

# **EDCA Differentiation HCCF Scheduling**

Renato Lo Cigno

[www.dit.unitn.it/locigno/didattica/NC/](http://www.dit.unitn.it/locigno/didattica/NC/)

# ... Copyright

Quest'opera è protetta dalla licenza *Creative Commons NoDerivs-NonCommercial*. Per vedere una copia di questa licenza, consultare:  
<http://creativecommons.org/licenses/nd-nc/1.0/>  
oppure inviare una lettera a:  
*Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.*

This work is licensed under the *Creative Commons NoDerivs-NonCommercial* License. To view a copy of this license, visit:  
<http://creativecommons.org/licenses/nd-nc/1.0/>  
or send a letter to  
*Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.*



# Thanks & Disclaimer

- These slides and results are based on the following recent research activities
  - “Performance Evaluation of Differentiated Access Mechanisms Effectiveness in 802.11 Networks”, Ilenia Tinnirello , Giuseppe Bianchi , Luca Scalia, IEEE Globecom 2004.
  - “Scheduling in 802.11e: Open-Loop or Closed-Loop?”, P. Larcheri, R. Lo Cigno, IEEE WONS 2006
- As such they must be considered examples of the possible performances and tradeoffs
- Thanks to Bianchi and all the other authors for providing copy of the papers graphics and slides



# EDCA or HCCA?

- How does EDCA support differentiation?
  - Is this enough for standard purposes?
  - Are parameters easy to tune and universal?
- 
- How can HCCA polling-based scheduling be implemented?
  - Do we need to use the feedback from the STA?
  - How can the traffic be described?



# **Performance Evaluation of Differentiated Access Mechanisms Effectiveness in 802.11 Networks**

*G. Bianchi, I. Tinnirello, L. Scalia*

**presented @ Globecom 2004**

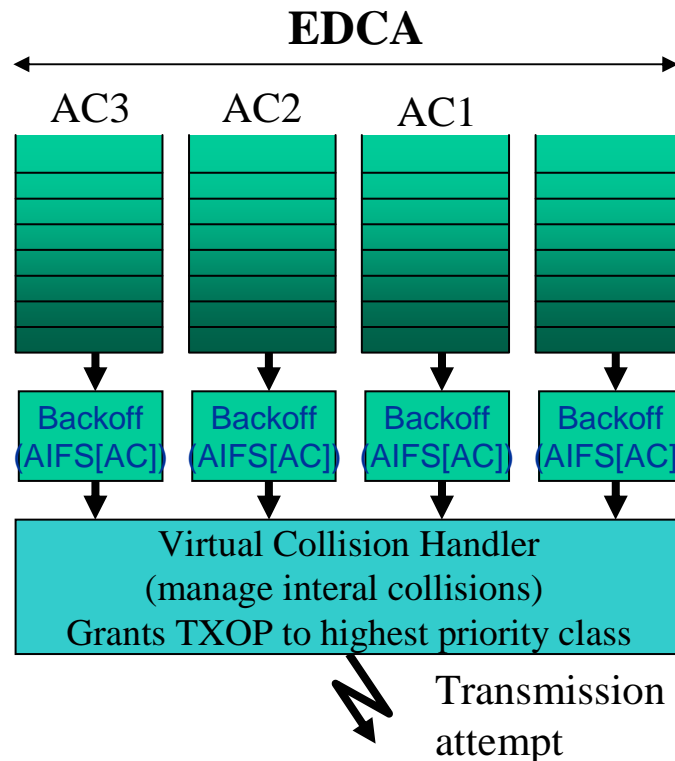
# QoS Support issues in legacy 802.11

- DCF is long term fair
  - Equal channel access probability among the stations
  - Averagely, the same channel holding time (for homogeneous packet sizes)
    - Solution: differentiate packet sizes?
    - Solution: differentiate channel holding times?
- NO WAY! QoS is not a matter of how long I hold the channel
  - It means more...
    - Need to manage access delay problems for real-time apps!!!
    - Need to modify 802.11 channel access fairness!!!



# QoS @ IEEE 802.11 MAC

- 802.11e defines different traffic classes onto map data flows
- Each traffic class behaves as an independent MAC entity
- Differentiated access priority is provided by:
  - Giving probabilistically lower backoff counters (**CWmin, CWmax, PF**)
  - Giving deterministically lower inter-frame spaces and backoff de-freezing times. (**AIFSN**)



Different MAC Access Parameters @ each class to differentiate channel access probability

Backoff based parameters: **CWmin, CWmax, PF**  
Channel monitoring based parameters: **AIFS**



# EDCA Performance Evaluation

- Performance Evaluation: answers we try to give...
  - Homogeneous sources
    - Performance effectiveness of each differentiation MAC parameter, individually taken
    - How each differentiation parameter reacts to different load conditions?
  - Heterogeneous sources
    - What are the most effective settings to manage high-priority delay requirements?



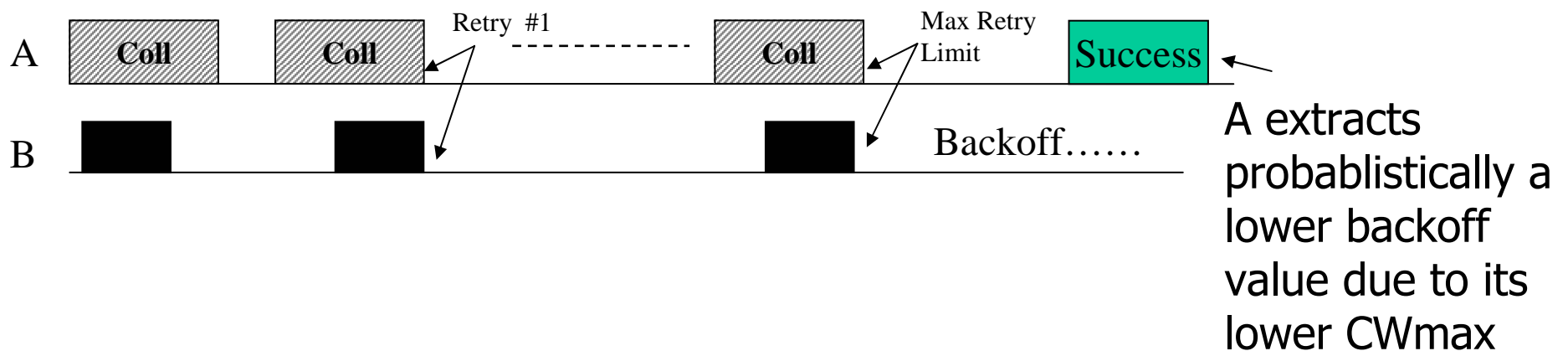
# EDCA Performance Evaluation

- Simulations
  - Same number of HP and LP stations
  - Same packet size (1024 bytes)
- Homogeneous sources scenario
  - Saturation conditions for HP and LP stations
    - Queues never empty
    - Data rate = Phy rate = 1 Mbps
- Heterogeneous sources scenario
  - 3 pkts/sec. for HP traffic
  - Saturation conditions for LP traffic
    - Data rate = Phy rate = 1 Mbps

