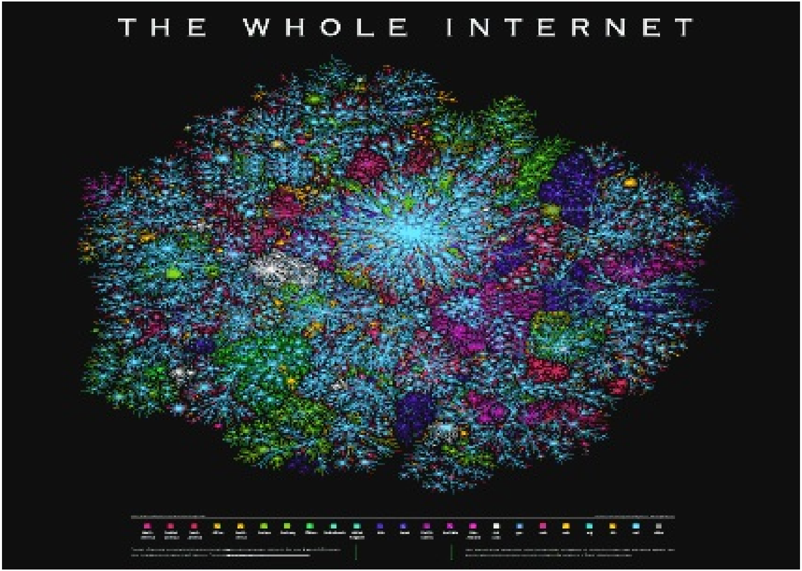
Optimization models for congestion control with multipath routing in TCP/IP networks

Roberto Cominetti
Cristóbal Guzmán

Departamento de Ingeniería Industrial
Universidad de Chile

Workshop on Optimization, Games and Dynamics

28-29 Novembre 2011, Institut Henri Poincaré, Paris

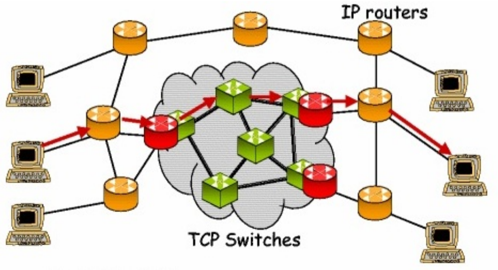

THE WHOLE INTERNET

Overview

1. TCP/IP communication protocols
2. Congestion control and network utility maximization
3. Congestion control with Markovian multipath routing

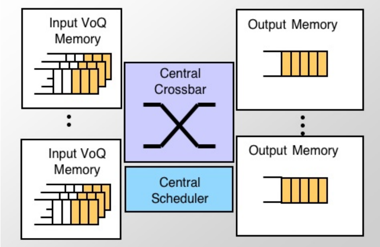
TCP/IP – Single path routing

- Communication network $G = (N, A)$
- Each source $s \in S$ transmits packets from origin o_s to destination d_s
- At which rate? Along which route?



Congestion measures: link delays / packet loss

Switch/Router



- Links have random delays $\tilde{\lambda}_a = \lambda_a + \epsilon_a$ with $\mathbb{E}(\epsilon_a) = 0$

$$\tilde{\lambda}_a = \text{Queuing} + \text{Transmission} + \text{Propagation}$$

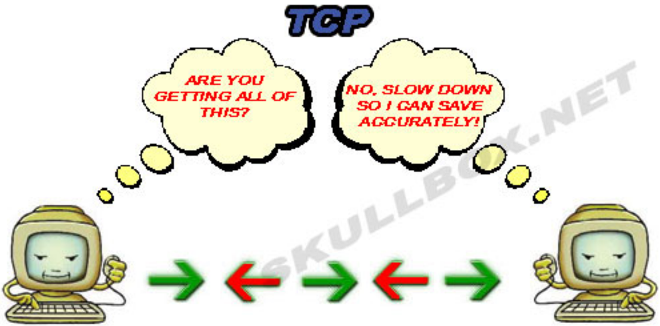
- Finite queuing buffers $\Rightarrow$ packet loss probability p_a

TCP/IP – Current protocols

- **Route selection** (RIP/OSPF/IGRP/BGP/EGP)
  Dynamic adjustment of routing tables
  Slow timescale evolution (15-30 seconds)
  Network Layer 3

