A Low-Latency and High-Throughput Scheduler for Emergency and Wireless Networks

Maurizio Casoni, Carlo Augusto Grazia, Paolo Valente

Department of Engineering Enzo Ferrari
University of Modena and Reggio Emilia

Sydney, 14 June 2014
International Workshop on Advances in Public Safety and Emergency Communications APSEC’2014
The PPDR-TC project: Public Protection and Disaster Relief - Transformation Center

PPDR-TC goals

- Effective Public Protection & Disaster Relief (PPDR) communications
- Preparation of the next generation of PPDR systems

The Consortium:
Talk overview

1. Introduction
   - Problem
   - State of the Art

2. Proposed solution
   - Modular Architecture
   - Benefits

3. Results
   - Test Environment
   - Reference Scenario
   - HFS packet scheduler

4. Conclusions

5. Future Works
**Problem**

**what**

- to support PPDR communications over wireless links
- 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 $\rightarrow$ IP level
  - link issues $\rightarrow$ 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
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)
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

IP layer - QoS guarantees

MAC-SAL layer - boost throughput
Architecture: double scheduler
Benefits

1. for QoS guarantees, existing packet schedulers for wired links can be used without modification

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

- UNIX-based open tool
- possibility to execute original scheduler alone or plugged into a double scheduler
- schedulers used:
  - W\textsuperscript{2}FQ\textsuperscript{+}: optimal service guarantees, $O(\log n)$ cost
  - DRR: $O(n)$ deviation from optimal service, $O(1)$ cost
  - QFQ\textsuperscript{+}: quasi-optimal service guarantees, execution time close to DRR
  - W\textsuperscript{2}F\textsuperscript{2}Q: 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
  - packets arrival pattern
Reference Scenario
Reference Scenario

- 20 first responders (FR)
- link rate 54 Mb/s
- one MAC-SAL flow per FR
- 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
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 $W^2F^2Q$
B-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

- HFS
- DRR
- WF²Q+
- WF²Q

Total execution time [ns] vs MAC-SAL buffer size Q [pkts]
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 $W^2F^2Q$
- B-WFI close to $WF^2Q+$
- execution time close to DRR
- low energy consumption due to:
  - increase throughput $\rightarrow$ more packets successfully transmitted per energy consumed $\rightarrow$ less retransmission $\rightarrow$ **less power consumption**
  - low execution time per packet processing $\rightarrow$ **less power consumption**
Future Works

- benefits for the transport layer (e.g. TCP goodput)
- implement and integrate different channel models (e.g. WiMAX, 3G/LTE)
- real testbed
thank you
for your attention
Workshop on Emergency Networks for Public Protection and Disaster Relief

October 8, 2014
Cyprus

Submission deadline: July 8, 2014 through edas
Workshop site: http://en4ppdr.ing.unimo.it/
extra slides
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. experimental
   - proof the effectiveness of the architecture through simulation
   - test environment UNIX-based
   - different schedulers tested
   - different parameters for a possible, realistic scenario
Reference Scenario
Normalized throughput for different MAC-SAL schedulers

- QFQ+
- DRR
- WF
- 2Q+

MAC-SAL buffer size $Q$ [pkts]

Normalized throughput

MAC-SAL buffer size $Q$ [pkts]
Queueing delay for different MAC-SAL schedulers

- QFQ+
- DRR
- WF

MAC-SAL delay [ms]
MAC-SAL buffer size Q [pkts]

MAC-SAL delay [ms]
MAC-SAL buffer size Q [pkts]