RT-QoS for Wireless ad-hoc Networks of Embedded Systems

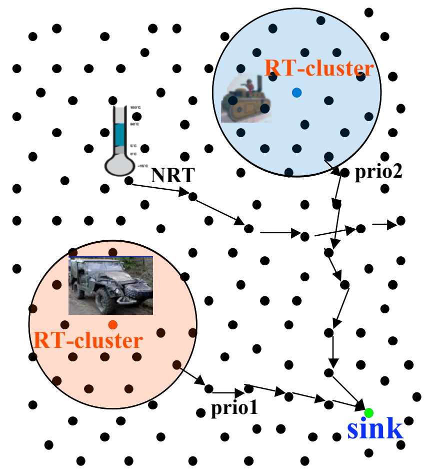
Marco Caccamo
University of Illinois
Urbana-Champaign

Outline

- Wireless RT-QoS: important MAC attributes and faced challenges
- Some new ideas and results for embedded systems:
  - Implicit contention & RI-EDF for single hop scenario
  - Real-time chains for multi-hop scenario

Can we deploy a WSN that reacts in real-time (RT-WSNs)?

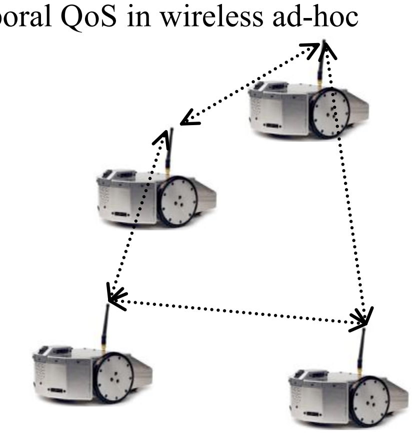
- Example of multi-hop scenario for RT-WSNs
- Whenever an event of interest is detected, create a RT cluster around it. The cluster should provide 1) **high bandwidth**, 2) **soft real-time guarantee**
- Geographic Forwarding (GF) routing protocol can be used to establish a communication flow between the RT cluster and the sink.
- A notion of priority is needed to properly schedule the shared wireless channel among different real-time flows.
- RT-WSNs will support audio/video streaming and enhance existing sensor network applications such as surveillance, environmental monitoring, etc.



3

Challenges when providing RT-QoS

- Important attributes when providing temporal QoS in wireless ad-hoc networks of embedded systems
  - **prioritization of Medium Access**
  - **robustness**
  - **power awareness**
  - **dynamic handling of mobile nodes**
  - **scalability (multi-hop)**
  - **adaptation to dynamic workloads**



- Two main challenges need to be addressed:

  a) Mitigate/avoid packet collisions on wireless channel,

  b) Avoid unbounded priority inversions when accessing wireless medium.

4

Some new ideas and results for embedded systems

- We developed RI-EDF protocol for the single-hop case. Major strengths are robustness, high bandwidth, soft RT guarantee, and low jitter
- We introduced the novel idea of Real-Time Chain for the multi-hop case. It allows to establish multi-hop soft real-time data flows on-demand.
- Real-Time Chains:
  - are characterized by a priority,
  - are compatible with IEEE 802.15.4 (after a minor modification to the standard).
  - do not require synchronized clocks or regular network structure.
  - support slow mobility (lifetime of existing routes is of the order of seconds);

5

RT Wireless: assumptions for the single hop case

- Most existing wireless protocols make the underlying assumption that the network traffic is intrinsically random

➔this assumption usually does not hold in real-time networks!

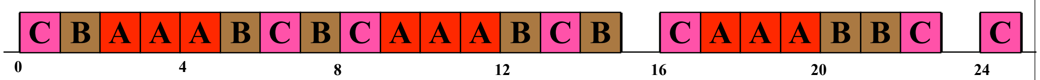
➔e.g.: nodes do not randomly connect with or download files (ftp) from remote nodes. *Traffic is rather predictable*

- **Assumptions (single hop scenario)**:
  - **clocks are not synchronized**
  - **nodes are fully linked (proven to be conflict free! ➔ low probability of collisions & graceful degradation if network is not fully-linked**
  - **if a node fails, it cannot use its transceiver**
  - **an attacker can jam the medium but cannot alter packet's content**
- **Requirements**: robustness against node failures, packet losses, and transient jamming

6

Robust Implicit EDF

- RI-EDF is a robust asynchronous real-time Medium Access protocol



- nodes use <u>*implicit contention*</u>:
  - ➔ periodic nature of sensor data streams, once initialized, allows for EDF scheduling via implicit contention
  - ➔ no contention phase, <u>no conflicts</u>, no backoff
  - ➔ it provides <u>resource reclaiming</u> of spare bandwidth (budget exchange)
  - ➔ <u>no single point of failure</u>: no matter what node fails, communication is always recovered by the other nodes

