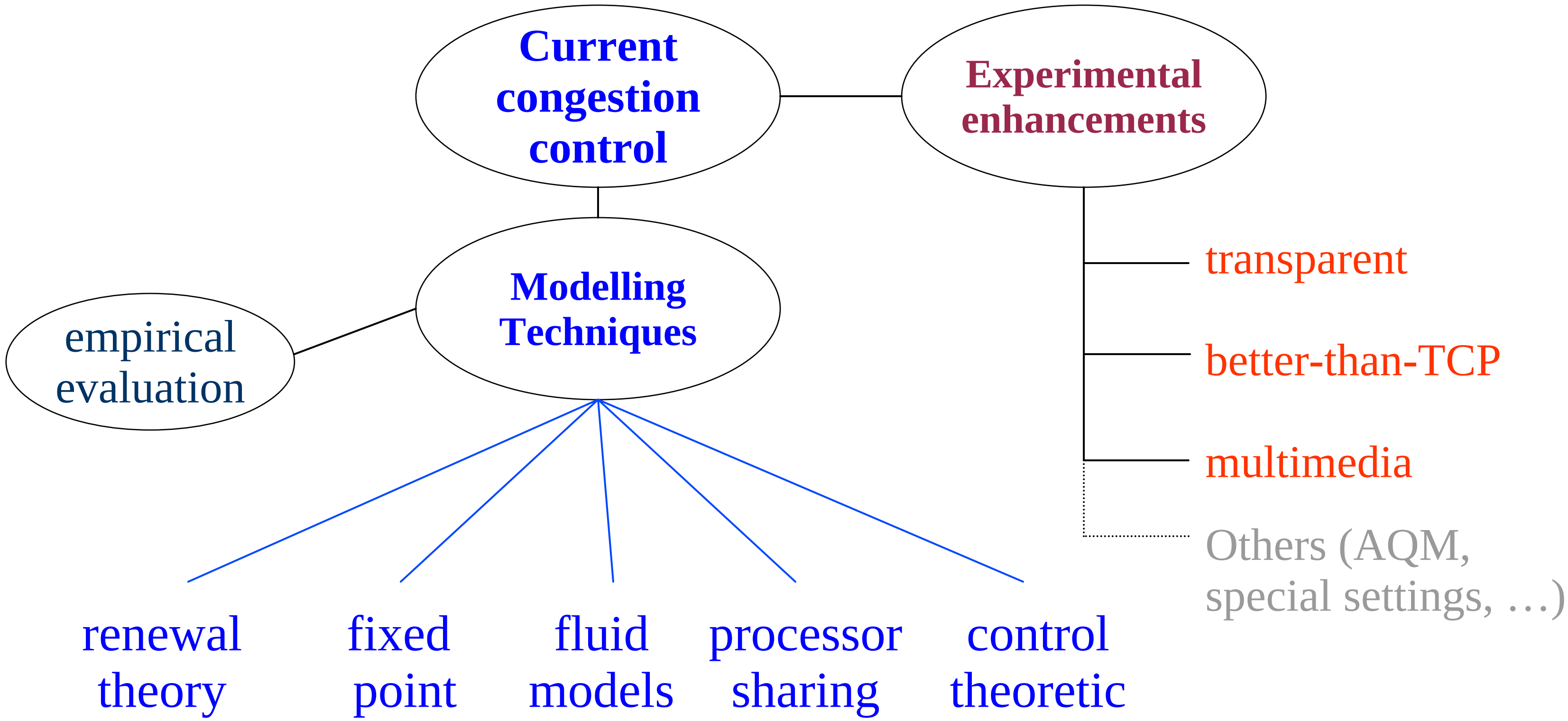
Stochastic Modeling of the TCP Protocol

PhD Candidacy
Eli Brosh
Dept. of Computer Science
Columbia University
Feb 2007

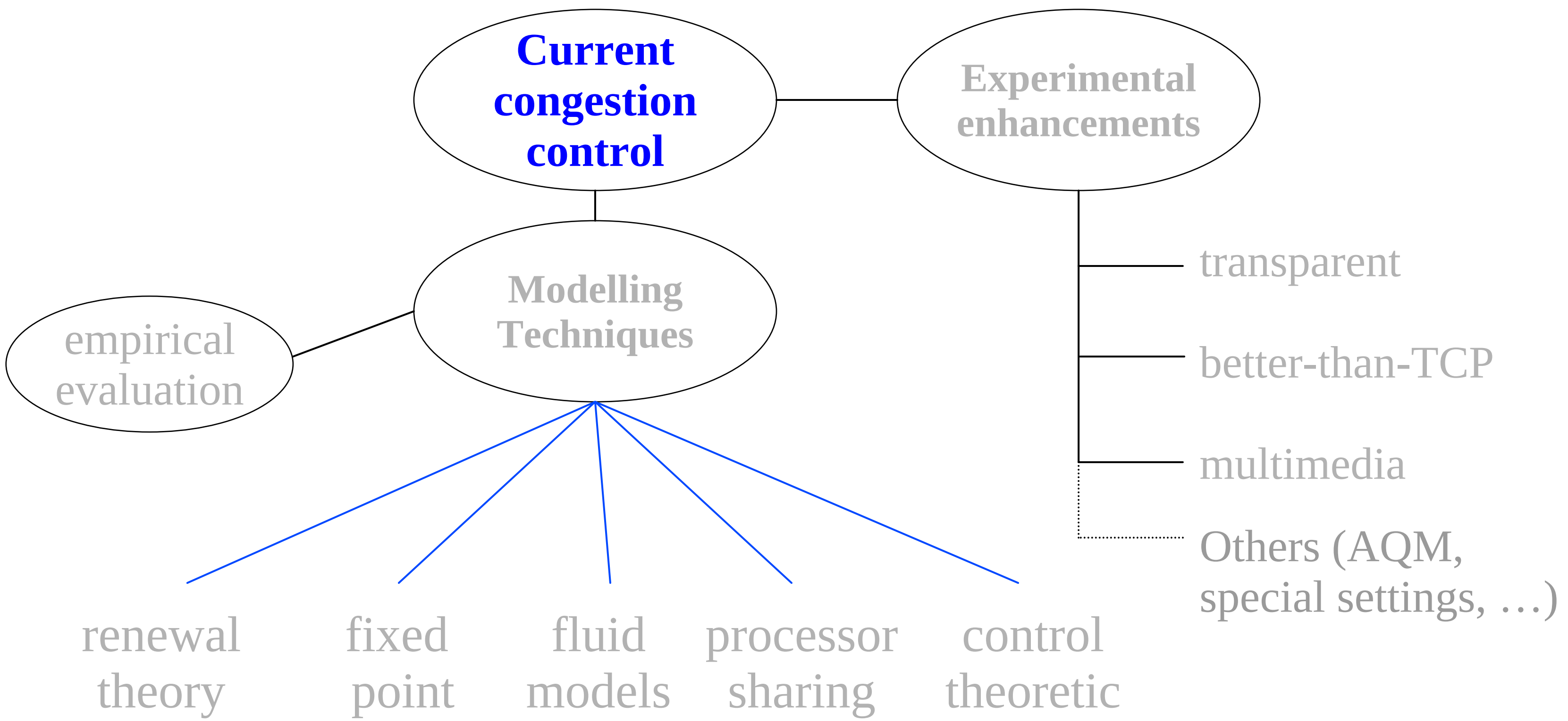
Motivation

- TCP is widely used!
  - Carries 80%-90% of internet traffic
- TCP models serve to compute (and hence to improve) network and application performance.
  - Reveal insights on the factors influencing TCP’s performance
  - Provide guidelines for designing and tuning AQM schemes
  - Form the basis for TCP-Friendly protocols