- **Rate control** (TCP Reno/Tahoe/Vegas)
  Dynamic adjustment of source rates – congestion window
  Fast timescale evolution (100-300 milliseconds)
  Transport Layer 4

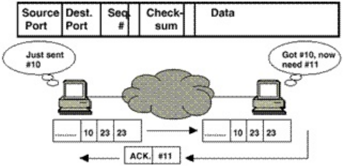
TCP – Congestion window control

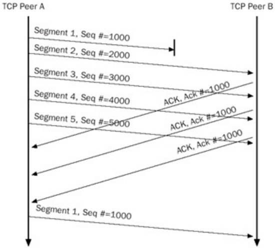


Packets $\longrightarrow$ $\longleftarrow$ Acks

TCP – Congestion window control

Packets $\longleftrightarrow$ Acks





$$x_s = \text{source rate} \sim \frac{\text{congestion window}}{\text{round-trip time}} = \frac{W_s}{\tau_s}$$

TCP – Congestion control

Sources adjust transmission rates in response to congestion

Basic principle: higher congestion $\Leftrightarrow$ smaller rates

x_s : source transmission rate [packets/sec]

λ_a : link congestion measure (loss pbb, queuing delay)

$y_a = \sum_{s \ni a} x_s$ (aggregate link loads)

$q_s = \sum_{a \in s} \lambda_a$ (end-to-end congestion)

Decentralized algorithms

$$\boxed{\begin{aligned} x_s^{t+1} &= F_s(x_s^t, q_s^t) \quad \text{(TCP – source dynamics)} \\ \lambda_a^{t+1} &= G_a(\lambda_a^t, y_a^t) \quad \text{(AQM – link dynamics)} \end{aligned}}$$

TCP – Congestion control

Sources adjust transmission rates in response to congestion

Basic principle: higher congestion $\Leftrightarrow$ *smaller rates*

x_s : source transmission rate [packets/sec]

λ_a : link congestion measure (loss pbb, queuing delay)

$$y_a = \sum_{s \ni a} x_s \qquad \text{(aggregate link loads)}$$

$$q_s = \sum_{a \in s} \lambda_a \qquad \text{(end-to-end congestion)}$$

Decentralized algorithms

$$\boxed{\begin{array}{rcll} x_s^{t+1} & = & F_s(x_s^t, q_s^t) & \text{(TCP – source dynamics)} \\ \lambda_a^{t+1} & = & G_a(\lambda_a^t, y_a^t) & \text{(AQM – link dynamics)} \end{array}}$$

TCP – Congestion control

Sources adjust transmission rates in response to congestion

Basic principle: higher congestion $\Leftrightarrow$ *smaller rates*

x_s : source transmission rate [packets/sec]

λ_a : link congestion measure (loss pbb, queuing delay)

$$y_a = \sum_{s \ni a} x_s \qquad \text{(aggregate link loads)}$$

$$q_s = \sum_{a \in s} \lambda_a \qquad \text{(end-to-end congestion)}$$

Decentralized algorithms

$$\boxed{\begin{array}{rcll} x_s^{t+1} & = & F_s(x_s^t, q_s^t) & \text{(TCP – source dynamics)} \\ \lambda_a^{t+1} & = & G_a(\lambda_a^t, y_a^t) & \text{(AQM – link dynamics)} \end{array}}$$

Example: TCP-Reno / packet loss probability

AIMD control

$$W_s^{t+\tau_s} = \begin{cases} W_s^t + 1 & \text{if } W_s^t \text{ packets are successfully transmitted} \\ \lceil W_s^t/2 \rceil & \text{one or more packets are lost (duplicate ack's)} \end{cases}$$

$$\pi_s^t = \prod_{a \in s}(1 - p_a^t) = \text{success probability (per packet)}$$

Additive congestion measure

$$\left.\begin{array}{l} q_s^t \triangleq -\ln(\pi_s^t) \\ \lambda_a^t \triangleq -\ln(1 - p_a^t) \end{array}\right\} \Rightarrow q_s^t = \sum_{a \in s} \lambda_a^t$$

