Defining an Effective and Green Wireless-Link Packet Scheduler through a Modular Architecture

Carlo Augusto Grazia

Department of Engineering *Enzo Ferrari*
University of Modena and Reggio Emilia

Modena, 15 October 2013
Workshop SFINGI
Software routers to Improve Next-Generation Internet (PRIN 2009)

Talk overview

Problem

what

to provide features over a wireless link

- throughput boosting and energy saving
- QoS guarantees

why

radio channels are unreliable

- burst channel error (multipath, fading, interference, noise, ecc...)
- user mobility

where

packet scheduler

State of the Art

typical solution

single **integrated** scheduler

weaknesses

- merge both QoS guarantees and wireless link issues
  - QoS $\longrightarrow$ IP level
  - link issues $\longrightarrow$ MAC/PHY level
- high-quality schedulers for wired links are unusable without modifications
- different technology or solution means to modify (again) the scheduler

Proposed solution 1/3 MAC-SAL *Scheduling&Abstraction Layer*

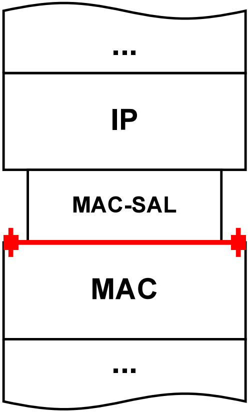
modular architecture

extends the network stack by adding a special **middle layer** on top of the MAC (decouple QoS and throughput problems)

bottom side

deals with the idiosyncrasies of the wireless link

- transmission reliability
- throughput boost using channel state information
- energy saving



Proposed solution 2/3 MAC-SAL *Scheduling&Abstraction Layer*

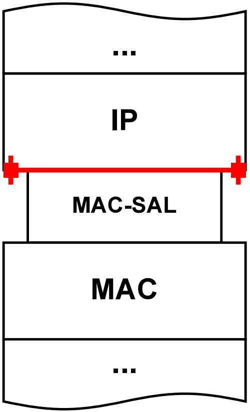
modular architecture

extends the network stack by adding a special **middle layer** on top of the MAC (decouple QoS and throughput problems)

top side

exports the abstraction of a link

- function *link_ready()*
- transparency for IP layer
- avoid cross-layering (IP-level)



Proposed solution 3/3 MAC-SAL *Scheduling&Abstraction Layer*

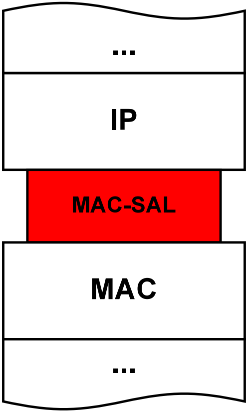
modular architecture

extends the network stack by adding a special **middle layer** on top of the MAC (decouple QoS and throughput problems)