Outline: TCP Modeling



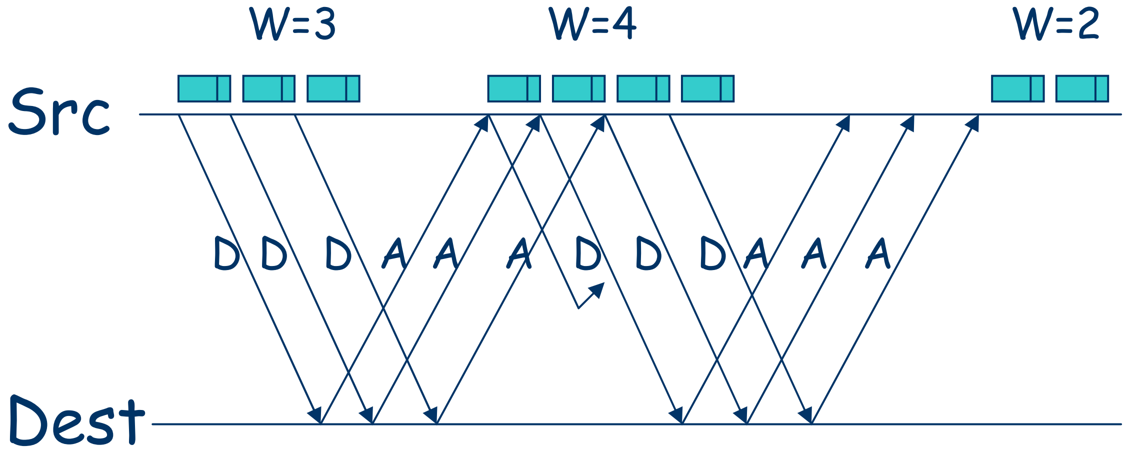
Outline: Overview of current TCP



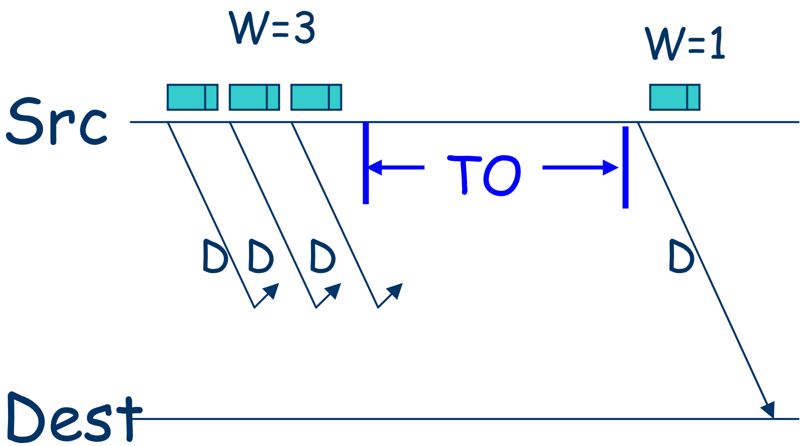
Basic TCP [J88]

- End-to-end congestion control
- Window algorithm: Can send W packets
  - ACK clocked, cumulative ACKs
- Increase window if no loss:
  - W <−W +1 per RTT ⟵ additive Increase
- Loss, indication of congestion
  - Triple-dup loss indication (TD)
  - Timeout loss indication (TO)
- Reduce window on loss:
  - Half window on TD loss, W <−W/2 ⟵ multiplicative decrease
  - Reduce to one on TO loss, W<-1

Triple-dup loss example



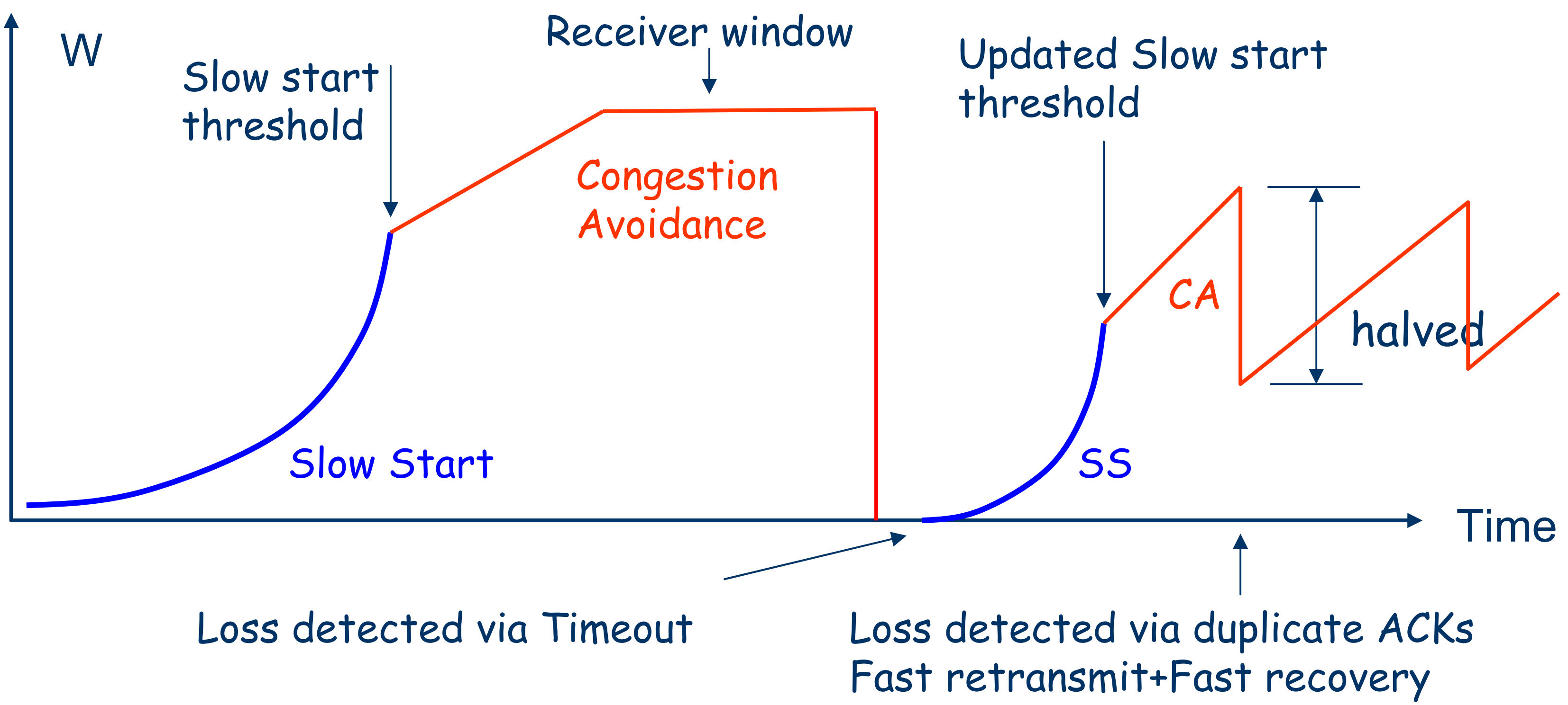
Timeout loss example



- Successive timeout intervals grow exponentially long up to six times

TCP Mechanisms
Congestion Avoidance (CA) and Slow-Start (SS)

- Slow-start phase at beginning of a session
- Sawtooth-like window evolution during CA