Approximate model for rate dynamics

$$\mathbb{E}(W_s^{t+\tau_s} | W_s^t) \sim e^{-q_s^t W_s^t}(W_s^t + 1) + (1 - e^{-q_s^t W_s^t})\lceil W_s^t/2 \rceil$$

$$\Rightarrow \boxed{x_s^{t+1} = x_s^t + \frac{1}{2\tau_s}\left[e^{-\tau_s q_s^t x_s^t}(x_s^t + \tfrac{2}{\tau_s}) - x_s^t\right]}$$

Example: TCP-Reno / packet loss probability

AIMD control

$$W_s^{t+\tau_s} = \begin{cases} W_s^t + 1 & \text{if } W_s^t \text{ packets are successfully transmitted} \\ \lceil W_s^t/2 \rceil & \text{one or more packets are lost (duplicate ack's)} \end{cases}$$

$$\pi_s^t = \prod_{a \in s}(1 - p_a^t) = \text{success probability (per packet)}$$

Additive congestion measure

$$\left.\begin{array}{l} q_s^t \triangleq -\ln(\pi_s^t) \\ \lambda_a^t \triangleq -\ln(1 - p_a^t) \end{array}\right\} \Rightarrow q_s^t = \sum_{a \in s} \lambda_a^t$$

Approximate model for rate dynamics

$$\mathbb{E}(W_s^{t+\tau_s} | W_s^t) \sim e^{-q_s^t W_s^t}(W_s^t + 1) + (1 - e^{-q_s^t W_s^t}) \lceil W_s^t/2 \rceil$$

$$\Rightarrow \boxed{x_s^{t+1} = x_s^t + \frac{1}{2\tau_s}\left[e^{-\tau_s q_s^t x_s^t}(x_s^t + \tfrac{2}{\tau_s}) - x_s^t\right]}$$

Example: TCP-Reno / packet loss probability

AIMD control

$$W_s^{t+\tau_s} = \begin{cases} W_s^t + 1 & \text{if } W_s^t \text{ packets are successfully transmitted} \\ \lceil W_s^t/2 \rceil & \text{one or more packets are lost (duplicate ack's)} \end{cases}$$

$$\pi_s^t = \textstyle\prod_{a\in s}(1-p_a^t) = \text{success probability (per packet)}$$

Additive congestion measure

$$\left.\begin{array}{l} q_s^t \triangleq -\ln(\pi_s^t) \\ \lambda_a^t \triangleq -\ln(1-p_a^t) \end{array}\right\} \Rightarrow q_s^t = \textstyle\sum_{a\in s} \lambda_a^t$$

Approximate model for rate dynamics

$$\mathbb{E}(W_s^{t+\tau_s}|W_s^t) \sim e^{-q_s^t W_s^t}(W_s^t + 1) + (1 - e^{-q_s^t W_s^t})\lceil W_s^t/2 \rceil$$

$$\Rightarrow \boxed{x_s^{t+1} = x_s^t + \frac{1}{2\tau_s}\left[e^{-\tau_s q_s^t x_s^t}(x_s^t + \tfrac{2}{\tau_s}) - x_s^t\right]}$$

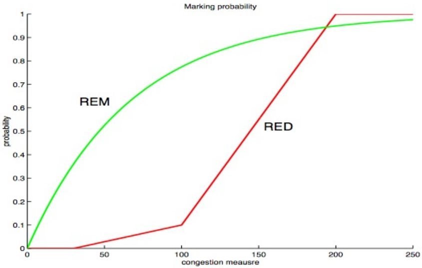
Example: AQM / Droptail $\longrightarrow$ RED-REM

Marking probability on links controlled by AQM

$$p_a^t = \varphi_a(r_a^t)$$

as a function of the link's average queue length

$$r_a^{t+1} = (1-\alpha) r_a^t + \alpha L_a^t$$



Loss probability vs. average queue length

Network Utility Maximization

- Kelly, Maullo and Tan (1998) proposed an optimization-based model for distributed rate control in networks.
- Low, Srikant, etc. (1999-2002) showed that current TCP/AQM control algorithms solve an implicit network optimization problem.
- During last decade, the model has been used and extended to study the performance of wired and wireless networks.