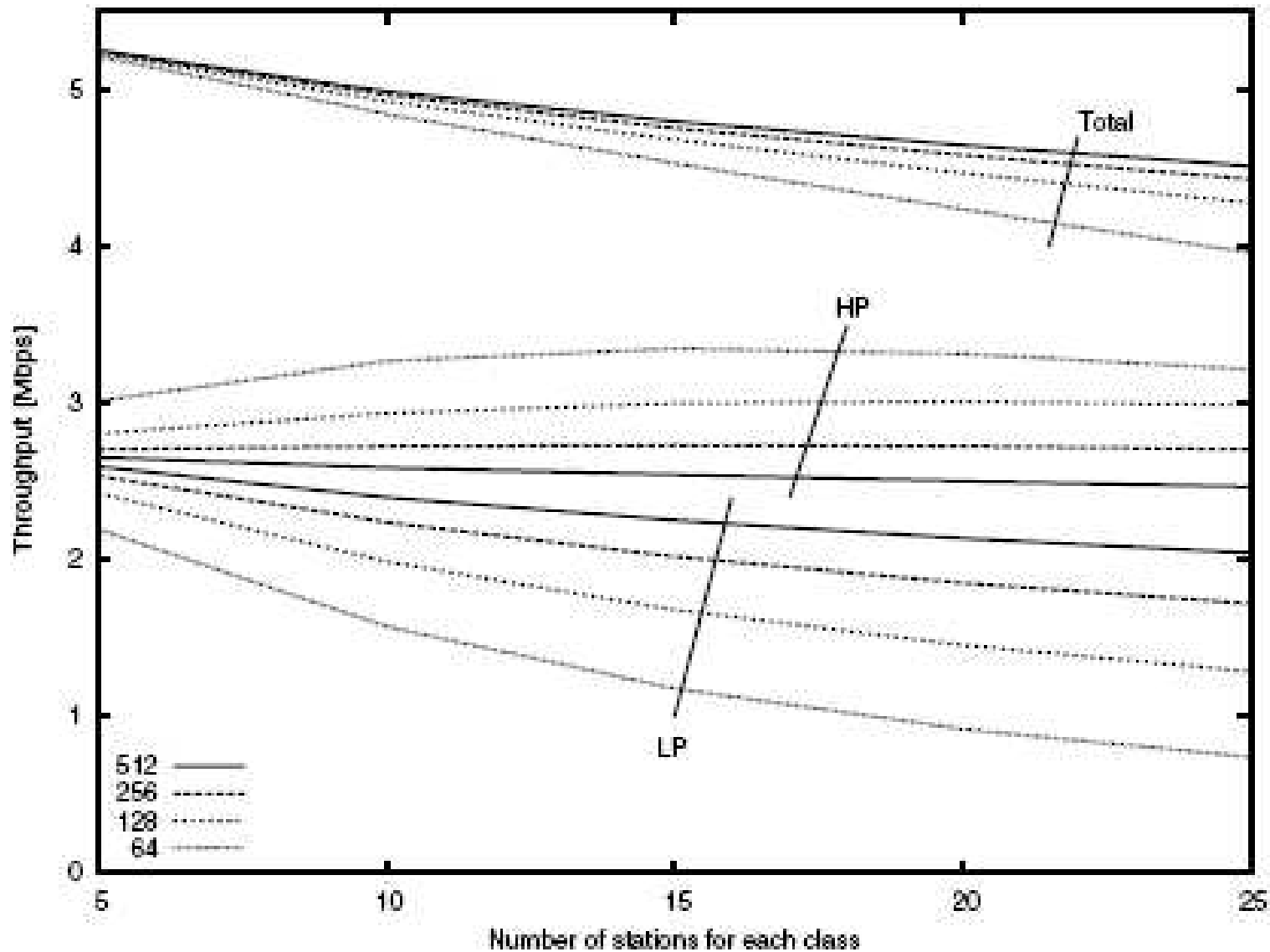


# CWmax Differentiation (1)

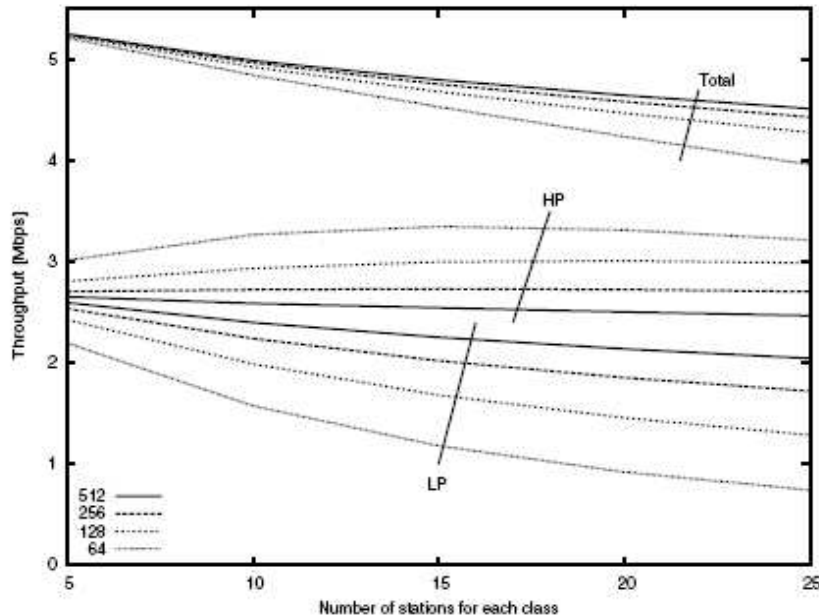
- $CW_{max}(A) < CW_{max}(B)$ 
  - Once reached  $CW_{max}$  (repeated collisions), A gets access priority over B



# CWmax Differentiation (2)



# CWmax Differentiation (3)

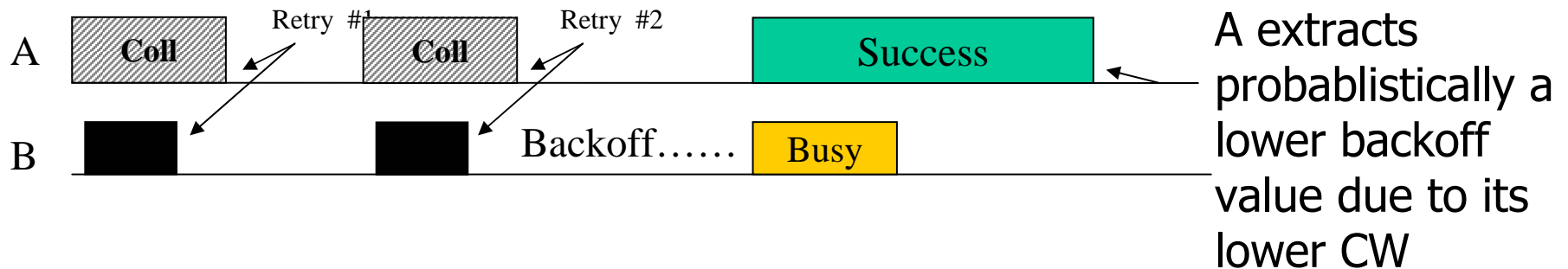


- Low throughput differentiation
  - Only with CWmax=64 effective
  - @ low loads poor performance
    - Few collisions
- Inefficient channel usage
  - Consecutive Collisions are needed for the differentiation effect
  - Overall throughput suffers @ high loads