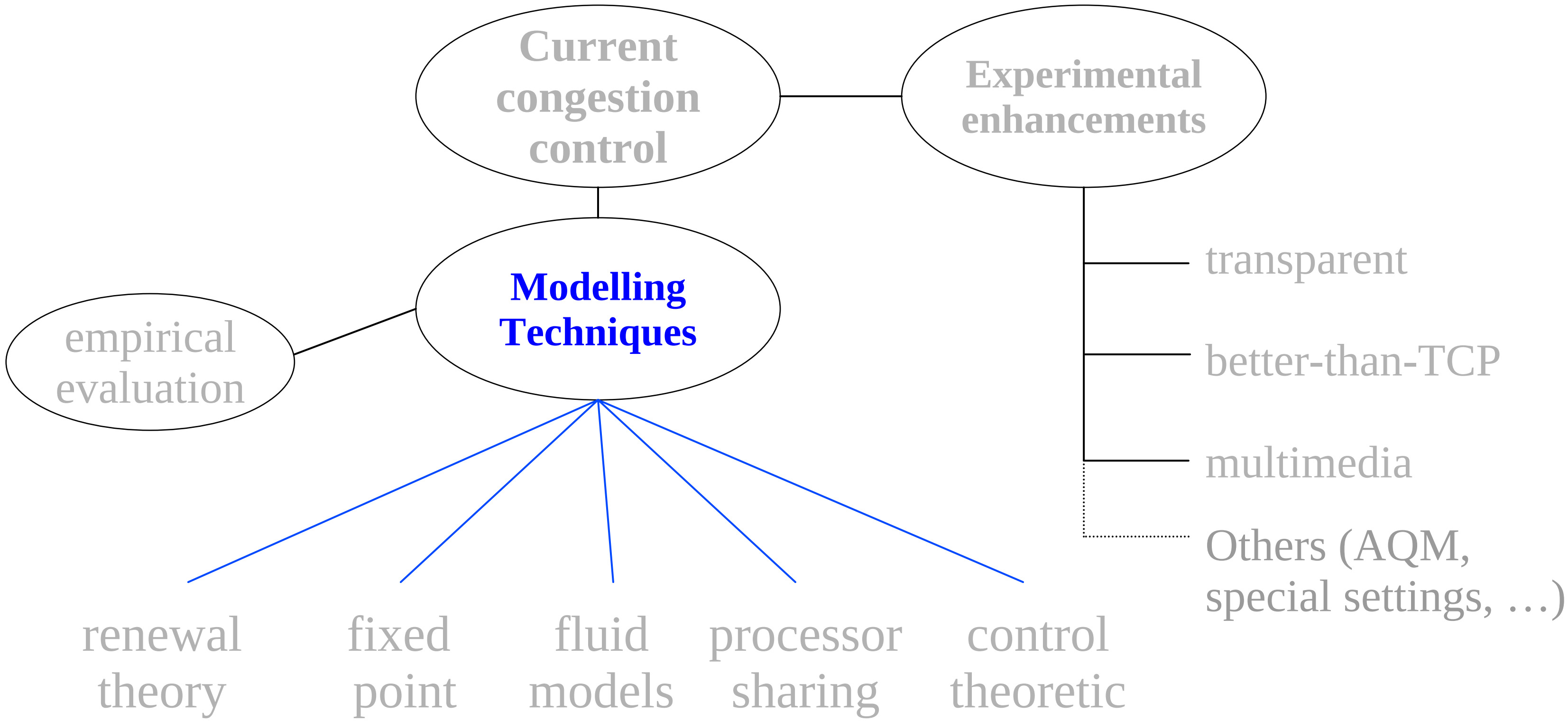


Overview: TCP Variants

- **Tahoe:** reduce window to one at loss indication, use slow-start to ramp up
- **Reno:** fast recovery without use of slow-start
- **NewReno:** react to only one loss per RTT
- **SACK:** receiver gives more information to sender about received packets allowing sender to recover from multiple-packet losses faster
- **Vegas:** delay-based congestion avoidance. Uses RTT variations as an early-congestion-feedback mechanism instead of losses
- **ECN** (explicit congestion notification) router marks packet; source treats like a TD loss

[RFCs 2581,2582,2883], [BP95]

Outline: Modeling techniques



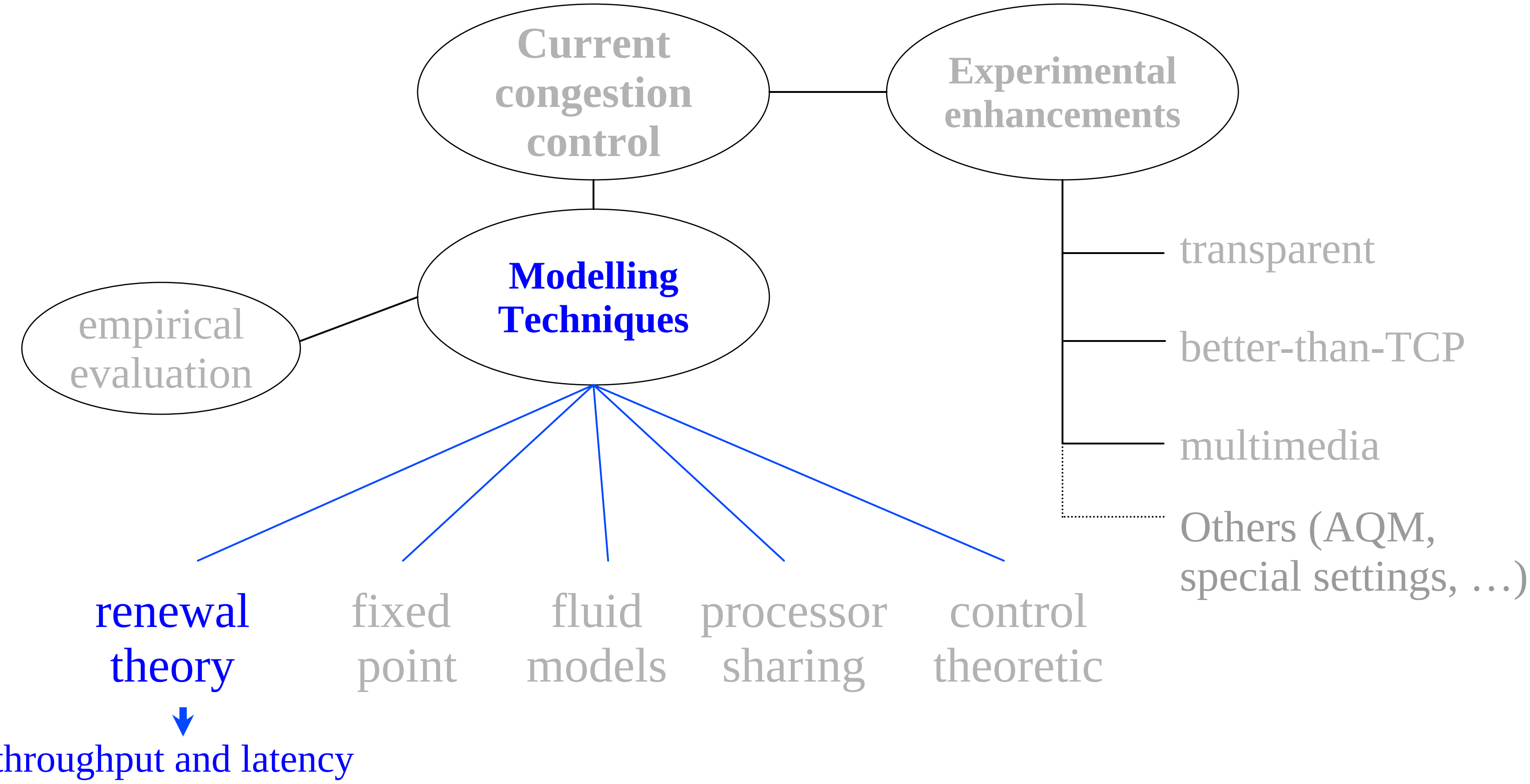
TCP Modeling: Objective

- Objective: to express the performance of a TCP transfer as a function of: packet loss rate, round-trip time, receiver advertised window, etc.

- TCP performance measures: Throughput, latency, fairness, etc.

- Basis for modeling TCP
  - Requires a model for TCP dynamics
    - At the packet-level, window-level, flow-level, etc.
  - Requires a model for the network
    - How do packets get dropped? What are the delays they experience?

Outline: Renewal Theory Models



Renewal Theory Models

- Renewal theory: study window evolution in terms of cycles
  - Cycle: period between two consecutive loss events
- Basic loss model is often used:
  - Bernoulli losses: packets are dropped with a fixed probability p, independently of others
  - Correlated losses: p until first packet lost, remaining window packets are lost
- Round trip time (RTT) is constant
- From renewal reward theory, the steady state TCP throughput:

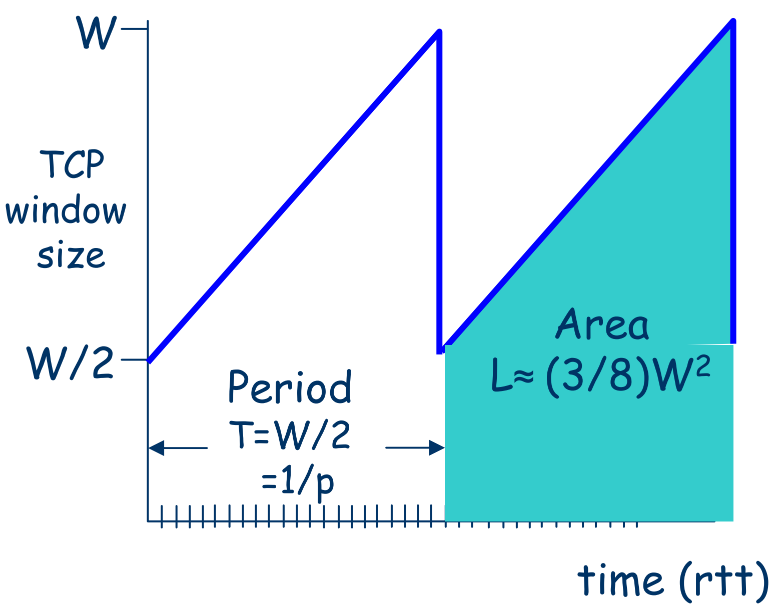
$$B = \frac{\text{Avg number of packets sent per cycle}}{\text{Avg duration of a cycle}}$$

A Simple Model for TCP Throughput [MSMO97]

Assumptions:

- Infinitely long TCP flow
- Periodic TD losses

$\Rightarrow$ window increases from W/2 to W at rate of one packet per RTT



Throughput:

$$B = \frac{L}{T} = \frac{(3/8)W^2}{RTT\, W/2} = \frac{1}{RTT\sqrt{(2/3)\,p}} \text{ pkts/sec}$$

Square root formula:

- Throughput is inversely proportional to RTT and p

PKFT Model [PKFT 98]

- Enhances the square root formula to account for
  - Timeouts
  - Receiver window
  - Delayed ACKs
- Correlated losses, drop-tail like behavior