7

Implicit prioritized medium access using RI-EDF

- Distributed scheduling among fully linked nodes
- EDF scheduler is replicated at each node

Static priority ↑

node	*message length*	*period*
A	3	8
B	1	6
	1	8
C	1	4

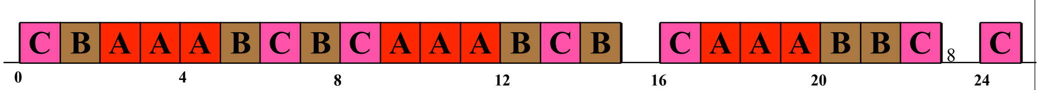
This table is stored in each node

max packet length (blocking time) → θ

$$\forall j, 1 \le j \le n \quad \sum_{k=1}^{j} \frac{Msg_k}{T_k} + \frac{\theta}{T_j} \le 1$$

⇩

Message set is schedulable!



8

Implicit prioritized medium access using RI-EDF

- What's new?

Two level packet scheduling:
- EDF packet schedule (normal operation) — Variable packet length, power awareness, no clock synchronization
- Static priority scheduling (recovery) — It recovers the EDF schedule in spite of packet loss, node failures

Static priority ↑

node	*message length*	*period*
A	3	8
B	1	6
	1	8
C	1	4

This table is stored in each node

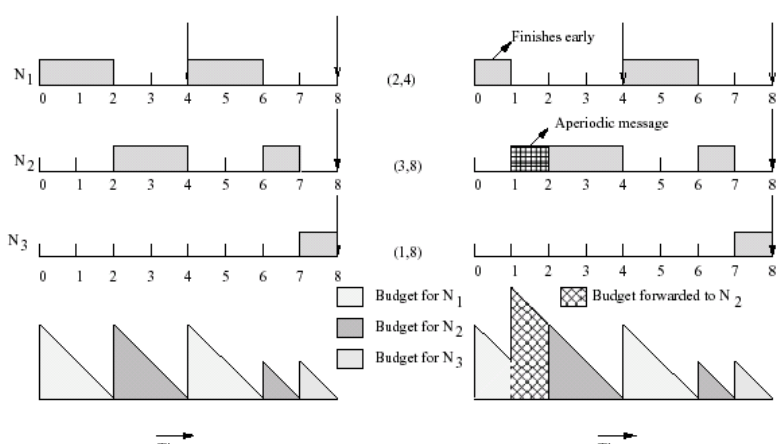
EDF schedule

C B A A A B C B C A A A B C B | C A A A B B C 9 | C

0 4 8 12 16 20 24

Bandwidth reservation for QoS provisioning

- Due to lack of clock synchronization, the transmission time is budgetted to avoid bandwidth stealing/starvation. Variable packets' size and early completions are handled by means of "**forwarded budget**"



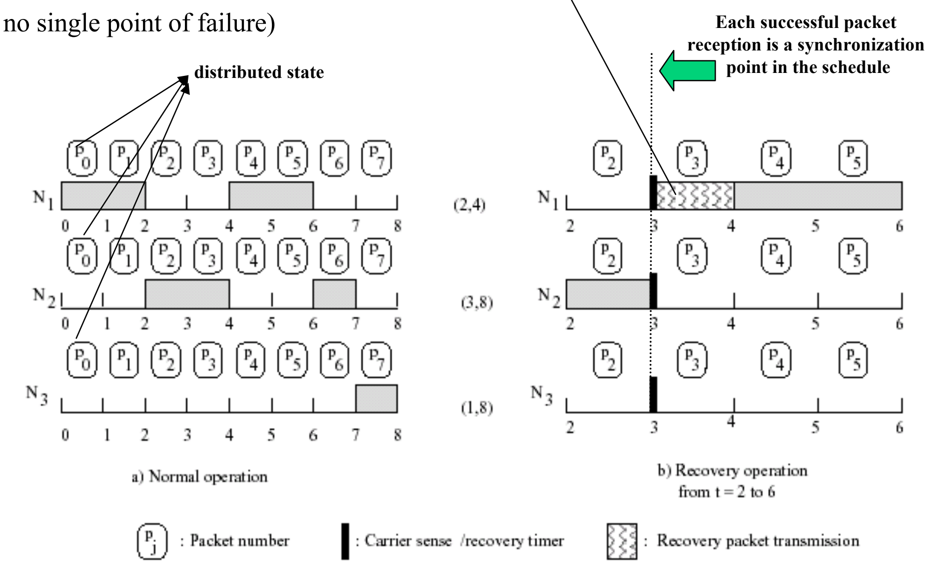
a) Normal operation

b) N1 finishes early and forwards its budget to N2

10

Schedule Recovery

- No matter what node fails, each alive node has the capability to recover the schedule (there is no single point of failure)



