [FJ93] S. Floyd and V. Jacobson Random Early Detection Gateways for Congestion Avoidance ToN 1993
- [F03] S. Floyd Internet Draft: HighSpeed TCP for large Congestion Windows 2003
- [GBMRDZ03] M. Guirguis, A. Bestavros I. Matta, N. Riga, G. Daimant and Y. Zhang Providing Soft Bandwidth Guarantees Using Elastic TCP-based Tunnels ISCC 2004
- [GBM03] M. Guirguis, A. Bestavros and I. Matta XQM: eXogenous-loss aware Queue Management ICNP 2003 Poster
- [GBM04] M. Guirguis, A. Bestavros and I. Matta Exploiting the Transients of Adaptation for RoQ Attacks on Internet Resources BU-TR 2004
- [HMTG101] C. Hollot,V. Misra, D. Towsley and W. Gong A Control Theoretic Analysis of RED INFCOM 2001
- [HMTG201] C. Hollot,V. Misra, D. Towsley and W. Gong On Designing Improved Controller for AQM Routers Supporting TCP Flows INFOCOM 2001
- [JWL04] C. Jin, D. Wei and S. Low FAST TCP: Motivation, Architecture, Algorithms, Performance INFOCOM 2004
- [KHR02] D. Katabi, M. Handley, and C. Rohrs Congestion Control for High Bandwidth-Delay Networks SIGCOMM 2002
- [K97] F. Kelly Charging and rate control for elastic traffic EToT 1997
- [K02] T. Kelly Scaleable TCP: Improving Performance in Highspeed Wide Area Networks 2002
- [kLB99] T. Kim, S. Lu and V. Bharghavan Improving Congestion Control Performance through Loss Differentiation ICCCN 1999

References (2/2)

Computer Science

- [KMT98] F. Kelly, A. Maulloo and D. Tan Rate control for communication networks: shadow prices, proportional fairness and stability J-ORS 1998
- [K99] F. Kelly Mathematical modeling of the Internet ICIAM 1999
- [KM99] P. Key and L. Massoulie User Policies in a network implementing Congestion Pricing ISQE 1999
- [KK03] A. Kuzmanovic and E. Knightly Low-Rate TCP-Targeted Denial of Service Attacks (The Shrew vs. the Mice and Elephants) SIGCOMM 2003
- [KS03] S. Kunniyur and R. Srikant End-to-End Congestion Control Schemes: Utility Function, Random Losses and ECN Marks ToN 2003
- [LM97] D. Lin and R. Morris Dynamics of Random Early Detection SIGCOMM 1997
- [LPW01] S. Low, F. Paganini, L. Wang Understanding TCP Vegas: A Duality Model SIGMETRICS 2001
- [LPWAD02] S. Low, F. Paganini, J. Wang, S. Adlakha and J. Doyle Dynamics of TCP/RED and Scalable Control INFOCOM 2002
- [MJV96] S. McCanne, V. Jacobson and M. Vetterli Receiver-driven Layered Multicast SIGCOMM 1996
- [MV95] J. Mackie-Mason and H. Varian Pricing congestible network resources IEEE JSAC 1995
- [MGT00] V. Misra, W. Gong and D. Towsley Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED SIGCOMM 2000
- [MW00] J. Mo and J. Walrand Fair End-to-End Window Based Congestion Control ToN 2000
- [O97] A. Odlyzko A modest proposal for preventing Internet Congestion 1997
- [PG93] A. Parekh and R. Gallanger A Generalized Processor Sharing Approach to Flow Control in Integrated Service Networks: The Single Node Case. ToN 1993
- [RFD01] K. Ramakrishnan, S. Floyd and D. Black The addition of Explicit Congestion Notification (ECN) to IP 2001
- [S95] S. Shenker Fundamental design issues for the future Internet IEEE JSAC 1995
- [SSZ98] I. Stoica, S. Shenker and H. Zhang Core-Stateless Fair Queuing: A Scalable Architecture to Approximate Fair Bandwidth Allocations in High Speed Networks SIGCOMM 1998
- [SP91] Applied Nonlinear Control J. Slotine and W. Li Prentice Hall