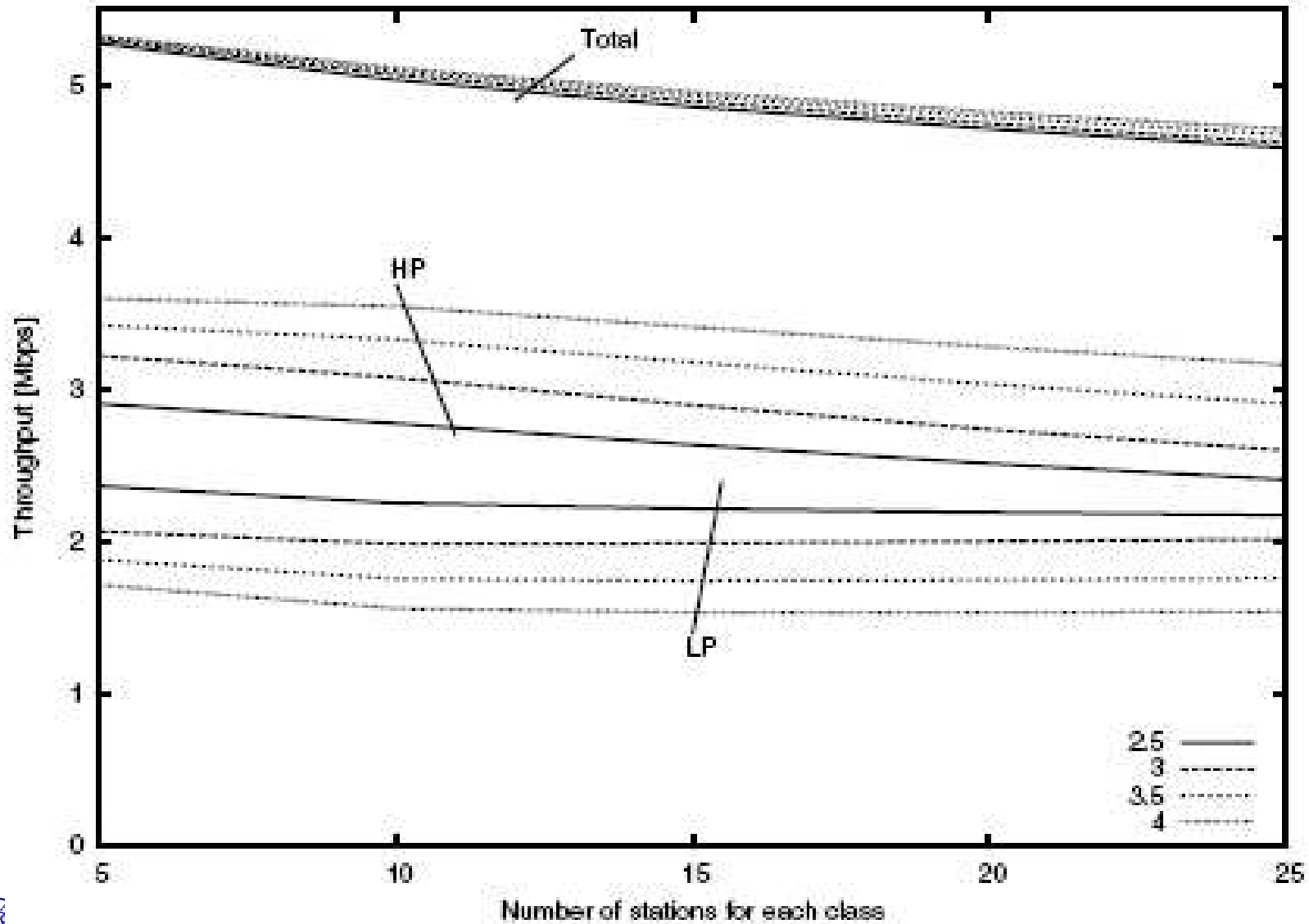


# PF Differentiation (1)

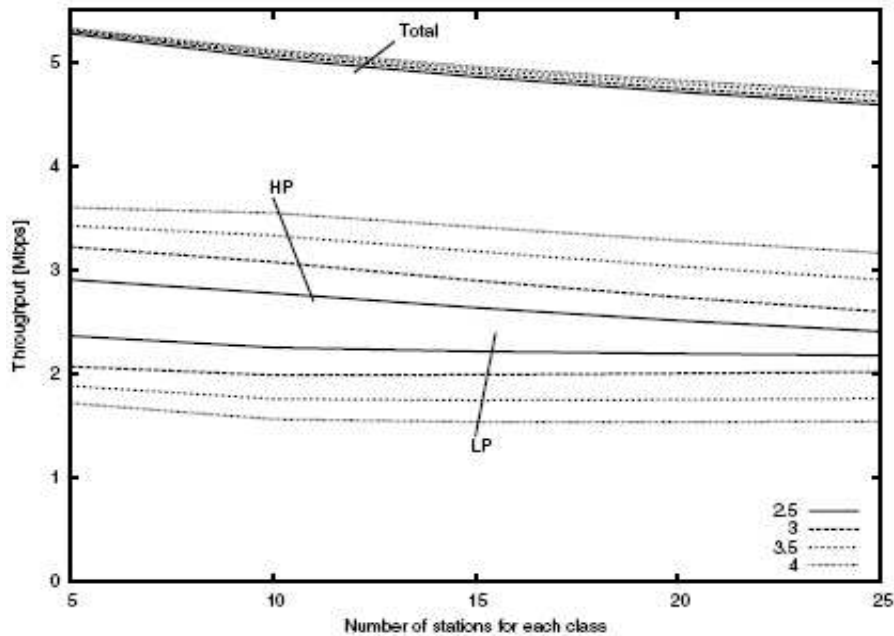
- $PF(A) < PF(B)$ 
  - once a collision occurs, station A has probabilistically an higher chance to extract a lower backoff value, thus it may retransmits first.



# PF Differentiation (2)



# PF Differentiation (3)

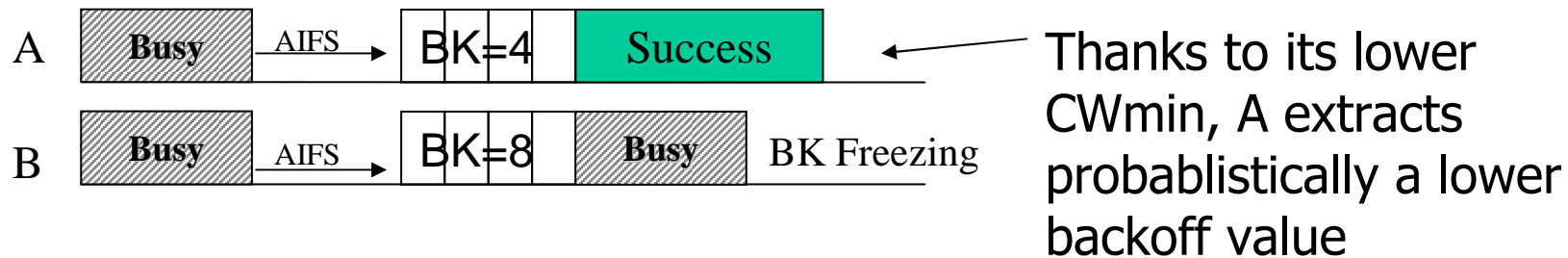


- PF is greater than 2 for LP stations.
- $CW_{new} = PF * CW_{old}$
- It is sufficient a single collision to begin the differentiation process.
- Impossible to force LP traffic to zero!
  - After a packet successful transmission, the PF effect is no more present

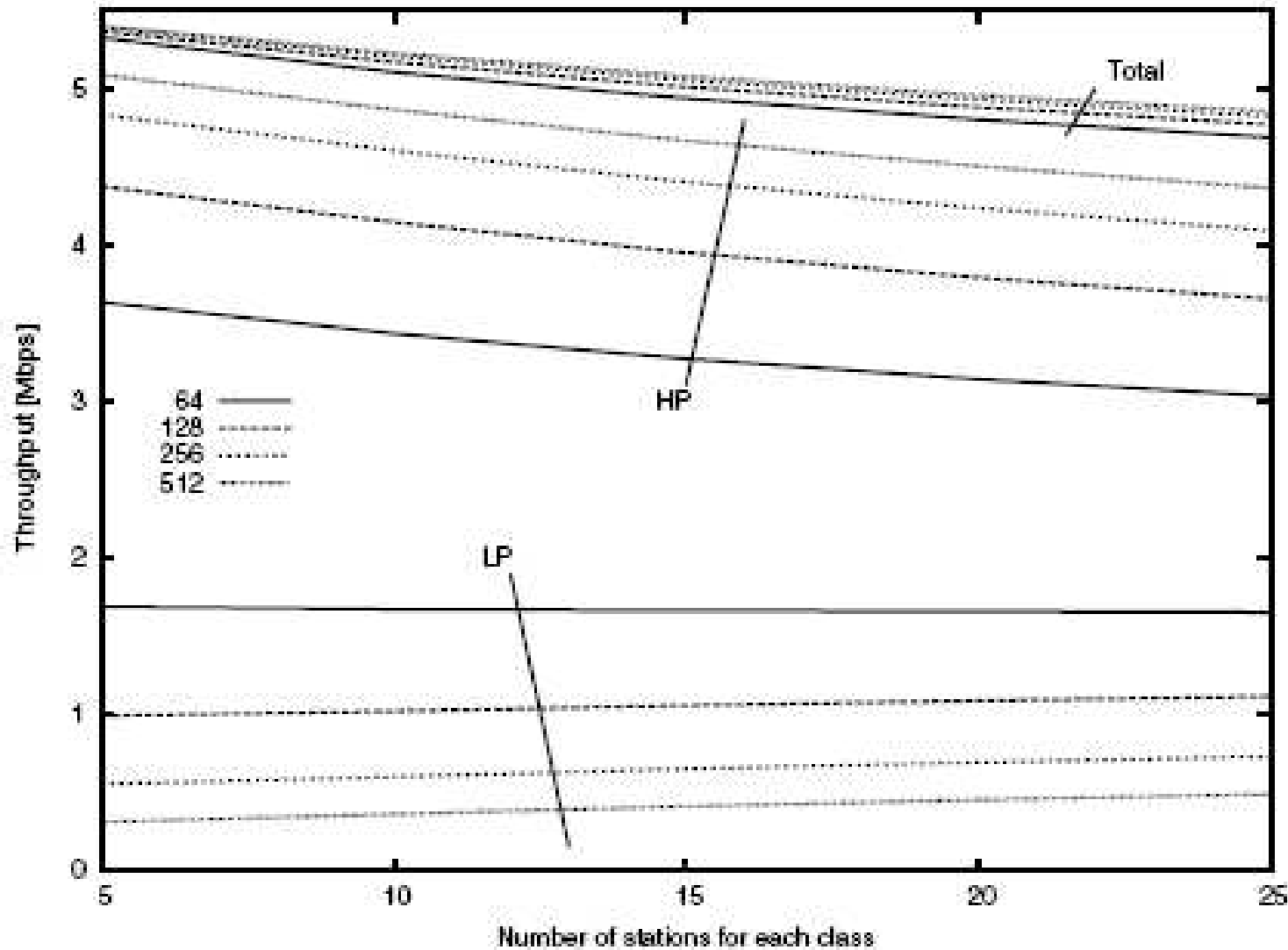


# CWmin Differentiation (1)

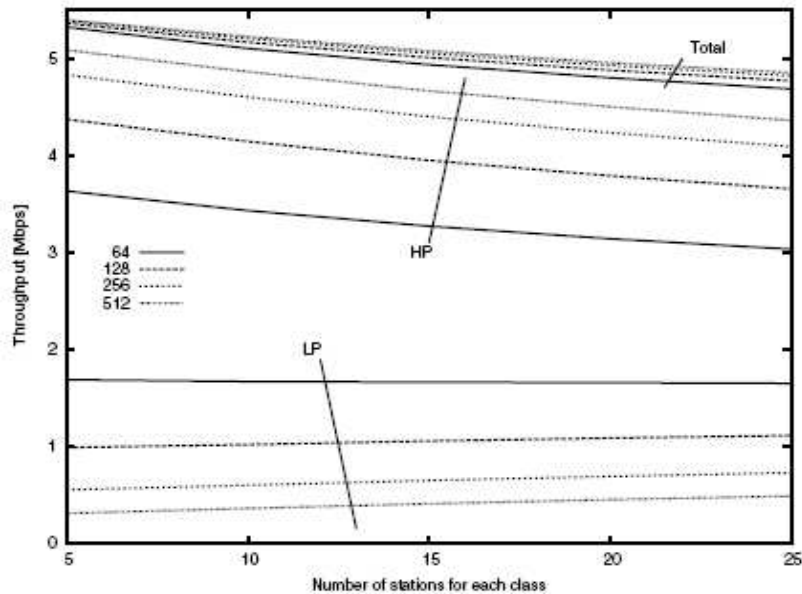
- $CWmin(A) < CWmin(B)$ 
  - In average, station A has a lower backoff than B



