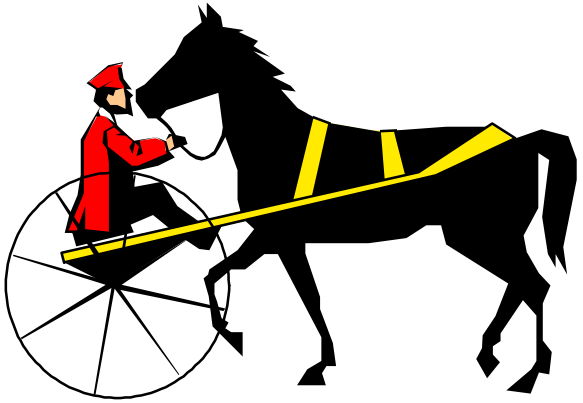
94-0882

Rate-Based Schemes: Mistakes to Avoid



Raj Jain, Shiv Kalyanraman, Ram Viswanathan
Department of Computer and Information Sci.
The Ohio State University
Columbus, OH 43210
Jain@ACM.Org

Disclaimers

- This presentation is in support of the Rate-based approach not against it.
- Some versions of rate-based approaches may have problems but they are not inherent to the approach.
- It is possible to design good or bad schemes with either rate or credit approach.
- Good simple rate-based schemes exist.

- When is the system overloaded?
- What control is the feedback related to?
- How to indicate your rate?
- How to achieve fairness?

