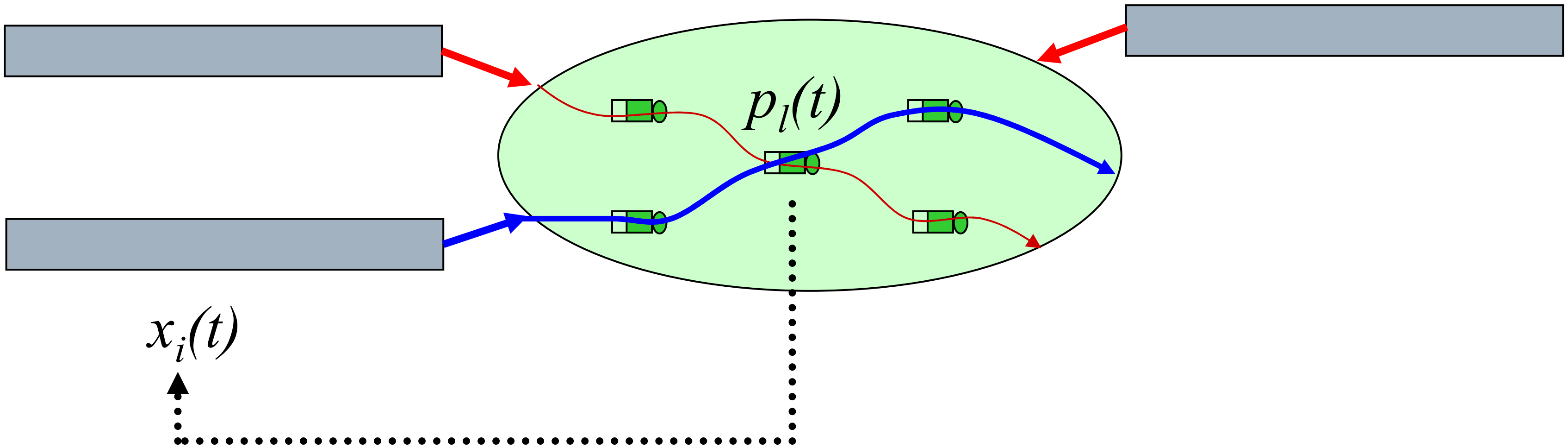
Internet Protocols

Steven Low

CS/EE
netlab.**CALTECH**.edu
July 2004

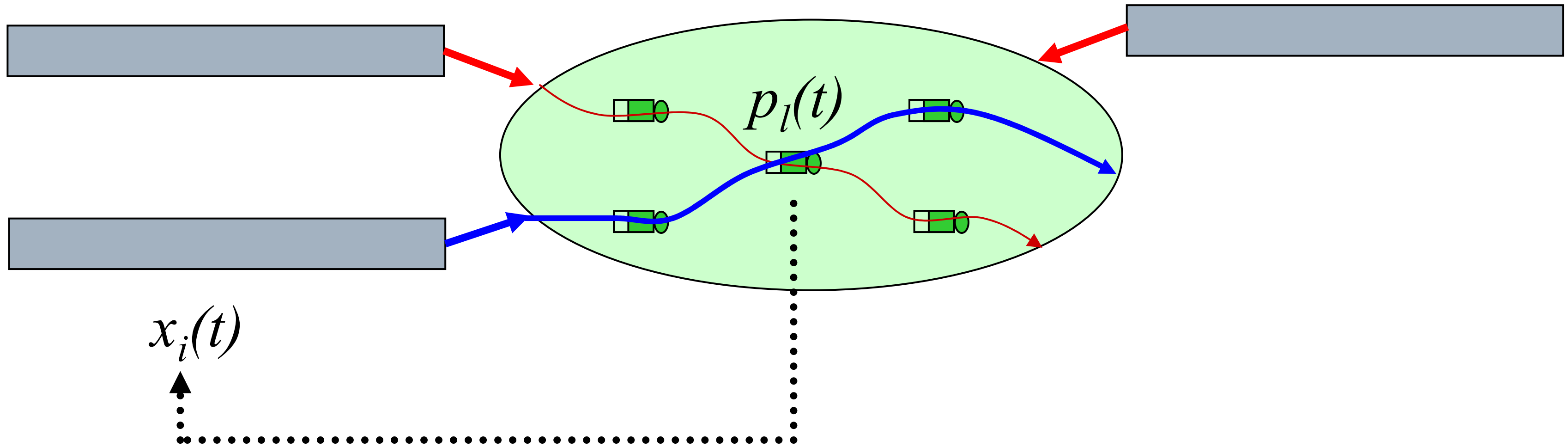
with J. Doyle, L. Li, A. Tang, J. Wang

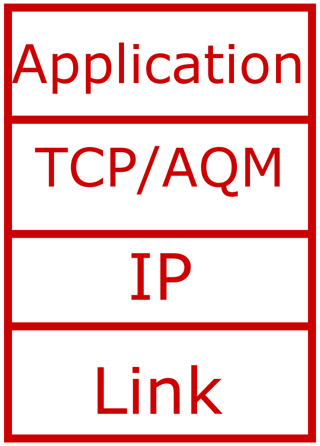
Internet Protocols



- 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,…

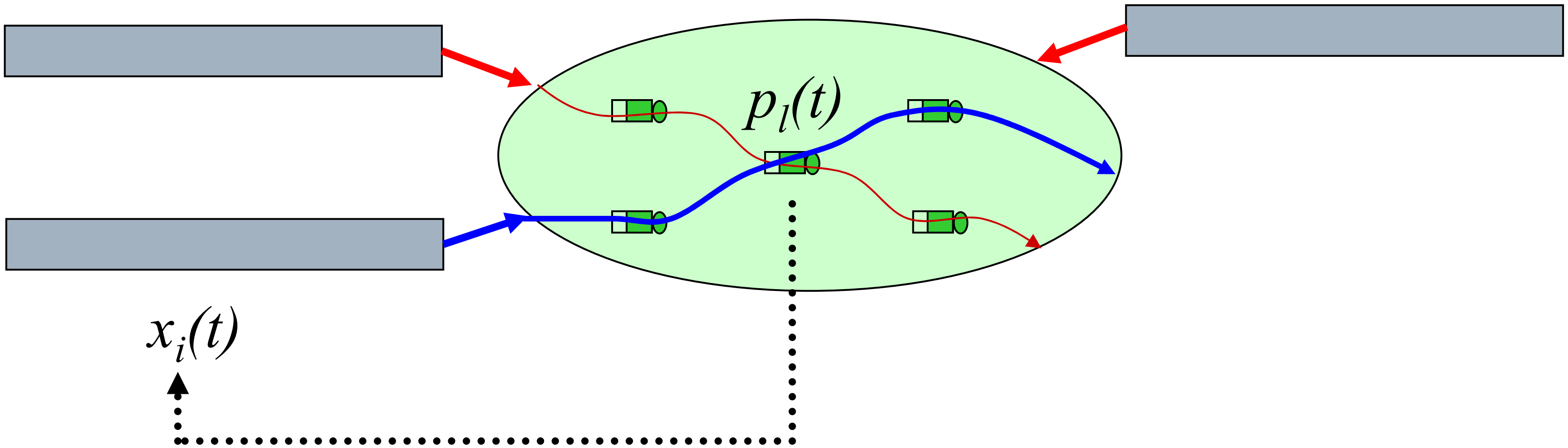
Internet Protocols





- Protocols determines 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