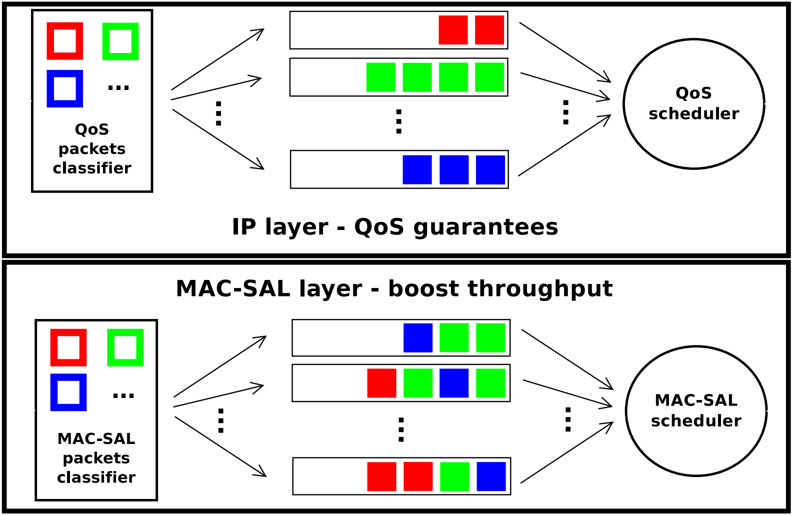
internally

MAC-SAL layer scheduler

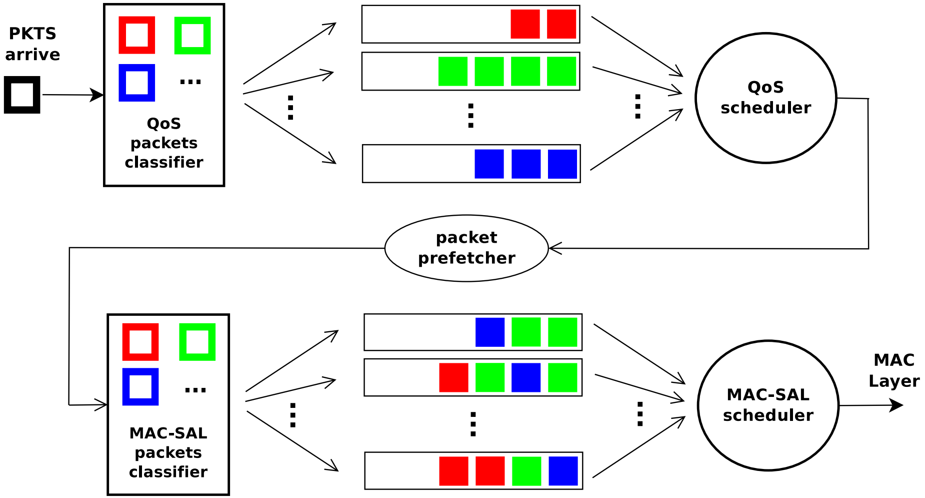
- shared buffer with **M** virtual queues
- buffer size equal to **Q** packets



Architecture: double scheduler



Architecture: double scheduler



Benefits

1. for QoS guarantees, existing packet schedulers for wired links can be used <u>without modification</u>
2. the same packet scheduler can be used
   - on heterogeneous wireless technologies
   - with different solutions to boost the throughput
   - only values/parameters of MAC-SAL scheduler change
3. high throughput through *cross-layering*, while still preserving *flexibility*

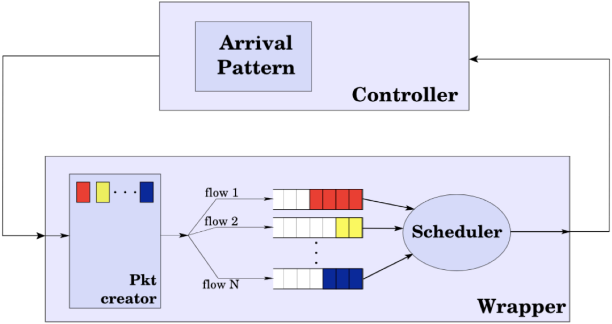
Test EnvironMent for Performance Evaluation of the Scheduling of packeTs

- UNIX-based open tool
- simulate both wired and wireless environment
- possibility to execute *original* scheduler alone or plugged into a *double* scheduler
  - different schedulers available by default
  - easy to add new schedulers
- performance measured
  - execution time
  - energy consumption
  - throughput
  - queueing delay, B-WFI, T-WFI, RFI

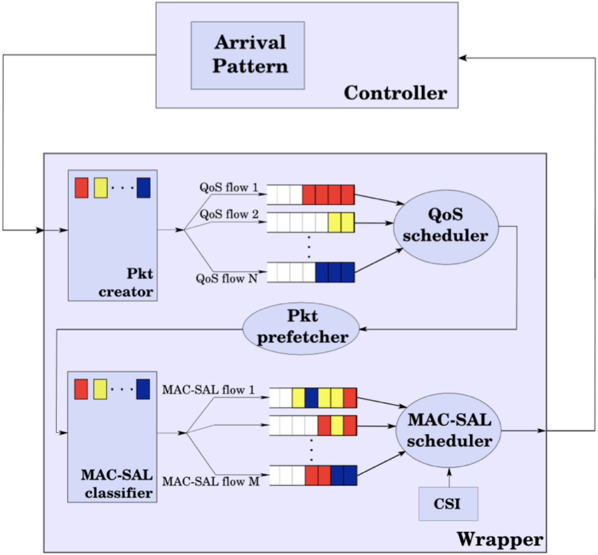
Test EnvironMent for Performance Evaluation of the Scheduling of packeTs

- schedulers used:
  - WF^2Q+: optimal service guarantees, $O(logn)$ cost
  - DRR: $O(n)$ deviation from optimal service, $O(1)$ cost
  - QFQ+: quasi-optimal service guarantees, execution time close to DRR
  - W^2F^2Q: best integrated scheduler with $O(n)$ cost
- easy run-time configuration
  - single/double scheduler mode
  - number of flows (QoS and/or MAC-SAL), weight distribution
  - Q buffer size
  - realistic packets arrival pattern

