Internet Protocols

Steven Low

CS/EE
netlab.CALTECH.edu
July 2004

with J. Doyle, L. Li, A. Tang, J. Wang
Internet Protocols

- Application: Selects user criteria, web layout, utility
- TCP/AQM: Selects source transmission rates
- IP: Selects paths from sources to dests
- Link: Selects topology, capacities, power,...
Internet Protocols

- Protocols determine network behavior
- Critical, yet difficult, to understand and optimize
- Local algorithms, distributed spatially and vertically → global behavior
- Designed separately, deployed asynchronously, evolves independently
Internet Protocols

Protocols determine network behavior

Critical, yet difficult, to understand and optimize

Local algorithms, distributed spatially and vertically

Designed separately, deployed asynchronously, evolves independently

Need to reverse engineer
... much easier than biology with full specs
Internet Protocols

- Minimize path costs (IP)
- Maximize utility (TCP/AQM)
- Minimize response time (web layout)
- Minimize SIR, max capacities, ...

Application
TCP/AQM
IP
Link
Internet Protocols

- Each layer is abstracted as an optimization problem
- Operation of a layer is a distributed solution
- Results of one problem (layer) are parameters of others
- Operate at different timescales

$$\max_{x \geq 0} \sum_{i} U_i(x_i)$$

subject to $Rx \leq c$
A Theory for the Internet?

General Approach:

1) Understand a single layer in isolation and assume other layers are designed nearly optimally.

2) Understand interactions across layers

3) Incorporate additional layers, with the ultimate goal of viewing entire protocol stack as solving one giant optimization problem (where individual layers are solving parts of it).
Outline

Applications

TCP/AQM

IP

Link

Carlson: HOT traffic & web layout

Low: equilibrium
Paganini: dynamics

Low: routing

Willinger: HOT topology
Outline

Applications

TCP/AQM

IP

Challenges

Details

Summary
Network model

\[ R_{li} = 1 \text{ if source } i \text{ uses link } l \]

\[ x(t+1) = F\left(R^T p(t), x(t)\right) \]

\[ p(t+1) = G\left(p(t), Rx(t)\right) \]
Protocol decomposition

TCP algorithms maximize utility with different utility functions

TCP-AQM

$$\max_{x \geq 0} \sum_i U_i(x_i)$$

subject to $Rx \leq c$ (Kelly, Malloo, Tan 98)

Congestion prices coordinate across protocol layers
TCP algorithms maximize utility with different utility functions.

IP shortest path routing is optimal using congestion prices as link costs, with given link capacities $c$.

Congestion prices coordinate across protocol layers.
TCP algorithms maximize utility with different utility functions.

IP shortest path routing is optimal using congestion prices as link costs, with given link capacities $c$.

With optimal provisioning, static routing is optimal using provisioning cost $\alpha$ as link costs.

Congestion prices coordinate across protocol layers.
Protocol decomposition – TCP/AQM

TCP-AQM

\[
\max_{x \geq 0} \sum_i U_i(x_i)
\]

subject to \( Rx \leq c \)

TCP/AQM:
- TCP maximizes aggregate utility (not throughput)
- Fair bandwidth allocation is not always inefficient
- Increasing capacity does not always raise throughput

Intricate network interactions → paradoxical behavior
Protocol decomposition – TCP/IP

\[
\begin{align*}
\max_R & \quad \max_{x \geq 0} \sum_i U_i(x_i) \\
\text{subject to} & \quad Rx \leq c
\end{align*}
\]

TCP/IP (fixed \( c \)):
- Equilibrium exists iff zero duality gap
- NP-hard, but subclass with zero duality gap is LP
- Equilibrium, if exists, can be unstable
- Can stabilize, but with reduced utility

Inevitable tradeoff bw utility max & routing stability
Outline

Applications

TCP/AQM

IP

TCP/IP

Summary

Details

TCP

TCP/IP

Challenges
Duality model

- Flow control problem (Kelly, Malloo, Tan 98)

\[ \max_{x \geq 0} \sum_{i} U_i(x_i) \]
\[ \text{s. t. } Rx \leq c \]

- Primal-dual algorithm

\[ x(t + 1) = F(x, R^Tp) \]
\[ p(t + 1) = G(p, Rx) \]

- TCP/AQM
  - Maximize utility with different utility functions
  - \((L 03)\): \((x^*, p^*)\) primal-dual optimal iff
\[ y_i^* \leq c_i \text{ with equality if } p_i^* > 0 \]
Duality model

- Historically, packet level implemented first
- Flow level understood as after-thought
- But flow level design determines
  - performance, fairness, stability

\[
U_i(x_i) = \begin{cases} 
    (1 - \alpha)x_i^{1-\alpha} & \text{if } \alpha \neq 1 \\
    \log x_i & \text{if } \alpha = 1
\end{cases}
\]

- \(\alpha = 1\) : Vegas, FAST, STCP
- \(\alpha = 1.2\) : HSTCP (homogeneous sources)
- \(\alpha = 2\) : Reno (homogeneous sources)
- \(\alpha = \text{infinity}\) : XCP (single link only)
Duality model