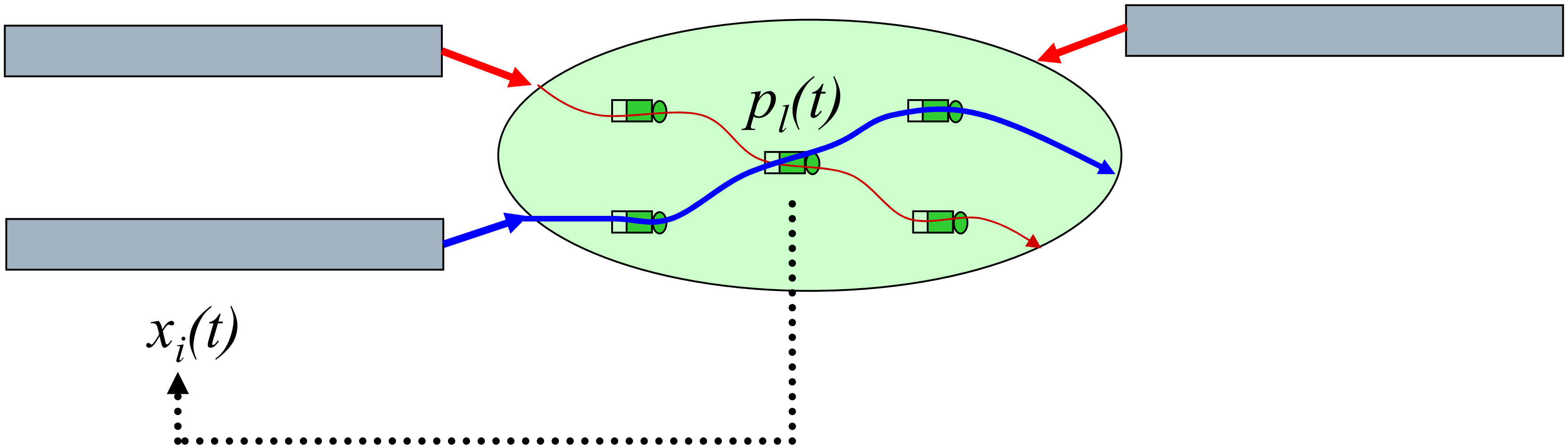


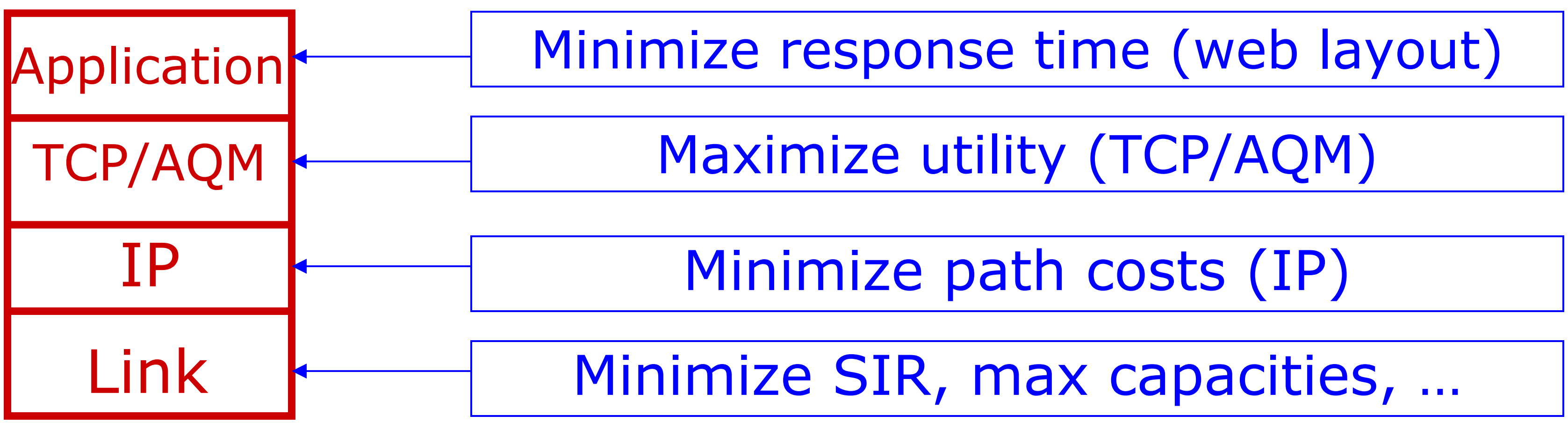
Application
TCP/AQM
IP
Link

Need to reverse engineer

… much easier than biology with full specs

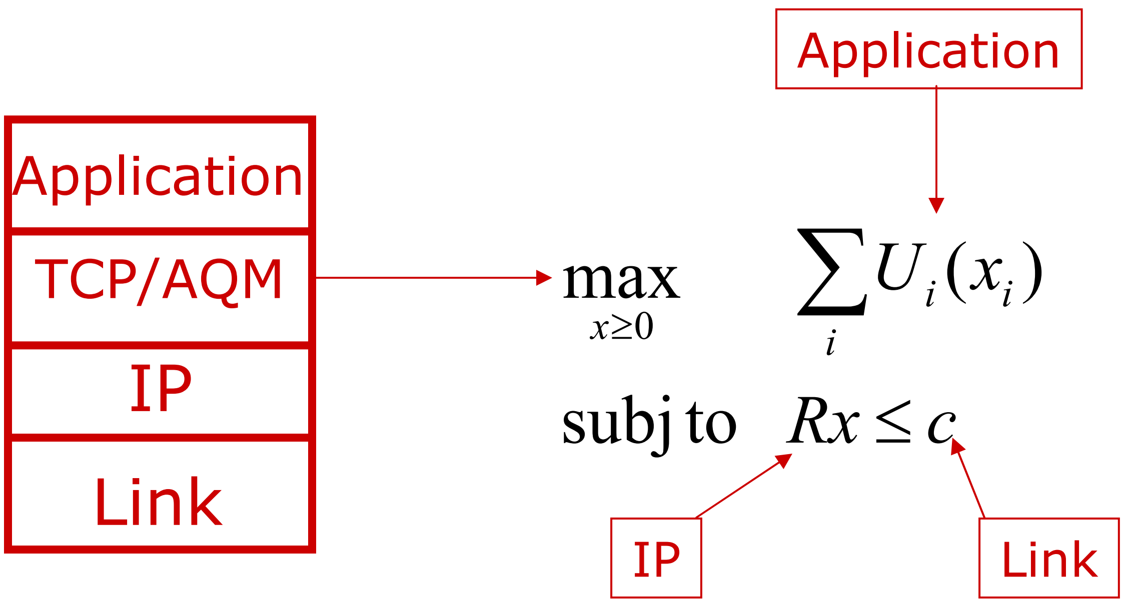
Internet Protocols





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



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

$$\text{subj to} \quad Rx \leq c$$

A Theory for the Internet?