Steady state equations

$$\begin{aligned} x_s^{t+1} &= F_s(x_s^t, q_s^t) && \text{(TCP – source dynamics)} \\ \lambda_a^{t+1} &= G_a(\lambda_a^t, y_a^t) && \text{(AQM – link dynamics)} \end{aligned}$$

Steady state equations

$$\begin{aligned} x_s &= F_s(x_s, q_s) \qquad &\text{(TCP – source equilibrium)} \\ \lambda_a &= G_a(\lambda_a, y_a) \qquad &\text{(AQM – link equilibrium)} \end{aligned}$$

Steady state equations

$$\begin{aligned} x_s &= F_s(x_s, q_s) && \text{(TCP – source equilibrium)} \\ \lambda_a &= G_a(\lambda_a, y_a) && \text{(AQM – link equilibrium)} \end{aligned}$$

$$\Updownarrow$$

$$\boxed{\begin{array}{ll} x_s = f_s(q_s) & \text{(decreasing)} \\ \lambda_a = \psi_a(y_a) & \text{(increasing)} \\ q_s = \sum_{a \in s} \lambda_a & \\ y_a = \sum_{s \ni a} x_s & \end{array}}$$

Steady state equations

$$\begin{array}{rcll} x_s & = & F_s(x_s, q_s) & \quad \text{(TCP – source equilibrium)} \\ \lambda_a & = & G_a(\lambda_a, y_a) & \quad \text{(AQM – link equilibrium)} \end{array}$$

$$\Updownarrow$$

$$\boxed{\begin{array}{ll} x_s = f_s(q_s) & \quad \text{(decreasing)} \\ \lambda_a = \psi_a(y_a) & \quad \text{(increasing)} \\ q_s = \sum_{a\in s} \lambda_a & \\ y_a = \sum_{s\ni a} x_s & \end{array}} \Leftrightarrow \boxed{\begin{array}{l} x_s = f_s(\sum_{a\in s} \lambda_a) \\ \lambda_a = \psi_a(\sum_{s\ni a} x_s) \end{array}}$$

Examples

$\boxed{\text{TCP-Reno}}$ (loss probability)

$$
\begin{aligned}
q_s &= f_s^{-1}(x_s) &\triangleq& \ \frac{1}{\tau_s x_s}\ln\left(1+\frac{2}{\tau_s x_s}\right)\\
\lambda_a &= \psi_a(y_a) &\triangleq& \ \frac{\delta y_a}{c_a - y_a}
\end{aligned}
$$

$\boxed{\text{TCP-Vegas}}$ (queueing delay)

$$
\begin{aligned}
q_s &= f_s^{-1}(x_s) &\triangleq& \ \frac{\alpha\tau_s}{x_s}\\
\lambda_a &= \psi_a(y_a) &\triangleq& \ \frac{y_a}{c_a - y_a}
\end{aligned}
$$

Steady state – Primal optimality

$$\boxed{\begin{aligned} x_s &= f_s(\textstyle\sum_{a\in s}\lambda_a) \\ \lambda_a &= \psi_a(\textstyle\sum_{s\ni a} x_s) \end{aligned}}$$

$$f_s^{-1}(x_s) = \textstyle\sum_{a\in s}\lambda_a = \sum_{a\in s}\psi_a(\sum_{u\ni a} x_u)$$

$\equiv$ optimal solution of strictly convex program

$$(P) \quad \boxed{\min_x \textstyle\sum_{s\in S} U_s(x_s) + \sum_{a\in A}\Psi_a(\sum_{s\ni a} x_s)}$$

$$\begin{aligned} U_s'(\cdot) &= -f_s^{-1}(\cdot) \\ \Psi_a'(\cdot) &= \psi_a(\cdot) \end{aligned}$$

Steady state – Primal optimality

$$\boxed{\begin{aligned} x_s &= f_s(\textstyle\sum_{a\in s}\lambda_a) \\ \lambda_a &= \psi_a(\textstyle\sum_{s\ni a} x_s) \end{aligned}}$$

$$f_s^{-1}(x_s) = \textstyle\sum_{a\in s}\lambda_a = \sum_{a\in s}\psi_a(\sum_{u\ni a} x_u)$$

$\equiv$ optimal solution of strictly convex program

$$(P) \quad \boxed{\min_x \ \textstyle\sum_{s\in S} U_s(x_s) + \sum_{a\in A}\Psi_a(\sum_{s\ni a} x_s)}$$

$$\begin{aligned} U_s'(\cdot) &= -f_s^{-1}(\cdot) \\ \Psi_a'(\cdot) &= \psi_a(\cdot) \end{aligned}$$