- Historically, packet level implemented first
- Flow level understood as after-thought
- But flow level design determines
  - performance, fairness, stability

Now

- Given (application) utility functions, can generate provably scalable TCP algorithms (Paganini)
Questions

- Is fair allocation always inefficient
- Does raising capacity always increase throughput

Intricate and surprising interactions in network ...
... unlike at single-link ......
Questions

- Is fair allocation always inefficient?
- Does raising capacity always increase throughput?
Fairness

\[
\begin{align*}
\max_{x \geq 0} & \quad \sum_{i} U_i(x_i) \\
\text{s. t.} & \quad Rx \leq c
\end{align*}
\]

\[
U_i(x_i) = \begin{cases} 
(1 - \alpha)x_i^{1-\alpha} & \text{if } \alpha \neq 1 \\
\log x_i & \text{if } \alpha = 1
\end{cases}
\]

(Mo, Walrand 00)

- Identify allocation with \( \alpha \)
- An allocation is fairer if its \( \alpha \) is larger
Fairness

\[
\max_{x \geq 0} \sum_i U_i(x_i)
\]

s. t. \( Rx \leq c \)

\[
U_i(x_i) = \begin{cases} 
(1 - \alpha)x_i^{1-\alpha} & \text{if } \alpha \neq 1 \\
\log x_i & \text{if } \alpha = 1 
\end{cases}
\]

(Mo, Walrand 00)

- \( \alpha = 0 \): maximum throughput
- \( \alpha = 1 \): proportional fairness
- \( \alpha = 2 \): min delay fairness (Reno)
- \( \alpha = \text{infinity} \): maxmin fairness
Fairness

\[
\max_{x \geq 0} \sum_{i} U_i(x_i)
\]

s. t. \( Rx \leq c \)

\[
U_i(x_i) = \begin{cases} 
(1 - \alpha)x_i^{1-\alpha} & \text{if } \alpha \neq 1 \\
\log x_i & \text{if } \alpha = 1 
\end{cases}
\]

(Mo, Walrand 00)

- \( \alpha = 1 \) : Vegas, FAST, STCP
- \( \alpha = 1.2 \) : HSTCP (homogeneous sources)
- \( \alpha = 2 \) : Reno (homogeneous sources)
- \( \alpha = \infty \) : XCP (single link only)
Efficiency

\[
\max_{x \geq 0} \sum_{i} U_i(x_i)
\]
\[
\text{s. t. } Rx \leq c
\]

\[
U_i(x_i) = \begin{cases} 
(1 - \alpha)x_i^{1-\alpha} & \text{if } \alpha \neq 1 \\
\log x_i & \text{if } \alpha = 1 
\end{cases}
\]

- Unique optimal rate \( x(\alpha) \)
- An allocation is **efficient** if \( T(\alpha) \) is large

throughput \( T(\alpha) := \sum_i x_i(\alpha) \)
Conjecture

\[
\max_{x \geq 0} \sum_i U_i(x_i)
\]

s. t. \( Rx \leq c \)

\[
U_i(x_i) = \begin{cases} 
(1 - \alpha)x_i^{1-\alpha} & \text{if } \alpha \neq 1 \\
\log x_i & \text{if } \alpha = 1 
\end{cases}
\]

Conjecture

\( T(\alpha) \) is nonincreasing

i.e. a fair allocation is always inefficient
Example 1

Conjecture

$T(\alpha)$ is nonincreasing

i.e. a fair allocation is always inefficient
**Example 2**

\[ T(1) = \frac{2}{3} \left( c_1 + c_2 + \sqrt{c_1^2 + c_2^2} - c_1 c_2 \right) \]

\[ T(\infty) = c_1 + \frac{c_2}{2} \]

\[ \Rightarrow \quad T(1) > T(\infty) \]

**Conjecture**

\[ T(\alpha) \] is nonincreasing

i.e. a fair allocation is always inefficient
Example 3

Conjecture

$T(\alpha)$ is nonincreasing

i.e. a fair allocation is always inefficient
Intuition

“The fundamental conflict between achieving flow fairness and maximizing overall system throughput….. The basic issue is thus the trade-off between these two conflicting criteria.”

Luo, etc. (2003), ACM MONET
Results

- **Theorem**: Necessary & sufficient condition for **general** networks $(R, c)$ provided every link has a 1-link flow

- **Corollary 1**: true if $N(R)=1$

\[
\frac{1}{L^{1/\alpha} + 1} + \frac{L^{1/\alpha}}{L^{1/\alpha} + 1} = c_{l} = 1
\]
Results

- **Theorem**: Necessary & sufficient condition for general networks \((R, c)\) provided every link has a 1-link flow.

- **Corollary 2**: true if
  - \(N(R) = 2\)
  - 2 long flows pass through same# links
Counter-example

- There exists a network such that $dT/d\alpha > 0$ for almost all $\alpha > 0$

Intuition
- Large $\alpha$ favors expensive flows
- Long flows may not be expensive

Max-min may be more efficient than proportional fairness
Counter-example

- **Theorem**: Given any $\alpha_0 > 0$, there exists a network where

\[ \frac{dT}{d\alpha} > 0 \quad \text{for all } \alpha > \alpha_0 \]

- Compact example
Questions

- Is fair allocation always inefficient?
- Does raising capacity always increase throughput?