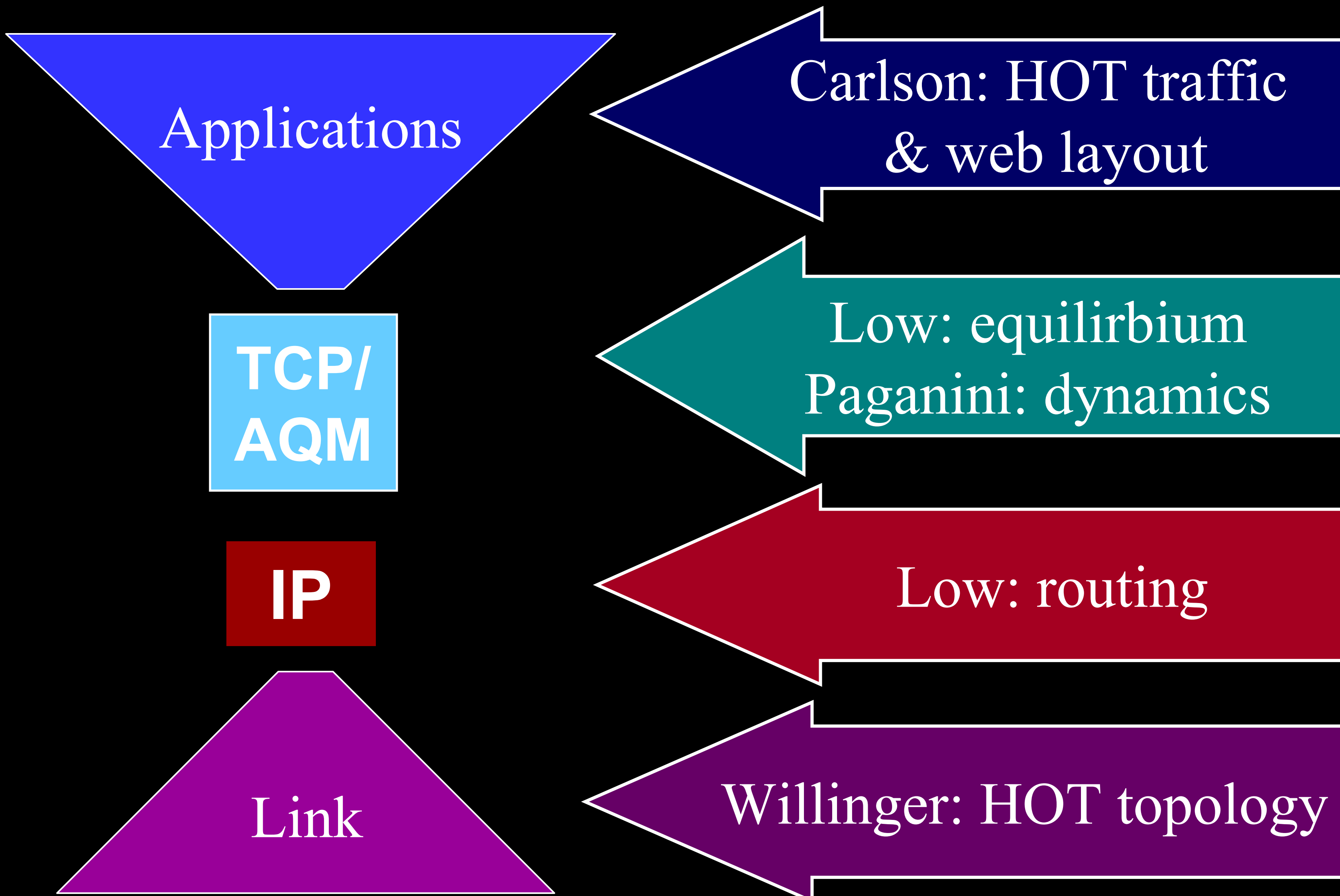
General Approach:

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

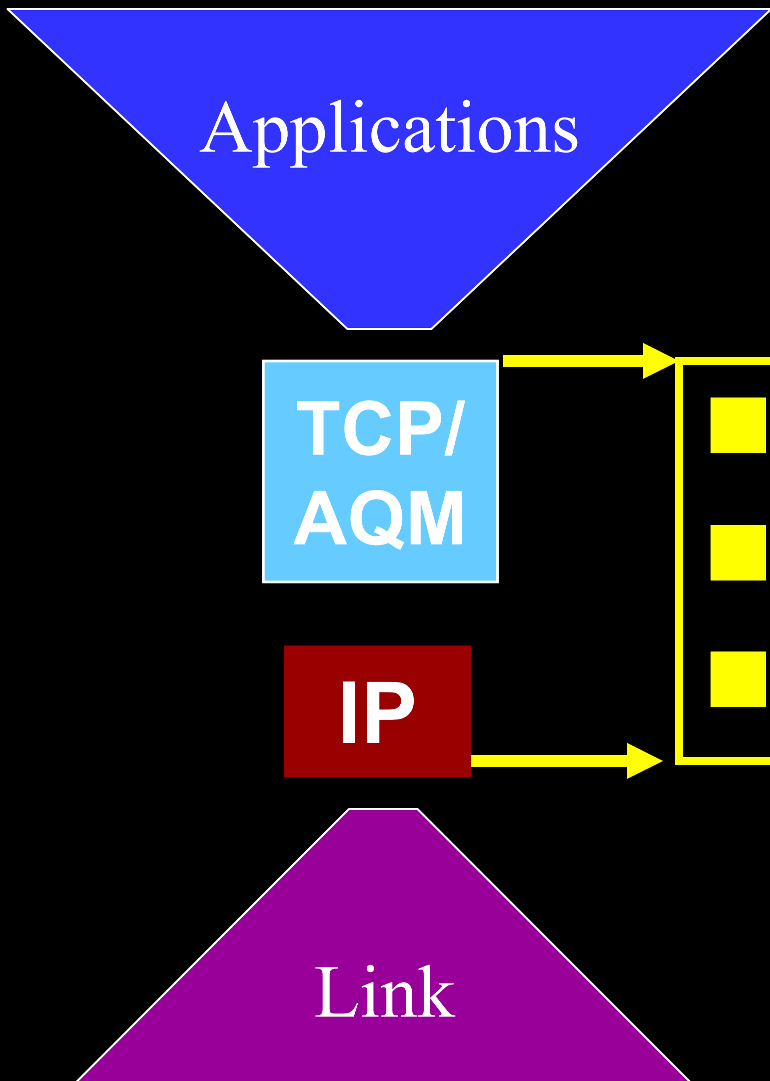
2) Understand interactions across layers

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

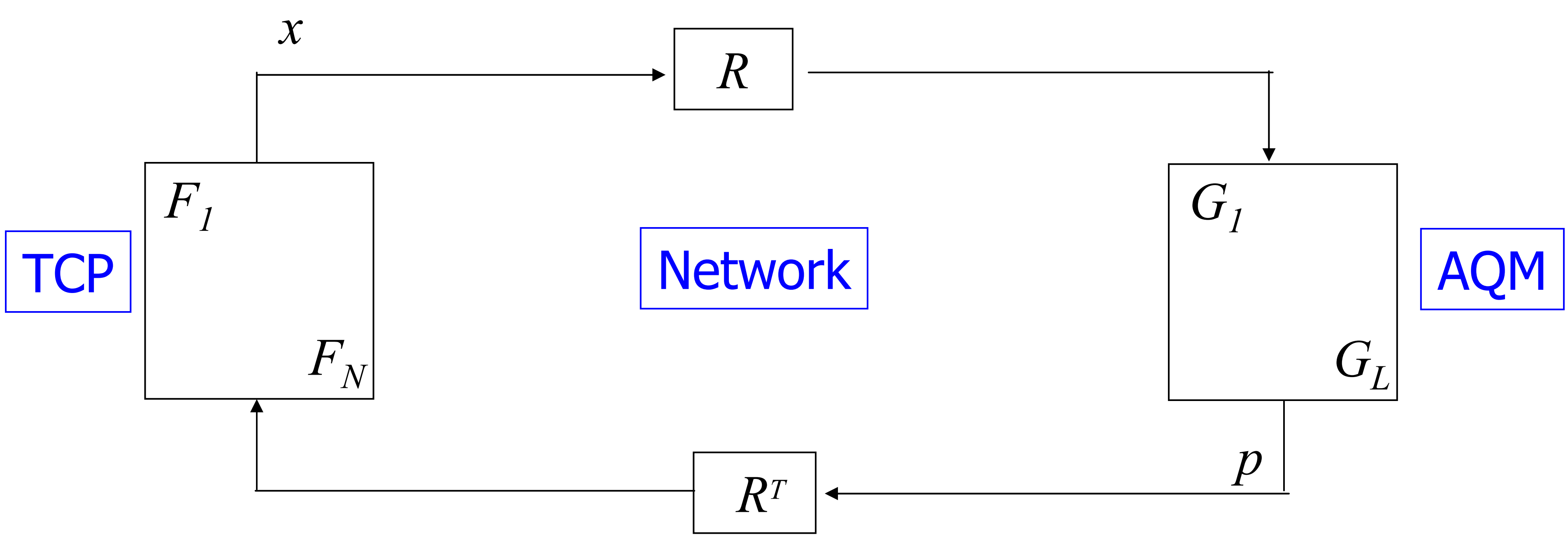

Applications
TCP/
AQM
IP
Link
Carlson: HOT traffic
& web layout
Low: equilirbium
Paganini: dynamics
Low: routing
Willinger: HOT topology