Power awareness

- Drawback of basic RI-EDF: channel is always fully utilized!
- The following set of rules extends RI-EDF to be power aware:
  - Nodes divided among sources and sinks (sinks collect & process source data). An example of source is a sensor node.
  - Packet header has information whether the transmitter is a source or sink.
  - Sinks have higher static priority compared to sources
  - Sinks send periodically a beacon to solicit data from sources
  - If a source node N does recovery, its recovery mechanism is disabled, it can still send *P* data packets before transmission is completely disabled. Recovery and normal transmission are re-enabled upon reception of a sink's packet.
  - Sink nodes don't disable their recovery mechanism unless they sleep

Power awareness

- Nodes can arbitrarily go to sleep and periodically wake up. The recovery mechanism efficiently fills up the gaps in the EDF packet schedule
- The described set of rules provides the following features:
  - Sink nodes can disable source nodes if they are not interested in actual transmitted data.
  - If a source node is out of the range of all sinks, its transmission capability will temporarily be disabled.
  - Sinks can suspend their normal communication (sleeping mode), transmitting only a beacon periodically.
  - Even if the real-time packet schedule is suspended, the periodic beacon can reactivate it on demand.

13

Experiments with Berkeley Motes

- Comparison of current consumption

  - 5 sources generated random real-time traffic
  - Throughput ranging from 200 to 1000 bytes/sec
  - Each source disabled its radio between transmission requests

TinyOS MAC

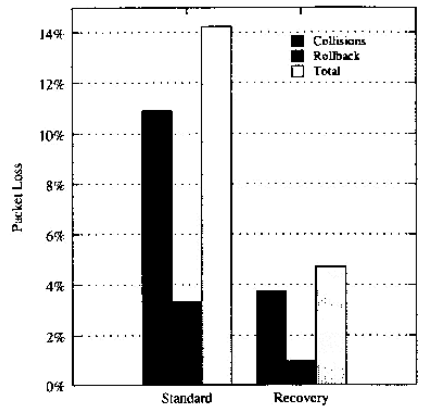
Target Bps	Actual Bps	% Collisions	% Missed Deadlines	% Time in Trans.	% Time in Recv.	Current (mA)
200	199.0	0.33	0	3.66	7.52	1.338
400	392.9	0.81	0	7.09	19.22	3.037
600	597.0	1.80	0	10.67	36.89	5.334
800	780.9	3.62	29.28	13.60	70.12	9.017
1000	820.9	4.32	69.60	14.22	81.79	10.240

RI-EDF

Target Bps	Actual Bps	% Collisions	% Missed Deadlines	% Time in Trans.	% Time in Recv.	Current (mA)
200	204.9	0	0	3.77	11.60	1.746
400	404.8	0	0	7.42	24.46	3.595
600	602.1	0	0	11.03	39.82	5.675
800	802.1	0	0	14.68	55.65	7.809
1000	1001.3	0	0.25	18.31	72.25	10.012

Relaxing RI-EDF assumptions & hidden node problem

- In adverse environment, RI-EDF faces the hidden node problem
- The medium is not anymore conflict free, but RI-EDF exploits the notion of **Recovery Group** to achieve robustness and low probability of conflicts



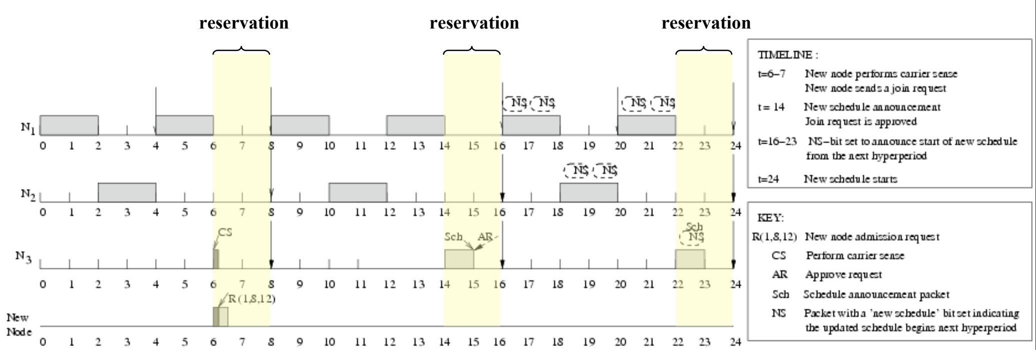
15

Recovery Groups Comparison

Dynamic schedule update

- It uses a packet train reservation at end of hyperperiod
- Prioritization of medium access during schedule update handshake:
  - Schedule announcement after join request (highest static priority for leader)
  - Schedule update requests and Join requests (new priority level)
  - Unrequested schedule announcement (driven by standard recovery priorities)