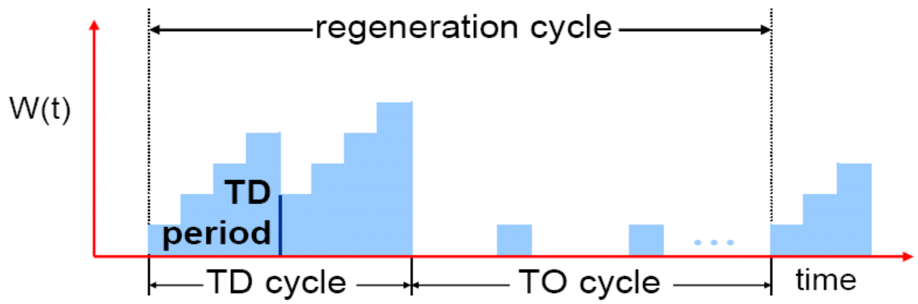
Throughput:

$$B(p) \approx \min\left(\frac{W_{\max}}{RTT}, \quad \frac{1}{RTT\sqrt{(2/3)bp} + T_0 \min(1, 3\sqrt{(3/8)bp})\, p(1+32p^2)}\right)$$

W_{max}: max. window size, T_0: initial TO interval, b: delayed ACK factor

- Validated using Internet measurements, and by many other studies
- Insensitive to TCP flavor

Analysis Technique



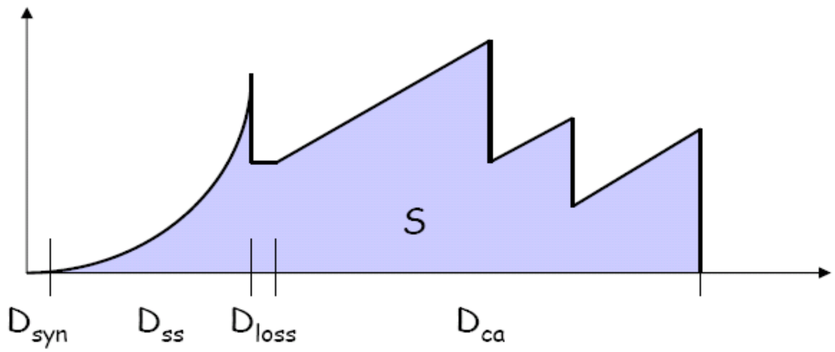
- Compute avg no. of TD periods per TD cycle
  - account for all possible events leading to TD
  - no. TD periods per TD cycle geometric r.v.
- Compute avg. length of TO cycle
  - no. timeouts geometric r.v.

Modeling TCP latency [CST00]

- Large portion of TCP flows are short-lived
  - For short transfers, TCP delay is dominated by slow-start
  - ⇒ PKFT formula may be inaccurate

- Model assumes finite size transfers (size S)
- Average latency:

  $D = D_{syn} + D_{ss} + D_{loss} + D_{CA}$
- Throughput: S/D



- For short transfers, large improvement in throughput prediction
- Further refined by [SKV01] to include independent losses

Markov Chain and TCP Vegas Models

- Markov chain approach allows more **"careful"** models
- Chain keeps track of TCP parameters, e.g., window size
  - Can be embedded at loss [K98] or window-size-change epochs [CM00]
- Little difference (specific environments?)

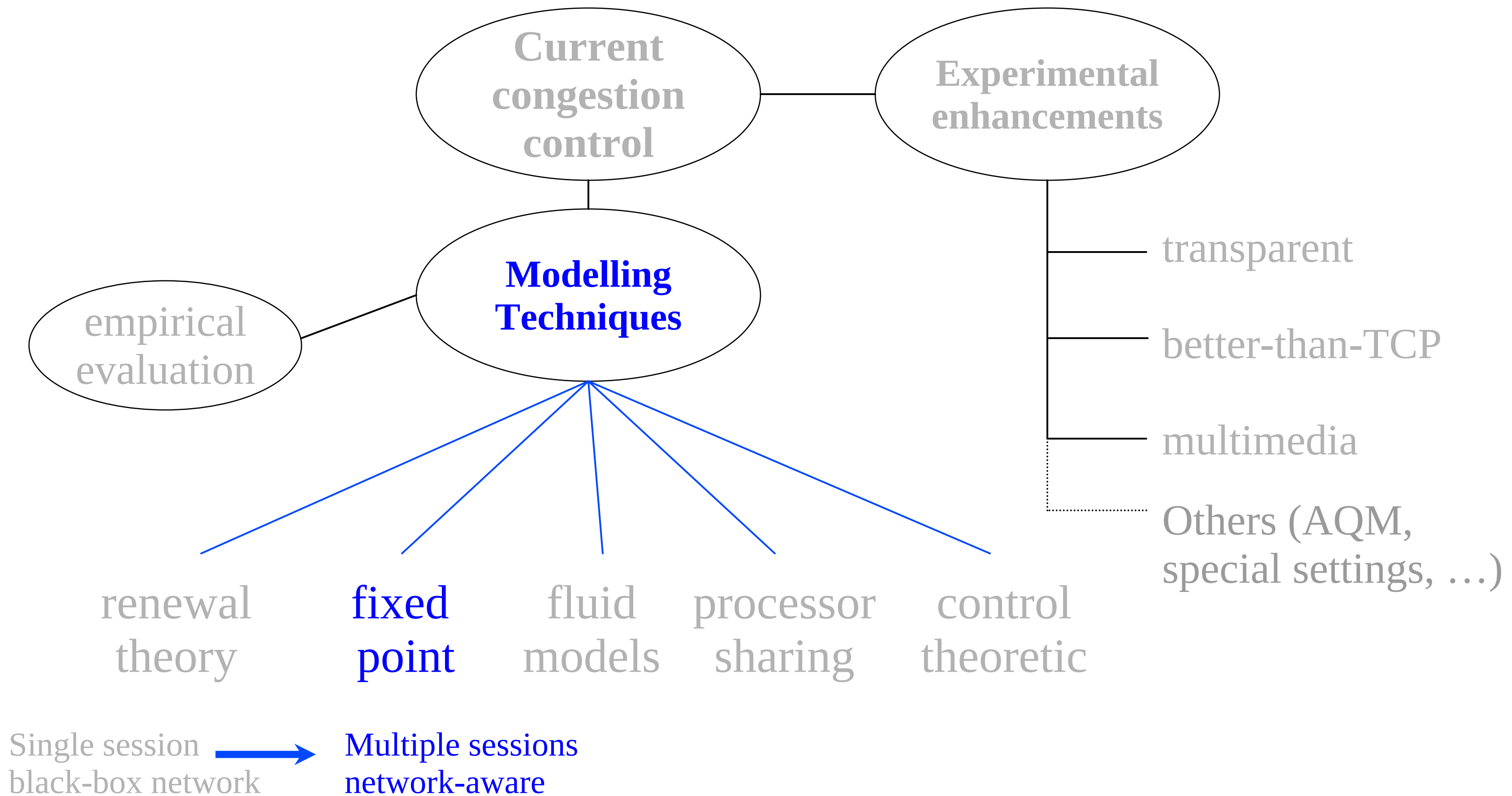
- [SV03]: modeled TCP-Vegas, which detects congestion based on no. of packets backlogged in network.
  - Simple model (similar to PKFT) that yields a closed-form expression
  - Reveals that Vegas's doesn't bias flows with large RTTs

So far: single session, black-box network models

- Lessons learned so far:
  - TCP's throughput appears to have a well-defined curve
  - Throughput is inversely proportional to RTT and p

- Problems with renewal based models:
  - Assume a single session and black-box network
    - E.g., requires knowledge of RTTs and loss rates

Outline: Renewal Theory Models

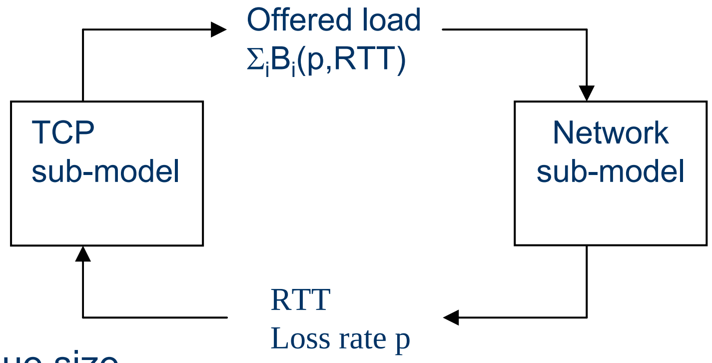


Fixed Point Models

- Network-aware method
  - Couples detailed TCP model with well known network model

