RNAP: A Framework for Congestion-based Pricing and Charging for Adaptive Multimedia Applications

Xin Wang and Henning Schulzrinne
Internet Real -Time Laboratory
Columbia University
http://www.cs.columbia.edu/~xinwang/RNAP.html

Outline

- Motivation
- Objectives
- Dynamic resource negotiation: architectures, messages, aggregation
- Pricing schemes
- User request adaptation
- Simulation
- Conclusions

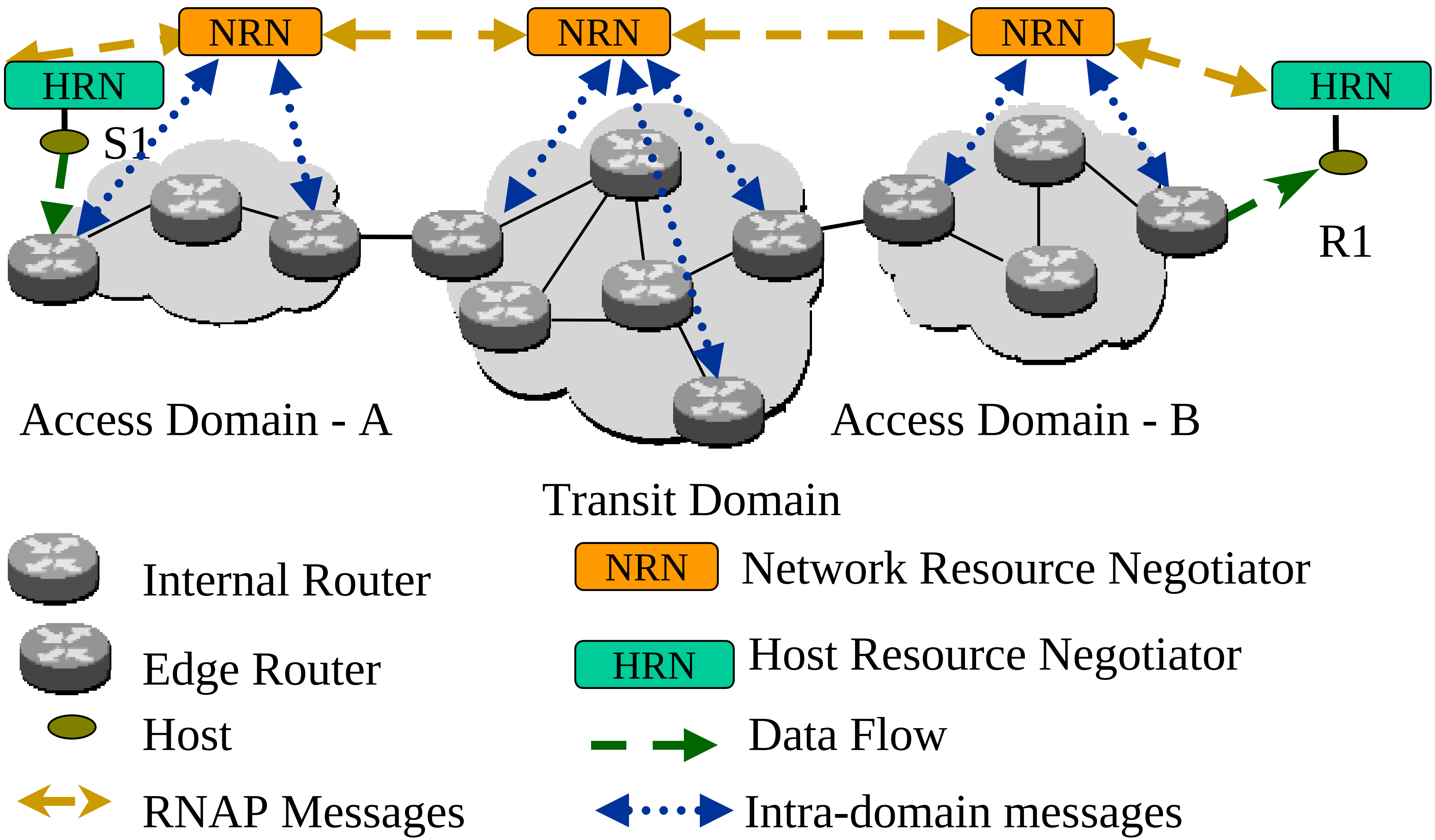
Motivation

- Current approaches for quality support
  - Resource reservation, admission control, differentiated services
    - Pros: QoS expectation
    - Cons: insufficient knowledge on data traffics, conservative, network dynamics not considered, lacks pricing support for multiple service levels
  - Multimedia adaptation to network conditions
    - Pros: efficient bandwidth usage
    - Cons: users have no motivation to adapt requests

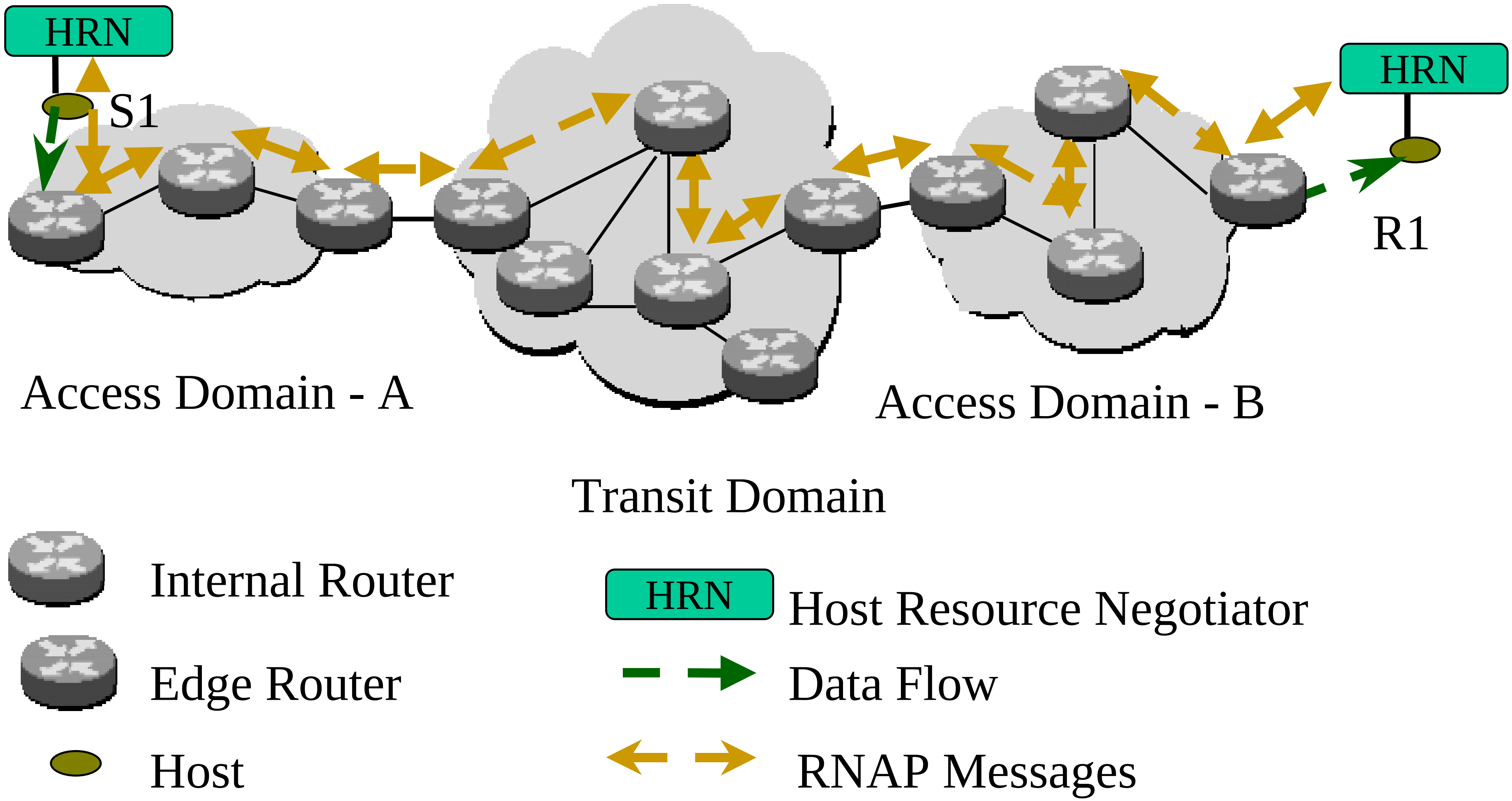
Objectives

- Develop a resource negotiation and pricing framework which
  - Combines QoS support and user adaptation
  - Allows resource commitment for short intervals
  - Provides differential pricing for differentiated services, and usage- and congestion-sensitive pricing to motivate user adaptation
  - Allows provider to trade-off blocking connections and raising prices
- RNAP: a Resource Negotiation And Pricing protocol through which the user and network (or two network domains) negotiate network delivery services.