Static priority

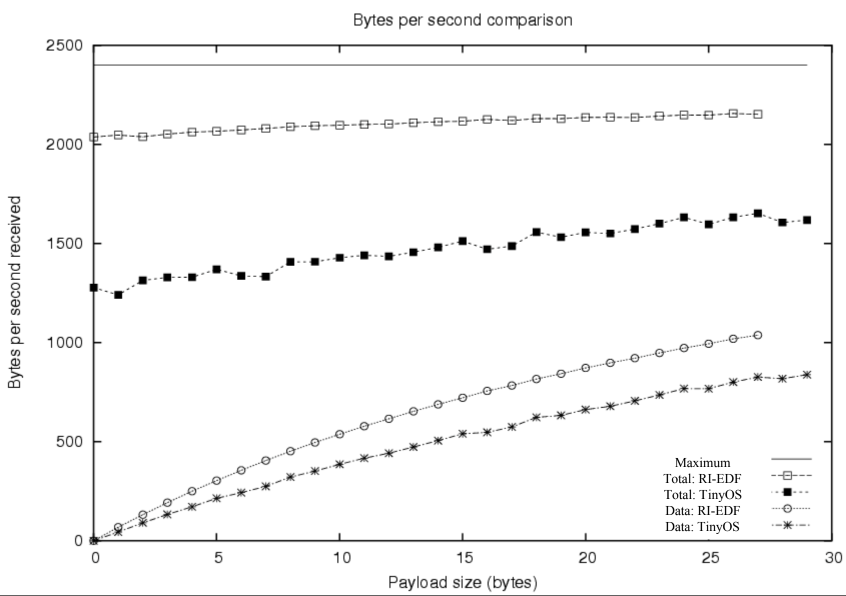


Experiments with Berkeley Motes

- Berkeley Mica2 Motes:7 sources / 1 sink
- Network throughput and packet jitters are evaluated comparing RI-EDF with original MAC protocol of TinyOS version 1.1.0
- Packet jitters are evaluated by timestamping packets in the MAC layer of the receiver.
- Packet overhead: RI-EDF uses 5 bytes for sender ID, schedule, and budget
  ➔ max. available payload 28 bytes. Overhead is two extra bytes

Experiments with Berkeley Motes

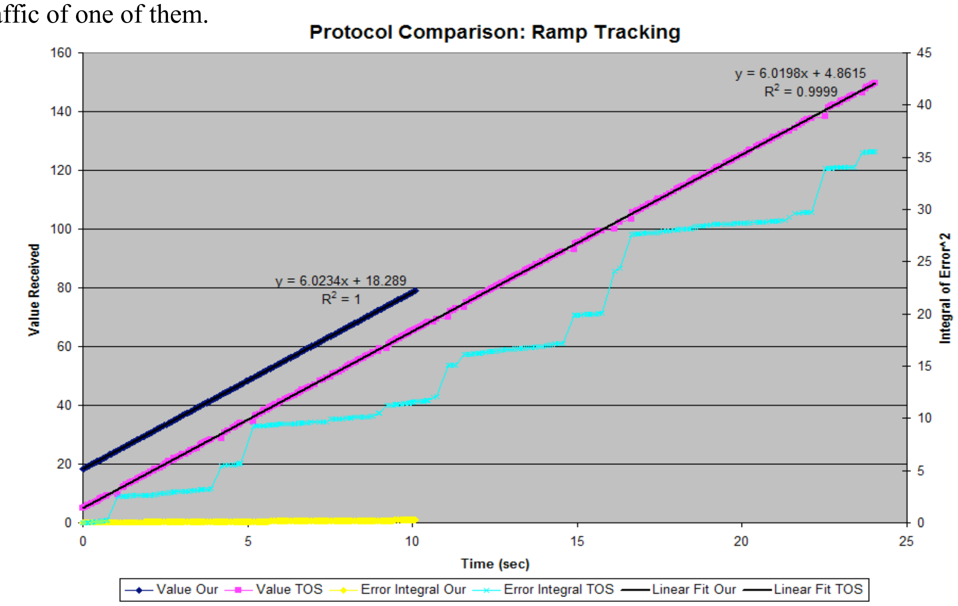
- throughput analysis ➔ RI-EDF allows 26% increase in Tx data