Outline



- Summary
- Details
- Challenges

Network model



$R_{li} = 1$ if source i uses link l ← IP routing

$x(t+1) = F\,(R^T p(t),\ x(t))$ ← Reno, Vegas

$p(t+1) = G\,(p(t),\ Rx(t))$ ← DT, RED, …

Protocol decomposition

TCP-AQM

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

$$\text{subject to} \quad Rx \leq c$$

(Kelly, Malloo, Tan 98)

- TCP algorithms maximize utility with different utility functions

Congestion prices coordinate across protocol layers

Protocol decomposition

IP → $\max_R$; TCP-AQM → $\max_{x \geq 0}$

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

$$\text{subject to} \quad Rx \leq c$$

(Wang, Li, Low, Doyle 03)

- 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

Protocol decomposition

Link → $\max_c$, IP → $\max_R$, TCP-AQM → $\max_{x \geq 0}$

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

$$\text{subject to} \quad Rx \leq c, \quad \alpha^T c \leq B$$

- 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 α as link costs

Congestion prices coordinate across protocol layers

Protocol decomposition – TCP/AQM

TCP-AQM

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

$$\text{subject to} \quad Rx \leq c$$

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

IP | TCP-AQM

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

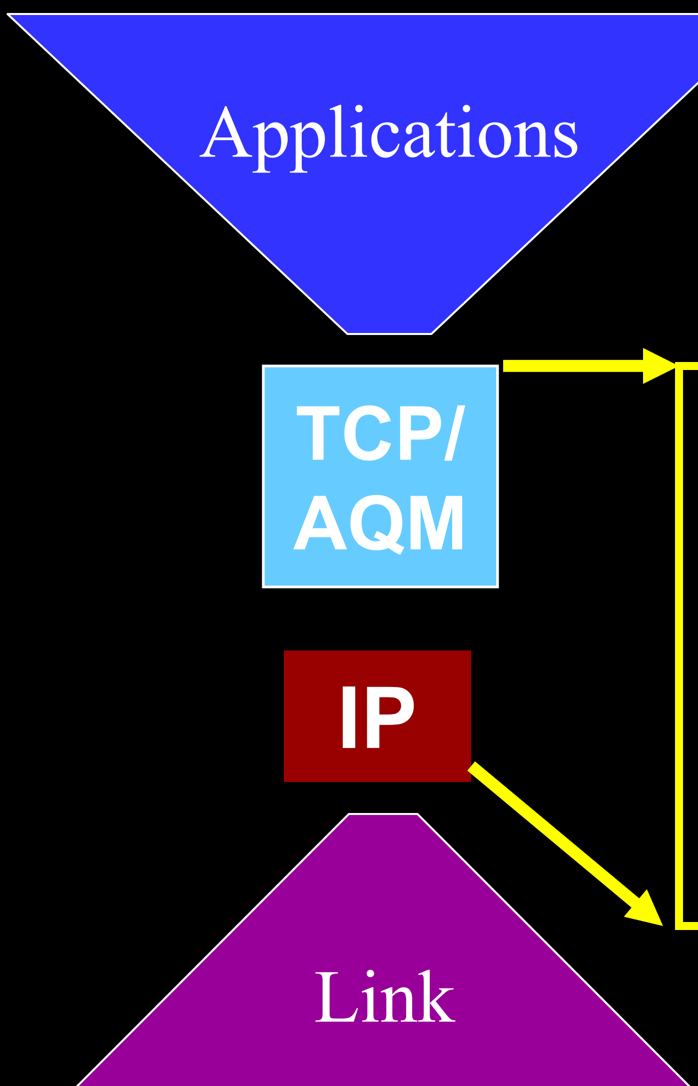
$$\text{subject to} \quad Rx \leq c$$

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



- Summary
- Details
  - TCP
  - TCP/IP
- Challenges

Duality model

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

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

$$\text{s. t.} \quad Rx \leq c$$

- Primal-dual algorithm

$$x(t+1) = F(x, R^T p)$$ ← Reno, Vegas, FAST

$$p(t+1) = G(p, Rx)$$ ← DT, RED, REM/PI, AVQ

- TCP/AQM
  - Maximize utility with different utility functions
  - (L 03): (x^*, p^*) primal-dual optimal iff

$$y_l^* \leq c_l \quad \text{with equality if } p_l^* > 0$$

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)
- $\alpha =$ 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

$$\max_{x \geq 0} \sum_i U_i(x_i)$$
$$\text{s. t.} \quad Rx \leq c$$

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

(Mo, Walrand 00)

- Identify allocation with α
- An allocation is fairer if its α is larger

Fairness

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

$$\text{s. t.} \quad Rx \leq c$$

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

(Mo, Walrand 00)

- $\alpha = 0$: maximum throughput
- $\alpha = 1$: proportional fairness
- $\alpha = 2$: min delay fairness (Reno)
- $\alpha =$ infinity: maxmin fairness

Fairness

$$\max_{x \geq 0} \sum_i U_i(x_i)$$
$$\text{s. t.} \quad Rx \leq c$$

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

(Mo, Walrand 00)

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

Efficiency

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

$$\text{s. t.} \quad Rx \leq c$$

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

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

$$\text{throughput} \quad T(\alpha) := \sum_i x_i(\alpha)$$

Conjecture

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

$$\text{s. t.} \quad Rx \leq c$$

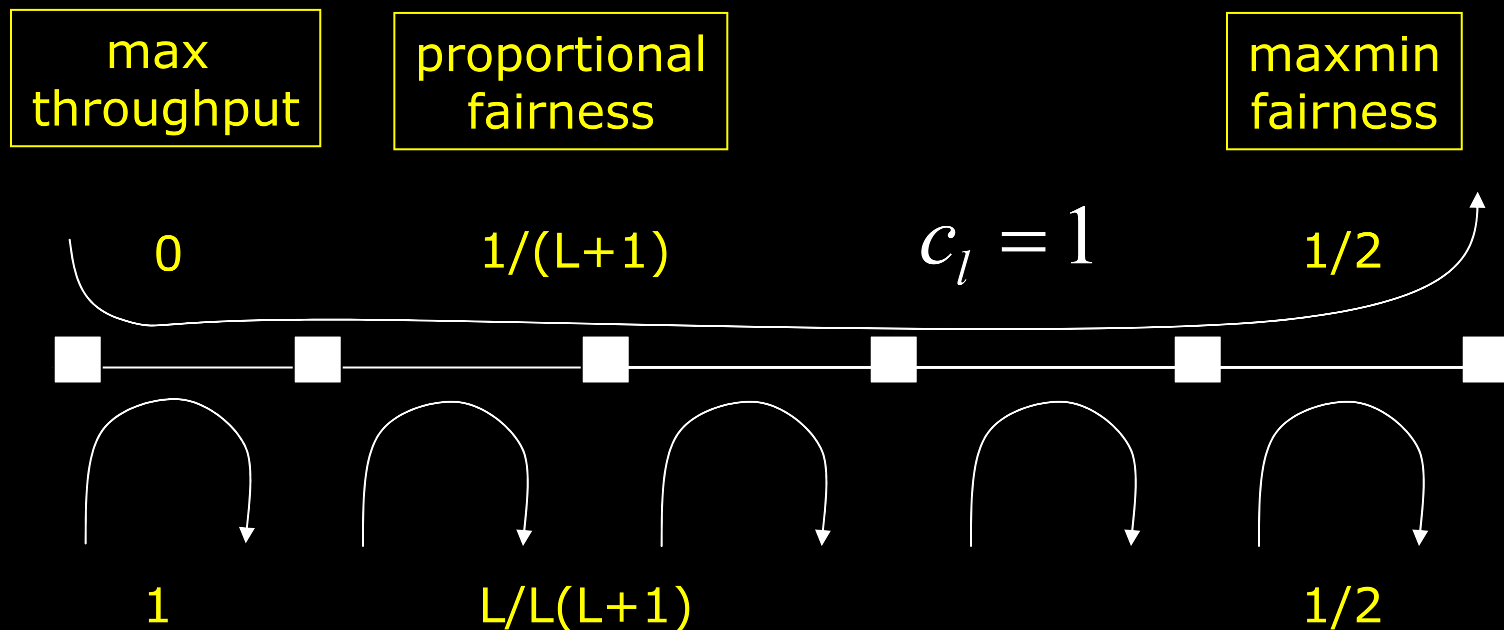
$$U_i(x_i) = \begin{cases} (1-\alpha)x_i^{1-\alpha} & \text{if } \alpha \neq 1 \\ \log x_i & \text{if } \alpha = 1 \end{cases}$$