# CWmin Differentiation (2)



# CWmin Differentiation (3)

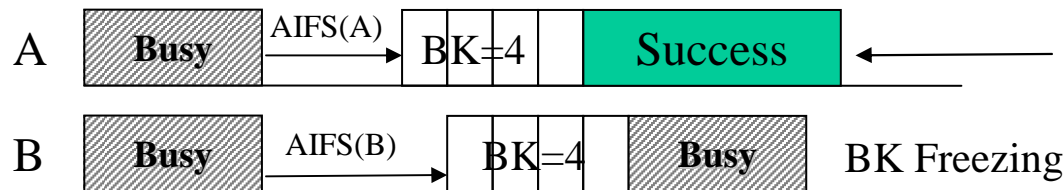


- Very High differentiation performance
- @ low loads performance is good
  - Collision effects among HPs not significant
- @ high loads collisions mainly involve HP stations (because of their small CW)
  - Degradations regard HP traffic - > bad!
  - LP traffic not affected
    - Collision effects un-altered



# AIFS Differentiation (1)

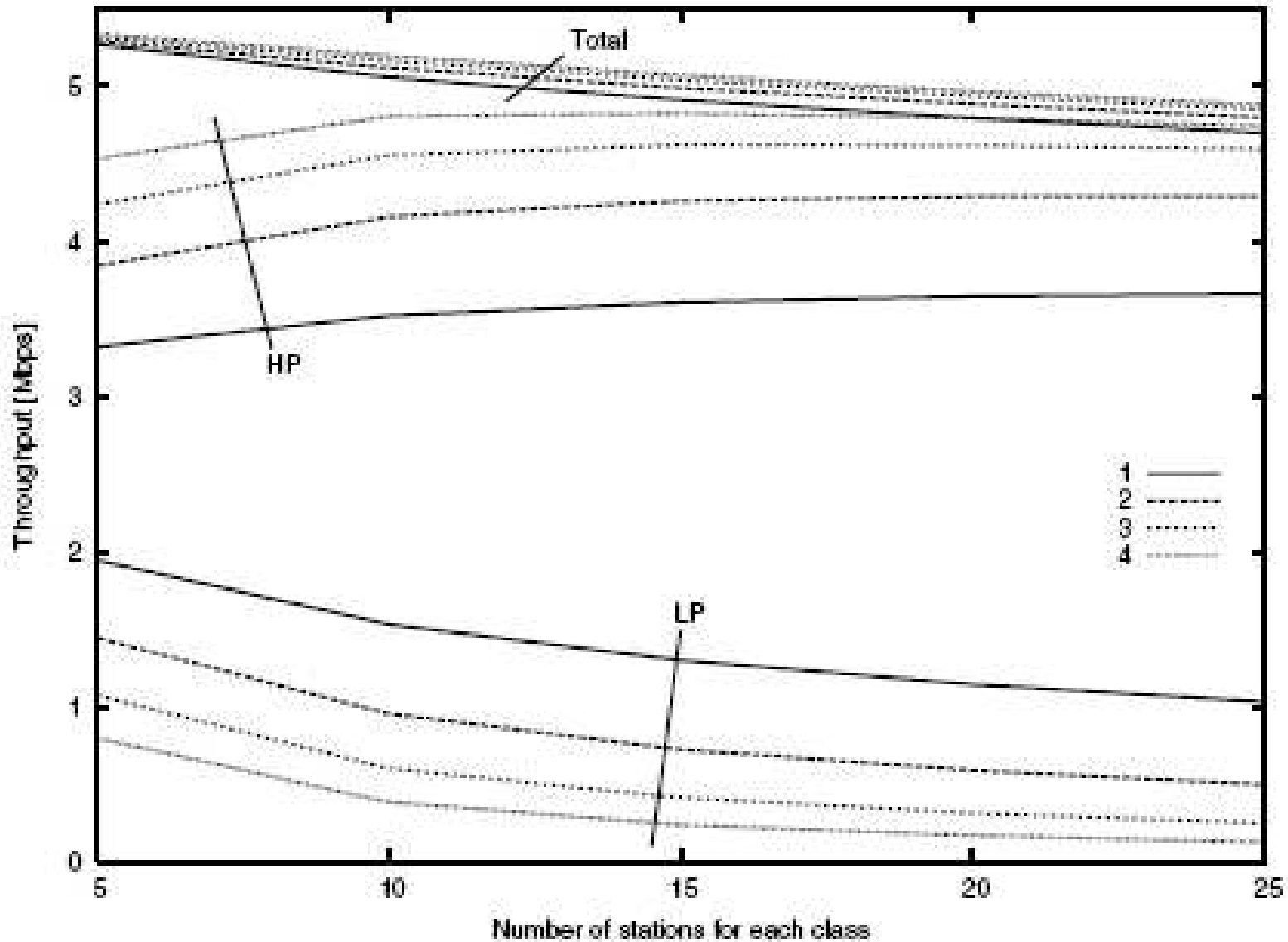
- $AIFS(A) < AIFS(B)$ 
  - station A decrements its backoff timer before than station B



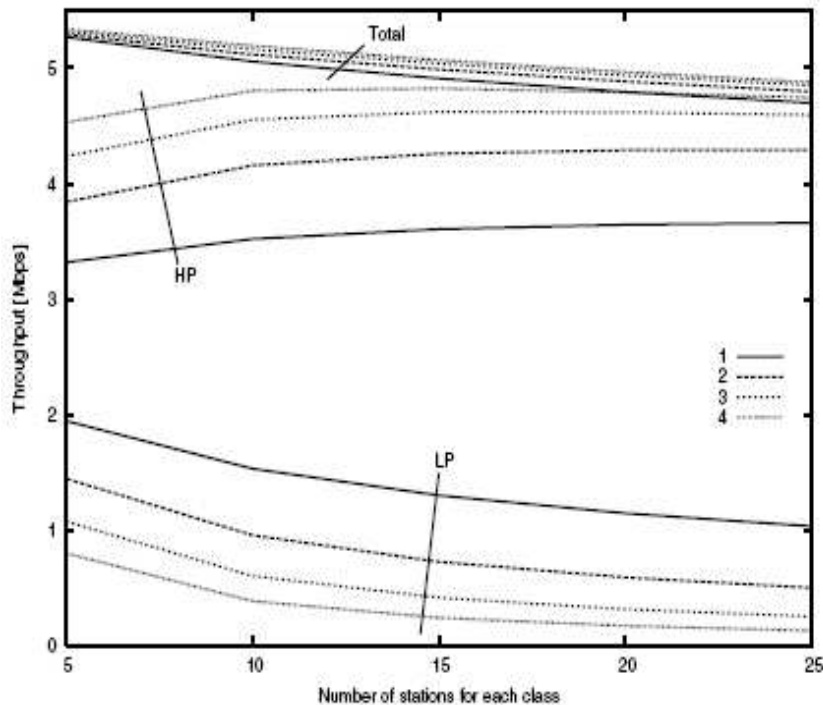
Thanks to its lower AIFS, A starts decrementing its backoff value before than B either after busy channel or idle channel conditions



# AIFS Differentiation (2)



# AIFS Differentiation (3)

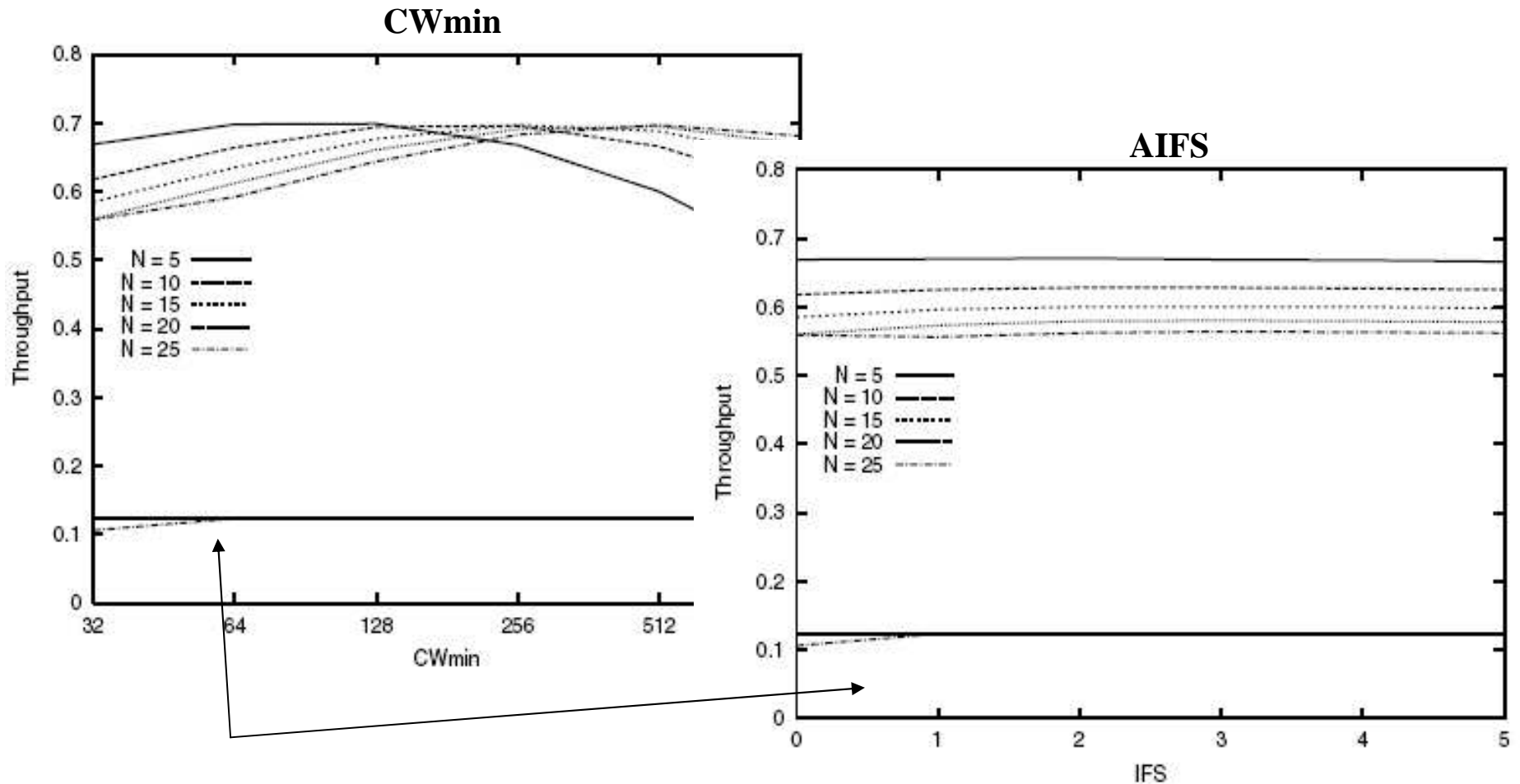


- Very High differentiation performance
  - Complementary to CWmin case
- @ low loads differentiation performance suffers
  - Collision are few ->
- @ high loads collisions mainly involve LP stations, since HP stations access first
  - Degradations regard LP traffic -> good!
  - HP traffic not affected