Steady state – Dual optimality

$$\boxed{\begin{aligned} x_s &= f_s(\textstyle\sum_{a\in s}\lambda_a) \\ \lambda_a &= \psi_a(\textstyle\sum_{s\ni a} x_s) \end{aligned}}$$

$$\psi_a^{-1}(\lambda_a) = \textstyle\sum_{s\ni a} x_s = \sum_{s\ni a} f_s(\sum_{b\in s}\lambda_b)$$

$\equiv$ optimal solution of strictly convex program

$$(D) \quad \boxed{\min_{\lambda} \ \textstyle\sum_{a\in A}\Psi_a^*(\lambda_a) + \sum_{s\in S} U_s^*(\sum_{a\in s}\lambda_a)}$$

Steady state – Dual optimality

$$\boxed{\begin{array}{l} x_s = f_s(\sum_{a\in s}\lambda_a) \\ \lambda_a = \psi_a(\sum_{s\ni a} x_s) \end{array}}$$

$$\psi_a^{-1}(\lambda_a) = \sum_{s\ni a} x_s = \sum_{s\ni a} f_s(\sum_{b\in s}\lambda_b)$$

$\equiv$ optimal solution of strictly convex program

$$(D) \quad \boxed{\min_{\lambda} \ \sum_{a\in A}\Psi_a^*(\lambda_a) + \sum_{s\in S} U_s^*(\sum_{a\in s}\lambda_a)}$$

Theorem (Low'2003)

$$\boxed{\begin{array}{l} x_s = f_s(\sum_{a \in s} \lambda_a) \\ \lambda_a = \psi_a(\sum_{s \ni a} x_s) \end{array}} \Leftrightarrow \boxed{\begin{array}{l} x \text{ and } \lambda \text{ are optimal solutions} \\ \text{for } (P) \text{ and } (D) \text{ respectively} \end{array}}$$

Relevance:

- Reverse engineering of existing protocols / forward engineering (f_s, ψ_a)
- Design distributed stable protocols to optimize prescribed utilities
- Flexible choice of congestion measure q_s

Limitations:

- Ignores delays in transmission of congestion signals
- Improper account of stochastic phenomena
- Single-path routing

Theorem (Low'2003)

$$\boxed{\begin{array}{l} x_s = f_s(\sum_{a\in s} \lambda_a) \\ \lambda_a = \psi_a(\sum_{s\ni a} x_s) \end{array}} \Leftrightarrow \boxed{\begin{array}{l} x \text{ and } \lambda \text{ are optimal solutions} \\ \text{for } (P) \text{ and } (D) \text{ respectively} \end{array}}$$

Relevance:

- Reverse engineering of existing protocols / forward engineering (f_s, ψ_a)
- Design distributed stable protocols to optimize prescribed utilities
- Flexible choice of congestion measure q_s

Limitations:

- Ignores delays in transmission of congestion signals
- Improper account of stochastic phenomena
- Single-path routing

Theorem (Low'2003)

$$\boxed{\begin{array}{l} x_s = f_s(\sum_{a\in s}\lambda_a) \\ \lambda_a = \psi_a(\sum_{s\ni a} x_s) \end{array}} \Leftrightarrow \boxed{\begin{array}{l} x \text{ and } \lambda \text{ are optimal solutions} \\ \text{for } (P) \text{ and } (D) \text{ respectively} \end{array}}$$

Relevance:

- Reverse engineering of existing protocols / forward engineering (f_s, ψ_a)
- Design distributed stable protocols to optimize prescribed utilities
- Flexible choice of congestion measure q_s

Limitations:

- Ignores delays in transmission of congestion signals
- Improper account of stochastic phenomena
- Single-path routing

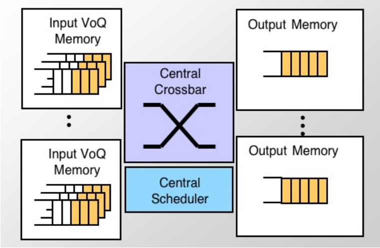
Markovian Network Utility Maximization (MNUM)

- Increase transmission rates: single path $\longrightarrow$ multi-path
- Goal: design distributed TCP protocols with multi-path routing
- Packet-level protocol that is stable and satisfies optimality criteria
- Model based on the notion of Markovian traffic equilibrium