- [FB00, BT01]: N flows going through a bottleneck router
  - Aggregated rate matches capacity
  - All flows see same loss prob

- Solve a fixed point problem for q
  $\Sigma_i B_i (RTT_i, p(q)) = C$
  $RTT_i = A_i + p(q)/C$



where $\mathbf{A_i}$ is propagation delay, $\mathbf{B}_i$ PKFT formula,
p(q) drop prob of AQM policy, **C** router capacity, **q** queue size

- Model is accurate in its predictions

Fixed Point Models

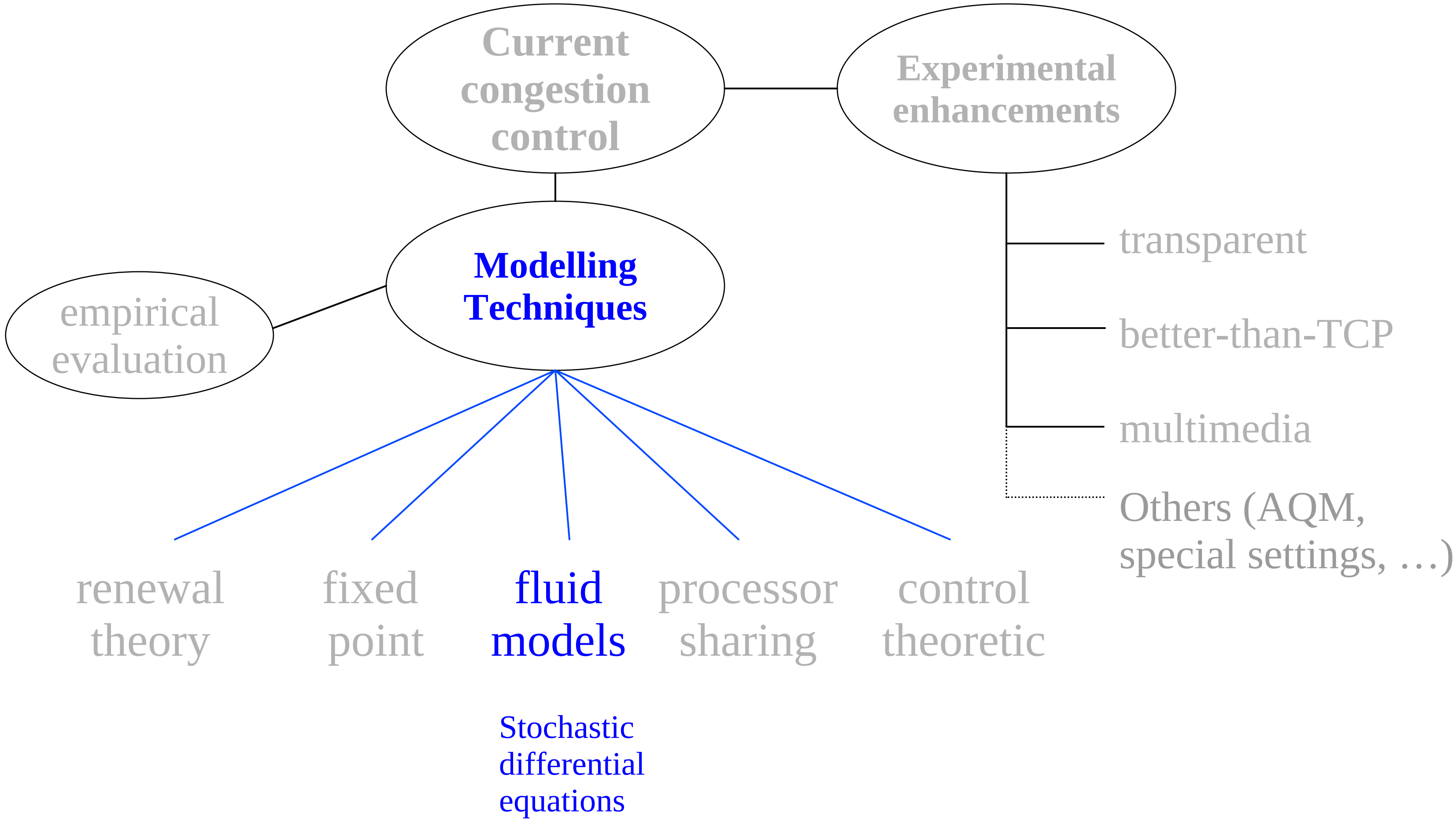
- [FB00]: showed that RED may be unstable
- [BT01]: extended method to a network of congested routers
- [CM00]: captures on-off application behavior (e.g., server activity); uses a Markov-based TCP model and a M/M/1 network model

Lessons

- Renewal theory models
  - Detailed models capable of distinguishing Drop-Tail/AQM variants
  - Single session, black-box network models

- Fixed point models
  - Multi session models that predict performance from natural in-parameters: network topology, no. of flows

Outline: Fluid Models



Fluid models [MGT99, MGT00]

- Model TCP as a fluid flowing through the network
- Losses are modeled by a Poisson process
  - Validated by WAN measurements
  - Poisson counter process N at rate λ:

$$dN = \begin{cases} 1 & \text{at possion event} \\ 0 & \text{elsewhere} \end{cases} \qquad E[dN] = \lambda$$

- Stochastic differential equation (SDE):

$$dW = dt/R - W/2\, dN_{TD} + (1-W) dN_{TO}$$

Additive increase | Mult. decrease | TO based decrease

λ_{TD}: triple-dup ACK Poisson process
λ_{T0}: timeout Poisson process
R: Round-trip time

- [MGT00] Closed loop model: Analysis of a network of AQM routers
  - Yields a system of differential equations, solved numerically
  - Captures transient performance of TCP
  - Insights on tuning RED parameters (flaw in RED avg mechanism)

More Fluid Models

- **Problem:** loss process in the internet can have a complex distribution (e.g., Poisson in WAN, Bursty in LAN)
- [AAC00]: SQRT formula is generalized to the case of stationary ergodic losses based on a fluid model
- Throughput: $$B = \frac{1}{RTT\sqrt{bp}}\sqrt{\frac{3}{2} + \frac{1}{2}V + \sum_{k=1}^{\infty}\frac{1}{2}^{k}C(k)}$$

V and C(k) are the variance and correlation of inter loss times, respectively.

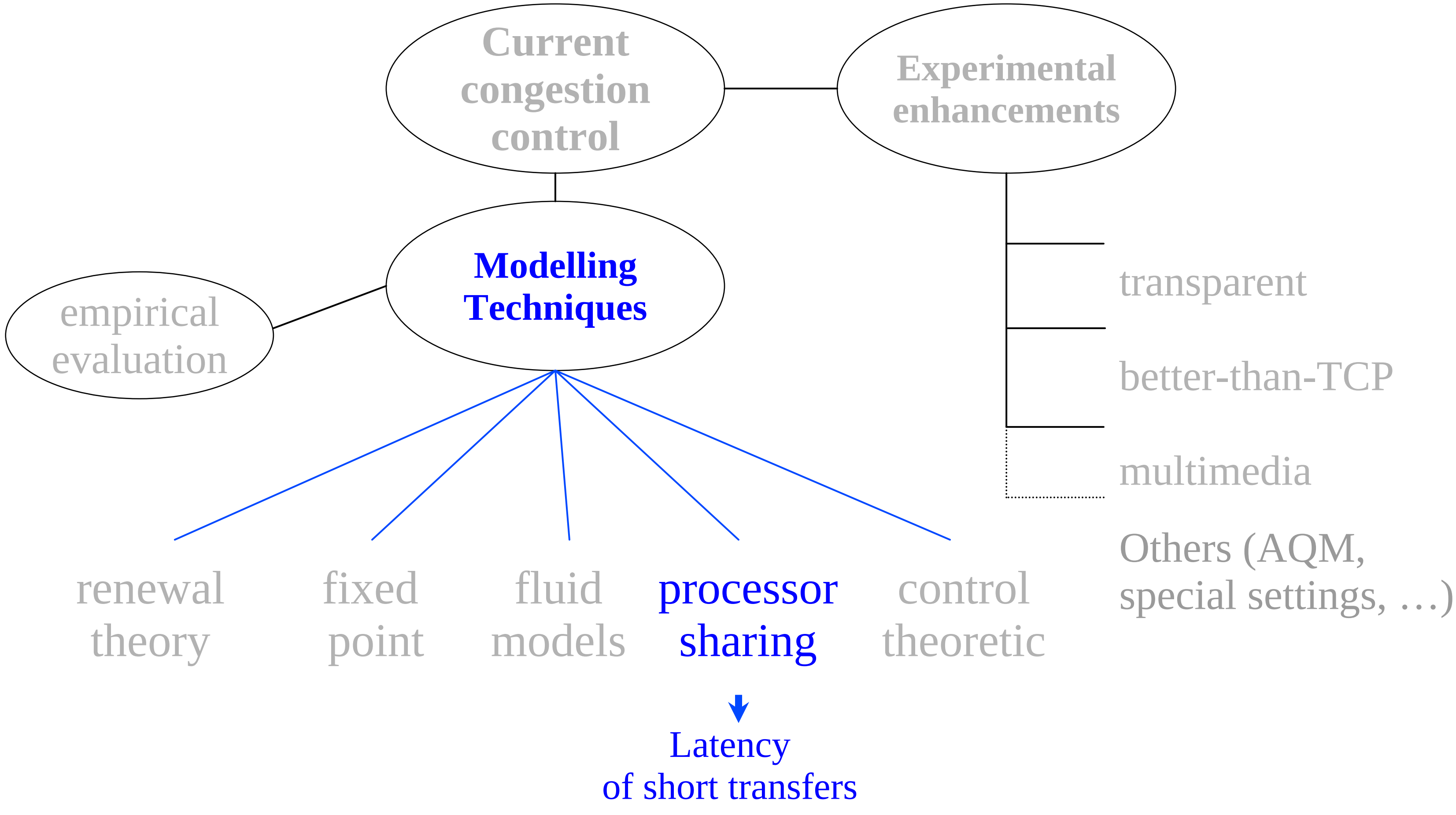
Parallel TCP Sockets [ABV06]