Conjecture

$T(\alpha)$ is nonincreasing

i.e. a fair allocation is always inefficient

Example 1



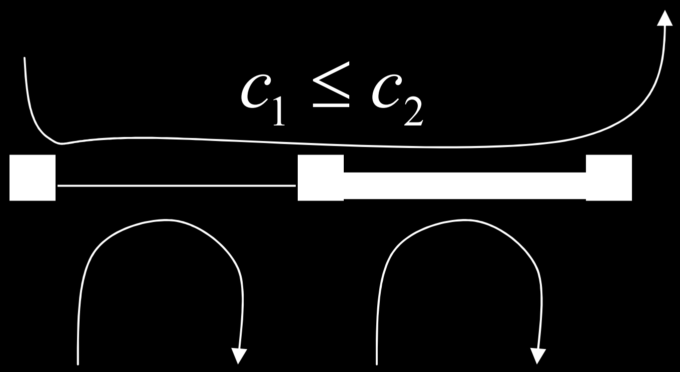
$$T(0) > T(1) > T(\infty)$$

Conjecture

$T(\alpha)$ is nonincreasing

i.e. a fair allocation is always inefficient

Example 2



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

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

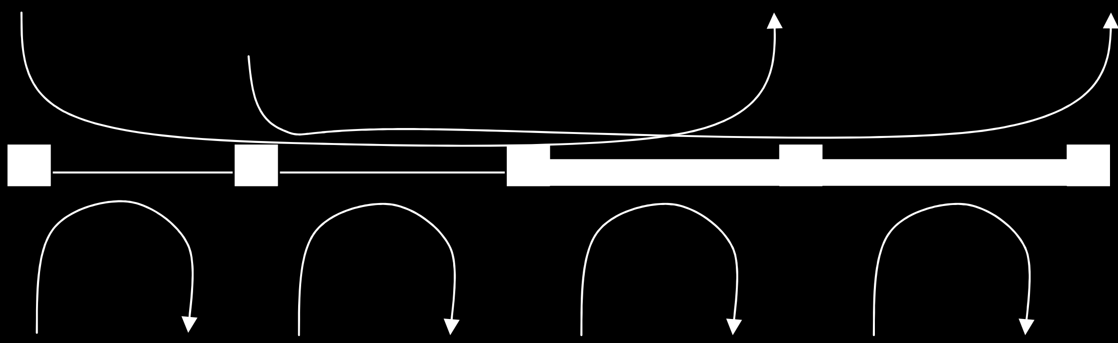
$$\Rightarrow \quad T(1) > T(\infty)$$

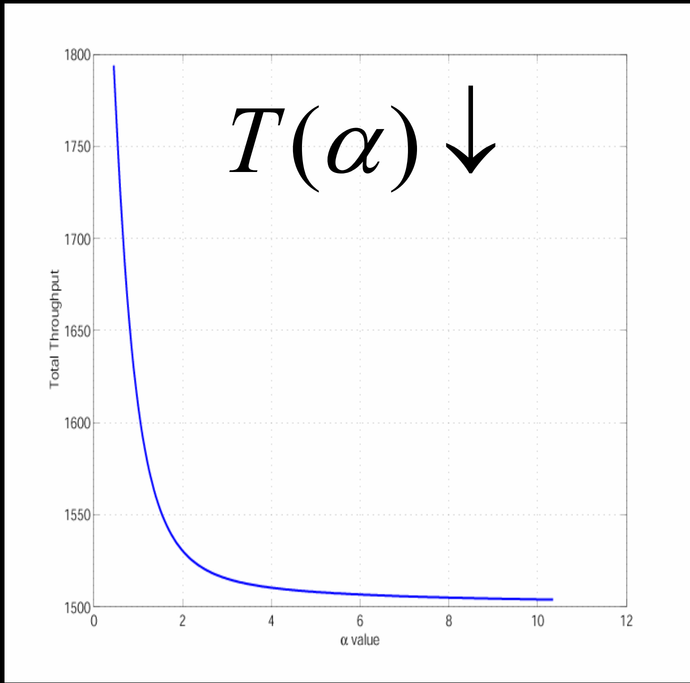
<u>Conjecture</u>

$T(\alpha)$ 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

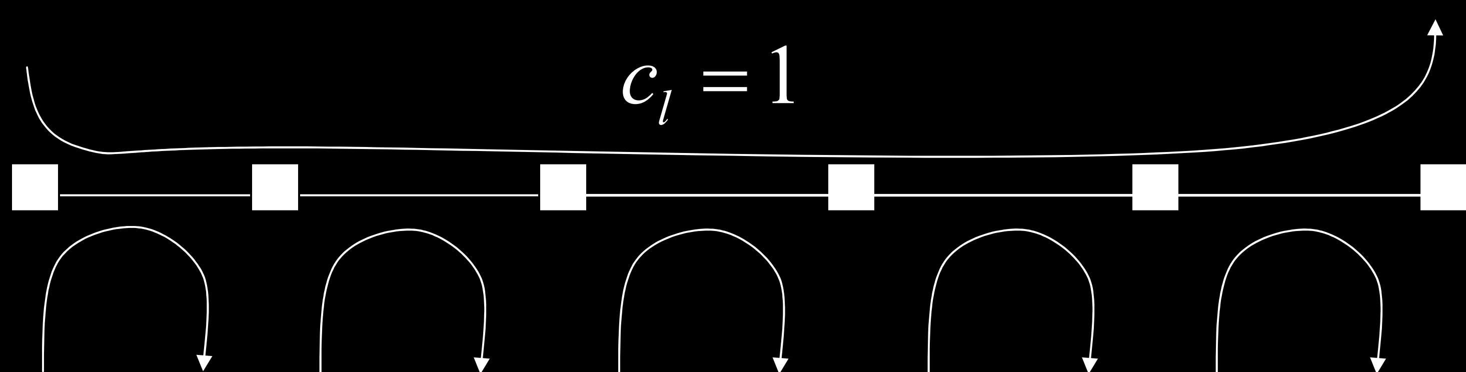
"*The fundamental conflict between achieving flow fairness and maximizing overall system throughput….. The basic issue is thus the trade-off between these two conflicting criteria*."