# Heterogeneous Sources: Throughput

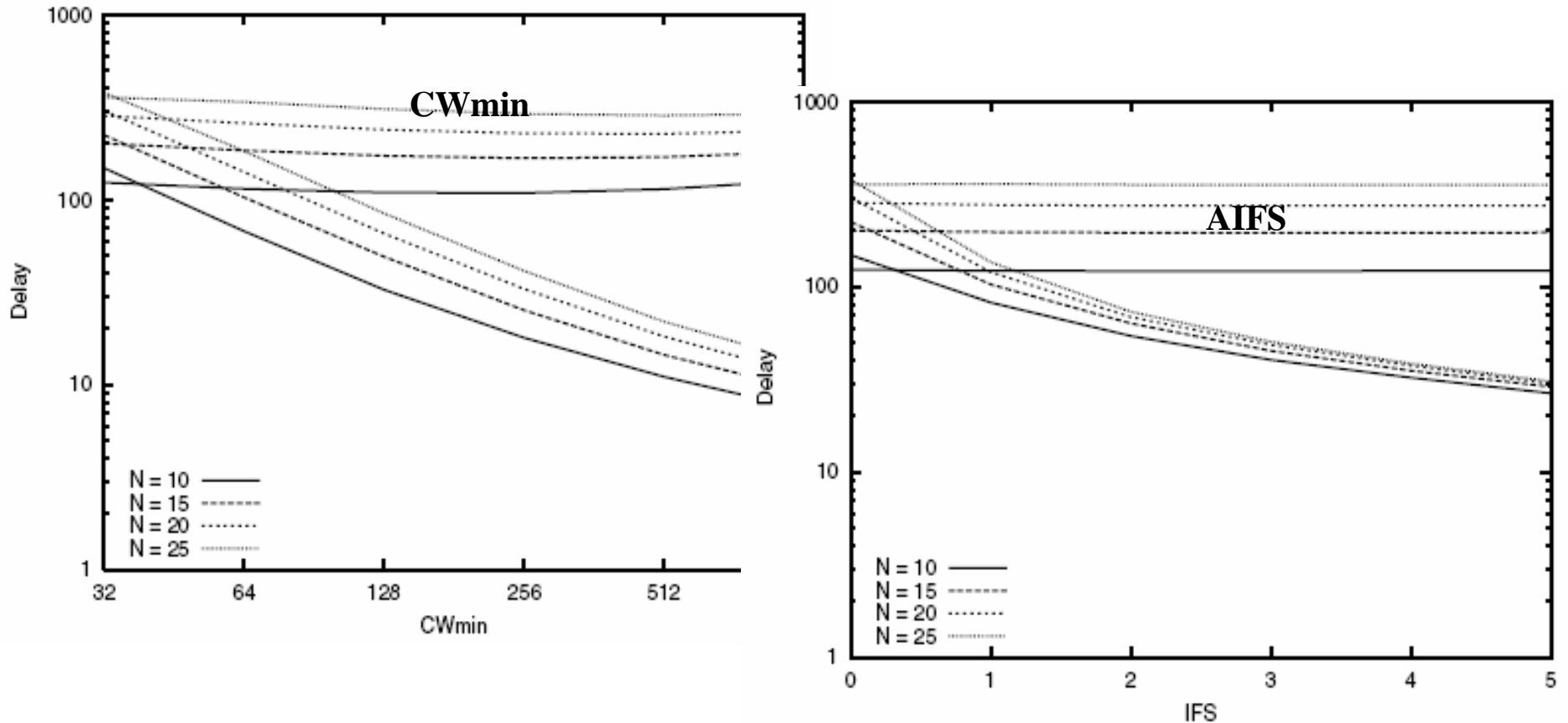
- Focus on AIFS and CWmin differentiation, seen to be most effective



**The minimum differentiation effect allows to guarantee HP traffic!!!**



# Heterogeneous Sources: Delay



- 1) CWmin more effective to manage delay behaviour than AIFS (see slopes)
- 2) AIFS differentiation slightly sensitive to load in terms of delay
- 3) Joint use: delay requirements satisfied with AIFS, throughput managed via CWmin (because of the maxima)



# Conclusions

- Cwmin and AIFS differentiation perform better than PF and CWmax differentiation
  - PF and CWmax differentiation operations allowed only by collisions
- CWmin and AIFS show a complementary behaviour
  - CWmin performance degrades @ high loads
  - AIFS performance degrades @ low loads
- Joint use of CWmin and AIFS
  - AIFS to meet delay requirements
  - CWmin to manage throughput performance
- Complex parameter setting
- Behavior hardly predictable



# Scheduling in 802.11e: Open-Loop or Closed-Loop?

**Paolo Larcheri**  
**Renato Lo Cigno**

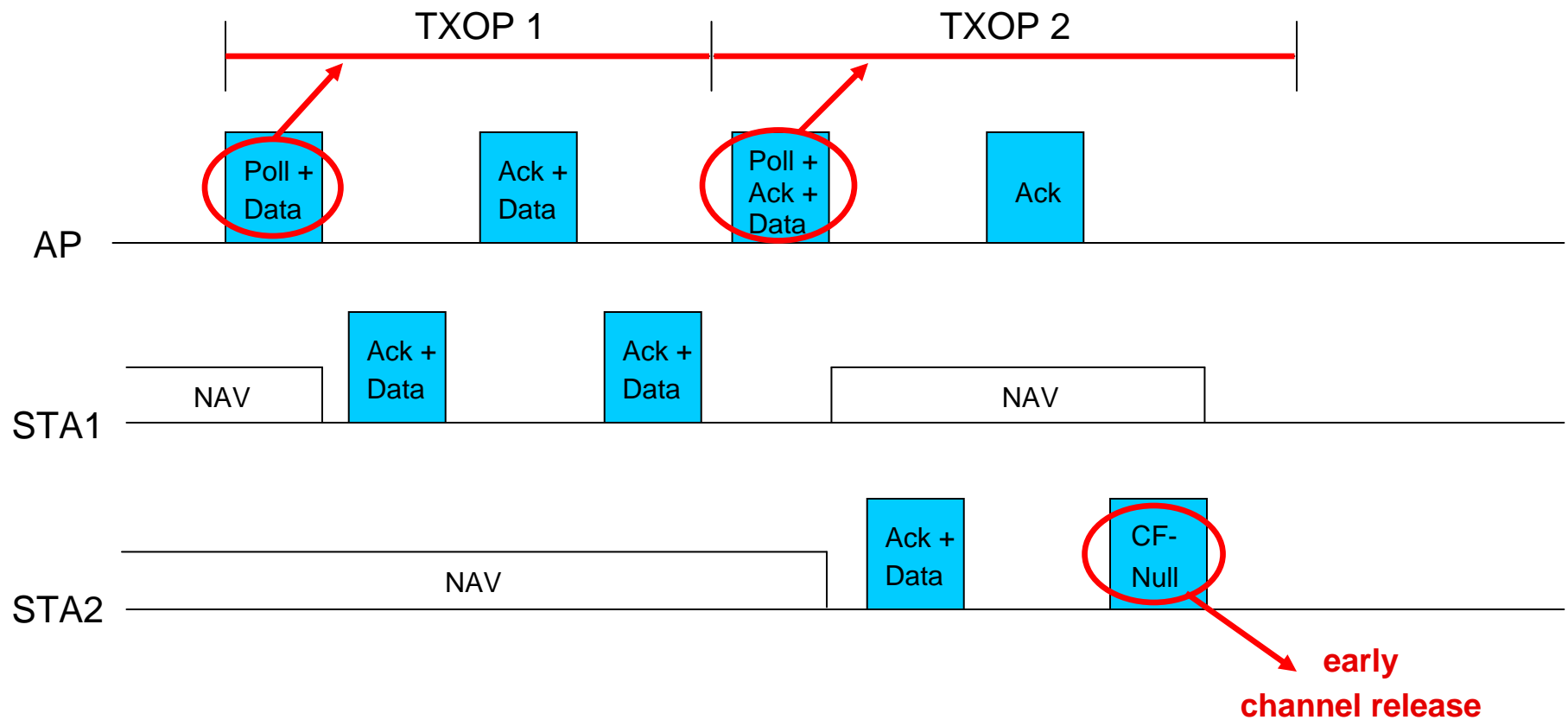
**Presented @ WONS 2006**

# Outline

- Scheduling: The Rules of The Game
- Sample (on the standard) Scheduler
- Equivalent Bandwidth Approach
- Closed Loop Scheduling: A Control Theoretic Approach



# MAC 802.11e: HCCA



# Resource Scheduling (1)

- Not essential to backward compatibility
  - The standard has just a reference impl. (SS)
- HCF is implemented in the AP
  - HCCA scheduling is a function of HCF
- Requirements of traffic flows are contained in the *Traffic Specifications (TSPEC)*:
  - Maximum, minimum and mean data rate
  - Maximum and nominal size of the MSDUs
  - Maximum Service Interval and *Delay Bound*
  - Inactivity Interval
  - ...