Single mode Test Environment



Double mode Test Environment

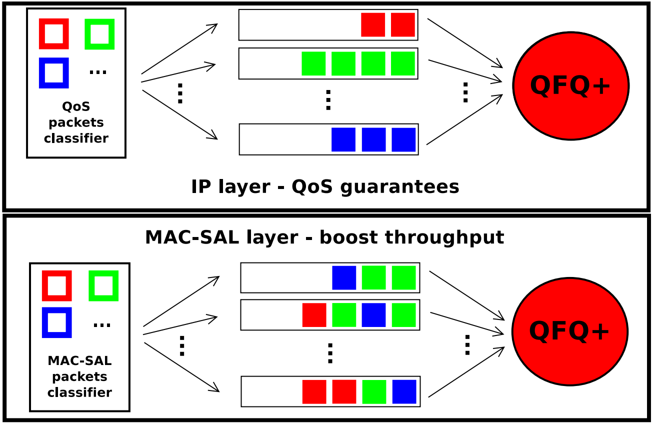


Reference Scenario

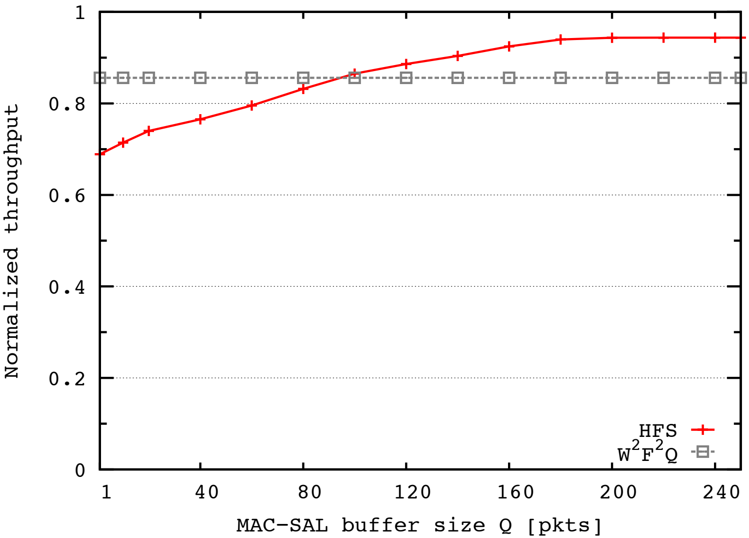
- 20 wireless stations
- link rate 54 Mb/s
- one MAC-SAL flow per wireless station
- MAC-SAL flow packet loss probability
  - ranging linearly from 10^0 to 10^{-1}
  - outsider values as 10^{-2}, 10^{-3} and 10^{-4}
  - static
- MAC-SAL flow weight distribution
  - analogical: $\phi_k = (1 - P_{loss_k}) \cdot 1000$
- 100 QoS flows with different weights

HFS: High-throughput twin Fair Scheduler

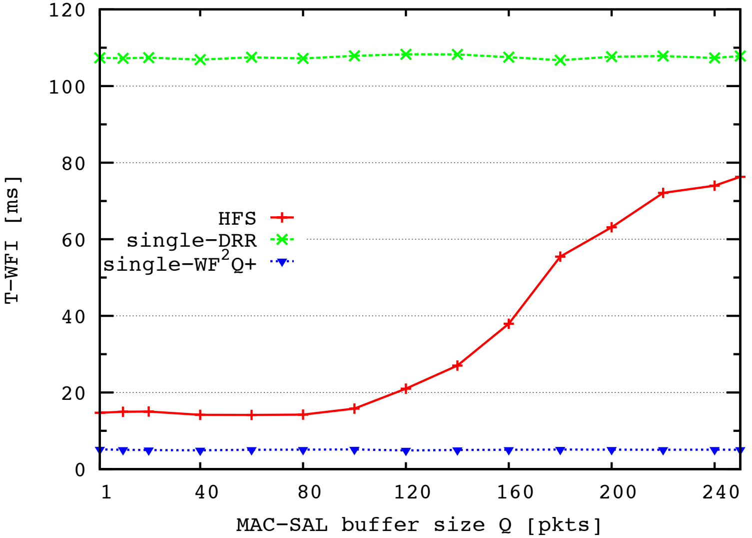
QoS layer: quasi-optimal service guarantees, cost close to DRR
MAC-SAL layer: high throughput, quasi-optimal service guarantees, cost close to DRR