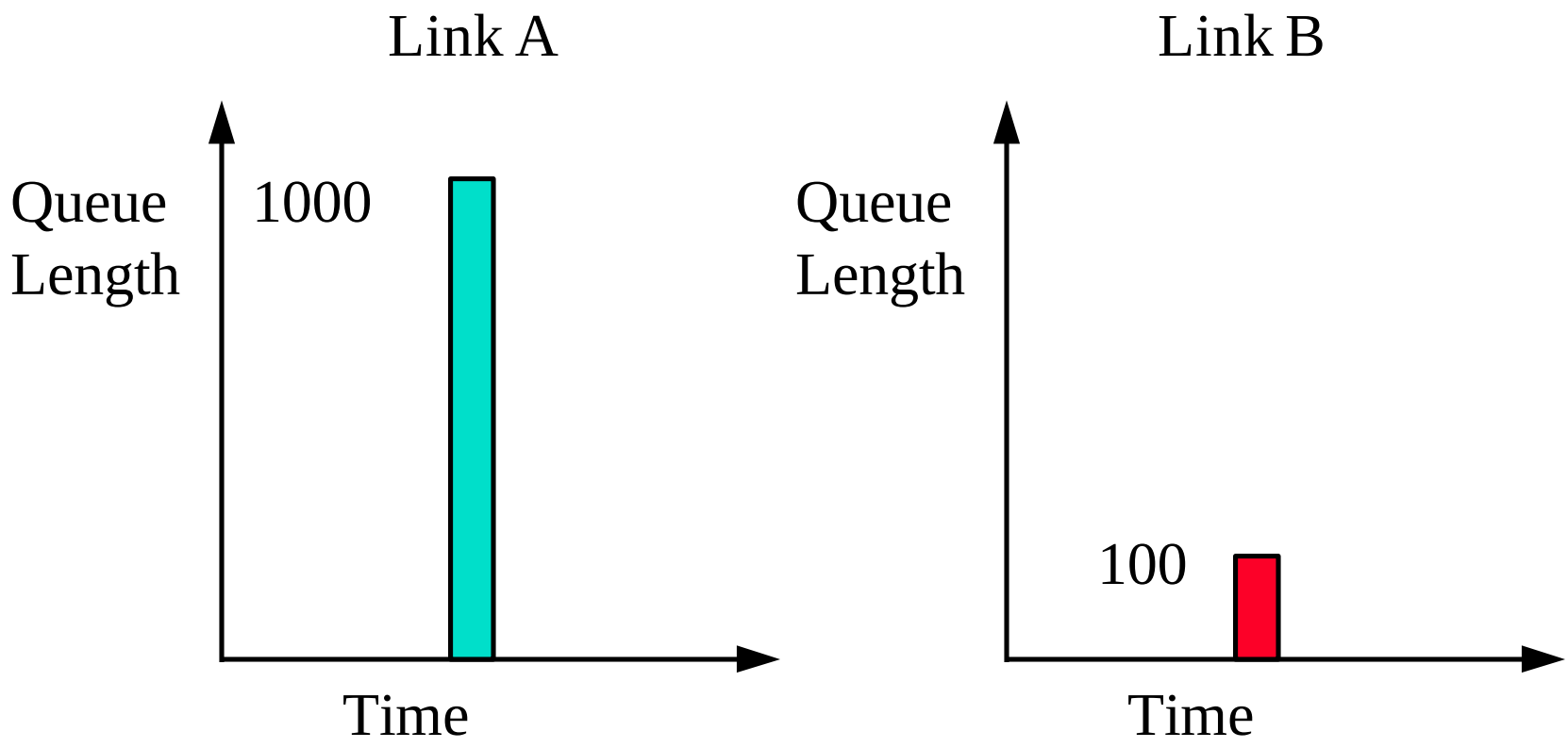
Fundamental Principles

- Information
  - Not using it ⇒ Worse performance
  - Using it correctly ⇒ Better performance
  - Misusing it ⇒ Disasters

Example: If a link is loaded by a factor of 2.5

- The link is overloaded
- The link is loaded by a factor of 2.5
- The link is not loaded

Which Link is More Overloaded?

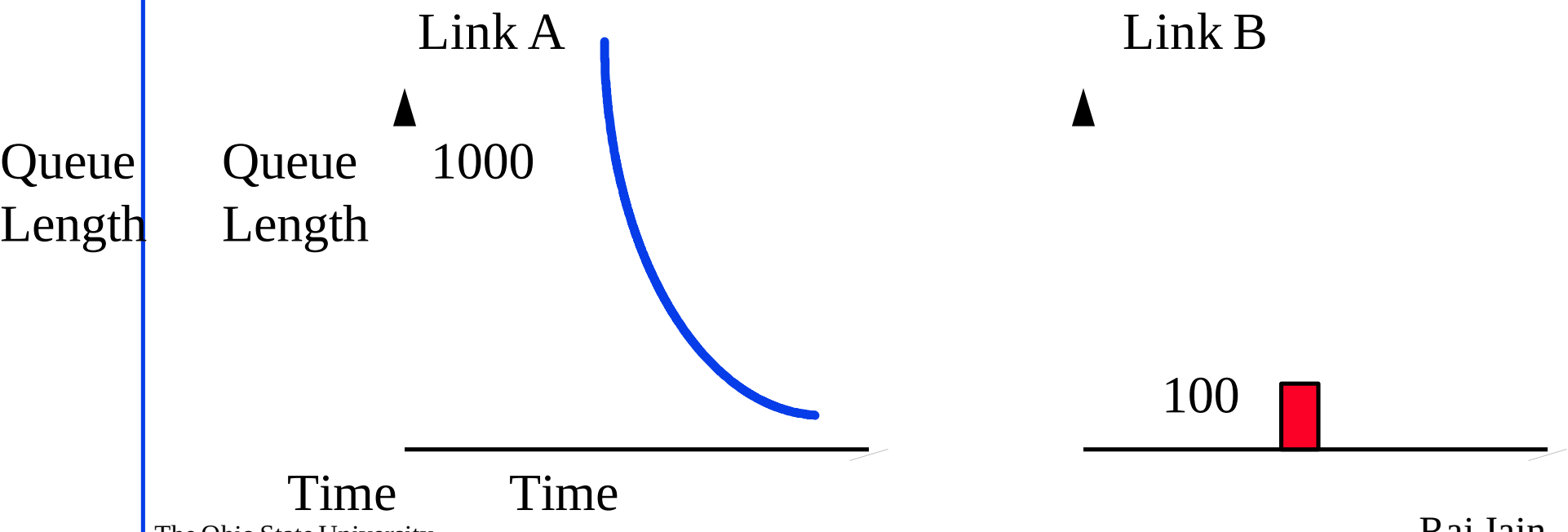


Answer: It Depends!

Control: Rate or Window?