# Resource Scheduling (2)

- KEY notions are
  - Service Interval -  $SI(j)$ : The maximum amount of time between successive polling to a station  $j$
  - Transmission Opportunities -  $TXOP(j)$ : The amount of resources (time) assigned to station  $j$  in a single polling
- Goals of scheduling:
  - Find suitable values of SIs and TXOPs
  - Fully exploit resources
  - Guarantee quality and differentiation of the TSPECs



# Reference Implementation (SS)

Service Interval  $m = \min_i (\text{MaximumServiceInterval}_i)$

$$Sl = \max(x) \text{ t.c. } x < m \text{ e } Bl \bmod x = 0$$

TXOP  $N_i = \left\lceil \frac{SI \times \rho_i}{L_i} \right\rceil$   $T_i = \max\left(\frac{N_i \times L_i}{R} + O, \frac{M_i}{R} + O\right)$

- $\rho_i$  Mean data rate
- $L_i$  Nominal MSDU size
- $M_i$  Maximum MSDU size
- $R$  TX rate
- $O$  Overhead (Ack, SIFS,...)



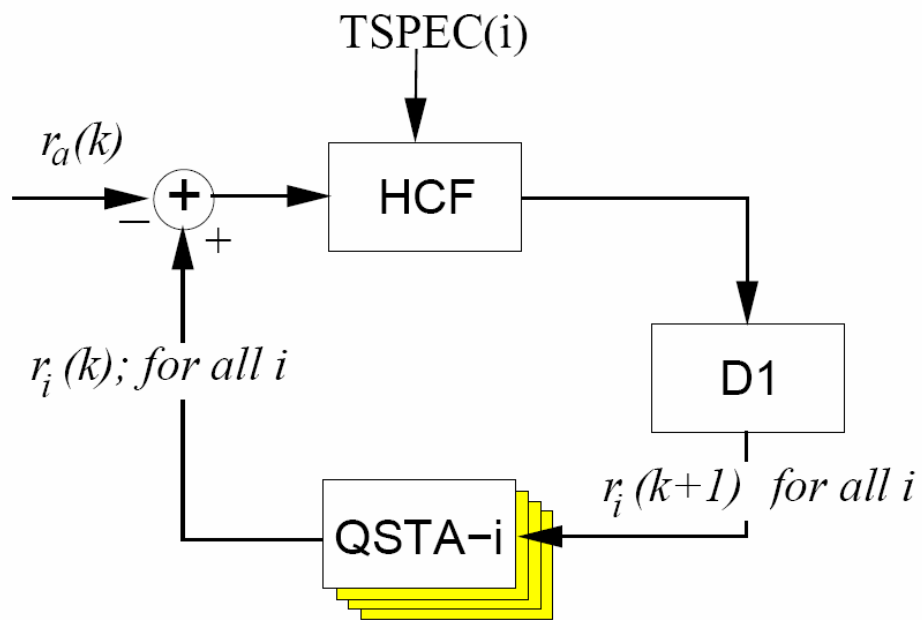
# Feedback Information ... or not?

- SS Schedules is open-loop:
  - Uses only TSPEC info
  - Assigns the mean rate: not suited for VBR ...
  - ... but you can assign a rate based on an **Equivalent Bandwidth** approach
- 802.11e has a field to feedback information about backlog (bytes or frames in queue)
  - Use this info for prediction or
  - Use this info for **closed-loop control?**

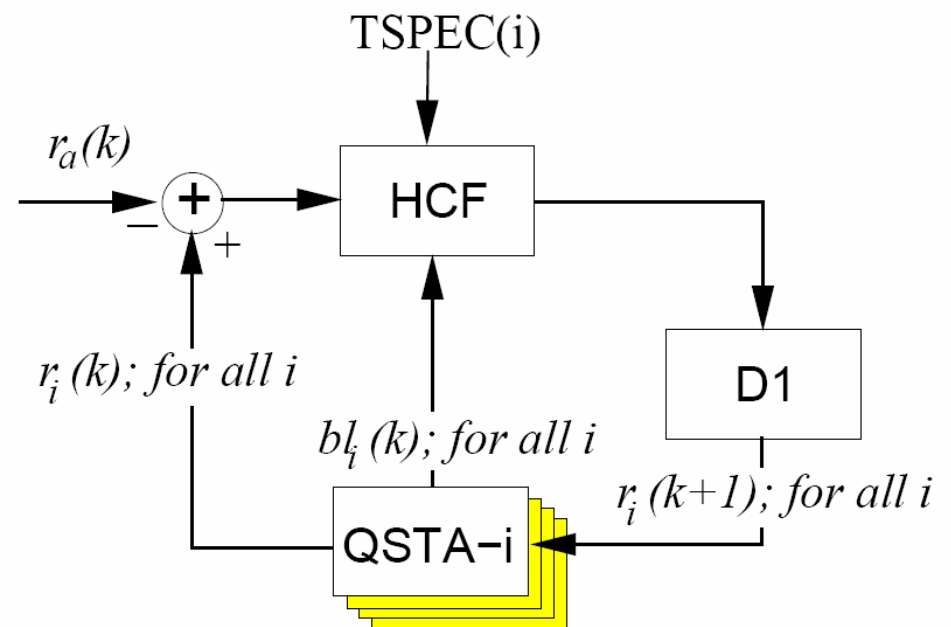


# Open/Closed Loop Scheduling

OPEN LOOP



CLOSED LOOP



# Equivalent Bandwidth

- Well known approach
  - Conceptually simple, just assign resources such that

$$P\left[\frac{\rho}{SI} > \frac{EB(p)}{SI}\right] = p$$

- $EB(p)$  is the assignment that guarantees  $p$  frame loss probability
- $\rho$  is the actual (time-depended) offered traffic
- **But** ... requires full stochastic knowledge of the traffic ☹️



# Closed-loop Scheduling: Basics

- Discrete time modeling
  - Just throw away time (creates a lot of problems)
  - The system evolves in cycles of SIs: 1,2,3...,k
- Goal: equalize (to zero) all queues
- Max/Min fair approach
  - Only resources above the minimum guarantee are "controlled"
- Assumption: There is a CAC function ensuring long-term stability
  - Can use large loop gains without oscillation risks



# Closed-loop Scheduling: Formulae

$$\frac{1}{K} \sum_{k=1}^K r_a(k) > \sum_{i=1}^{N_{QS}} \bar{r}_i$$

CAC based long term stability:  
the average available resources  
over a finite time K are larger  
than the average assigned resources

$$r_j(k) = r_j^{\min}(k) + r_j^+(k)$$

$$r_j^+(k+1) = \frac{B_j(k)}{\sum_{j=1}^{N_{TS}} B_j(k)} \left[ r_a(k+1) - \sum_{j=1}^{N_{TS}} r_j^{\min}(k+1) \right]$$



# Closed-loop Scheduling: Formulae

$$\frac{1}{K} \sum_{k=1}^K r_a(k) > \sum_{i=1}^{N_{QS}} \bar{r}_i$$

$$r_j(k) = r_j^{\min}(k) + r_j^+(k)$$

Max/Min Fairness  
 $\mathbf{r}^{\min}$  are guaranteed  
and not subject to control  
 $\mathbf{r}^+$  is strictly non negative

$$r_j^+(k+1) = \frac{B_j(k)}{\sum_{j=1}^{N_{TS}} B_j(k)} \left[ r_a(k+1) - \sum_{j=1}^{N_{TS}} r_j^{\min}(k+1) \right]$$



# Closed-loop Scheduling: Formulae

$$\frac{1}{K} \sum_{k=1}^K r_a(k) > \sum_{i=1}^{N_{QS}} \bar{r}_i$$

Simple proportional controller  
splitting excess resources  
among all the flows that are  
backlogged