8

Experiments with Berkeley Motes

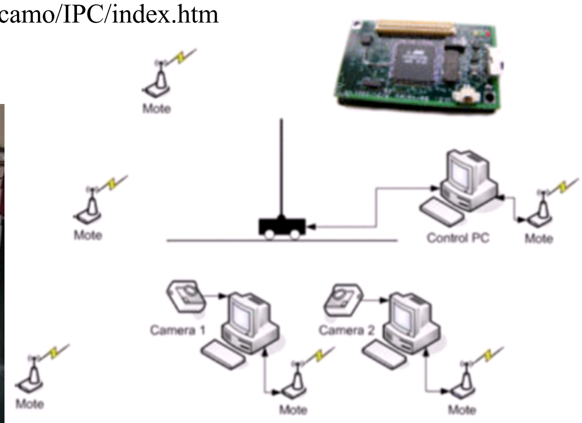
- Jitter analysis: 3 Motes are sending traffic at 30Hz, 30Hz, and 10Hz. We monitored the traffic of one of them.



Testbed for wireless distributed control

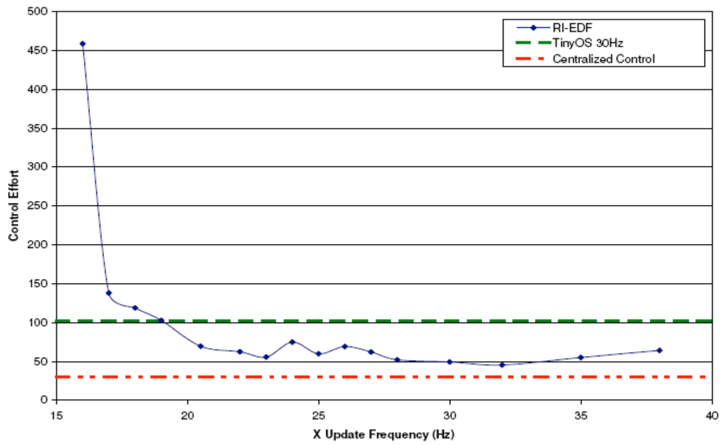
- Inverted pendulum uses a remote camera to track the cart position.
- Images are processed locally and cart position is transmitted by wireless
- Additional real-time flows can be guaranteed by means of RI-EDF
- Motes are used either as standalone units or as PC transceiver

See http://pertsserver.cs.uiuc.edu/~mcaccamo/IPC/index.htm



Experiments with Berkeley Motes

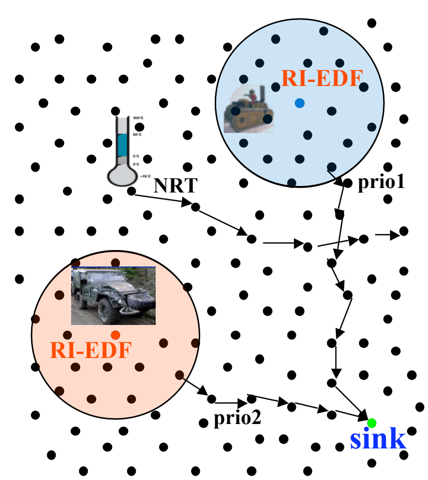
- Quadratic Error Index: $QEI = \frac{1}{T}\int_0^T w_1\delta x^2(t) + w_2\delta\omega^2(t) + w_3\delta V^2(t)dt$
- TinyOS MAC protocol could only provide stable control at 30Hz. The default backoff period had to be reduced to provide stable control since the default value could not keep the inverted pendulum balanced. RI-EDF provides stable control at a lower frequency compared to TinyOS.