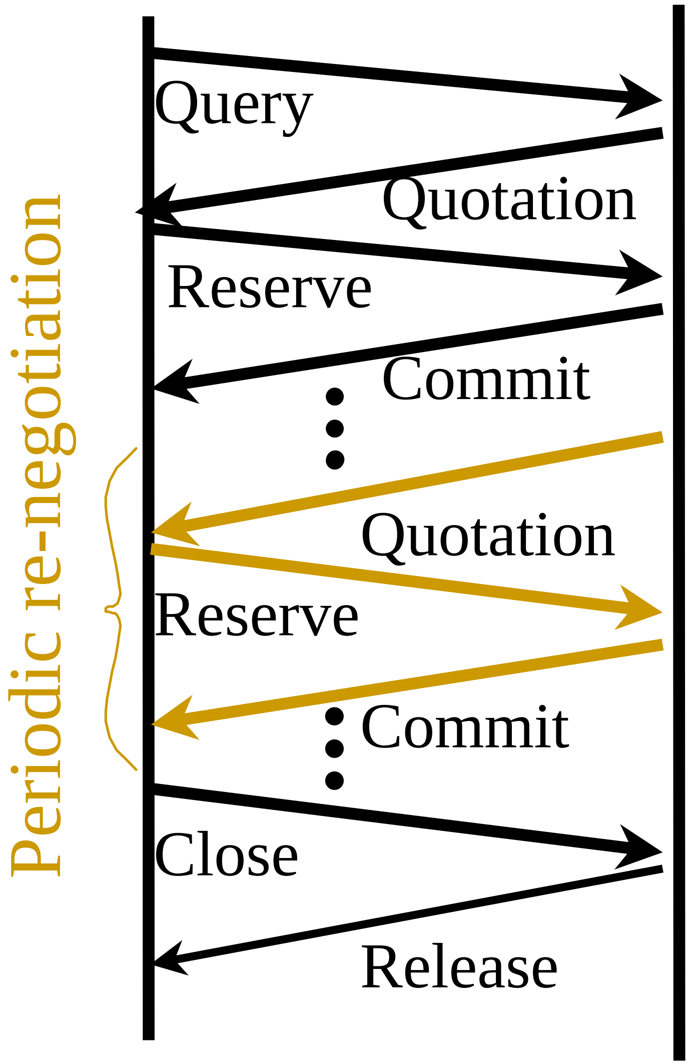
Protocol Architectures: Centralized (RNAP-C)



Protocol Architectures: Distributed (RNAP-D)



RNAP Messages



Query: Inquires about available services, prices

Quotation: Specifies service availability, accumulates service statistics and prices

Reserve: Requests service(s), resources

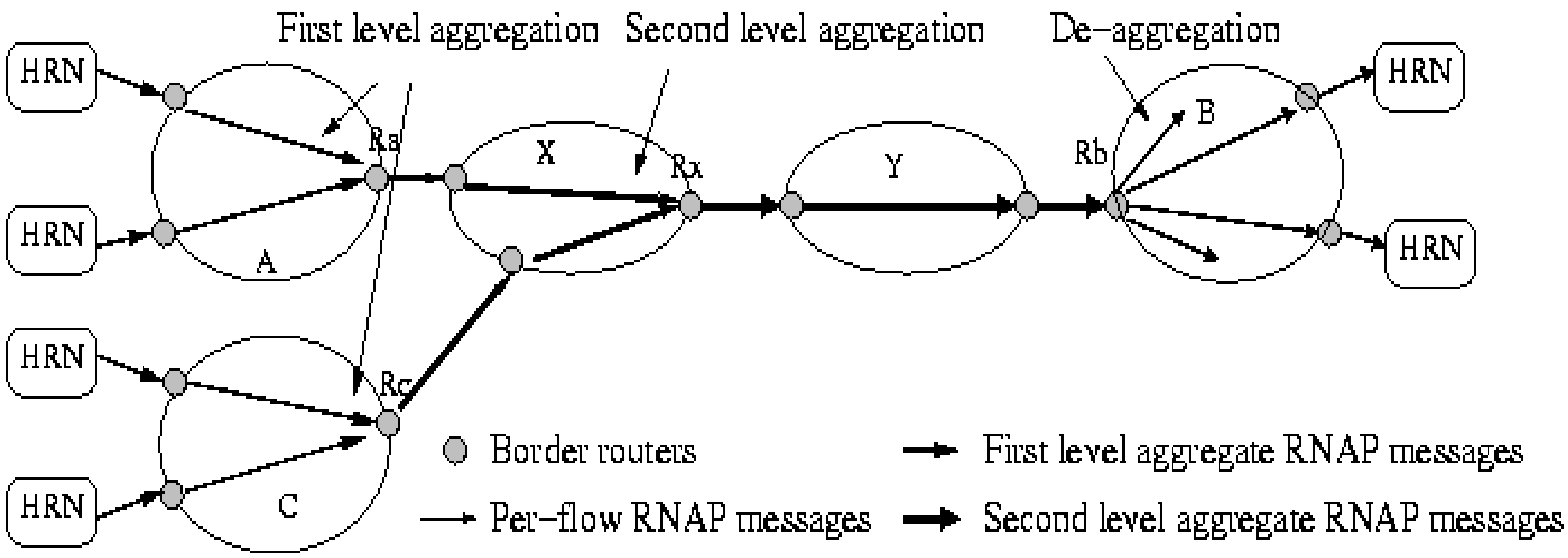
Commit: Admits the service request at a specific price or denies it.

Close: Tears down negotiation session

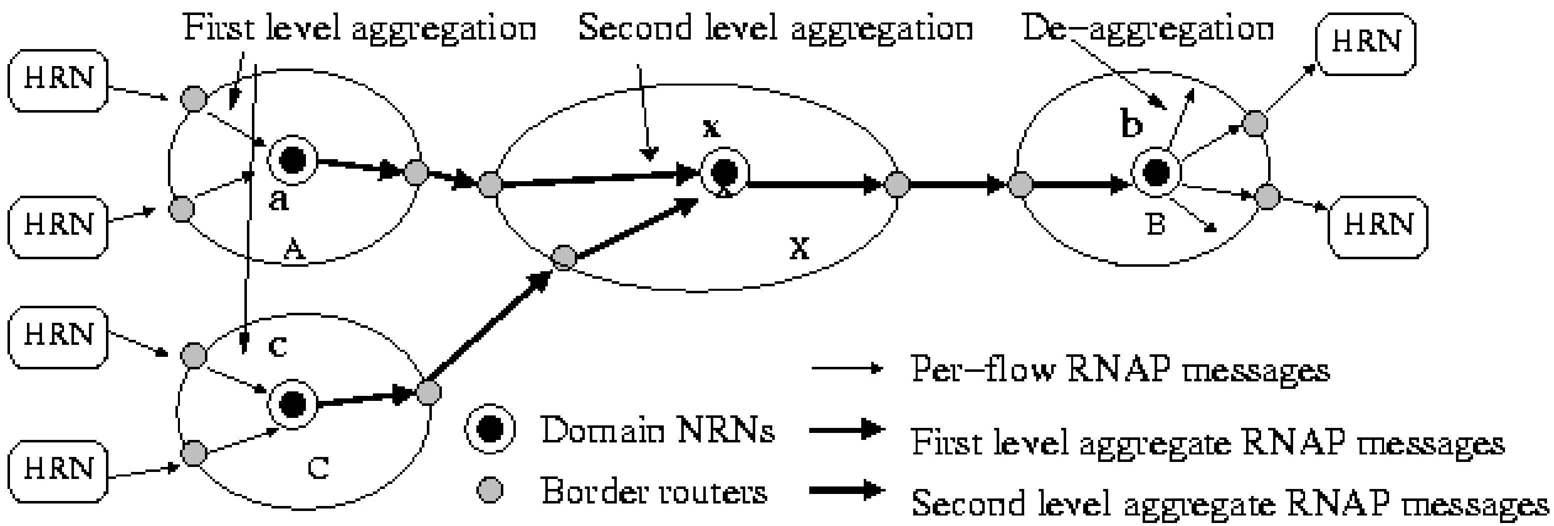
Release: Releases the resources

RNAP Message Aggregation

RNAP-D



RNAP-C

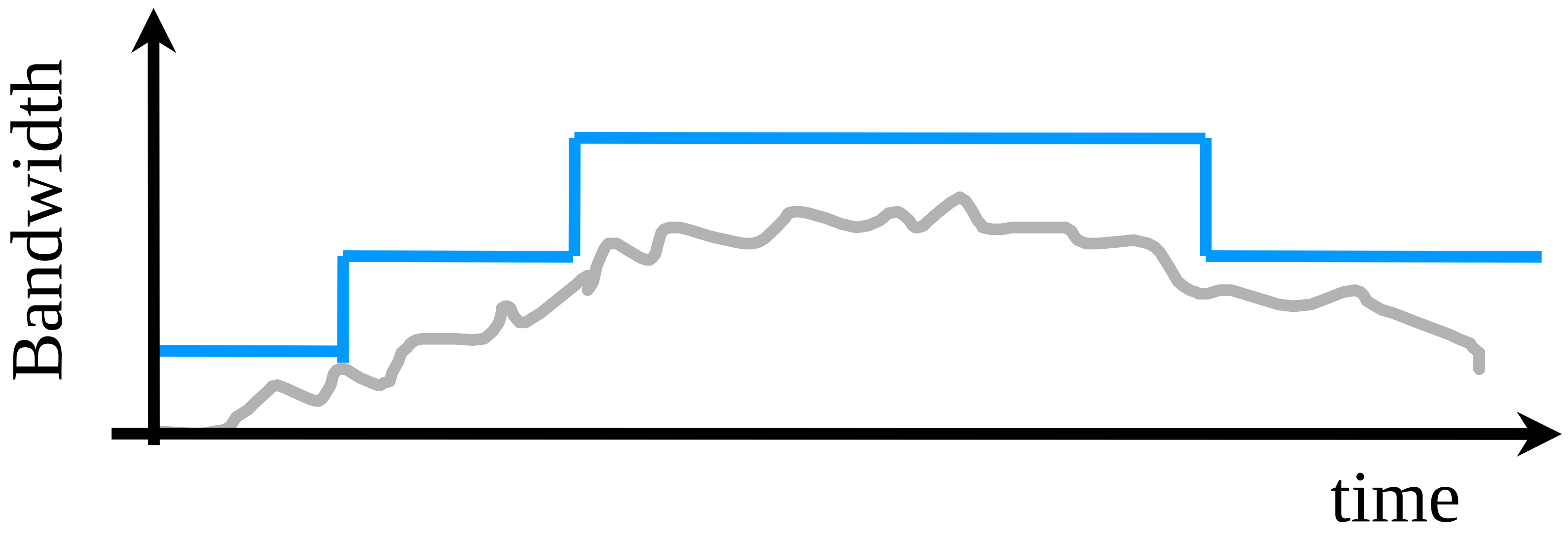


RNAP Message Aggregation (cont'd)