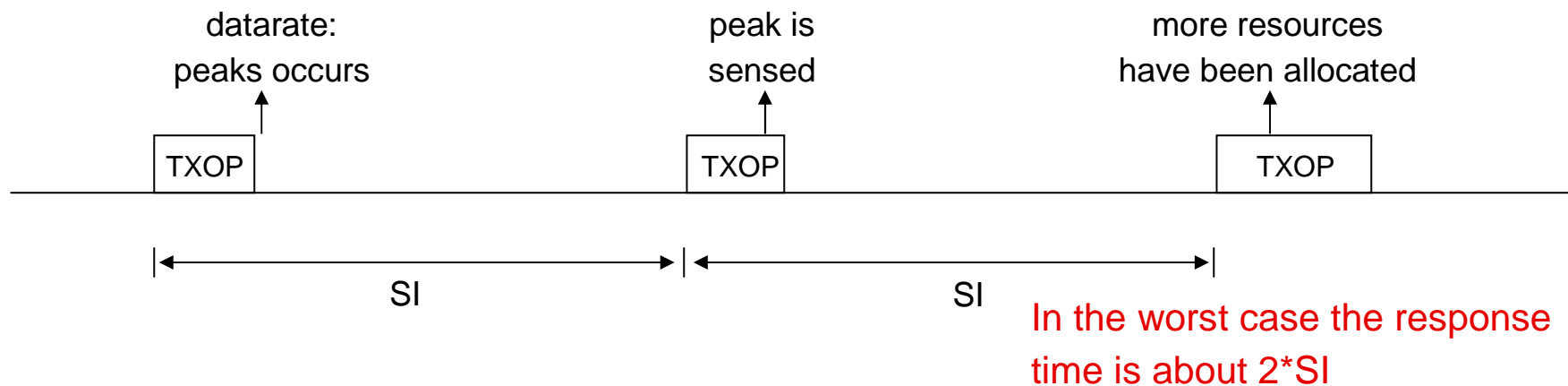
$$r_j(k) = r_j^{\min}(k) + r_j^+(k)$$

$$r_j^+(k+1) = \frac{B_j(k)}{\sum_{j=1}^{N_{TS}} B_j(k)} \left[ r_a(k+1) - \sum_{j=1}^{N_{TS}} r_j^{\min}(k+1) \right]$$



# Details ... the real doom!

- Highly quantized resource assignment
  - A minimum assignment of one maximum size segment is mandatory ... what if the station transmits at low rate?
  - "Fragments" of frames might lead to waste resources
- Reaction of the controller can be sluggish



# Closed-loop Schedulers

- MMF-A
  - Implements the formulae above
  - Have quantization and response problems
- MMF-AR
  - Dynamically changes the SI 'on-demand' 😊
  - Reassign spare resources at the end of the CFP
  - Violates proportional assignment to avoid quantization problems



# Simulations

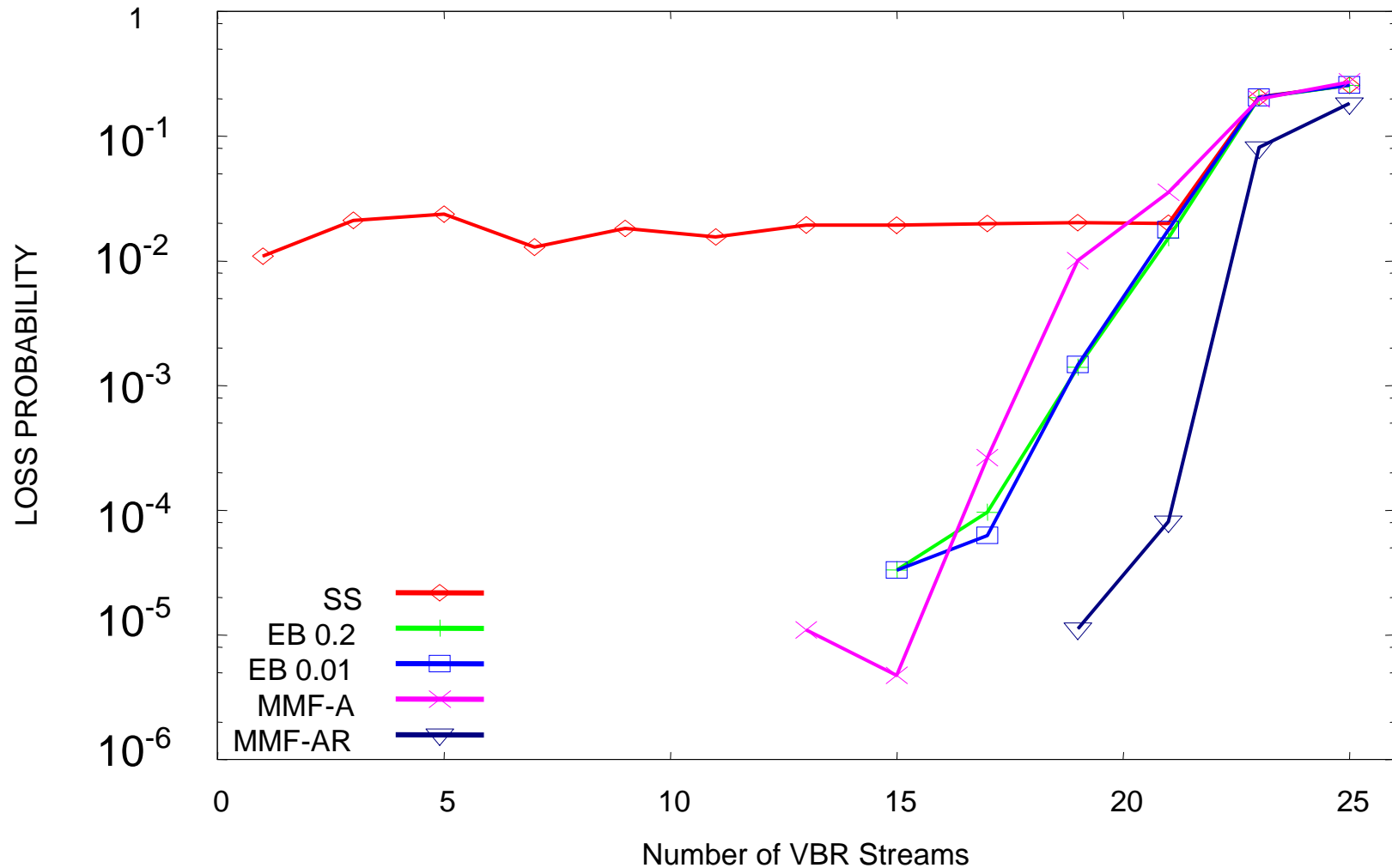
- Schedulers implemented in ns2
- ns2 adapted to draft 11.0 of 802.11e standard
- 2 and 3 state VBR sources implemented as MMDP processes
- EB(p) exact computation for the MMDP sources
- The package use for simulations is available at <http://twelve.unitn.it/tools.html>



# Results

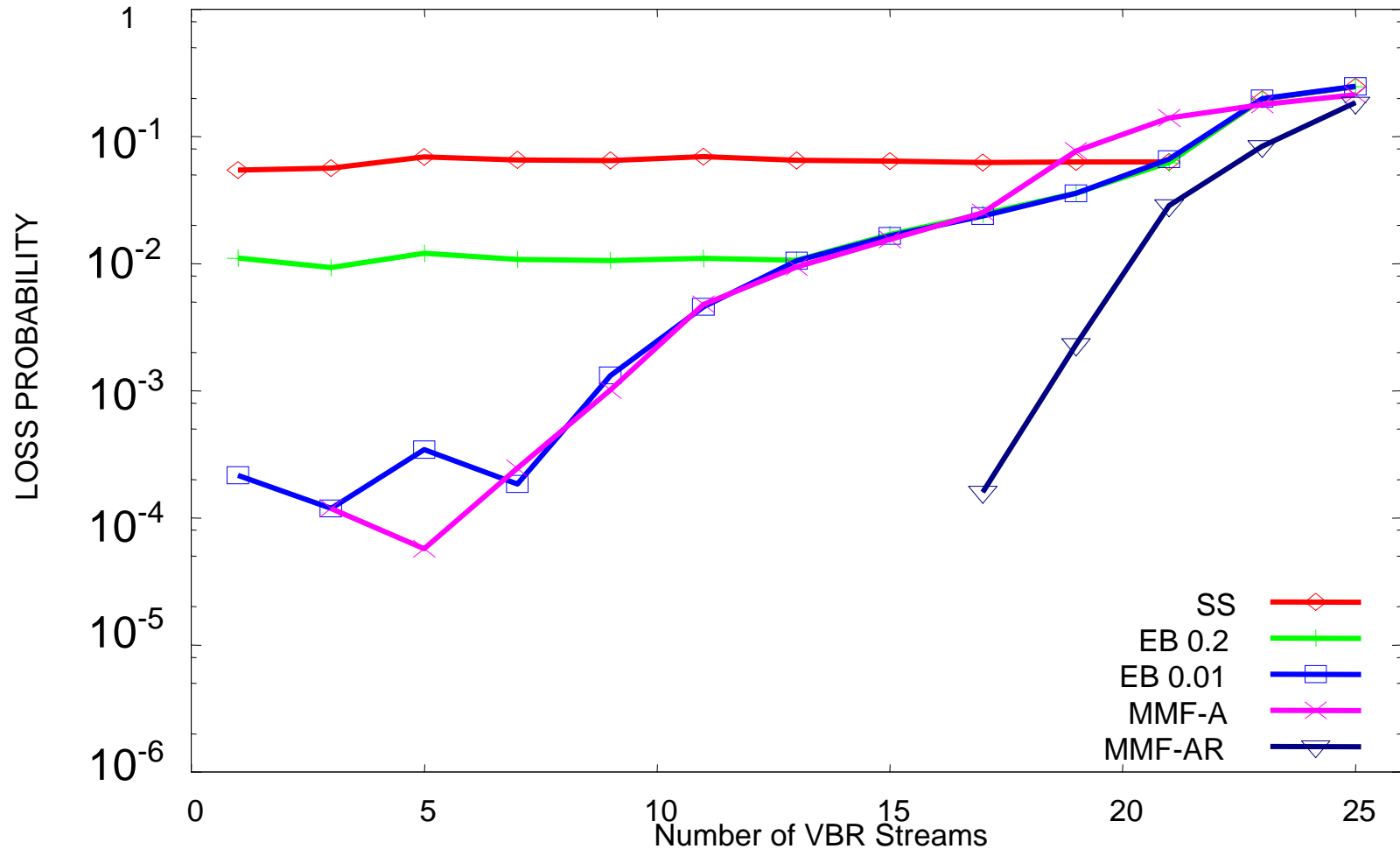
Traffic VBR-3: both packet size and interarrival time change