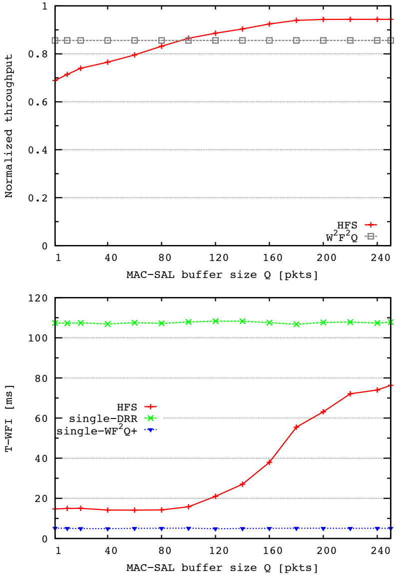
Throughput of HFS against W^2F^2Q



T-WFI of HFS against WF^{2}Q+ and DRR



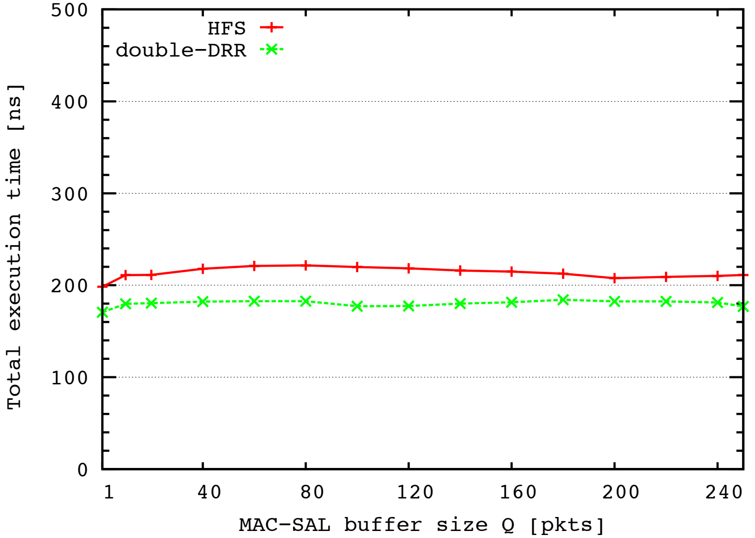
Tradeoff between QoS guarantees and throughput boosting



Tunable parameter:

- the higher is Q, the higher is the throughput
- the lower is Q, the higher is QoS guarantees

Execution time of HFS against DRR



Conclusions

Architecture

we defined a feasible, flexible and modular architecture which decouples QoS guarantees and link issues tasks

HFS

we implemented a new flexible, efficient and green packet scheduler for wireless links

- throughput higher than W^2F^2Q
- T-WFI close to WF^2Q+
- execution time close to DRR
- low energy consumption due to:
  - increase throughput → more packets successfully transmitted per energy consumed → less retransmission → **less power consumption**
  - low execution time per packet processing → **less power consumption**

Future Works

- benefits for the transport layer (e.g. TCP goodput)
- dynamic weight distribution
- implement and integrate different channel models (e.g. WiMAX, 3G/LTE)

References

Conferences:

M. Casoni, A. Paganelli, P. Valente
A Modular Architecture for QoS Provisioning over Wireless Links.
PAEWN'13. (2013) 95-100, Barcelona

M. Casoni, C.A. Grazia, P. Valente
A Flexible and Green Scheduler for providing QoS and High Throughput over Wireless Links.
ICCST'13. (2013), Hammamet

Journals submissions:

