

Modeling of the Shrew: the Quest for a "Model" Network Model

Jim Kurose
Department of Computer Science
University of Massachusetts

The question should *not* be

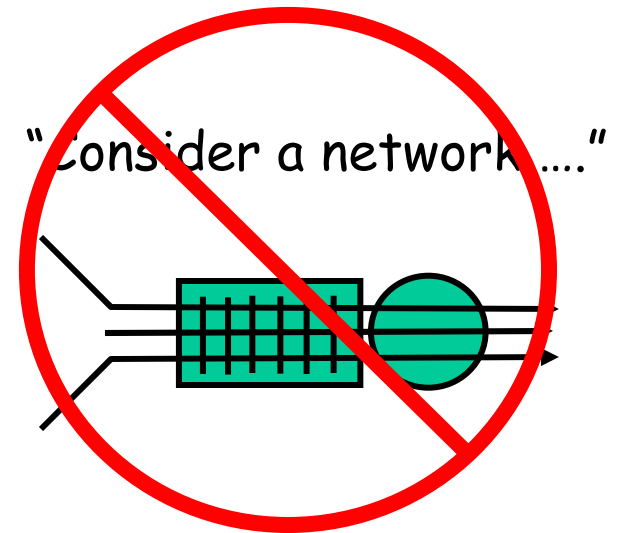
“What is in the network model?”

but rather...

“What to model in the network?”

Looking back: successes!

- ❑ open loop networks:
 - loss, delay, throughput
 - "Kleinrock legacy"
- ❑ bounding techniques:
 - network calculi
- ❑ self-similarity, LRD
- ❑ small, closed-loop nets
 - TCP models



No more queues
in isolation!

Raising the level of abstraction

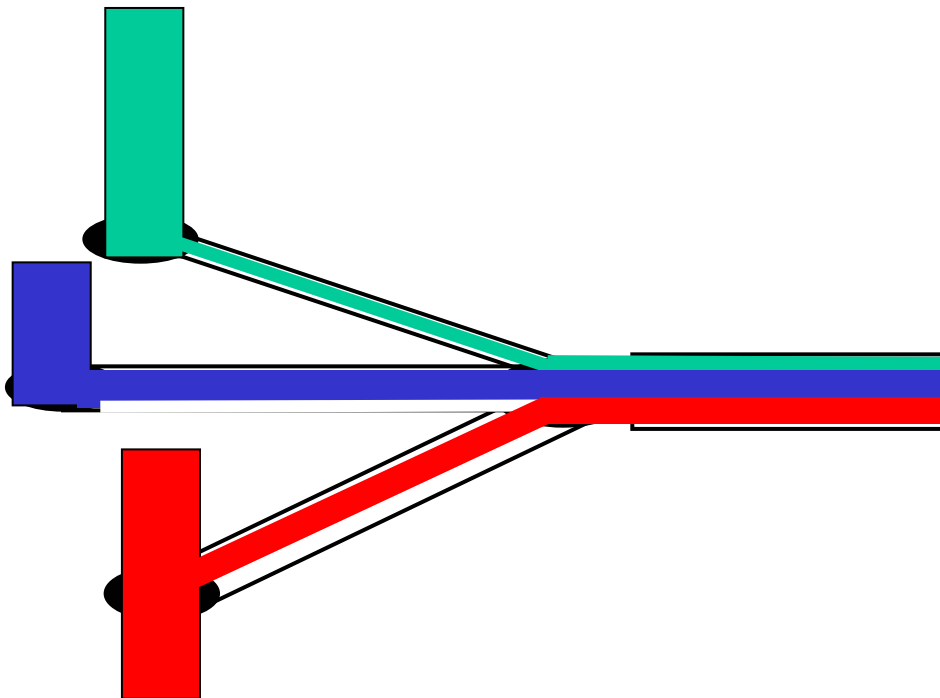
- ❑ packets: too microscopic
- ❑ coarser-grained models
 - large scale networks
 - dimensioning, provisioning
- ❑ flow-level abstractions
 - call models in telephony
 - WWW transfer:
document is workload unit
 - fluid models

Challenges

- ❑ multiple bottlenecks
- ❑ open and closed loop control
- ❑ long and short-lived flows
- ❑ micro-macro-micro
 - effects of aggregation, de-aggregation

Flow-level modeling: simple example

- *flow rate*: determined by link capacity, sharing requirements
- *networks* of processor-sharing-like queues



- Roberts, Massoulié, Gibbens, ...
- Bu, Towsley

What to model?

- we excel in data plane
 - loss, throughput, delay
- Q: What measures do people really care about?
 - "robustness"
 - "complexity of control"
 - maintainability
 - adaptability
 - reconfigurability
 - security
- modeling these is hard!
 - "efficiency" not the most important measure!
 - little/no past work!
 - metrics and models undefined!
- example: modeling a soft state protocol

Example: soft state control

- Conventional wisdom: “soft-state is robust, less complex than hard-state signaling”
 - really?
 - how to define “robustness”?
 - how to define “complexity”?
- Posing/answering such a question is:
 - *hard*: no well-accepted models, paradigms
 - *easy*: little/no past research
 - *important*: a fundamental question

Summary

- *lots* of successes to be proud of!
- new frontiers:
 - application-driven modeling
 - higher level of modeling abstraction
 - modeling “on beyond performance”