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 $N(R)=1$



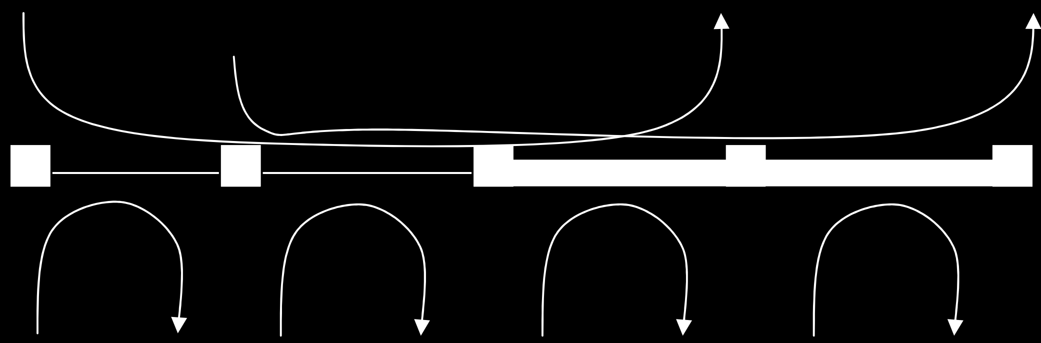
$$\frac{1}{L^{1/\alpha}+1}$$

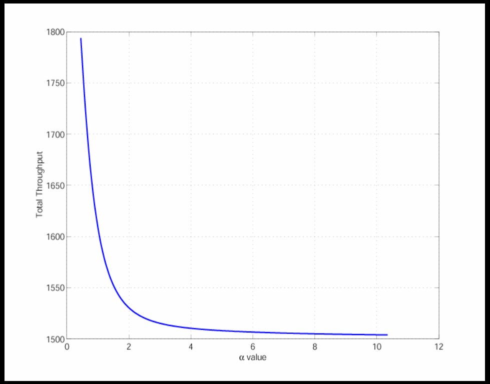
$$\frac{L^{1/\alpha}}{L^{1/\alpha}+1}$$

Results

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

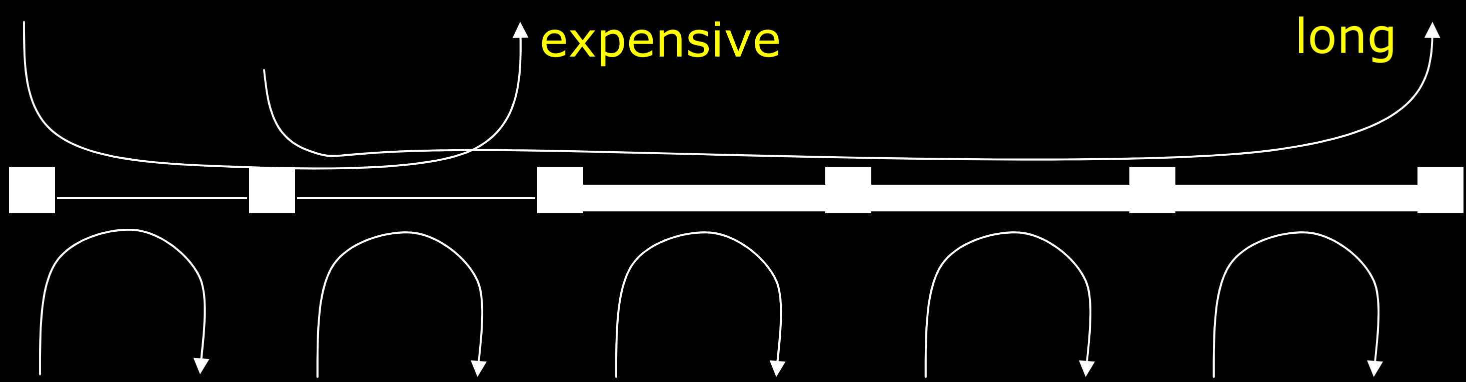
- Corollary 2: true if
  - $N(R)=2$
  - 2 long flows pass through same# links





Counter-example

- There exists a network such that $dT/d\alpha > 0$ for almost all $\alpha > 0$
- Intuition
  - Large α favors expensive flows
  - Long flows may not be expensive
- Max-min may be more efficient than proportional fairness

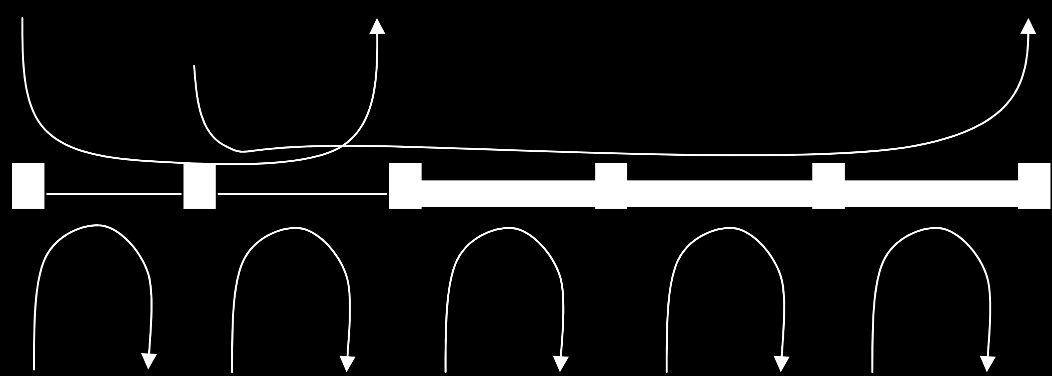


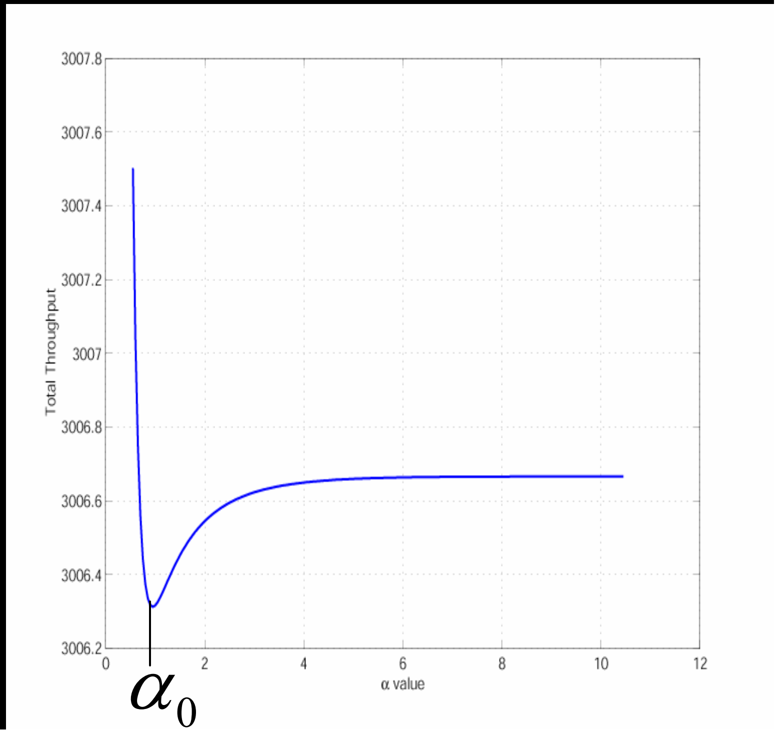
Counter-example

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

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

- Compact example





Questions

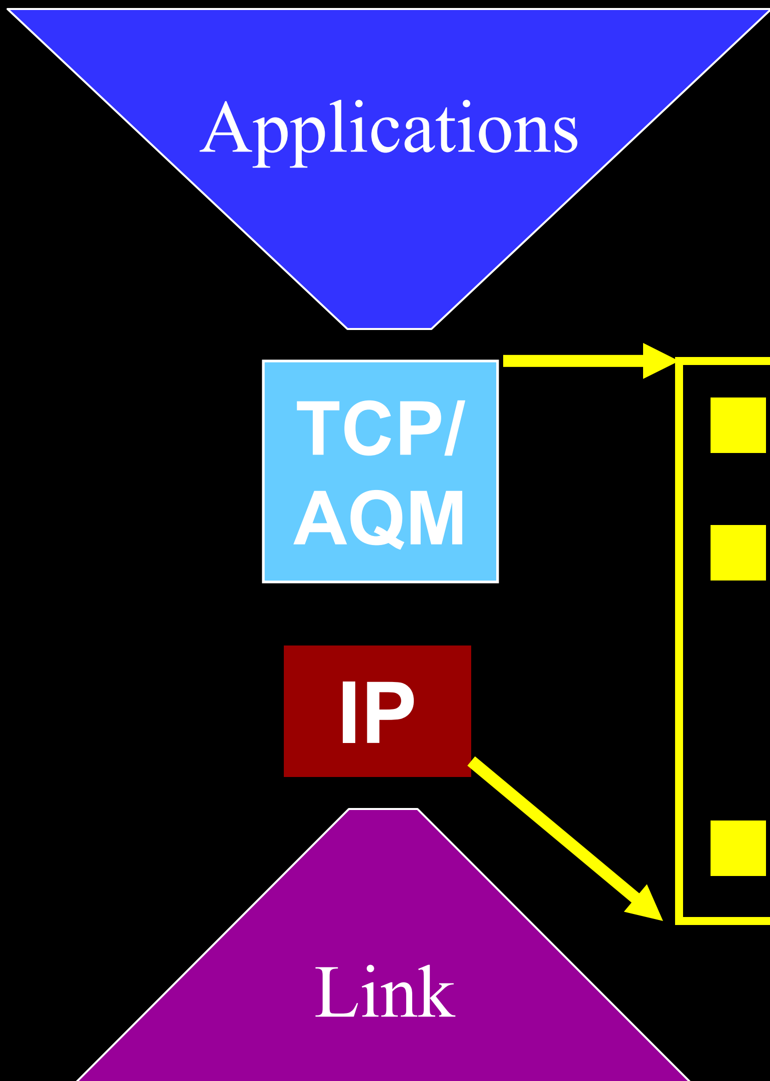
- Is fair allocation always inefficeint
- Does raising capacity always increase throughput

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 α, increasing
  - a link's capacity can reduce T
  - all links' capacities equally can reduce T
  - all links' capacities proportionally raises T

Outline



- Summary
- Details
  - TCP
  - TCP/IP
- Challenges

Motivation

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

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

Motivation

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

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

Shortest path routing!