- Aggregation when senders share the same destination network
- Messages merged by source or intermediate domains
- Messages de-aggregated at destination border routers (RNAP-D) , or NRNs (RNAP-C)
- Original messages sent directly to destination/source domains without interception by intermediate RNAP agents; aggregate message reserves and collects price at intermediate nodes/domains

Block Negotiation

- Block Negotiation
  - Aggregated resources are added/removed in large blocks to minimize negotiation overhead and reduce network dynamics



Two Volume-based Pricing Policies

- Fixed-Price (FP)
  - FP-FL: same for all services
  - FP-PR: service class dependent
  - FP-T: time-of-day dependent
  - FP-PR-T: FP-PR + FP-T
  - During congestion: higher blocking rate OR higher dropping rate and delay
- Congestion-Price-based Adaptation (CPA)
  - FP + congestion-sensitive price
  - CP-FL, CP-PR, CP-T, CP-PR-T
  - During congestion: users maintain service by paying more OR reduce sending rate or lower service class

Proposed Pricing Strategies

- Holding price and charge:
  - $p_h^j = \alpha^j \ (p_u^j - p_u^{j-1})$
  - $c_h^{ij}(n) = p_h^j \, r^{ij}(n) \tau^j$
- Usage price and charge:
  - $\max \ [\Sigma_l \, x^j (p_u^1, p_u^2, \ldots, p_u^J) \, p_u^j - f(C)]$,
    s.t. $r(x(p_u^2, p_u^2, \ldots, p_u^J)) \leq R, \ j \in J$
  - $c_u^{ij}(n) = p_u^j \ v^{ij}(n)$
- Congestion price and charge:
  - $p_c^j(n) = \min \ [\{p_c^j(n\text{-}1) + \sigma^j (D^j, S^j) \ x \ (D^j\text{-}S^j)/S^j, 0\}^+, \mathrm{p}_{\max}^j]$
  - $c_c^{ij}(n) = p_c^j \ v^{ij}(n)$

Usage Price for Differentiated Services

- Usage price for a service class based on cost of class bandwidth: lower target load -> higher QoS , but higher per unit bandwidth cost
- Parameters:
  - p_{basic} basic rate for fully used bandwidth
  - ρ^{j}: expected load ratio of class j
  - x^{ij}: effective bandwidth consumption of application i
  - A^{j}: constant elasticity demand parameter

Usage Price for Differentiated Services (cont'd)

- Price for class j: $p_u^j = p_{basic} / \rho^j$
- Demand of class j: $x^j(p_u^j) = A^j / p_u^j$
- Effective bandwidth consumption:
  - $x_e^j(p_u^j) = A^j / (p_u^j \rho^j)$
- Network maximizes profit
  - max $[\Sigma_l (A^j / p_u^j)\, p_u^j - f(C)]$, $p_u^j = p_{basic} / \rho^j$,
    s. t. $\Sigma_l A^j / (p_u^j \rho^j) \leq C$
- Hence:
  - $p_{basic} = \Sigma_l A^j / C$, $p_u^j = \Sigma_l A^j / (C\rho^j)$

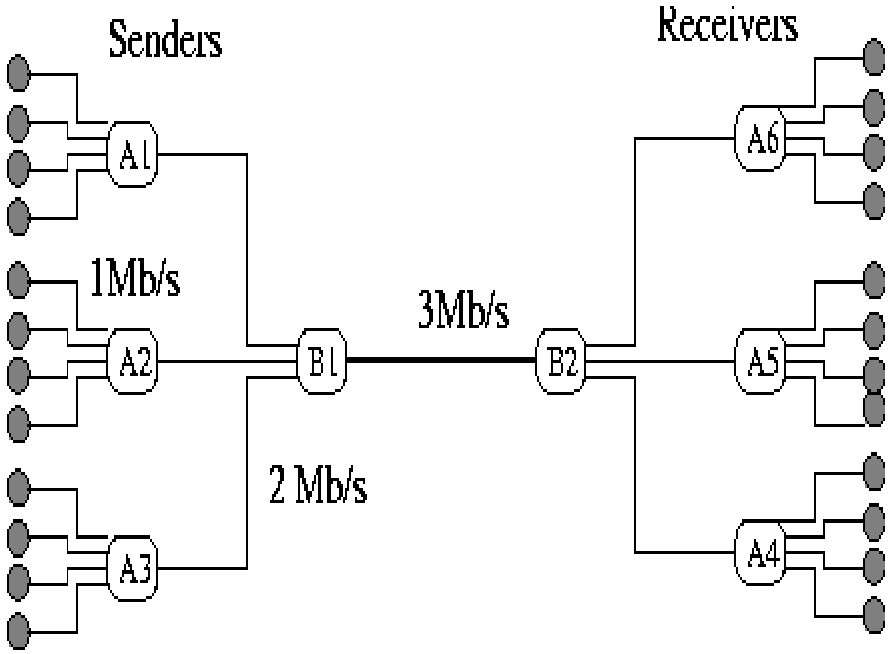
User Adaptation based on Utility

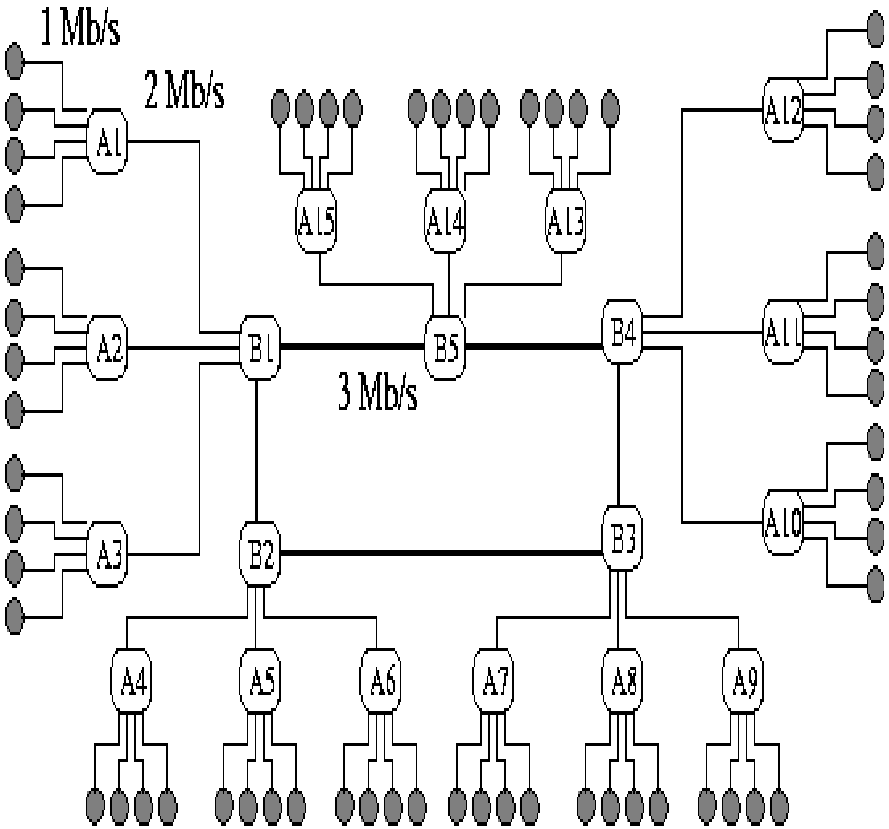
- Users adapt service selection and data rate based on utility which is associated with QoS
- Utility expressed in terms of perceived value, e.g.,15 cents /min
- Multi-application task (e.g., video-conference) - maximize total utility of task subject to budget -> dynamic resource allocation among component applications
- User utility optimization:
  - $U = \Sigma_i U^i(x^i(Tspec, Rspec)]$
  - max $[\Sigma_l U^i(x^i) - C^i(x^i)]$, s. t. $\Sigma_l C^i(x^i) \leq b$, $x_{min}^i \leq x^i \leq x_{max}^i$
  - Determine optimal Tspec and Rspec
- Not need to reveal utility to the network

User adaptation based on utility: example

- User defines utility at discrete bandwidth, QoS levels
- Utility is a function of bandwidth at fixed QoS
  - An example utility function: $U(x) = U_0 + \omega \log(x / x_m)$
  - U_0 : perceived (opportunity) value at minimum bandwidth
  - ω : sensitivity of the utility to bandwidth
- Function of both bandwidth and QoS
  - $U(x) = U_0 + \omega \log(x / x_m) - k_d\, d - k_l\, l$, for $x \geq x_m$
  - k_d : sensitivity to delay
  - k_l : sensitivity to loss
- Optimization:
  - max $[\Sigma_l\, U_0^i + \omega^i \log(x^i / x_m^i) - k_d^i\, d - k_l^i\, l - p^i x^i]$,
    s. t. $\Sigma_l\, p^i x^i \leq b$, $x \geq x_m$, $d \leq D$, $l \leq L$
  - Without budget constraint: $x^i = \omega^i / p^i$
  - With budget constraint: $b^i = b(\omega^i / \Sigma_l\, \omega^k)$

Simulation Model





Topology 1

Topology 2

Simulation Model

- Network Simulator (NS-2)
- Weighted Round Robin (WRR) scheduler
- Three classes: EF, AF, BE
  - EF:
    - tail dropping, limited to 50 packets
    - expected load threshold 40%, delay bound 2 ms, loss bound 10^{-6}
  - AF:
    - RED-with-In-Out (RIO), limited to 100 packets
    - expected load threshold 60%, delay bound 5 ms, loss bound 10^{-4}
  - BE:
    - Random Early Detection (RED), limited to 200 packets
    - expected load threshold 90%, delay bound 100 ms, loss bound 10^{-2}

Simulation Model (cont'd)

- Parameter Set-up
  - topology1: 60 users; topology 2: 360 users
  - sources: on-off or Pareto on-off (shape parameter: 1.5)
  - price adjustment factor: $\sigma = 0.06$; update threshold: $\theta = 0.05$
  - negotiation period: 30 seconds
  - price (for a 64 kb/s transmission):
    - usage price $p_{basic} = \$0.08$ / min, $p_{EF} = \$0.20$ / min, $p_{AF} = \$0.13$ / min, $p_{BE} = \$0.09$ / min
    - holding price: $p_{EF} = \$0.067$ / min, $p_{AF} = \$0.044$ / min
  - ω: 64 kb/s as reference, randomly set based on service type
    - EF: \$0.13 / min - \$0.20 / min; AF: \$0.09/ min - \$0.26 / min ; BE: \$0.06 / min - \$0.18 / min.
  - average session length 10 minutes,exponentially distributed.