Q = Window, dQ/dt = Rate

For Rate Control: Monitor Q growth rate



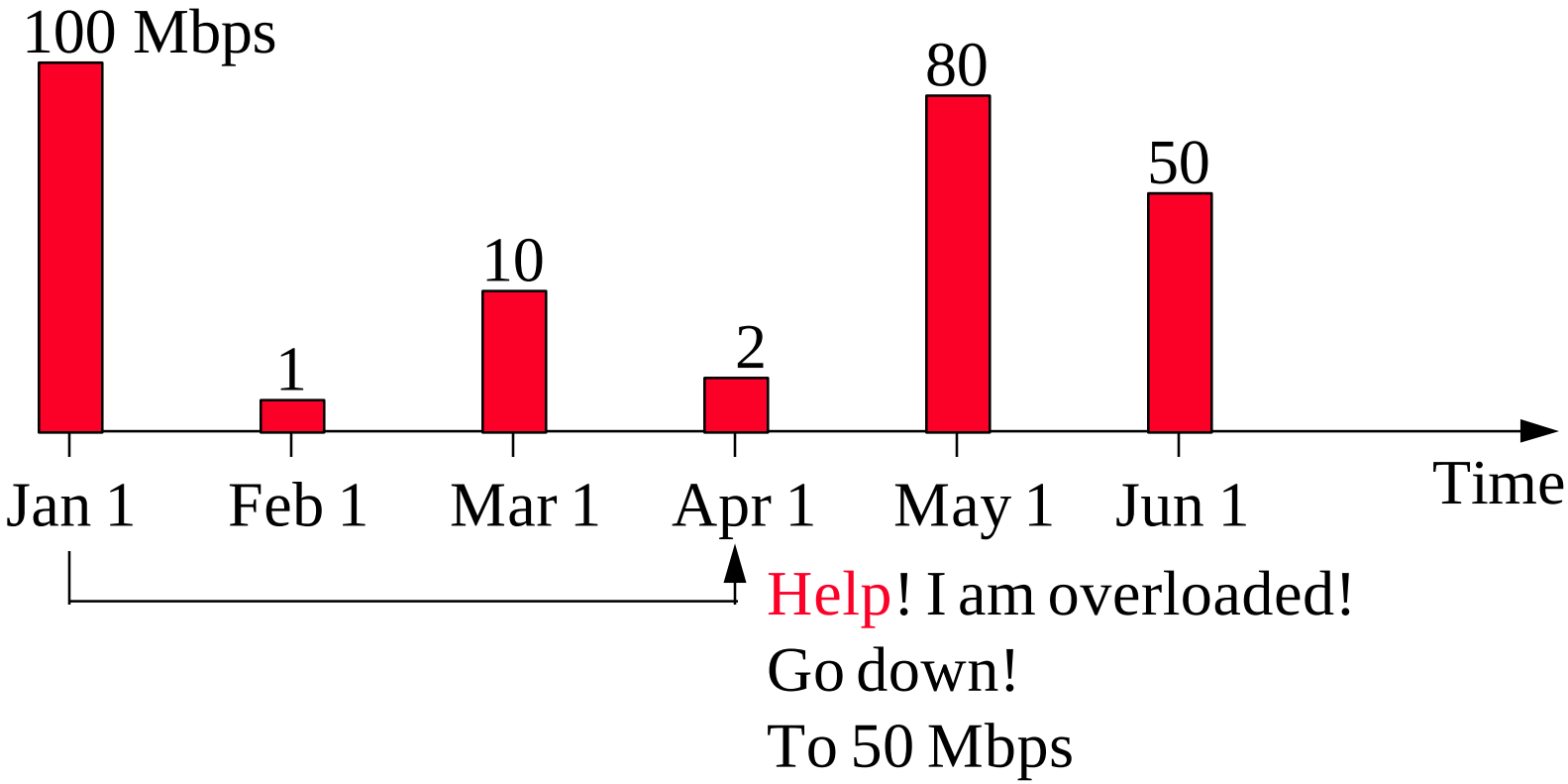
Link Speed: OC-12 or T1?

Conclusions I

- Instantaneous queue length is not a good indicator of load for a rate controlled system. Q(t) = Q(t-1) + Input rate - Service rate
- Using queue length as the load indicator in a rate controlled system leads to unnecessary oscillations.
- Input rate monitoring not only correctly tells whether the system is overloaded, it also tells by what factor.