Intricate and surprising interactions in network... unlike at single-link......
Throughput & capacity

- **Intuition:** Increasing link capacities always raises throughput $T$

- **Theorem:** Necessary & sufficient condition for general networks $(R, c)$

- **Corollary:** For all $\alpha$, increasing
  - a link’s capacity can reduce $T$
  - all links’ capacities equally can reduce $T$
  - all links’ capacities proportionally raises $T$
Motivation

Primal: \[ \max_{x \geq 0} \sum U_i(x_i) \quad \text{subject to } Rx \leq c \]

Dual: \[ \min_{p \geq 0} \left( \sum \max_{x_i \geq 0} \left( U_i(x_i) - x_i \sum R_{li} p_l \right) + \sum p_l c_l \right) \]
Motivation

Primal: \[ \max_R \max_{x \geq 0} \sum_i U_i(x_i) \quad \text{subject to } Rx \leq c \]

Dual: \[ \min_{p \geq 0} \left( \sum_i \max_{x_i \geq 0} \left( U_i(x_i) - x_i \max_R \sum_l R_{li} p_l \right) + \sum_l p_l c_l \right) \]

Shortest path routing!

Can TCP/IP maximize utility?
Two timescales

- Instant convergence of TCP/IP
- Link cost = \( a p_l(t) + b d_l \)
- Shortest path routing \( R(t) \)

TCP/AQM

| IP | \( a \ p(0) \) | \( a \ p(1) \) | \( R(0) \) | \( R(1) \) | \( \cdots \ R(t), \ R(t+1) ; \cdots \) | static | price |
Questions

- Does equilibrium routing $R_a$ exist?
- What is utility at $R_a$?
- Is $R_a$ stable?
- Can it be stabilized?

TCP/AQM

<table>
<thead>
<tr>
<th></th>
<th>TCP/AQM</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP</td>
<td></td>
</tr>
</tbody>
</table>

\[ \begin{array}{c}
R(0) \\
\uparrow \\
a \cdot p(0) \\
\downarrow \\
R(1) \\
\uparrow \\
a \cdot p(1) \\
\downarrow \\
\vdots \\
R(t), R(t+1), \ldots
\end{array} \]
Equilibrium routing

Theorem

1. If $b=0$, $R_a$ exists iff zero duality gap
   - Shortest-path routing is optimal with congestion prices
   - No penalty for not splitting

Primal: $\max_{R} \max_{x \geq 0} \sum_{i} U_i(x_i)$ subject to $Rx \leq c$

Dual: $\min_{p \geq 0} \left( \sum_{i} \max_{x_i \geq 0} \left( U_i(x_i) - x_i \max_{R_i} \sum_{l} R_{li}p_l \right) + \sum_{l} p_lc_l \right)$
Equilibrium routing

**Theorem**
1. If $b=0$, $R_a$ exists iff zero duality gap
   - Shortest-path routing is optimal with congestion prices
   - No penalty for not splitting
2. Primal problem is NP-hard
   - Subclass of problems with zero gap is LP

**Proof**
Reduce integer partition to primal problem
Given: integers $\{c_1, ..., c_n\}$
Find: set $A$ s.t. $\sum_{i \in A} c_i = \sum_{i \notin A} c_i$
Theorem

1. If $b=0$, $R_a$ exists iff zero duality gap
   - Shortest-path routing is optimal with congestion prices
   - No penalty for not splitting
2. Primal problem is NP-hard
   - Subclass of problems with zero gap is LP
3. Tradeoff between routing stability and achievable utility
Theorem

1. If \( b=0 \), \( R_a \) exists iff zero duality gap
   - Shortest-path routing is optimal with congestion prices
   - No penalty for not splitting
2. Primal problem is NP-hard
   - Subclass of problems with zero gap is LP
3. Tradeoff between routing stability and achievable utility
4. Static routing is optimal if capacity \( c \) is optimally provisioned
Protocol decomposition

TCP algorithms maximize utility with different utility functions.

IP shortest path routing is optimal using congestion prices as link costs, with given link capacities $c$.

With optimal provisioning, static routing is optimal using provisioning cost $\alpha$ as link costs.

Congestion prices coordinate across protocol layers.
Challenges

- IP
- TCP/AQM
- Applications

- Link
- Separation theorem
- Heterogeneous protocols
- • Duality & complexity
- • Routing dynamics
- • Integration of topology
- ... wireless ad hoc
Challenges

What is the ultimate underlying problem?

Integration of control, communication, computing?
(nonlinear, distributed, delayed, random, ...)

Applications

TCP/AQM

IP

Link
Challenges

Applications

TCP/AQM

IP

Link

Separation theorem

Are these the right questions?
(scalability, evolvability, verifiability, fragility, simplicity, ...)

- Integration of topology
- ... wireless ad hoc