MNUM: integrated routing & rate control

- Cross-layer design: routing + rate control
- Based on a common congestion measure: delay
- Link random delays $\tilde{\lambda}_a = \lambda_a + \epsilon_a$ with $\mathbb{E}(\epsilon_a) = 0$

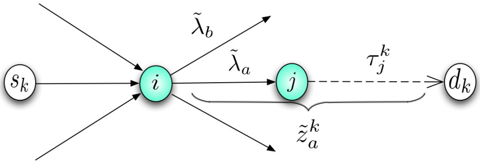
$$\tilde{\lambda}_a = \text{Queuing} + \text{Transmission} + \text{Propagation}$$



MNUM: Markovian multipath routing

At switch i, packets headed to destination d are routed through the outgoing link $a \in A_i^+$ that minimizes the *"observed"* cost-to-go

$$\boxed{\tilde{\tau}_i^d = \min_{a \in A_i^+} \underbrace{\tilde{\lambda}_a + \tau_{j_a}^d}_{\tilde{z}_a^d}}$$



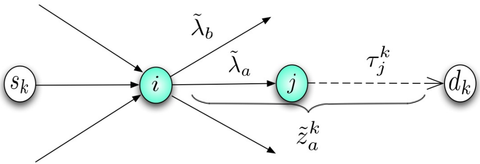
Markov chain with transition matrix

$$P_{ij}^d = \begin{cases} \mathbb{P}(\tilde{z}_a^d \leq \tilde{z}_b^d, \forall b \in A_i^+) & \text{if } i = i_a, j = j_a \\ 0 & \text{otherwise} \end{cases}$$

MNUM: Markovian multipath routing

At switch i, packets headed to destination d are routed through the outgoing link $a \in A_i^+$ that minimizes the *"observed"* cost-to-go

$$\boxed{\tilde{\tau}_i^d = \min_{a \in A_i^+} \underbrace{\tilde{\lambda}_a + \tau_{j_a}^d}_{\tilde{z}_a^d}}$$



Markov chain with transition matrix

$$P_{ij}^d = \begin{cases} \mathbb{P}(\tilde{z}_a^d \leq \tilde{z}_b^d, \forall b \in A_i^+) & \text{if } i = i_a,\ j = j_a \\ 0 & \text{otherwise} \end{cases}$$

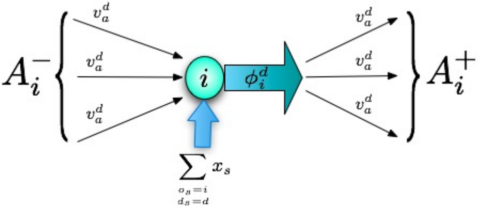
Expected flows (invariant measures)

The flow ϕ_i^d entering node i and directed towards d

$$\boxed{\phi_i^d = \sum_{\substack{o_s=i\\ d_s=d}} x_s + \sum_{a\in A_i^-} v_a^d}$$

splits among the outgoing links $a=(i,j)$ according to

$$\boxed{v_a^d = \phi_i^d P_{ij}^d}$$



Expected costs

Letting $z_a^d = \mathbb{E}(\tilde{z}_a^d)$ and $\tau_i^d = \mathbb{E}(\tilde{\tau}_i^d)$, we have

$$\begin{aligned} z_a^d &= \lambda_a + \tau_{j_a}^d \\ \tau_i^d &= \varphi_i^d(z^d) \end{aligned}$$

with

$$\boxed{\varphi_i^d(z^d) \triangleq \mathbb{E}(\min_{a \in A_i^+}[z_a^d + \epsilon_a^d])}$$

Moreover

$$\mathbb{P}\left(\tilde{z}_a^d \leq \tilde{z}_b^d, \forall b \in A_i^+\right) = \frac{\partial \varphi_i^d}{\partial z_a^d}(z^d)$$

Expected costs

Letting $z_a^d = \mathbb{E}(\tilde{z}_a^d)$ and $\tau_i^d = \mathbb{E}(\tilde{\tau}_i^d)$, we have

$$\begin{aligned} z_a^d &= \lambda_a + \tau_{j_a}^d \\ \tau_i^d &= \varphi_i^d(z^d) \end{aligned}$$

with

$$\boxed{\varphi_i^d(z^d) \triangleq \mathbb{E}(\min_{a \in A_i^+}[z_a^d + \epsilon_a^d])}$$