Simulation Model (cont'd.)

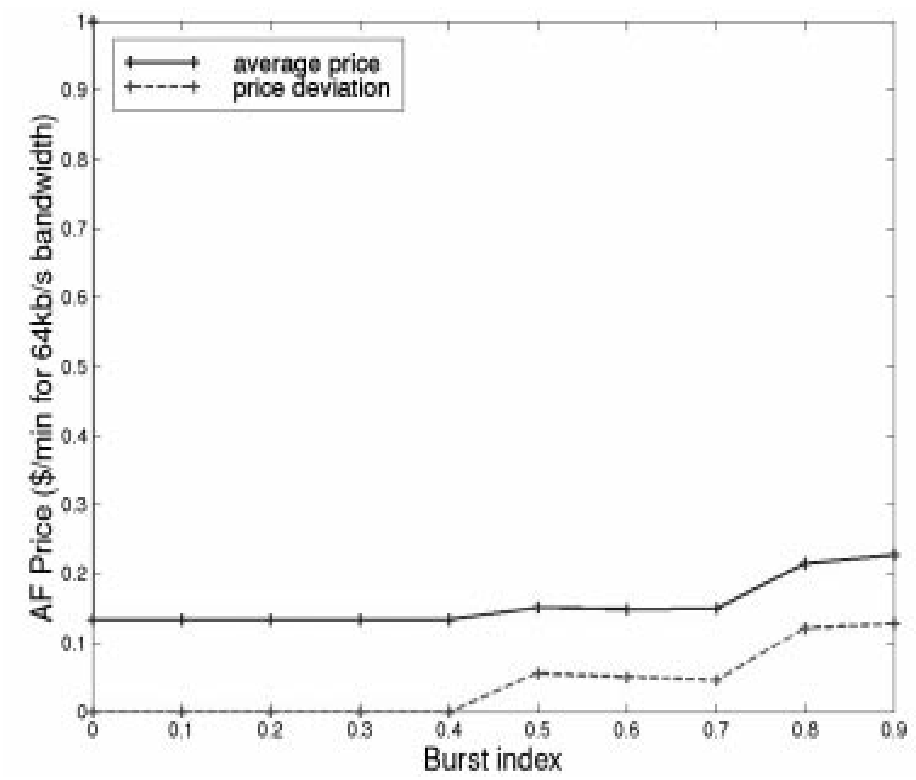
- Performance measures
  - Engineering metrics
    - Bottleneck traffic arrival rate
    - Average packet loss and delay
    - User request blocking probability
  - Economic metrics
    - Average user benefit
    - End to end price, and it standard deviation

Design of Experiments

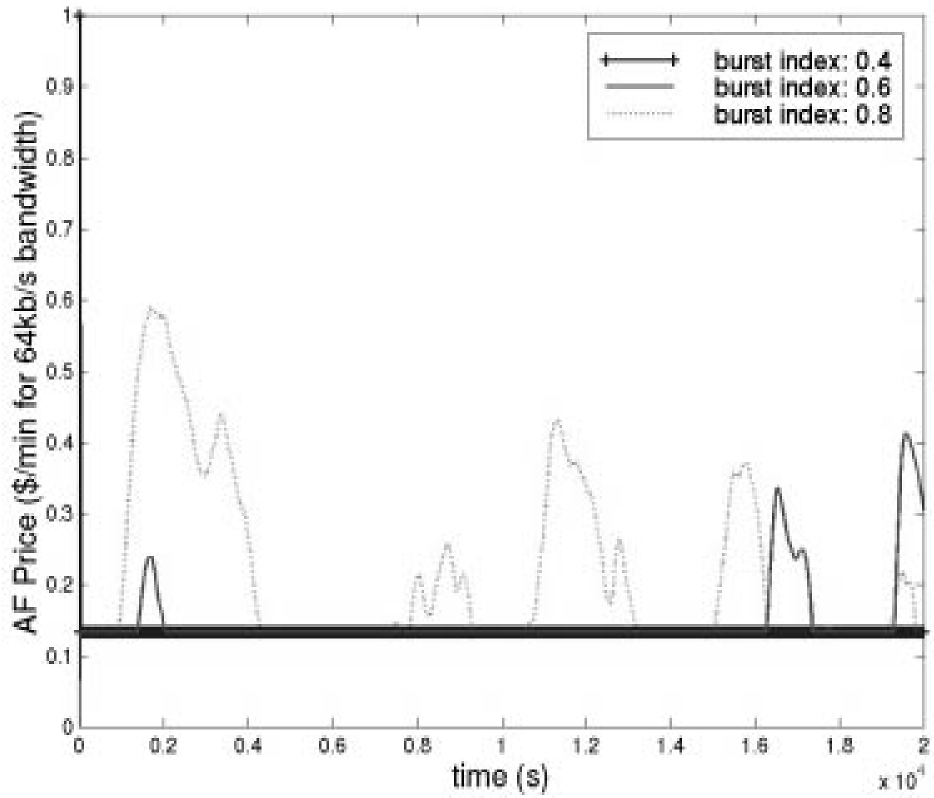
- Performance comparison: FP (usage price + holding price) and CPA (usage price + holding price + congestion price)
- Four groups of experiments:
  - Effect of traffic burstiness
  - Effect of traffic load
  - Load balance between classes
  - Effect of admission control
- Other experiments (see web page for references):
  - Effect of system control parameters: target reservation rate, price adjustment step, price adjustment threshold
  - Effect of user demand elasticity, session multiplexing
  - Effect when part of users adapt, session adaptation and adaptive reservation

Effect of Traffic Burstiness

Price average and standard deviation of AF class

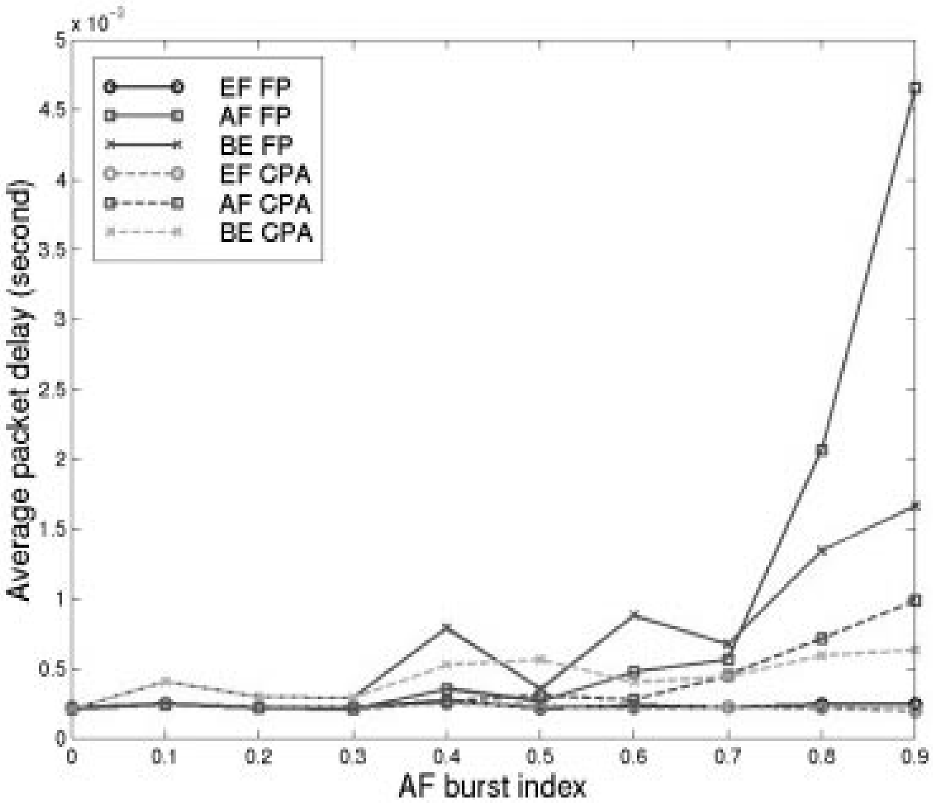


Variation over time of the price of AF class

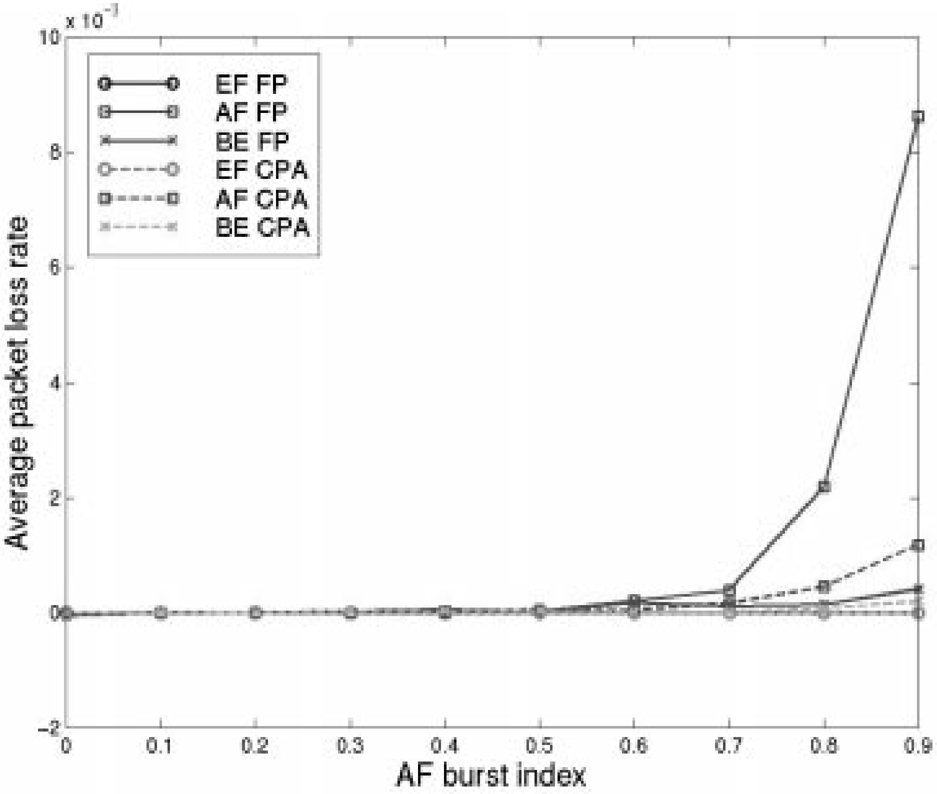


Effect of Traffic Burstiness (cont'd)