- Parallel TCP sockets used for bulk-data transfers
  - Throughput improvements, e.g., GridFTP
- Previous fluid model is extended to account for N TCP connections competing for bottleneck bandwidth
  - At each congestion event, a single connection is signalled to reduce rate
- Model yields a throughput formula for any given no. of flows (N)
  - Throughput-invariance (loss policy is irrelevant)
  - N = 1 : Utilization = 0.75 c
  - N = 3 : Util. > 90%
  - N = 6 : Util. > 95%

Lessons

- Fluid models
  - Accounts for the statistics of the inter-loss process
  - Provides insights on configuring AQM mechanisms
  - May not be suitable for detailed protocol modeling

Outline: Processor Sharing Models



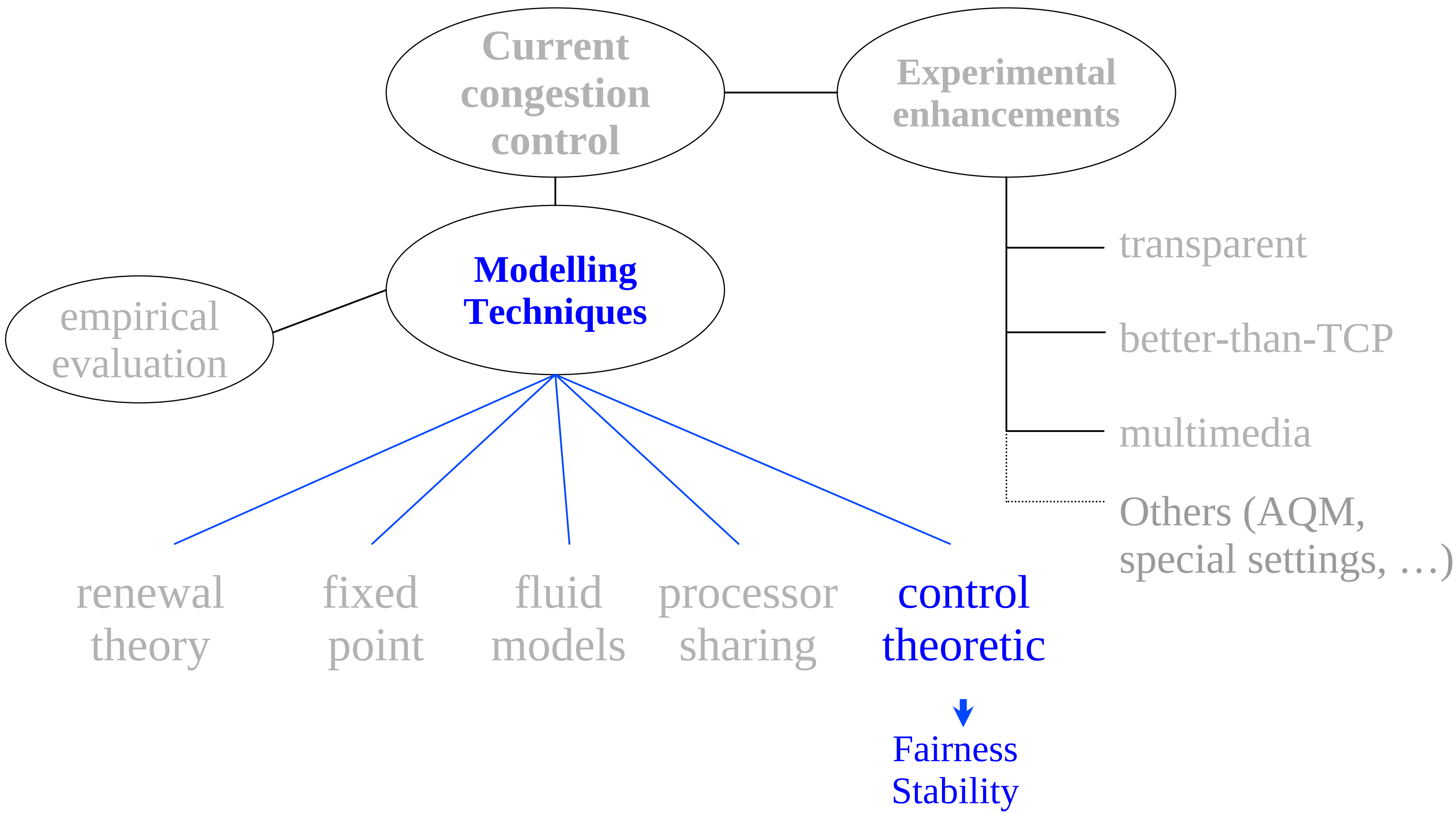
Processor Sharing Models [FBW01,BHM01]

- Focuses on short-lived connections:
  - Poisson arrivals of connections λ
  - Transfer size 1/μ
  - Single bottleneck link
- Can model as M/G/1 Processor Sharing queue
  - No. of simultaneous flows = No. of customers in queue
  - Download time = mean sojourn time in queue
- Upper limit on TCP's sending rate is captured by generalized processor sharing queue

Lessons

- Processor sharing models
  - Provide simple dimensioning guidelines
  - Model remains simple when extended
  - May be inaccurate for short transfers
  - Lacks high load results

Outline: Control Theoretic Models



Theoretical Foundations of Congestion Avoidance Mechanisms [CJ89]

- Assume distributed system
  - binary signal of congestion
  - x_i: rate after i-th feedback

- Simplest control strategy

$$x_i(t+1) = \begin{cases} a_I + b_I x_i(t) & increase \\ a_D + b_D x_i(t) & decrease \end{cases}$$

Design Space

	Additive **D**ecrease	**M**ultiplicative **D**ecrease
Additive **I**ncrease	AIAD ($b_I=b_D=1$)	**AIMD ($b_I=1$, $a_D=0$)**
Multiplicative **I**ncrease	MIAD ($a_I=0$, $b_I>1$, $b_D=1$)	MIMD ($a_I=a_D=0$)

- Which strategy? AIMD achieves conditions for both efficiency (bandwidth util.) and fairness (bandwidth is equally shared between competing flows)

$\Rightarrow$ AIMD: basic building block of most congestion control alg.,e.g., TCP