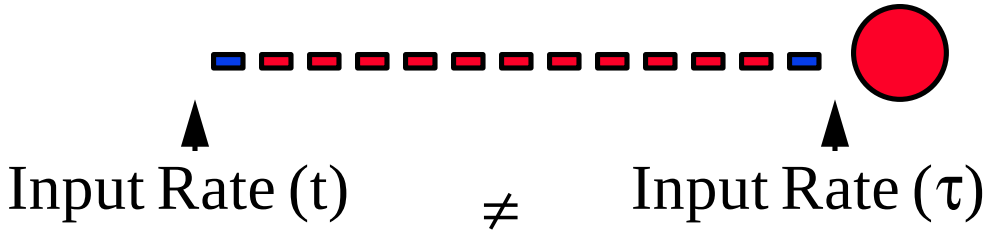
Control-Feedback Relation

- Long feedback delay + Dynamic control



Conclusions II

- It is important to indicate which control the feedback is related to.
- Switch should correctly correlate the source control with the feedback.



- Unrelated feedback messages result in confusion (oscillations).

How to Indicate Rate?

- Both streams are 5 cells/s but different.
- For constant bit rate traffic, rate can be specified by one number.
- For bursty traffic, rate has to be specified by at least two quantities. For example,
  Leaky bucket uses bucket size and PCR
  However, leaky buckets are non-additive
  $\text{Bucket}(\text{PCR}_1, \text{Size}_1)+\text{Bucket}(\text{PCR}_2, \text{Size}_2) = ?$

Conclusions III

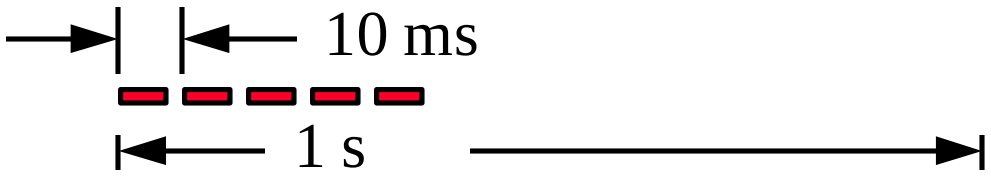
- A scheme based on single-number specification will work for infinite sources but not for bursty sources.
- Need control specifications that are additive.
  - Example 1: (r, T) smooth traffic.
    r cells per T secs

$(r_1, T_1) + (r_2, T_2) = (r, T)$

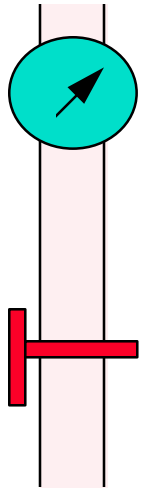
Where $r = r_1/T_1 + r_2/T_2$, $T = GCD(T_1, T_2)$

OSU Rate Specification

- Offered average Cell Rate (OCR)



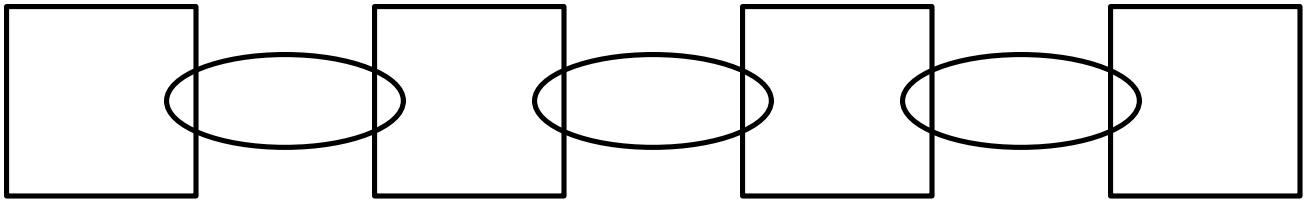
  - Averaged over a suitable interval
  - Is additive: 10 cps + 20 cps = 30 cps
  - But not easy to control
- Transmitted Cell Rate (TCR)
  - 1/Inter-cell time
  - Good for control