Average packet delay

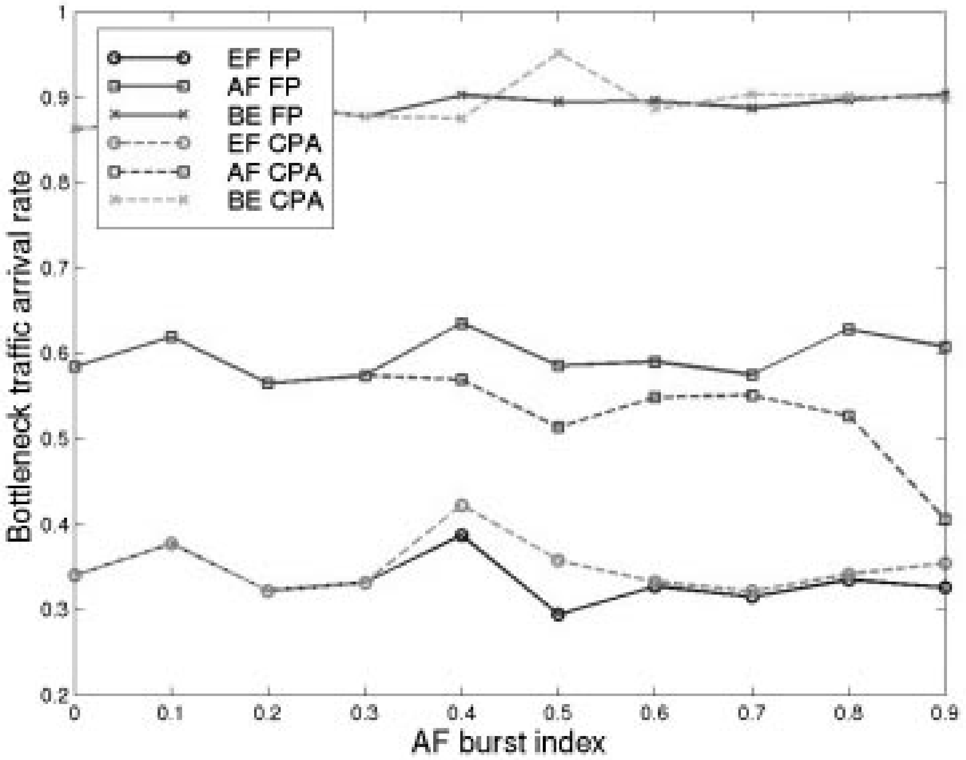


Average packet loss

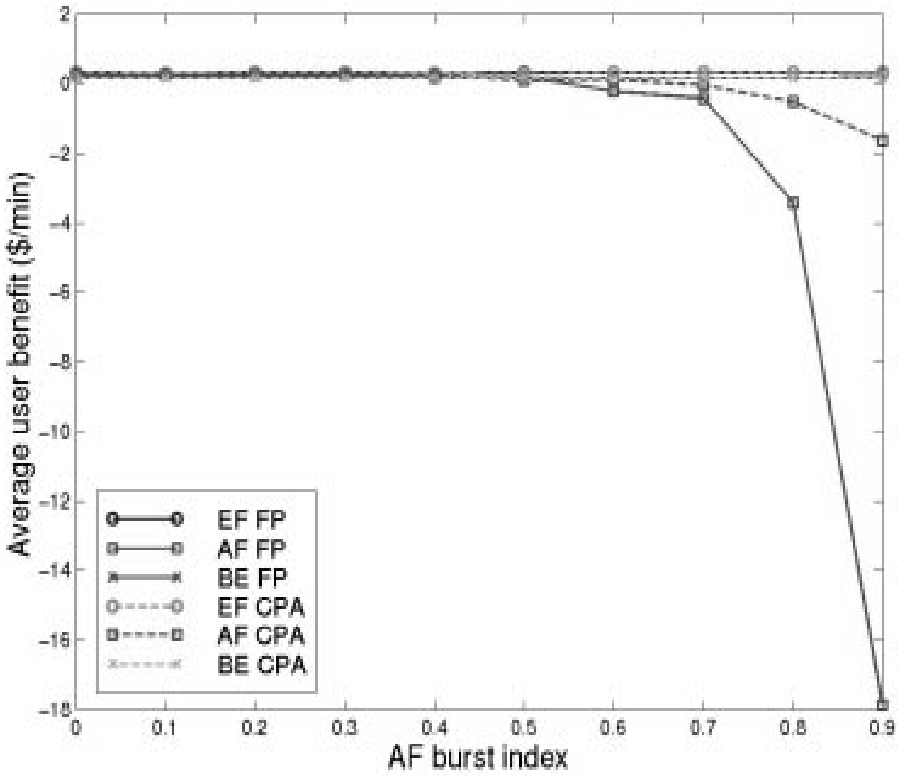


Effect of Traffic Burstiness (cont'd)

