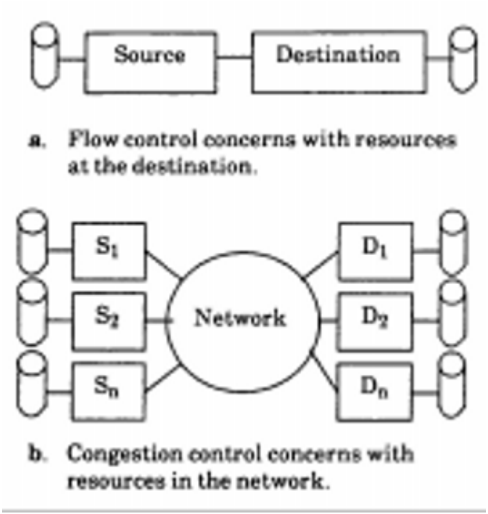
Analysis of the Increase and Decrease Algorithms for Congestion Avoidance in Computer Networks

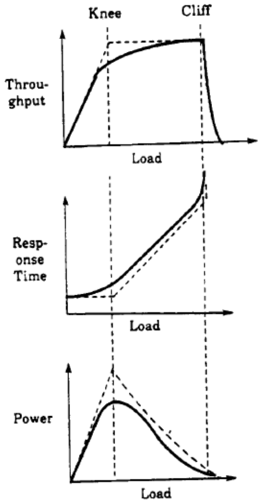
Dah-Ming Chiu, Raj Jain

Presented by: Ashish Vulimiri

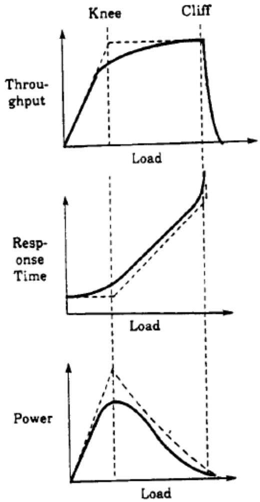
Congestion Control



Congestion Avoidance

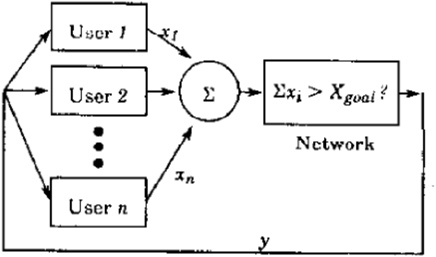


Congestion Avoidance



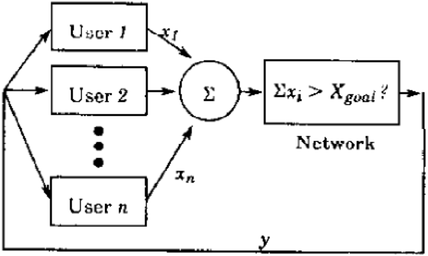
Congestion control vs congestion avoidance

System Model



- Single shared bottleneck resource
- Synchronous feedback
- Control applied by all users
- Feedback type: binary

System Model



- Feedback mechanism: set signal bit on packets, processed by destination
  - Rejected alternatives: extra signal packets, induce routing changes, signal bit processed by source
  - TCP Tahoe (and derivatives): signal = packet loss

Feedback

- Binary feedback

$$y(t) = \begin{cases} 0 & \Rightarrow \text{Increase load} \\ 1 & \Rightarrow \text{Decrease load} \end{cases}$$

- $x_i(t+1) = x_i(t) + u_i(t)$
- u_i can be an arbitrary function of $x_i(t)$ and $y(t)$
- We focus on a linear model

$$x_i(t+1) = \begin{cases} a_I + b_I x_i(t) & \text{if } y(t) \Rightarrow \text{Increase} \\ a_D + b_D x_i(t) & \text{if } y(t) \Rightarrow \text{Decrease} \end{cases}$$