Moreover

$$\mathbb{P}\left(\tilde{z}_a^d \leq \tilde{z}_b^d, \forall b \in A_i^+\right) = \frac{\partial \varphi_i^d}{\partial z_a^d}(z^d)$$

Markovian NUM – Definition

$$
\begin{array}{ll}
x_s = f_s(q_s) & \text{(source rate control)} \\
\lambda_a = \psi_a(y_a) & \text{(link congestion)} \\
y_a = \sum_d v_a^d & \text{(total link flows)} \\
q_s = \tau_s - \tau_s^0 & \text{(end-to-end queuing delay)}
\end{array}
$$

where $\tau_s = \tau_{o_s}^{d_s}$ with expected costs given by

$$
(ZQ) \quad \begin{cases} z_a^d = \lambda_a + \tau_{j_a}^d \\ \tau_i^d = \varphi_i^d(z^d) \end{cases}
$$

and expected flows v^d satisfying

$$
(FC) \quad \begin{cases} \phi_i^d = \sum_{\substack{o_s=i \\ d_s=d}} x_s + \sum_{a \in A_i^-} v_a^d & \forall i \neq d \\ v_a^d = \phi_i^d \frac{\partial \varphi_i^d}{\partial z_a^d}(z^d) & \forall a \in A_i^+ \end{cases}
$$

MNUM Characterization: Dual problem

- (ZQ) defines implicitly z_a^d and τ_i^d as concave functions of λ
- $x_s = f_s(q_s)$ with $q_s = \tau_{o_s}^{d_s}(\lambda) - \tau_{o_s}^{d_s}(\lambda^0)$ yields x_s as a function of λ
- (FC) then defines v_a^d as functions of λ

$$\text{MNUM conditions} \quad \Leftrightarrow \quad \psi_a^{-1}(\lambda_a) = y_a = \textstyle\sum_d v_a^d(\lambda)$$

Theorem

MNUM $\Leftrightarrow$ optimal solution of the strictly convex program

$$(D) \quad \min_{\lambda} \sum_{a \in A} \Psi_a^*(\lambda_a) + \sum_{s \in S} U_s^*(q_s(\lambda))$$

MNUM Characterization: Primal problem

Theorem

MNUM $\Leftrightarrow$ *optimal solution of*

$$\min_{(x,y,v)\in P} \sum_{s\in S} U_s(x_s) + \sum_{a\in A} \Psi_a(y_a) + \sum_{d\in D} \chi^d(v^d)$$

where

$$\chi^d(v^d) = \sup_{z^d} \sum_{a\in A} (\varphi^d_{i_a}(z^d) - z^d_a) v^d_a$$

and P is the polyhedron defined by flow conservation constraints.

SUMMARY

- Described an optimization model for TCP/IP equilibrium rates
- Model extended to multipath routing & rate control (MNUM)
- Inspired from packet-level distributed protocols
- Implementable under current TCP/IP standards

FUTURE WORK

- Simulation and testing of MNUM-based protocols
- Investigate stochastic-stability of protocols
- Investigate delay-stability of protocols
- ECN mechanisms for congestion signals

SUMMARY

- Described an optimization model for TCP/IP equilibrium rates
- Model extended to multipath routing & rate control (MNUM)
- Inspired from packet-level distributed protocols
- Implementable under current TCP/IP standards

FUTURE WORK

- Simulation and testing of MNUM-based protocols
- Investigate stochastic-stability of protocols
- Investigate delay-stability of protocols
- ECN mechanisms for congestion signals

Some references

- F. Kelly, A. Maulloo, D. Tan: *Rate control for communication networks: Shadow prices, proportional fairness and stability*, Journal of Operation Research, (1998).
- H. Yaïche, R. Mazumdar and C. Rosenberg: *A game theoretic framework for bandwidth allocation and pricing in broadband networks*, IEEE/ACM Transactions on Networking, (2000).
- M. Chiang, S. H. Low, A. R. Calderbank, J. C. Doyle: *Layering as optimization decomposition*, Proceedings of IEEE, (2006).
- J. B. Baillon, R. Cominetti: *Markovian traffic equilibrium*, Mathematical Programming (2007).
- R. Cominetti, C. Guzmán: *Network congestion control with Markovian multipath routing*, Submitted (2011).