Delay Bound =  $\infty$  Buffer Size = 50 pck Service Interval = 50 ms



# Results

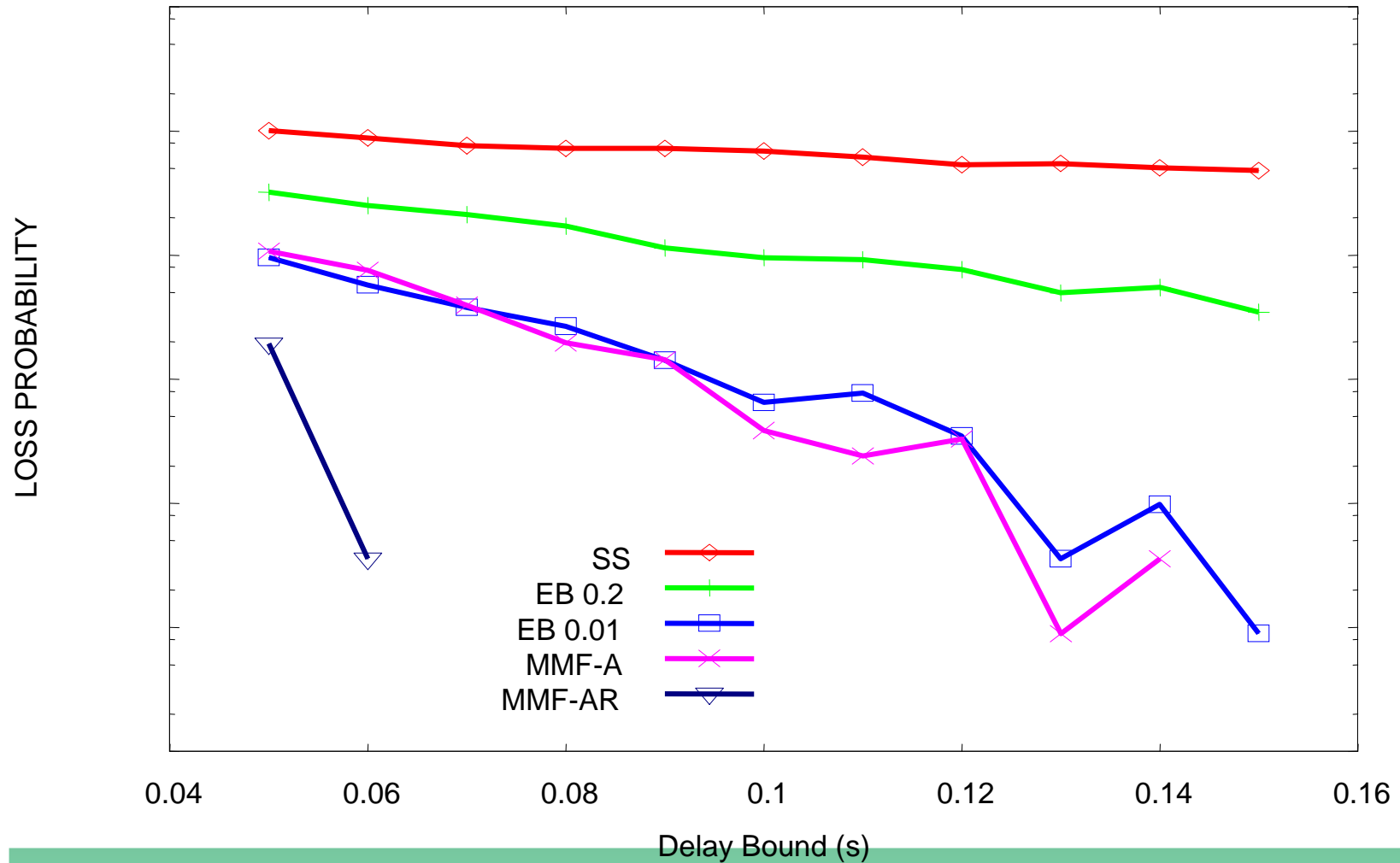
Traffic VBR-3: both packet size and interarrival time change  
Delay Bound = 100ms Buffer Size = 50 pck Service Interval = 50 ms



# Results

Sorgenti VBR-3: both packet size and interarrival time change

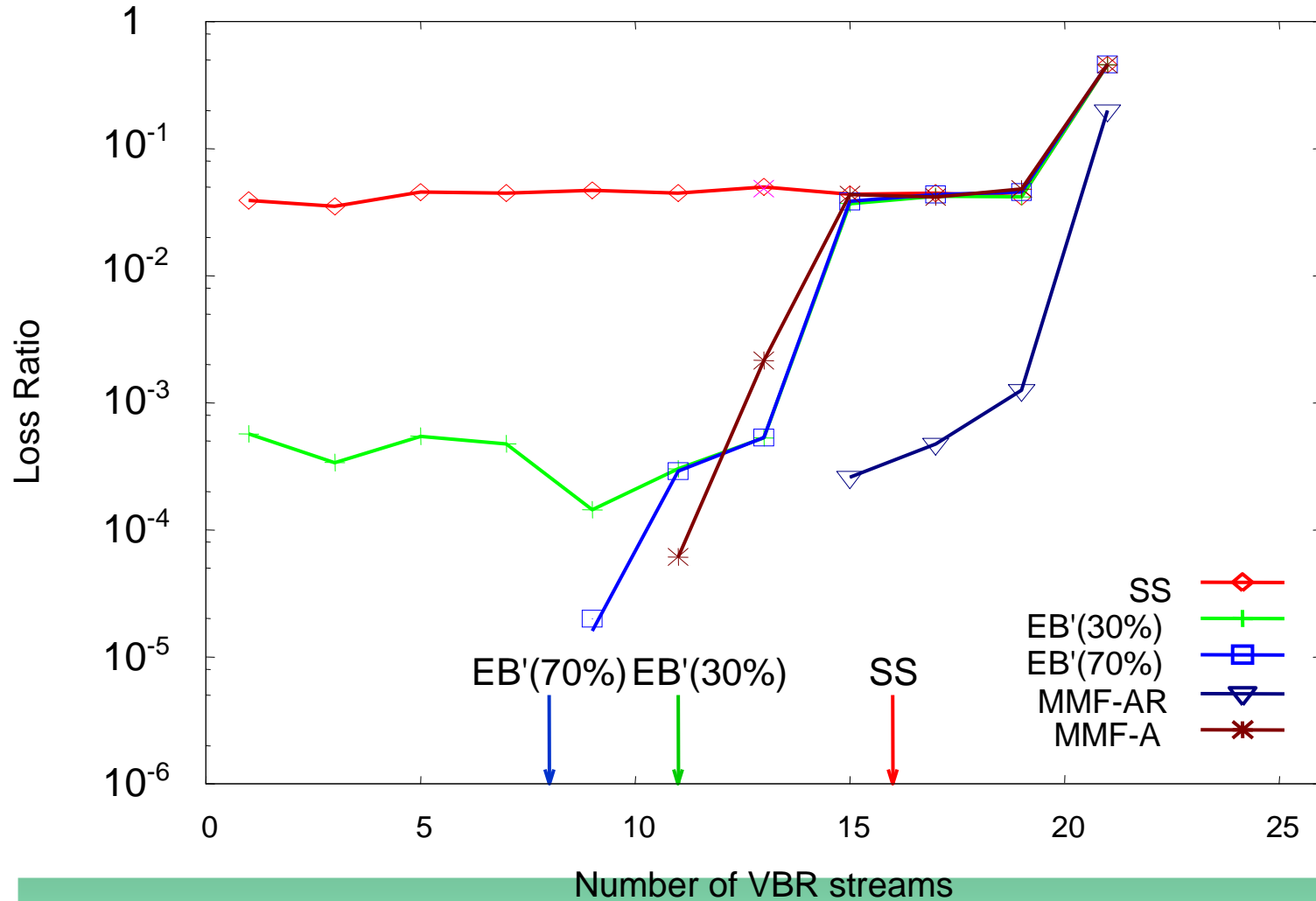
Buff. = 50pck    #stream = 8    Service Interval = 50 ms



# Results

Real Video Traces: h.263 codec → **EB???**

Delay bound = 150ms      Service Interval = 100 ms



# Conclusions

- Different HCCA scheduling explored
- HCCA complexity is manageable, performances are better than EDCA, configuration is easier
- Closed-loop scheduling:
  - Viable alternative to open-loop or predictive scheduling
  - Complexity much simpler and effective than Equivalent Bandwidth approaches
- The BIG problem are details
  - Quantization, Normalization, Spare Resource Collection, ...