Criteria

Criteria

1. Efficiency: load must be as close to knee point as possible

Criteria

1. Efficiency: load must be as close to knee point as possible
2. Fairness
   - In a perfectly fair system, $\forall i \forall j x_i = x_j$
   - We use a fairness criterion:

$$F = \frac{\left(\sum x_i\right)^2}{n\left(\sum x_i^2\right)}$$

   - F lies between 0 and 1, higher values indicating greater fairness

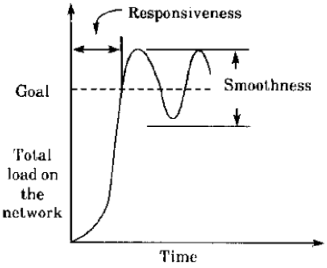
Criteria

1. Efficiency: load must be as close to knee point as possible
2. Fairness
   - In a perfectly fair system, $\forall i \forall j x_i = x_j$
   - We use a fairness criterion:

   $$F = \frac{\left(\sum x_i\right)^2}{n\left(\sum x_i^2\right)}$$

   - F lies between 0 and 1, higher values indicating greater fairness
3. Distributedness
   - Users do not know anything about state of system
   - Resource delivers only binary feedback to limit overhead

Criteria



4. Convergence
   - Constant-state convergence unlikely since we use only binary feedback
   - We use two parameters: responsiveness (how quickly the steady state is reached), and smoothness (size of oscillations)

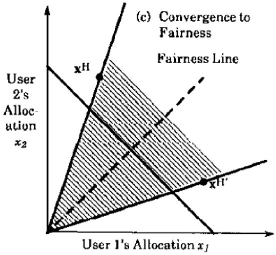
Constraints

Constraints: Fairness

- Fairness must improve in at least one of increase and decrease, and not degrade in either
- This implies: both $\frac{a_I}{b_I}$ and $\frac{a_D}{b_D}$ must be non-negative, and at least one must be positive

Constraints: Fairness

- Fairness must improve in at least one of increase and decrease, and not degrade in either
- This implies: both $\frac{a_I}{b_I}$ and $\frac{a_D}{b_D}$ must be non-negative, and at least one must be positive

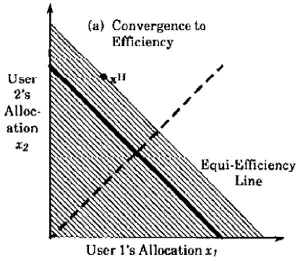


Constraints: Efficiency

- Feedback must be negative

Constraints: Efficiency

- Feedback must be negative



Efficiency: Example

- Consider a 2-user system with $x_1(t) = x_2(t) = 100$

Efficiency: Example

- Consider a 2-user system with $x_1(t) = x_2(t) = 100$
- Let $y(t) = \mathit{decrease}$, $a_D = 10$, $b_D = 1/2$

Efficiency: Example

- Consider a 2-user system with $x_1(t) = x_2(t) = 100$
- Let $y(t) = \mathit{decrease}$, $a_D = 10$, $b_D = 1/2$
- Then $x_1(t+1) = x_2(t+1) = 60$

Efficiency: Example

- Consider a 2-user system with $x_1(t) = x_2(t) = 100$
- Let $y(t) = \mathit{decrease}$, $a_D = 10$, $b_D = 1/2$
- Then $x_1(t+1) = x_2(t+1) = 60 \rightarrow$ -ve feedback

Efficiency: Example

- Consider a 2-user system with $x_1(t) = x_2(t) = 100$
- Let $y(t) = decrease$, $a_D = 10$, $b_D = 1/2$
- Then $x_1(t+1) = x_2(t+1) = 60 \rightarrow$ -ve feedback
- Suppose instead $x_1(t) = x_2(t) = 10$

Efficiency: Example

- Consider a 2-user system with $x_1(t) = x_2(t) = 100$
- Let $y(t) = \textit{decrease}$, $a_D = 10$, $b_D = 1/2$
- Then $x_1(t+1) = x_2(t+1) = 60 \rightarrow$ -ve feedback
- Suppose instead $x_1(t) = x_2(t) = 10$
- Then $x_1(t+1) = x_2(t+1) = 15$