Average traffic arrival rate

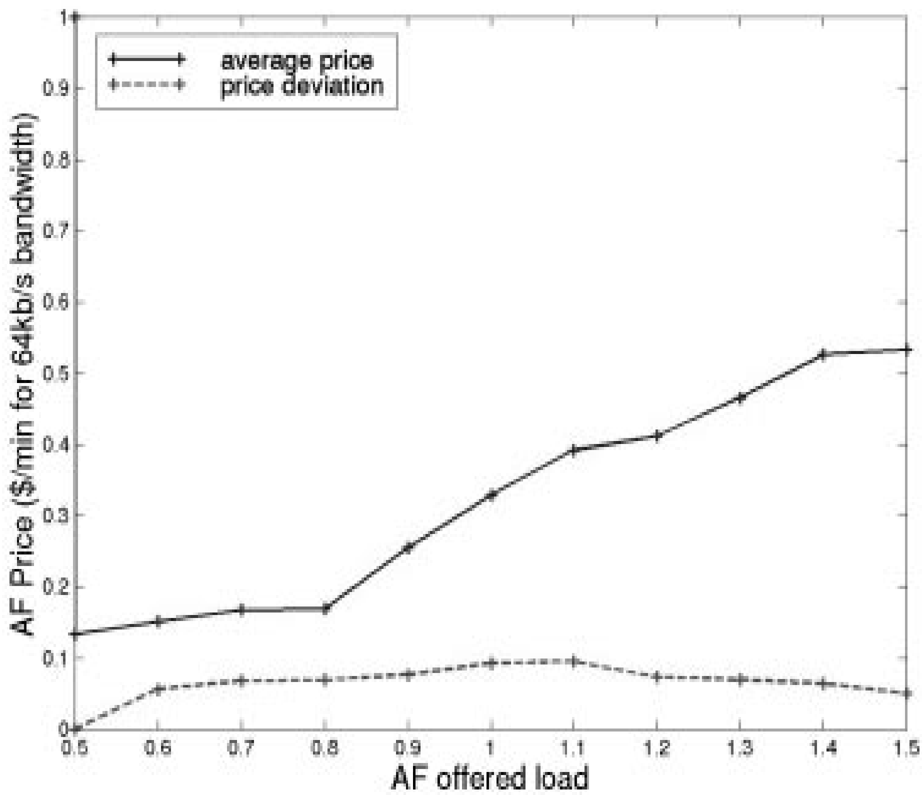


Average user benefit

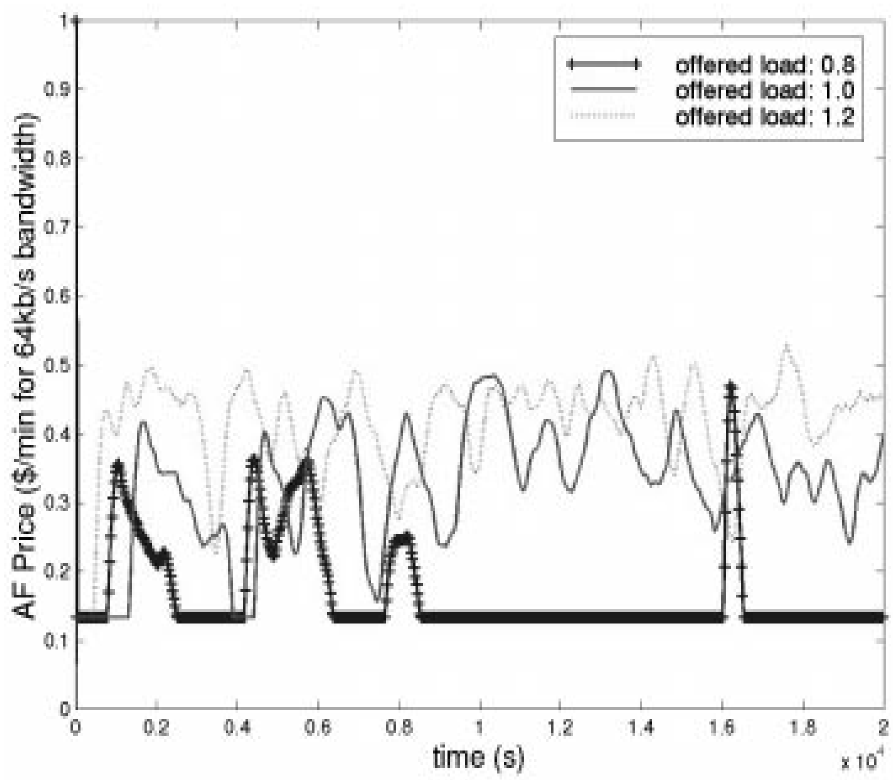


Effect of Traffic Load

Price average and standard deviation of AF class

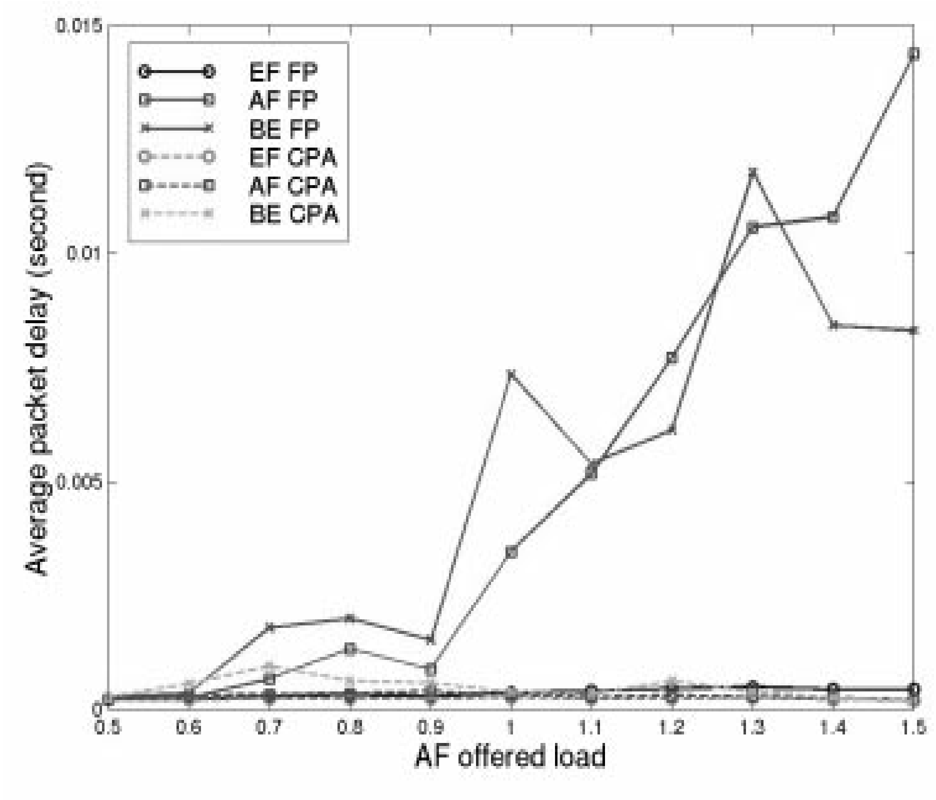


Variation over time of the price of AF class

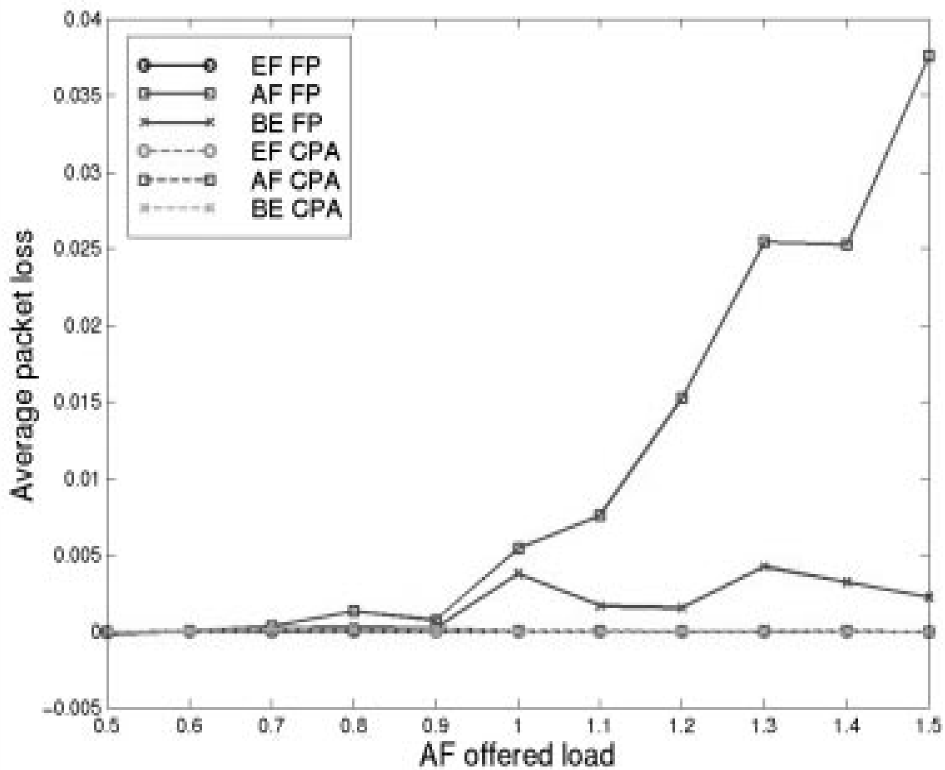


Effect of Traffic Load (cont'd)

Average packet delay

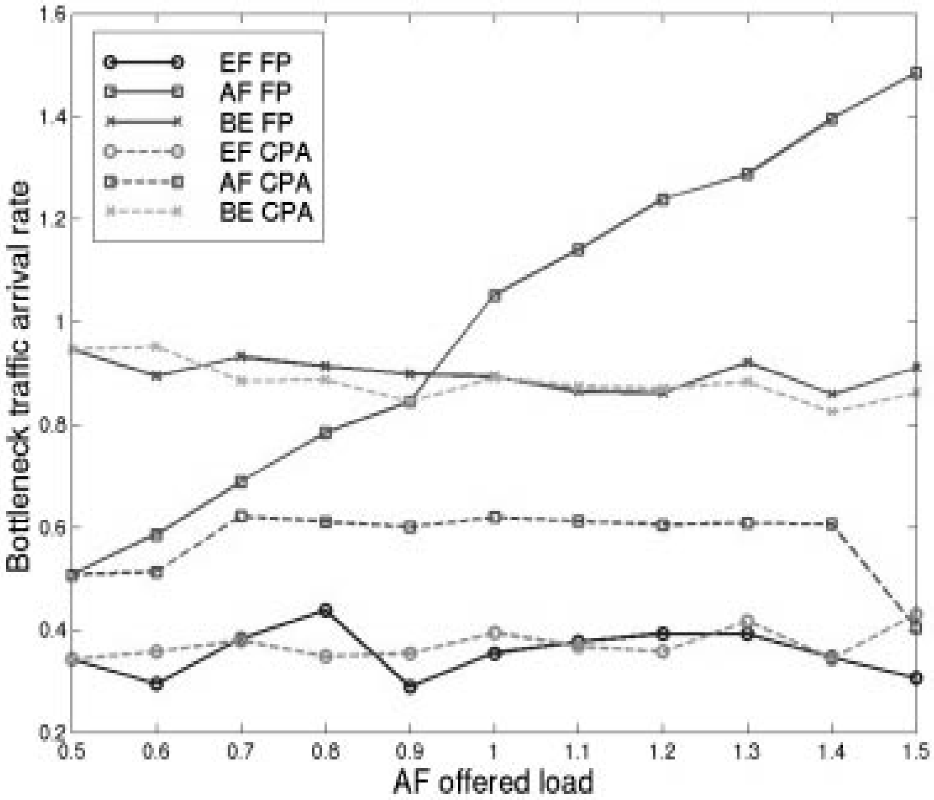


Average packet loss

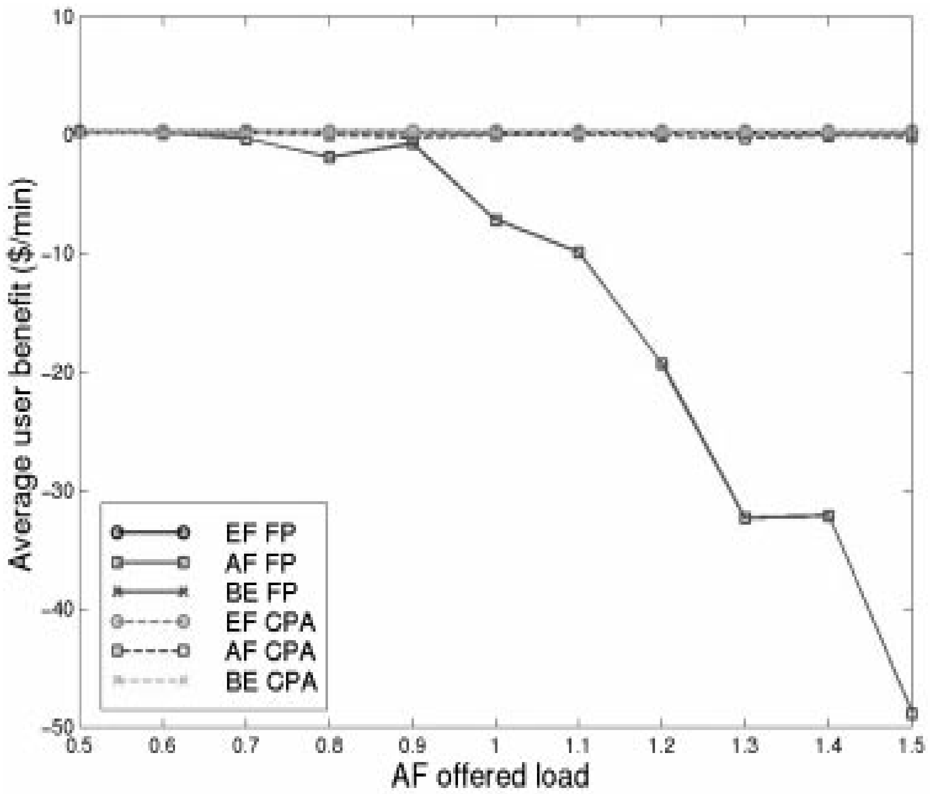


Effect of Traffic Load (cont'd)