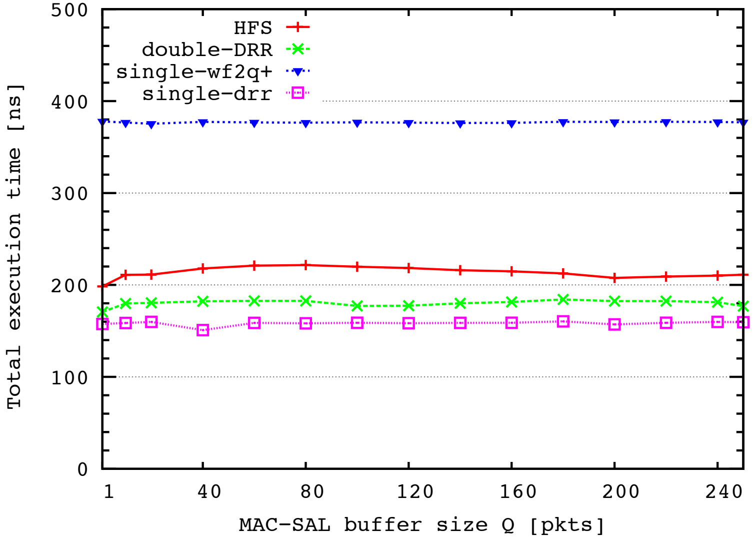
M. Casoni, C.A. Grazia, P. Valente
Defining an Effective Wireless-Link Packet Scheduler through a Modular Architecture.
COMNET. Elsevier

M. Casoni, C.A. Grazia, P. Valente
TEMPEST: a new Test EnvironMent for Performance Evaluation of the Scheduling of packeTs.
SIMPAT. Elsevier

thank you
for the attention

extra slides

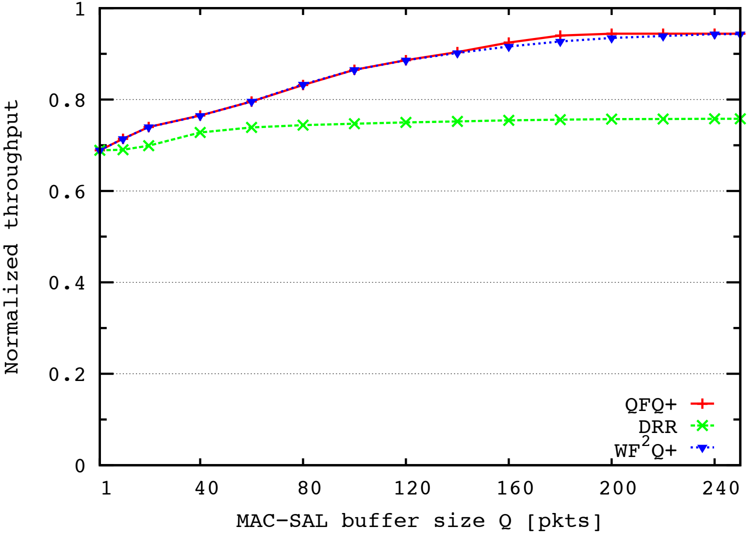
Execution time of HFS against all



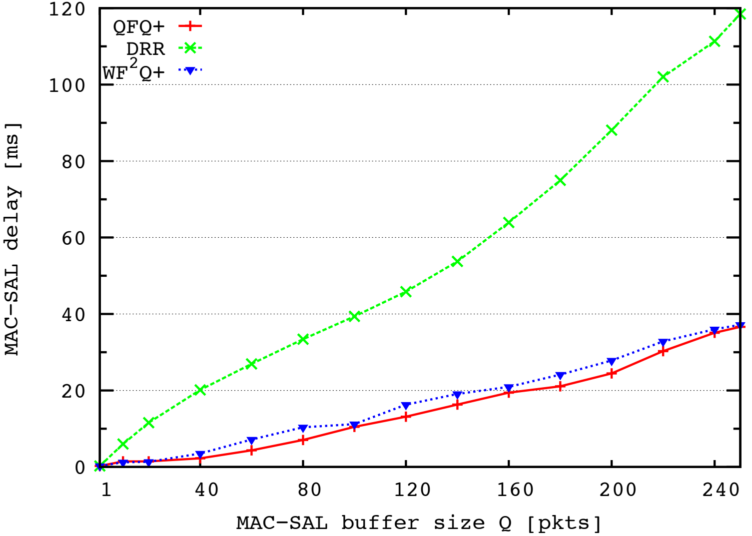
Guarantees

1. analytical
   - Deficit Round Robin scheduler in MAC-SAL
   - weight per-flow proportional to the max possible throughput
   - worst-case bandwidth displacement
   - MAC-SAL additional delay
2. sperimental
   - proof the effectiveness of the architecture through simulation
   - test environment UNIX-based
   - different schedulers tested
   - different parameters for a possible, realistic scenario

Normalized throughput for different MAC-SAL schedulers



Queueing delay for different MAC-SAL schedulers



Execution time for different MAC-SAL schedulers