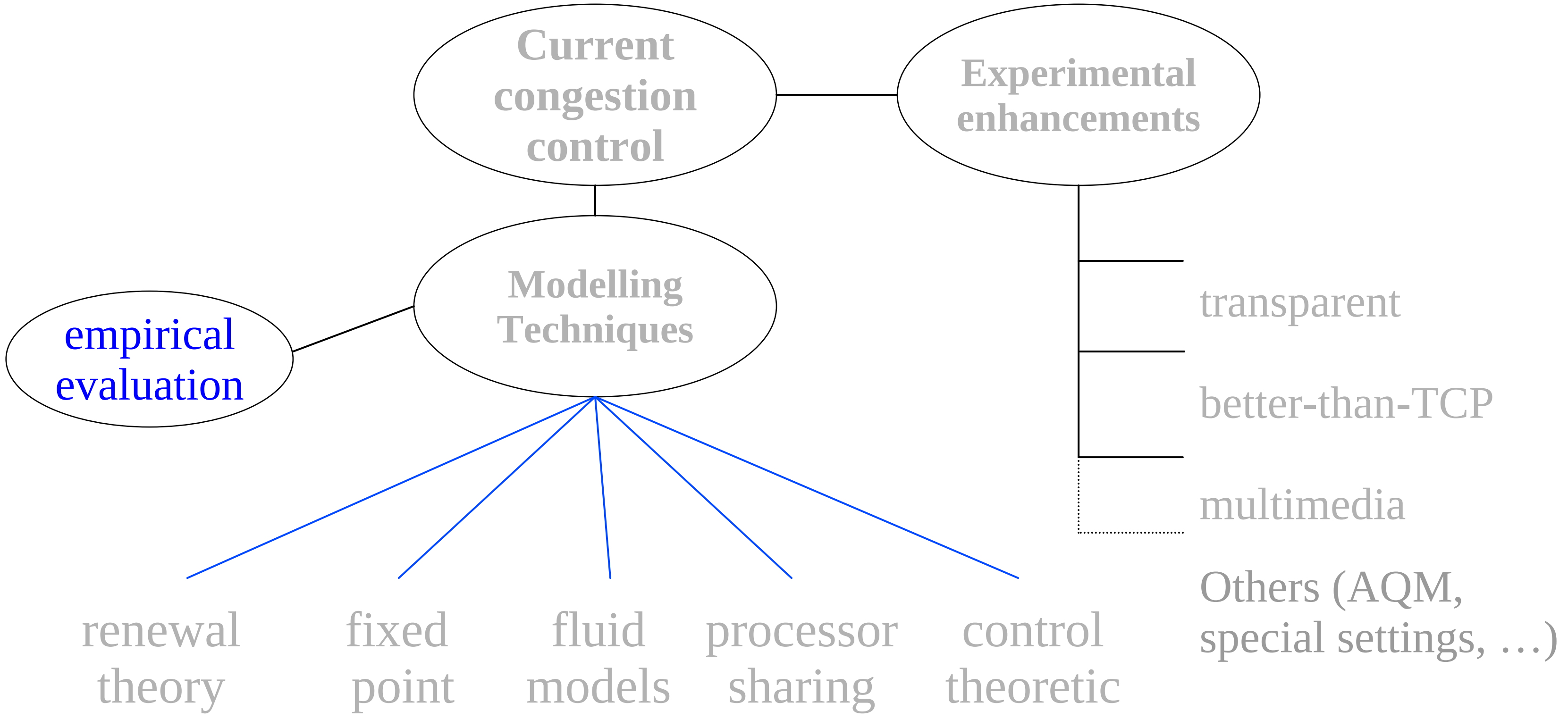
Control Theoretic Analysis of RED [HMTG01]

- TCP fluid model is analyzed from control theoretic viewpoint
- Linearization applied to analyze the non-linear system model
- Frequency domain analysis predicts system stability:
  - Decreases as number of flows decreases
  - Decreases as link capacity increases
  - Decreases as RTT increases

Lessons

- Control theoretic models
  - Can leverage well established stability and convergence analysis techniques
  - Allows design of new congestion control and AQM schemes
  - Less suitable for modeling transfer of files from general distribution due to the transient results obtained

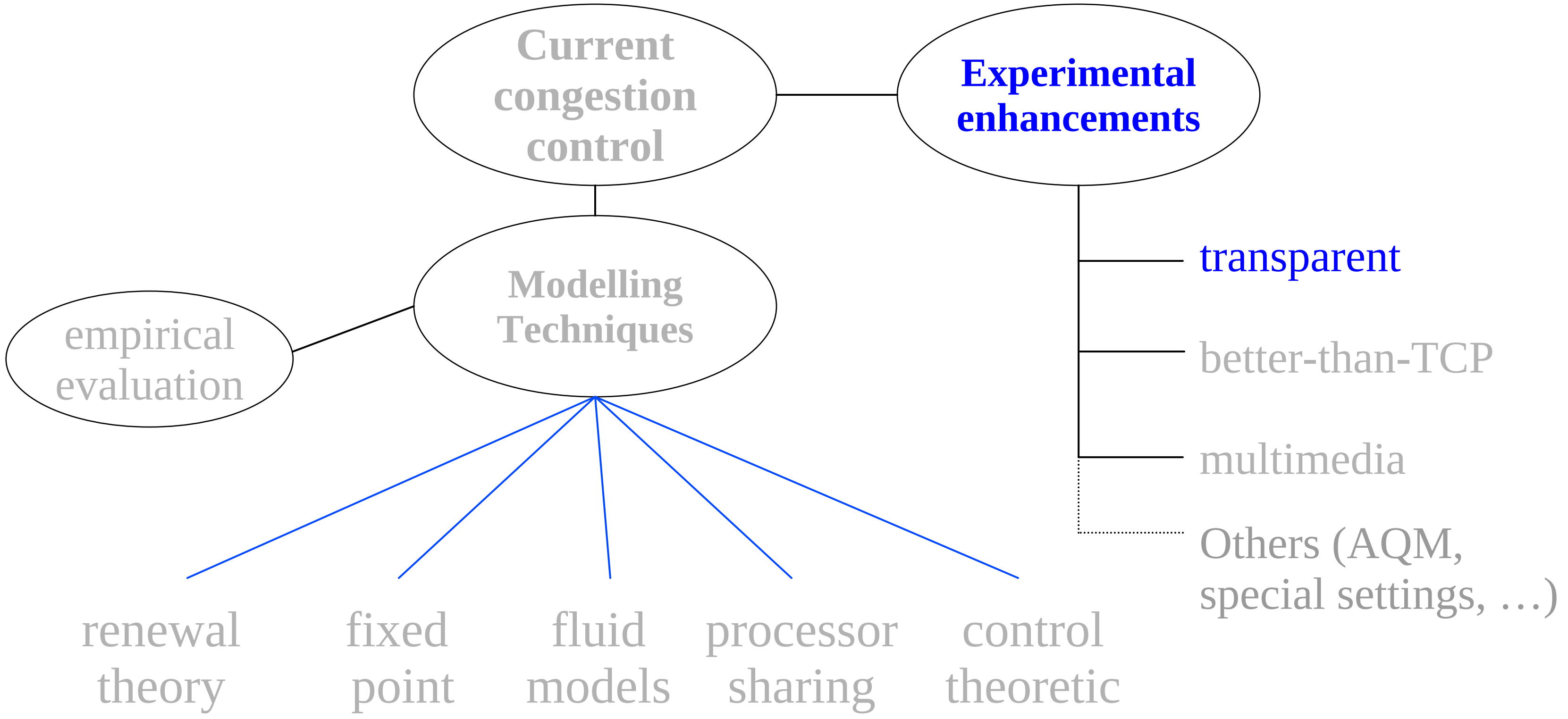
Outline: Empirical evaluation of TCP



Inferring TCP Characteristics [JIDKT03, JIDKT04]

- Crucial for understanding operation of deployed protocols (TCP)
- Variety of approaches
  - Active vs. passive
  - Where measurements taken: edge vs. routers
  - What metrics: loss, delay, per hop vs. per path
- Papers provide new methodologies and measurements:
  - out-of-sequence classification
  - tracking cwnd, TCP flavors
  - RTT estimation
- Uses passive measurements at single router
  - main challenge: incomplete observability