Average traffic arrival rate

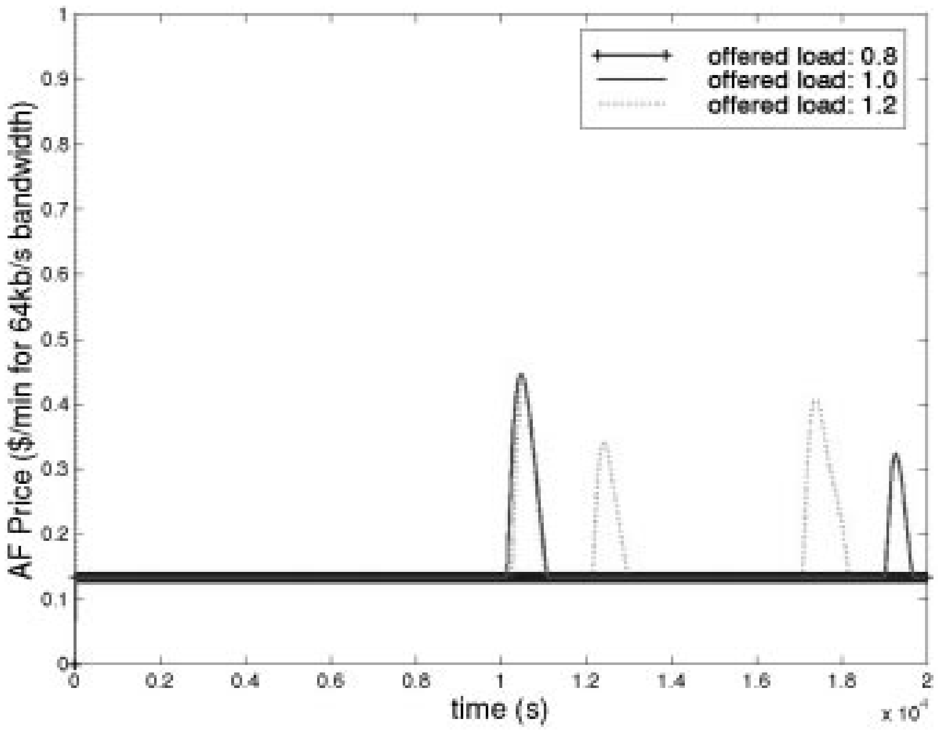


Average user benefit

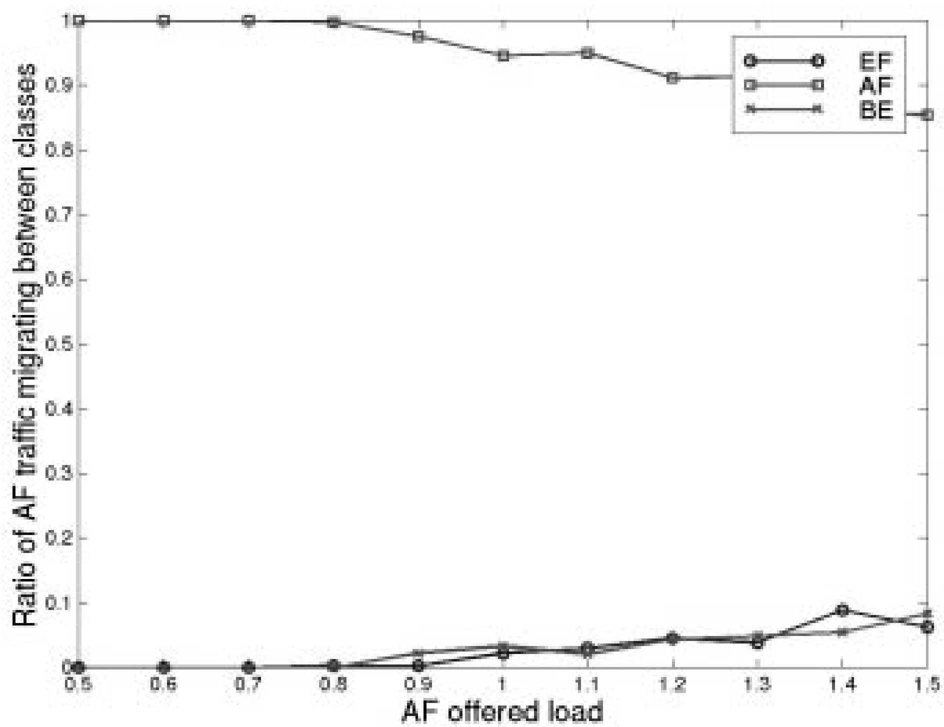


Load Balance between Classes

Variation over time of the price of AF class



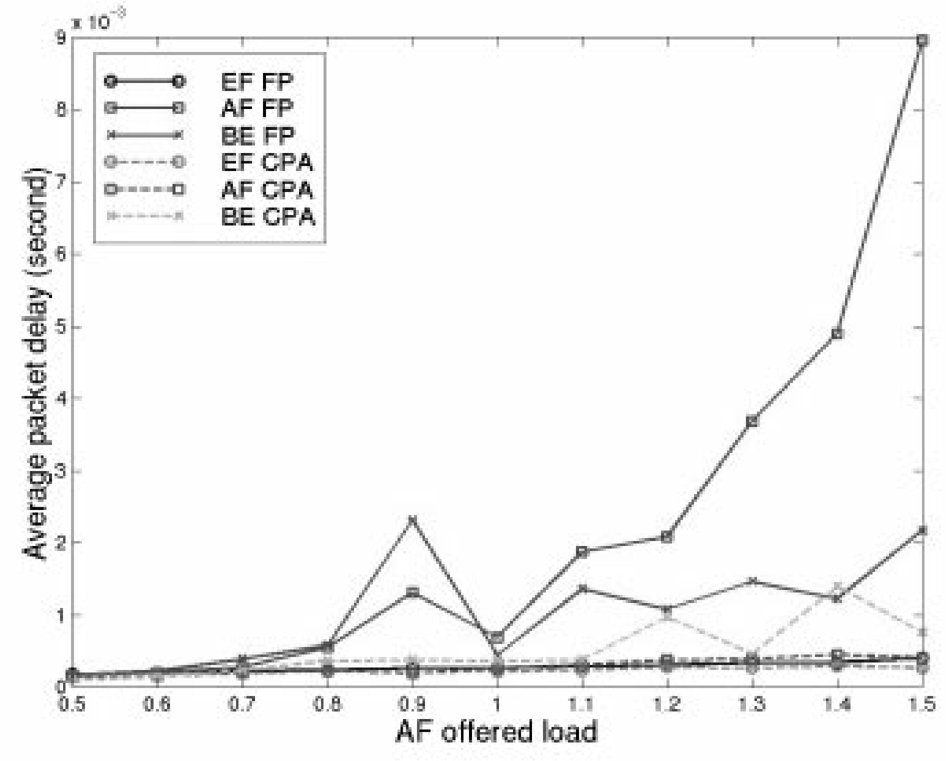
Ratio of AF class traffic migrating through class re-selection

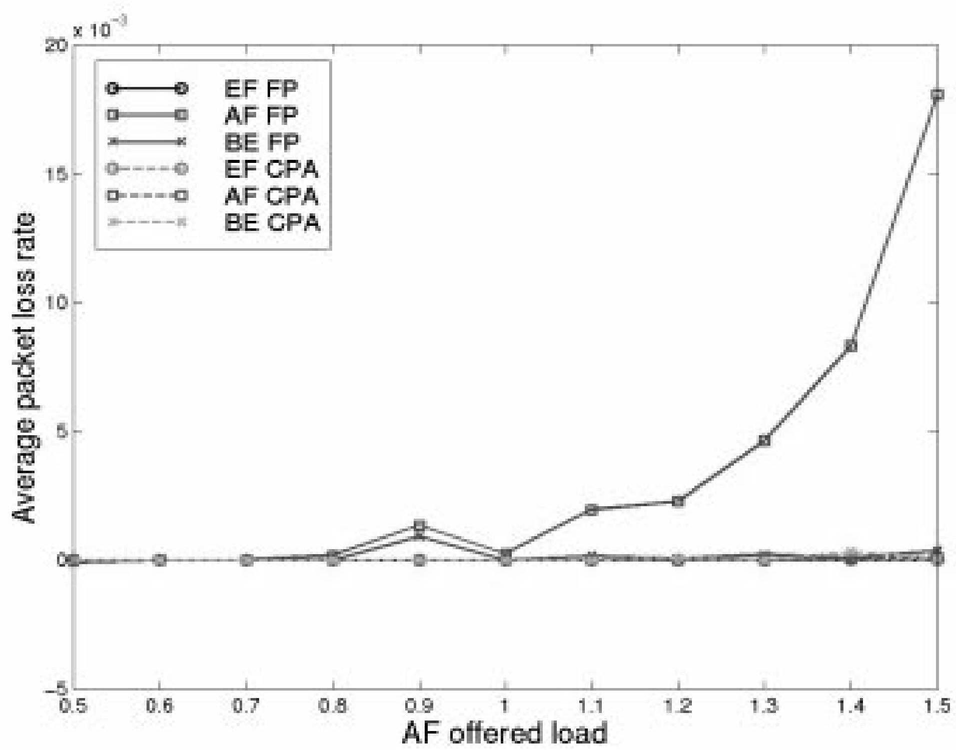


Load Balance between Classes (cont'd)