21

The multi-hop scenario: Real-Time Chains

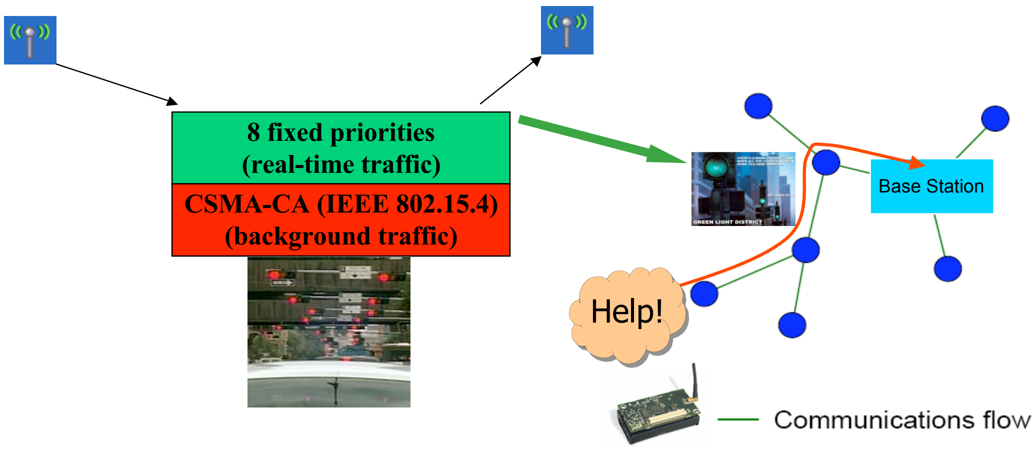
- Multi-hop scenario for RT-WSNs
- Real-time chains are prioritized real-time data flows (subject to soft real-time guarantee) that can coexist with non-realtime (CSMA/CA like) traffic.
- Assumptions:
  - network structure is NOT regular,
  - nodes are statically located or slow moving (lifetime of existing routes is of the order of seconds);
  - do not require synchronized clocks



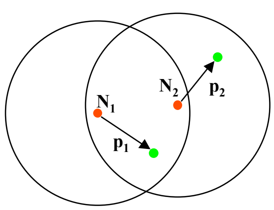
22

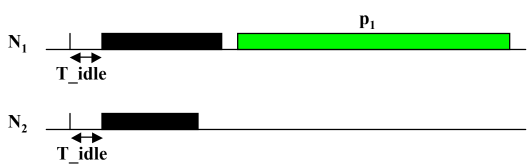
The multi-hop scenario: Real-Time Chains

- Each soft real-time flow is characterized by a priority
- Real-time packets as a group are given precedence over best effort traffic



Background: Black-Burst contention scheme



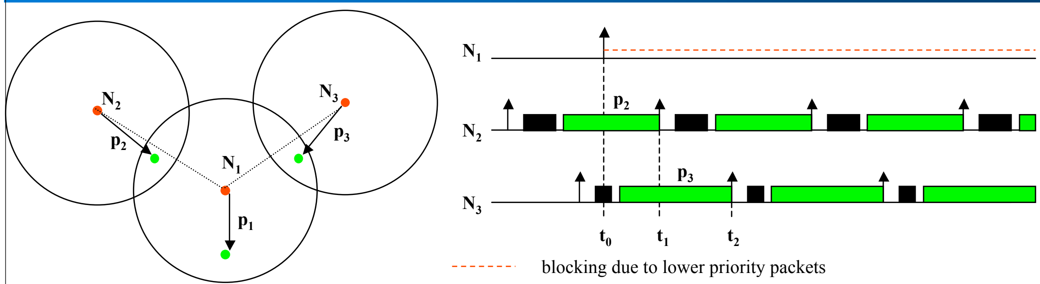


- Senders synchronize on transition from busy to idle channel.
- Each sender transmits a BB with length proportional to its packet priority.
- The longest BB wins the contention.
- Transmissions using different priorities are collision-free in the absence of hidden nodes. Hidden nodes are avoided when sensing range (R_S), interference range (R_I) and communication range (R_C) satisfy the following condition: $R_C + R_I \leq R_S$

J. L. Sobrinho and A. S. Krishna kumar. Quality-of-Service in ad hoc carrier sense multiple access networks. *IEEE Journal on Selected Areas in Communications*, 17(8):1353–1368, August 1999.

29

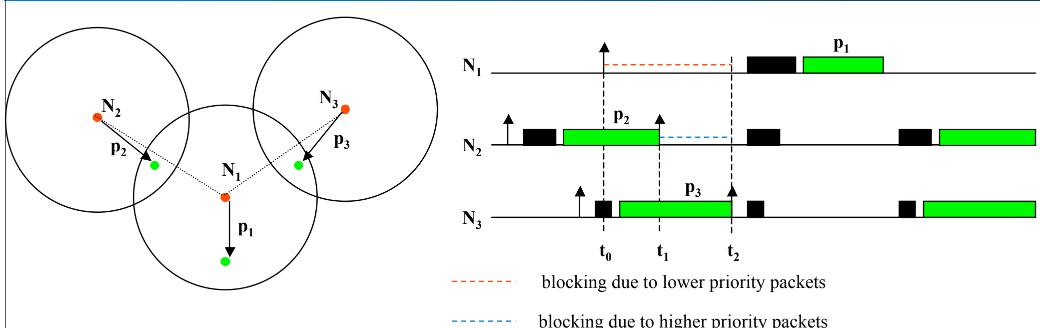
MAC & Real-Time: problem of unbounded priority inversion



- When p_2, p_3 end (at t_1,t_2, etc.), the channel is never idle for N_1
  ➔ unbounded priority inversion
- Assumptions:
  - work conserving policy that waits for idle channel before transmitting
  - local knowledge about traffic, no future knowledge about packet arrivals.