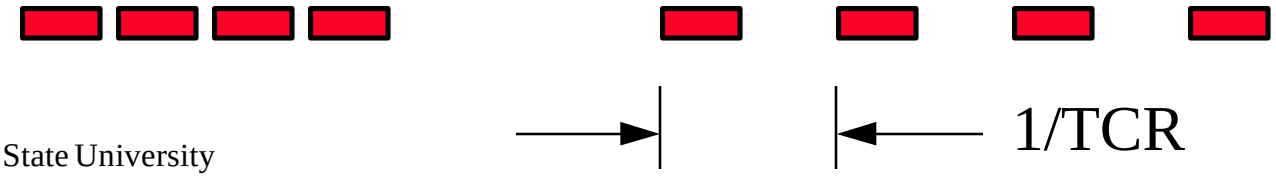
Alphabet Soup of Cell Rates

- ACR = Average, Allowed, Actual
- GCR = Generalized
- ICR = Initial
- MCR = Minimum
- MACR = Maximum
- PCR = Peak
- SCR = Sustained

Segment-by-Segment Version



- TCR, OCR, T $\Rightarrow$ Credit of OCR $\times$ T
  - No Loss
  - per-VC or some VCs
  - per-hop, multiple hops, or only lossy hops
  - Not bursty

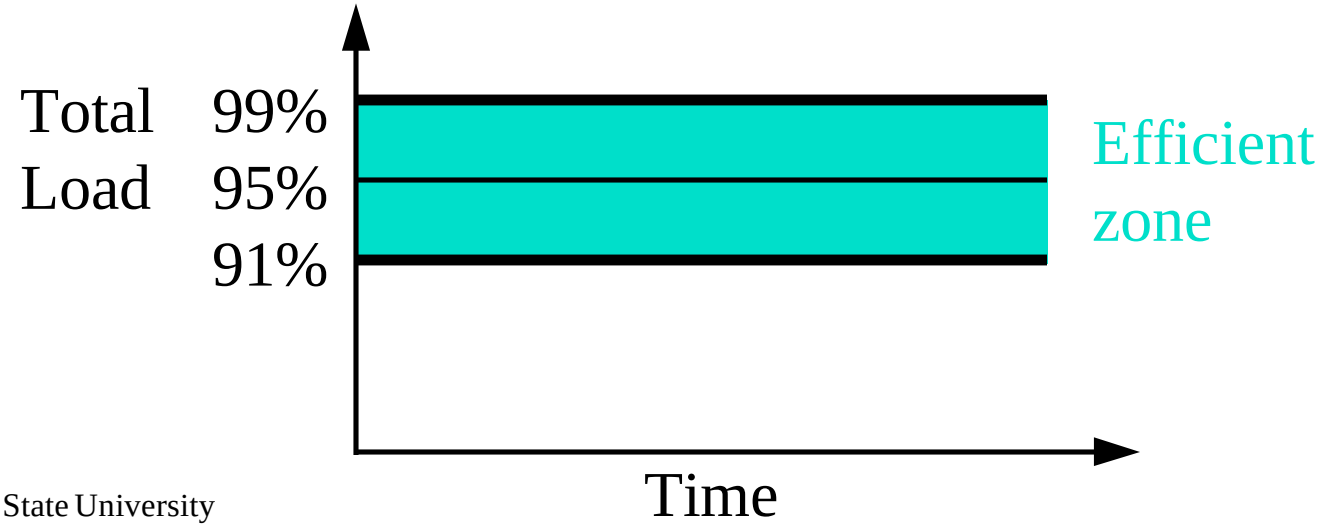


Fairness

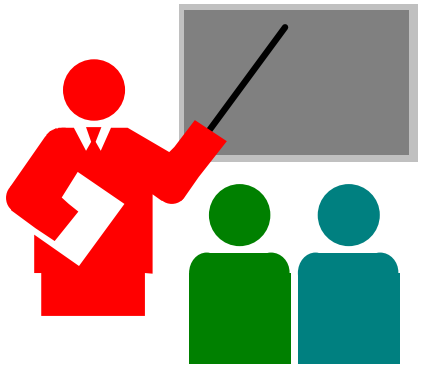
- Principle: Fairness cannot be achieved without per-VC accounting, queueing, monitoring, scheduling, or service.
- Credit based does per-VC queueing, monitoring, round-robin service at every switch.
- Rate-based: Average rate does not change

Fairness: OSU Solution

- Move monitoring to the source or entry-switch
- Feedback is a function of a VC's load
  Particularly, in efficient zone
- May or may not use all VC's individual rates



OSU Design Principles For Rate-Based Approach



- Monitor queue growth rate not queue length
- All feedback messages are related to control
- Use two rates: Instantaneous + Average
- Provide different feedback in the efficient zone