Can TCP/IP maximize utility?

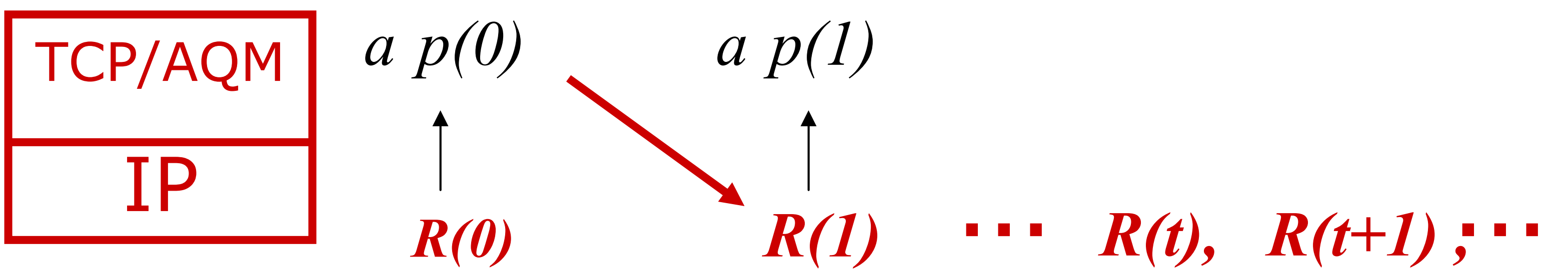
Two timescales

- Instant convergence of TCP/IP
- Link cost = $a\, p_l(t) + b\, d_l$

static

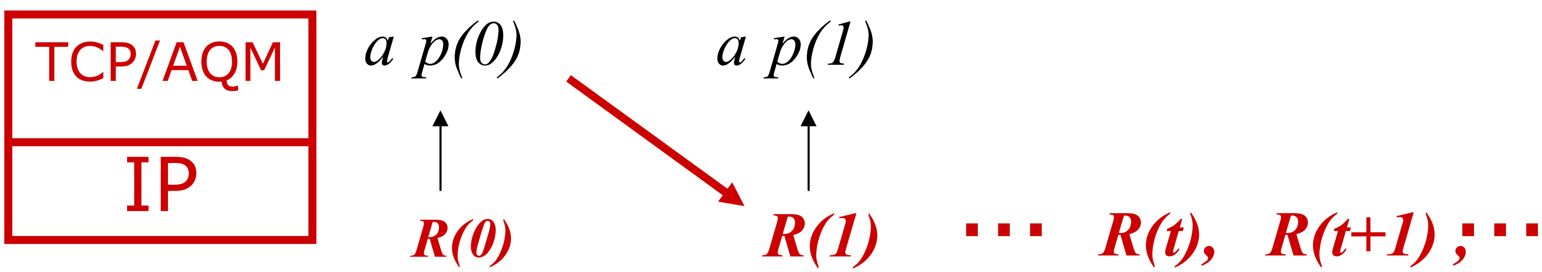
price

- Shortest path routing $R(t)$



Questions

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



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

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

$$\text{Dual:} \quad \min_{p\geq 0} \left(\sum_i \max_{x_i\geq 0} \left(U_i(x_i) - x_i \max_{R_i} \sum_l R_{li} p_l \right) + \sum_l p_l c_l \right)$$

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 $\{c_1, \ldots, c_n\}$

  Find: set A s.t. $\sum_{i \in A} c_i = \sum_{i \notin A} c_i$

TCP/IP

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

TCP/IP/provisioning

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

Link → $\max_c$; IP → $\max_R$; TCP-AQM → $\max_{x \geq 0}$

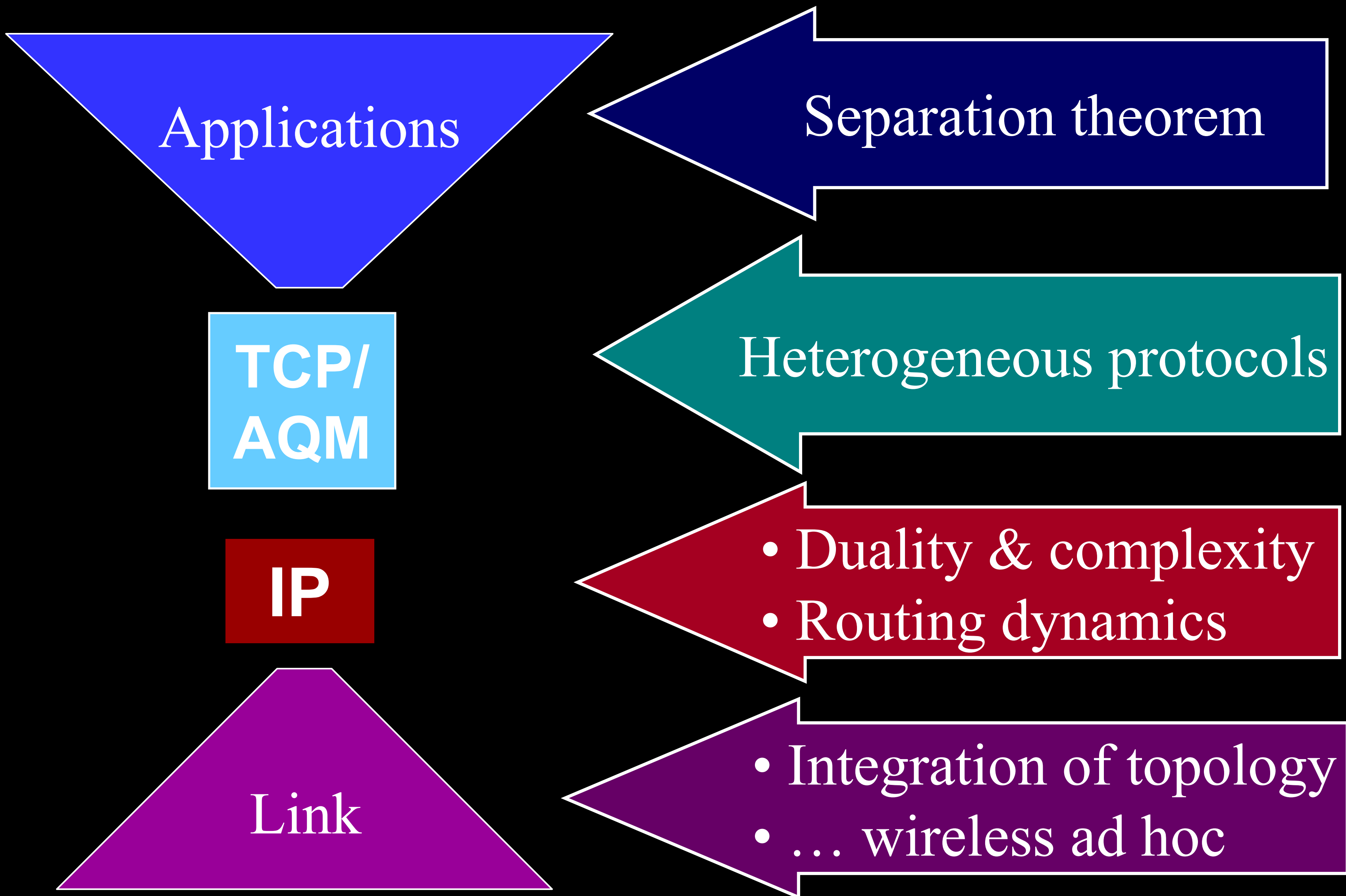
$$\max_{c} \; \max_{R} \; \max_{x \geq 0} \; \sum_i U_i(x_i)$$