25

MAC & Real-Time: problem of unbounded priority inversion



- N_2 should be blocked at t_1 to free the channel for N_1.
- Requirement for a real-time MAC: N_1 should force a new synchronization point within bounded time.
- *Real-time chains approximate this behavior.*

26

Real-time chains for multi-hop ad-hoc wireless

	Indoor	Outdoor
Communication Range (m)	3	10
Interference Range (m)	18	60
Sensing Range (m)	22	70

Ranges for MICAz motes

- **Design choices driven by experimental data:**
  a) Since the condition $R_C+R_I \leq R_S$ holds, the hidden node problem wasn't a main concern

  ➔ A MAC protocol like Black-Burst eliminates packet collisions under the assumption that two different nodes do not try to contend for the channel with the same priority at the same time.

  ➔ During our experimental testing of intersecting real-time chains, we did not experience collisions due to the hidden node problem.

  ➔ Remark: Even when $R_C+R_I > R_S$, the number of potential hidden nodes is expected to be limited and only due to intersecting chains contending on the same channel (more experiments are needed!).

27

Real-time chains for multi-hop ad-hoc wireless

	Indoor	Outdoor
Communication Range (m)	3	10
Interference Range (m)	18	60
Sensing Range (m)	22	70

Ranges for MICAz motes

- **Design choices driven by experimental data:**

  b) Both sensing and interference ranges were much larger than what commonly assumed,

  ➔ Sensing range was about 7 times larger than the reliable communication range. (we tested low power, 802.15.4 compliant transceiver of MICAz)

  ➔ channel reuse was extremely low and flow bandwidth could not be higher than B/8 unless multiple channels were used (no more than one intermediate node every 8 could simultaneously transmit)

28

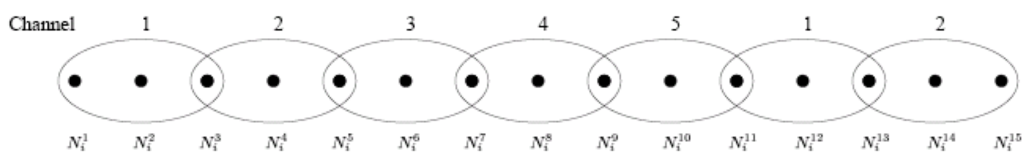
Real-time chains for multi-hop ad-hoc wireless

- **What is a real-time chain?**

  a) it uses a set of channels reserved for real-time traffic: [1,..,C]

  b) Each chain has a real-time priority (we have an implementation with 4 distinct priorities)

  c) All best effort traffic shares channel 0; hence, best effort traffic cannot interfere with real-time traffic.



Real-time chains for multi-hop ad-hoc wireless

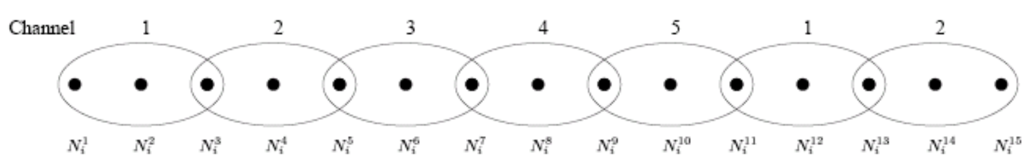
- **What is a real-time chain?**

  d) The chain opening request is transmitted on channel 0 using the BB scheme: it has higher priority over best effort traffic (CSMA/CA of 802.15.4) and it uses geographic forwarding as routing protocol.

  e) Nodes used by a chain are not available for other real-time/non real-time communication until the chain is closed.

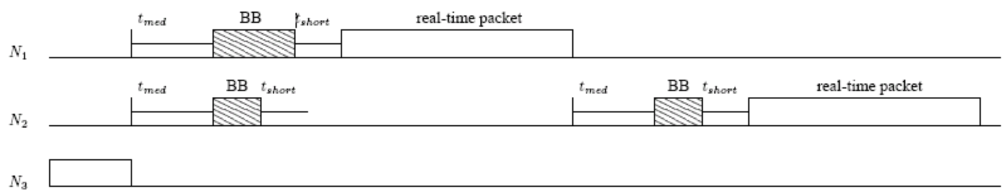
  f) it allows good spatial reuse of the wireless medium by exploiting multiple channels

  g) MICAz transceiver can support three groups of chain channels [1-5], [6-10], [11-15] → at most 12 different chains can co-exist within the same region without conflicting.