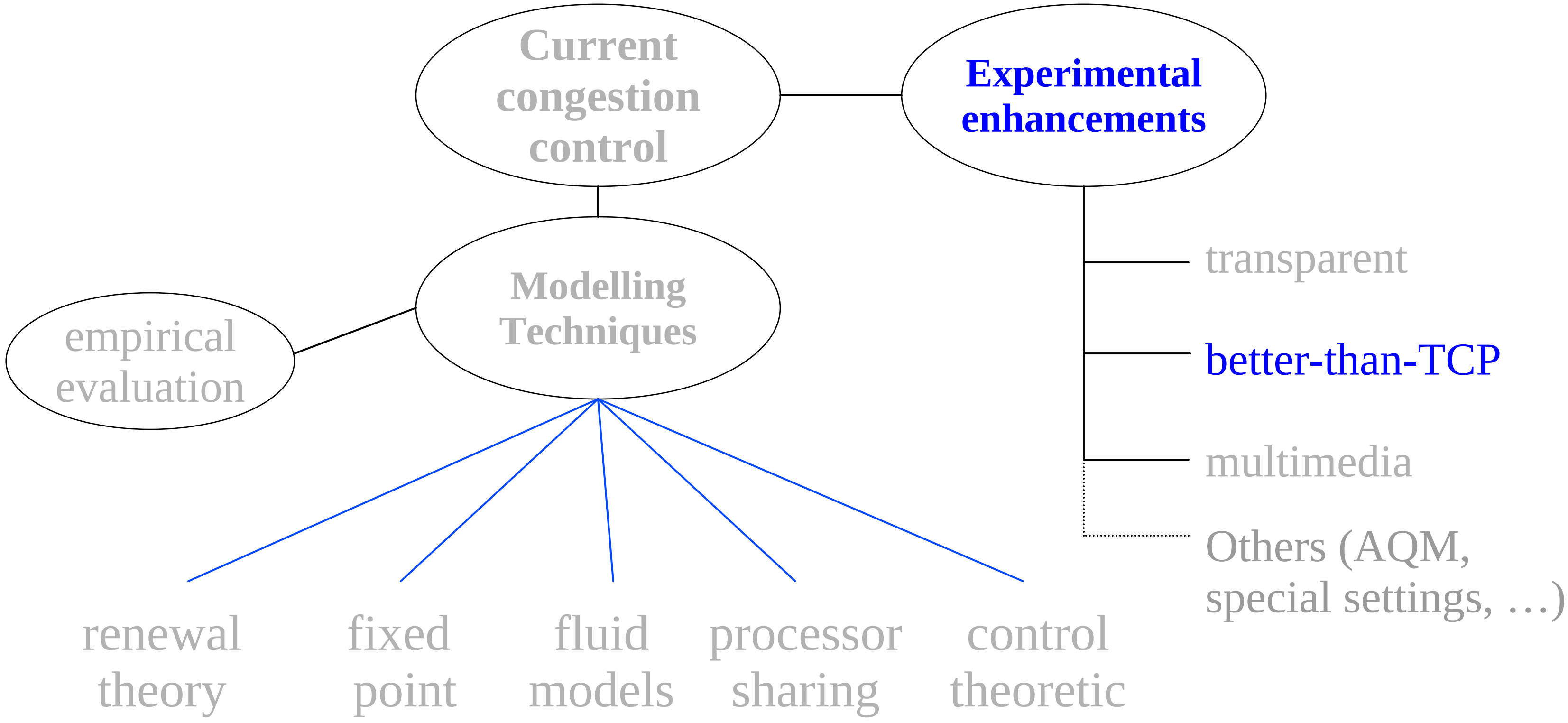
Outline: Control Theoretic Models



Performance of TCP Pacing [AST00]

- TCP is bursty (slow start, losses, ack compression, etc.)
- Bursty traffic is undesirable since it produces:
  - Higher queuing delays and losses
- A natural solution is to evenly space, or “pace”, TCP packets over an entire round-trip time
- Contribution: quantitatively evaluate the impact of pacing
  - Pacing improves fairness and drop rates when buffering is limited
  - In other cases: pacing leads to performance degradation
    - Due to mixing of traffic, synchronizes drops occur.

Outline: Control Theoretic Models



Probe Control Protocol (PCP) [ACKZ06]
Efficient Endpoint Congestion Control

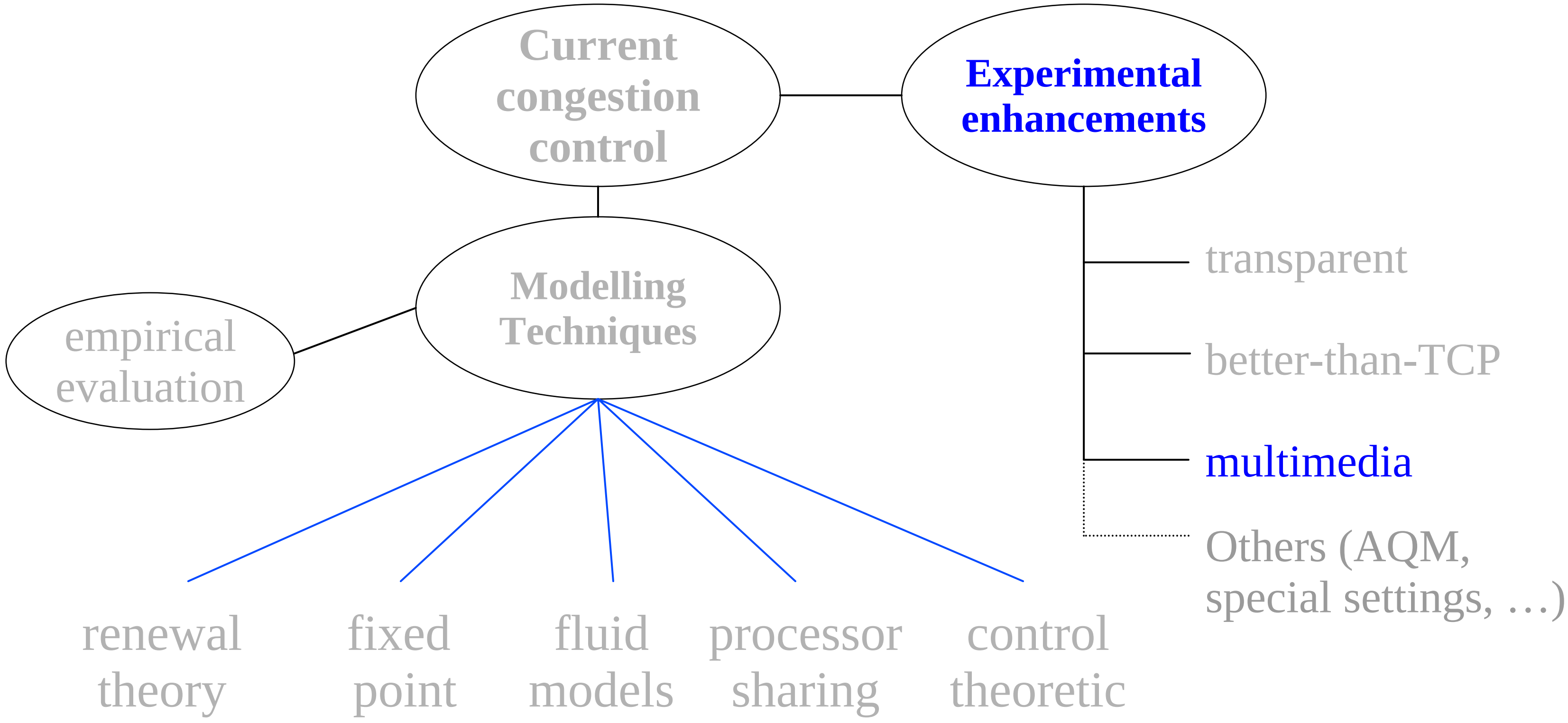
- TCP allocates resources without requiring network support
  - Uses "Try and Backoff" strategy
  - **Problem:** link capacity is not fully utilized for short and medium flows
- Network assisted congestion control
  - Routers provide feedback to end-systems
  - Routers explicitly allocate bandwidth to flows
  - **Problem:** makes routers complicated

Design Space

	Endpoint	Router Support
Try and Backoff	TCP, Vegas, RAP, FastTCP, Scalable TCP	DecBit, ECN, RED, AQM
Request and Set	**PCP**	ATM, XCP, WFQ, RCP

- How to improve performance in all likely circumstances?
- **Solution:** emulate network-based control by explicit short probes
- Initial results: PCP outperforms TCP by an avg factor of 2 for 200k transfers (with min impact on TCP traffic)

Outline: Control Theoretic Models



Multimedia Congestion Control

- TCP's congestion control may be inappropriate for real-time applications:
  - Rate adaptations may be unnecessarily severe
  - TCP reliability mechanism may incur additional delay
- Congestion control for multimedia streaming over UDP
  - Maintain same long term rate as TCP (TCP-friendly)
  - Smoother rate variations than TCP
  - [FHPW00] TFRC: TCP-Friendly rate control protocol
    - Uses TCP throughput formula (PFTK) as its control equation
    - Shown to coexists well with many kinds of TCP traffic of different flavors across various settings

The TCP-Friendliness of VoIP Traffic [BLT06]

- The stability of the current Internet is largely maintained by TCP
- Q: with the increase in VoIP users, are we facing an increasing danger of congestion collapse?
- A: Probably not since VoIP may be viewed as TCP-Friendly due to user back-off
  - User back-off: call drop due to unacceptable user-perceived quality
- Solution technique: use TCP and VoIP models to evaluate how bandwidth is shared among VoIP flows and TCP flows.
  - User back-off is quantified by approximating call drop probability as a function of network loss and delay using subjective test results.

Conclusions

- Overview of the main techniques for modeling TCP
- Further challenges
  - TCP's performance in specific environments
    - E.g., paths where the window size and the RTT are correlated
  - Analysis of multimedia streaming over TCP
  - Need to better understand how to model internet losses:
    - Is it Bernoulli? is it Poisson? Is it in bps or pbs?
  - New applications: design routing scheme based on TCP's throughput?
- And finally, perhaps the simplest models are the most useful ones…

Questions?