Average packet delay

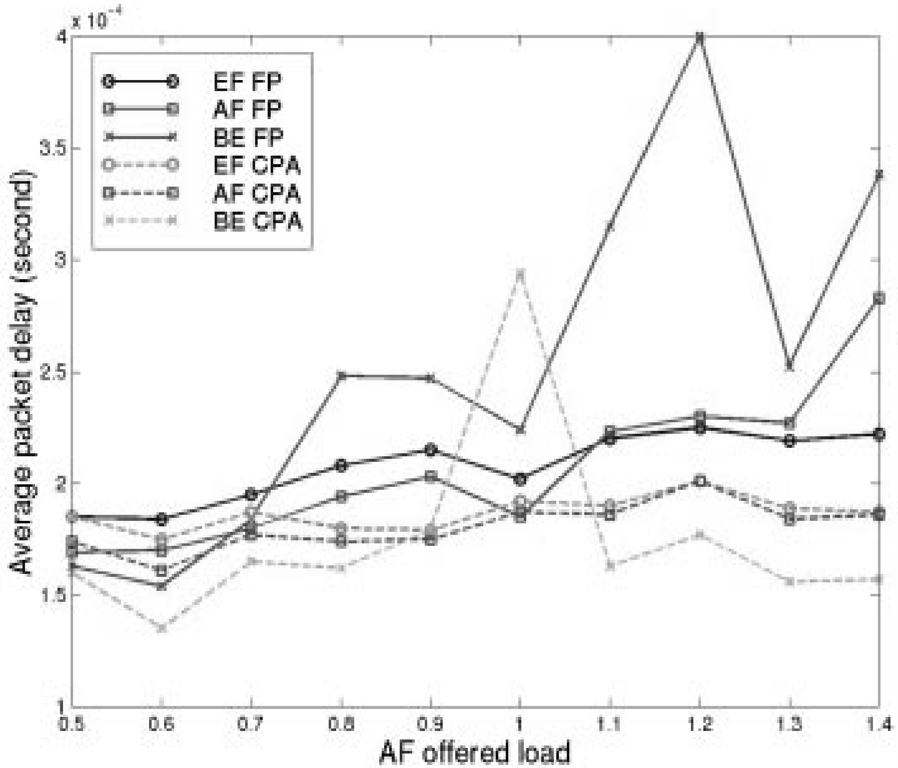
Average packet loss



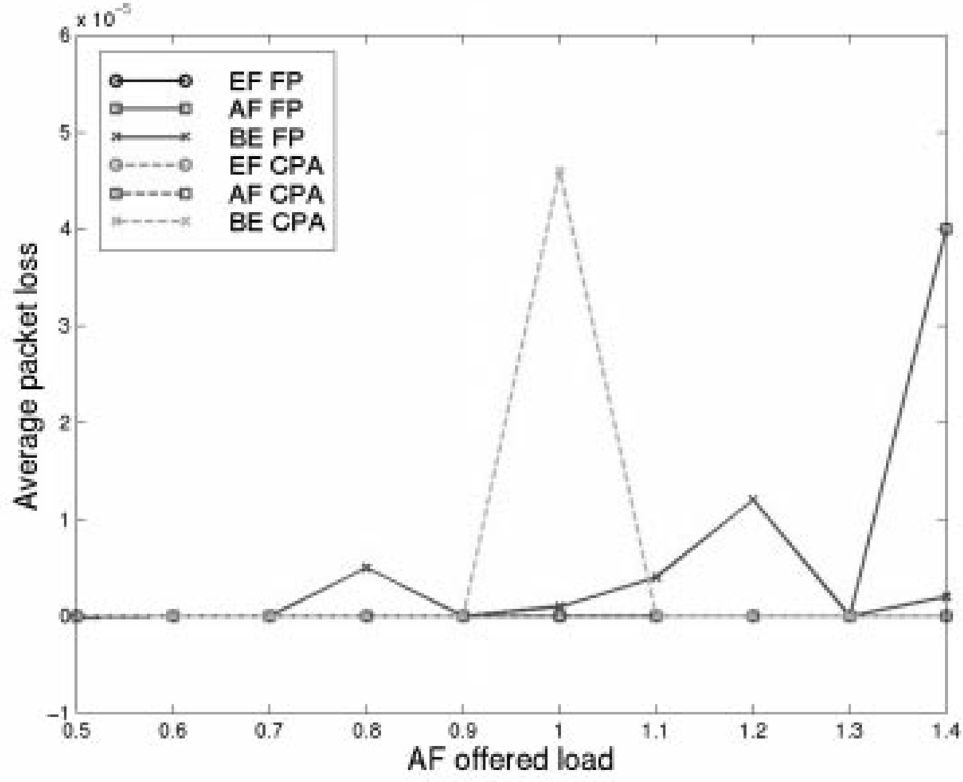


Effect of Admission Control

Average packet delay



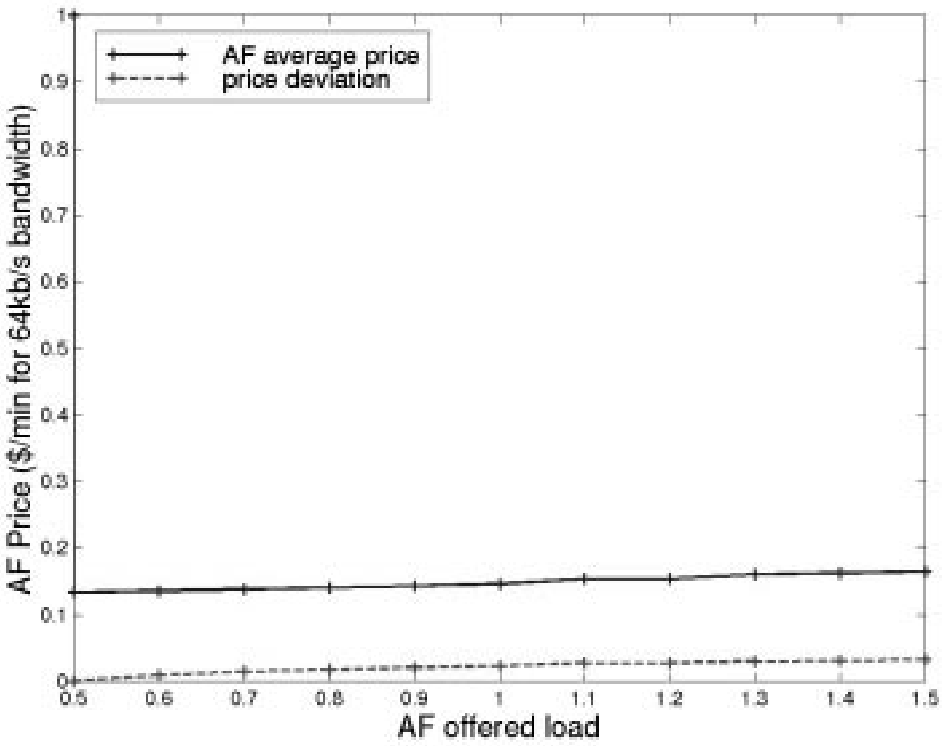
Average packet loss

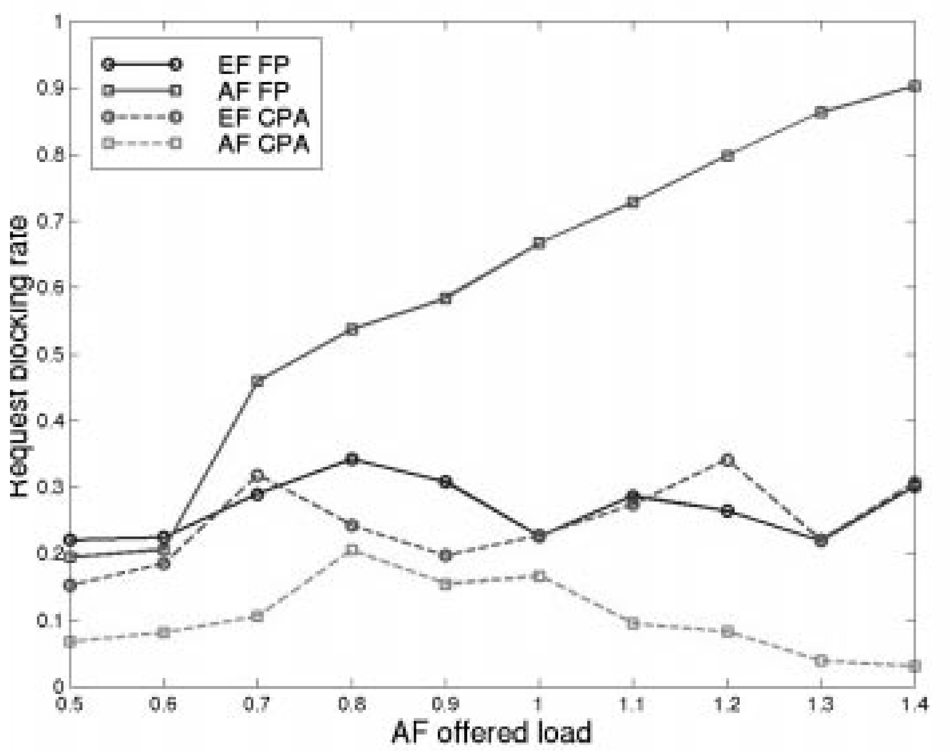


Effect of Admission Control (cont'd.)

Average and standard deviation of AF class price

User request blocking rate





Conclusions

- RNAP
  - Supports dynamic service negotiation, mechanisms for price and charge collation
  - Allows for both centralized and distributed architectures
  - Multi-party negotiation: senders, receivers, both
  - Can be stand alone, or embedded inside other protocols
  - Reliable and scalable
- Pricing
  - Consider both long-term user demand and short-term traffic fluctuation; use congestion-sensitive component to drive adaptation in congested network
- Application adaptation
  - Bandwidth proportional to user's willingness to pay

Conclusions (cont'd)

- Simulation results:
  - Differentiated service requires different target loads in each class
  - Without admission control, CPA coupled with user adaptation allows congestion control, and service assurances by restricting the load to the targeted level
  - With admission control, performance bounds can be assured even with FP policy, but CPA reduces the request blocking rate greatly and helps to stabilize price
  - Allowing service class migration further stabilizes price
- Future work
  - Refine the RNAP protocol, stand alone RNAP implementation in progress, experiments over Internet2