$$\text{subject to} \quad Rx \leq c, \quad \alpha^T c \leq B$$

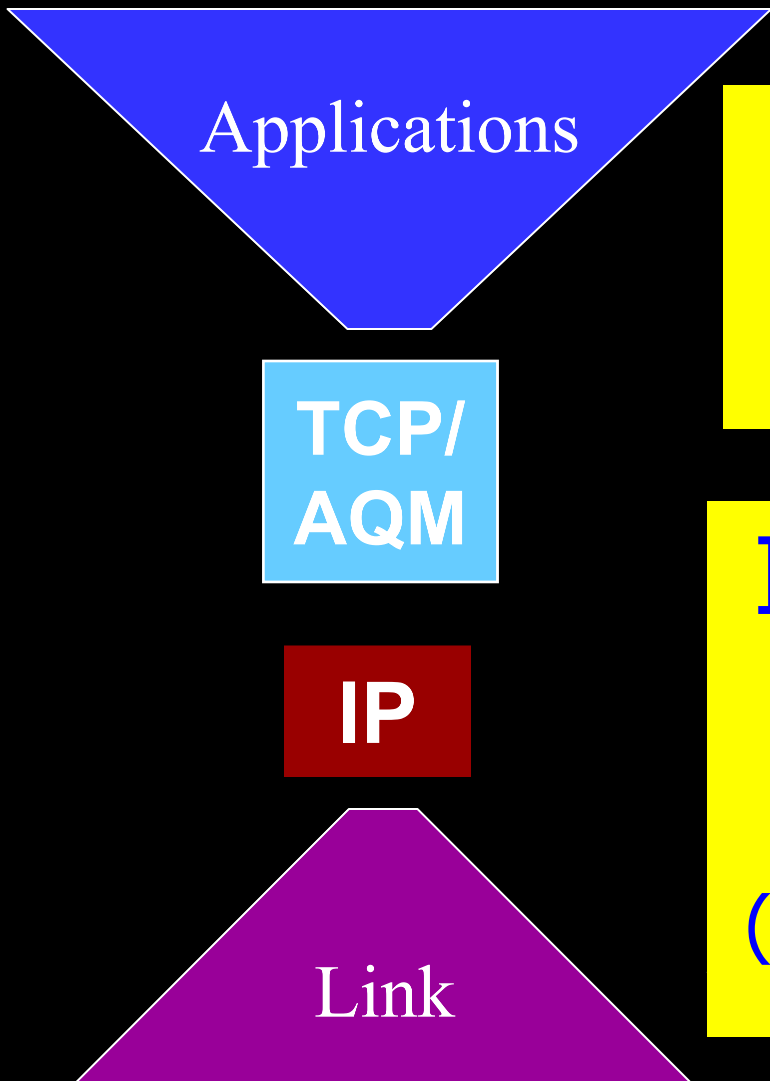
- TCP algorithms maximize utility with different utility functions
- IP shortest path routing is optimal using congestion prices as link costs, with given link capacities c
- With optimal provisioning, static routing is optimal using provisioning cost α as link costs

Congestion prices coordinate across protocol layers

Challenges


Applications
Separation theorem
TCP/
AQM
Heterogeneous protocols
IP
• Duality & complexity
• Routing dynamics
Link
• Integration of topology
• … wireless ad hoc

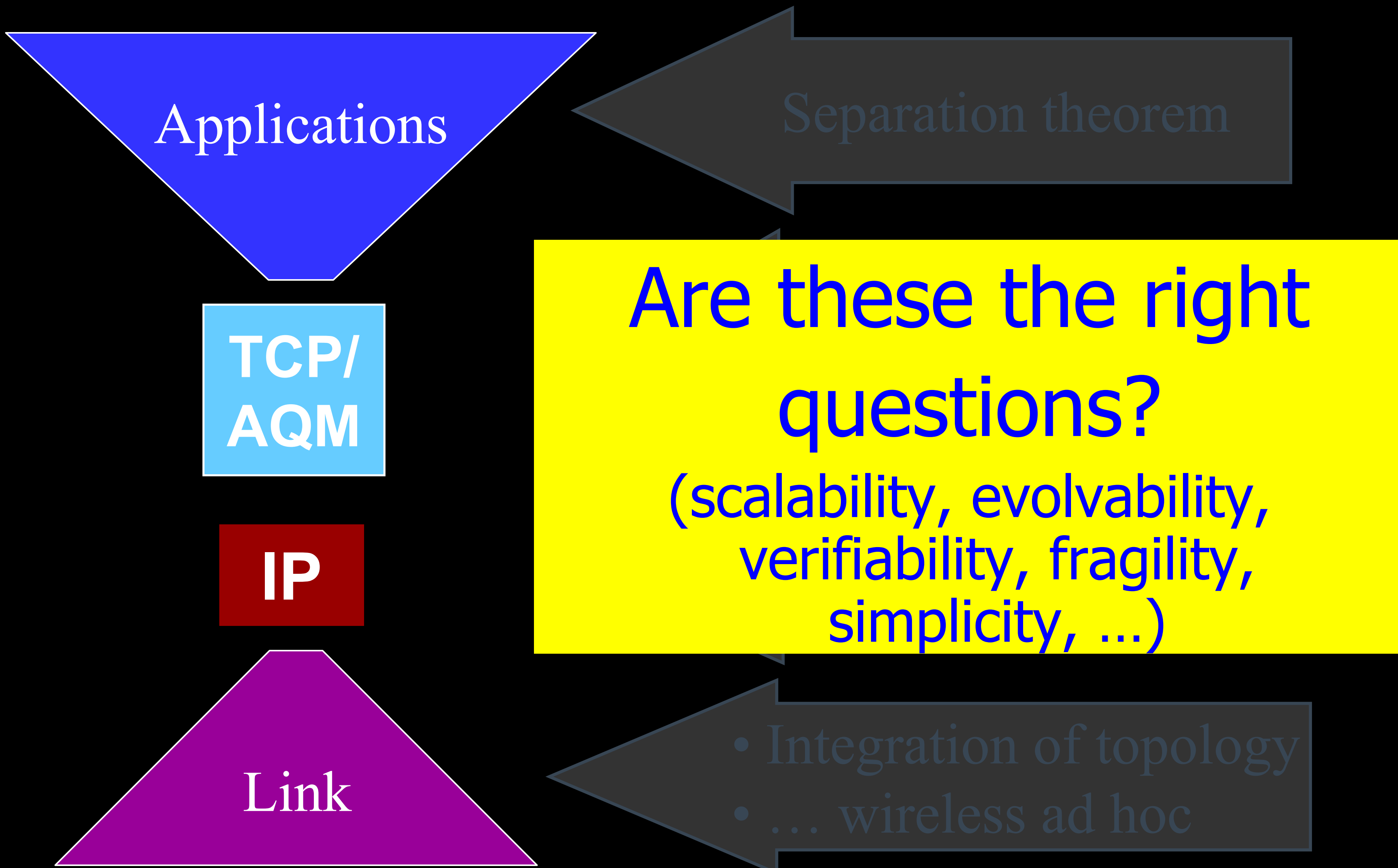
Challenges



What is the ultimate underlying problem?

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

Challenges


Applications
TCP/
AQM
IP
Link
Separation theorem
• Integration of topology
• … wireless ad hoc
Are these the right questions?
(scalability, evolvability, verifiability, fragility, simplicity, …)