Efficiency: Example

- Consider a 2-user system with $x_1(t) = x_2(t) = 100$
- Let $y(t) = \textit{decrease}$, $a_D = 10$, $b_D = 1/2$
- Then $x_1(t+1) = x_2(t+1) = 60 \rightarrow$ -ve feedback
- Suppose instead $x_1(t) = x_2(t) = 10$
- Then $x_1(t+1) = x_2(t+1) = 15 \rightarrow$ +ve feedback

Constraints: Efficiency + Distributedness

- Feedback must be negative no matter what the load distribution at the other users
- This implies

$$a_I > 0 \qquad b_I \geq 1$$

$$a_D = 0 \qquad 0 \leq b_d < 1$$

- Constraints can be loosened if bounds on X_{goal} and number of users are known

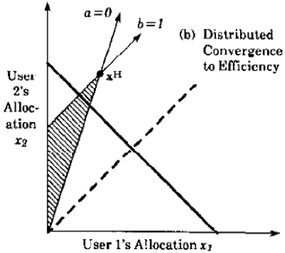
Constraints: Efficiency + Distributedness

- Feedback must be negative no matter what the load distribution at the other users
- This implies

$$a_I > 0 \qquad b_I \geq 1$$

$$a_D = 0 \qquad 0 \leq b_d < 1$$

- Constraints can be loosened if bounds on X_{goal} and number of users are known

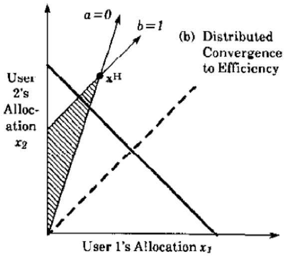


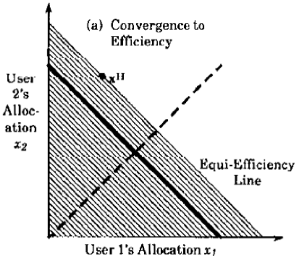
- Feedback must be negative no matter what the load distribution at the other users
- This implies

$$a_I > 0 \qquad b_I \geq 1$$

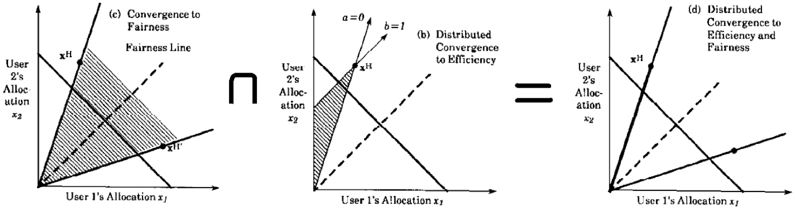
$$a_D = 0 \qquad 0 \leq b_d < 1$$

- Constraints can be loosened if bounds on X_{goal} and number of users are known

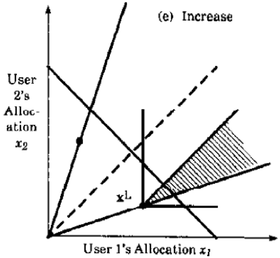




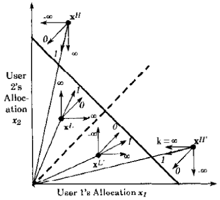
Combined Constraints
Decrease



Linear decrease must be purely multiplicative



- Increase must have an additive component, and optionally a multiplicative component
- Multiplicative component 0 for optimal fairness convergence



- Suppose

$$x_i(t+1) = x_i(t) + \sum_{k=-\infty}^{\infty} \alpha_k \left(x_i(t)\right)^k$$

- Can derive constraints on α_k and k by evaluating the criteria as done for linear feedback
- Advantage: more flexibility
- Disadvantage: less robust – higher sensitivity to system parameters like X_{goal} and n

Unresolved Issues

- Effect of delayed feedback
- Utility of non-binary feedback
- X_{goal} can't be known, but should users attempt to guess n?
- Asynchronous feedback

Other Issues

- Effect of packet loss due to errors
- Fault-resillience
- Can (and should) a scheme tailored to Internet traffic patterns be designed?

Other Issues

- Effect of packet loss due to errors
- Fault-resillience
- Can (and should) a scheme tailored to Internet traffic patterns be designed?
- And . . . ?