Real-time chains for multi-hop ad-hoc wireless

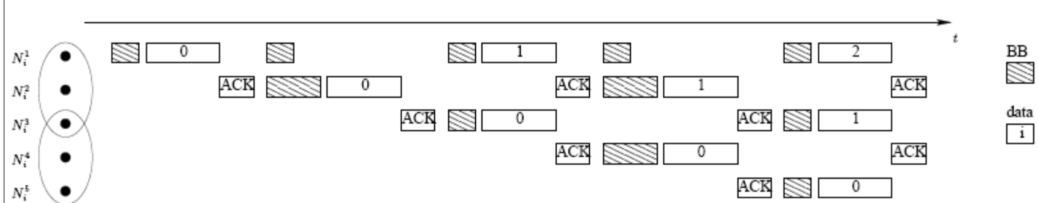
- **Real-time chain is compatible with IEEE 802.15.4 (after a minor modification to the IEEE standard)**



- Each node with best effort traffic must sense idle channel for a long interframe spacing $t_{long} > t_{med}$ before having the right of transmitting.
- This modification ensures that real-time packets as a group are given precedence over best effort traffic.

Real-time chains for multi-hop ad-hoc wireless

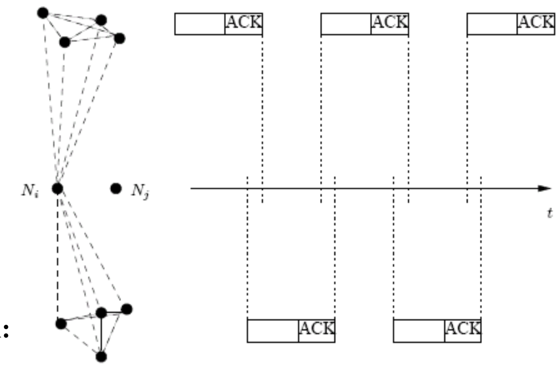
- **Example of a single real-time chain**



- Whenever the buffer is empty, the node listens on its reception channel and immediately acknowledges any packet sent to it. If the id of the received packet is greater than the counter, its value is updated and the packet is copied in the buffer.
- Upon copying a packet in the buffer, the node switches to its transmission channel and uses the BB contention scheme to transmit the packet. While the buffer is full, the node does not acknowledge any packet sent to it.
- If the node receives an ACK after winning a BB contention and sending the packet, it removes it from the buffer and switches back to listening. Otherwise, it contends again on the transmission channel until it correctly receives an ACK.

Real-time chains for multi-hop ad-hoc wireless

- **Starvation problem during chain opening**

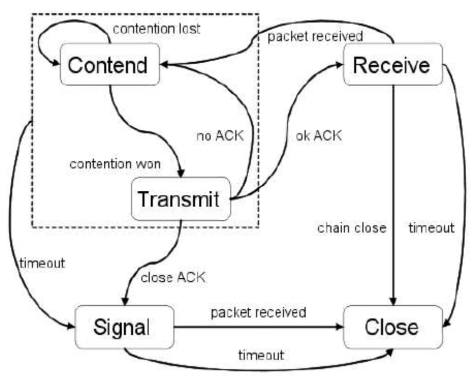


Heuristic to mitigate starvation problem:

- If N_i perceives the channel busy for more than t_{max}, it jams channel 0 with a high-power jamming signal. The signal lasts for t_{max} and the power is high enough that it is reliably perceived by all nodes that N_i is able to sense.
- After the jamming signal, all communication within the neighborhood of N_i should have stopped. In the absence of higher priority packets, N_i wins the channel contention and transmits the packet.

Real-time chains for multi-hop ad-hoc wireless

- A timeout mechanism together with Signal and Close states allow to detect and quickly recover a failed chain (e.g. conflicting priorities, disconnected route, etc.)
- A node transitions to Signal state upon receiving a close ACK from next node; it then tries to send a close ACK to previous node and switches to ch0.
- Upon failure, the chain is automatically closed by signaling to the source or by timeout; then, a new opening phase re-establishes the chain through a new route or by using different set of channels



Soft real-time guarantee

p_i	1	2	3	4
Measured per-hop delay (ms)	13.8	15.6	17.8	19.6
$2t^i_{pack} + t^i_{over}$ (ms)	12.4	14.7	17.1	19.2

Per-hop delay for a single chain

- If priority and route of each existing flow is known, we can easily compute the flows' throughput by building a set of linear constraints in $\rho_1, \ldots, \rho_m$ as follows
  - For each active flow:

$$\rho_k \leq r_k$$

  - For each set of flows belonging to an interference point I:

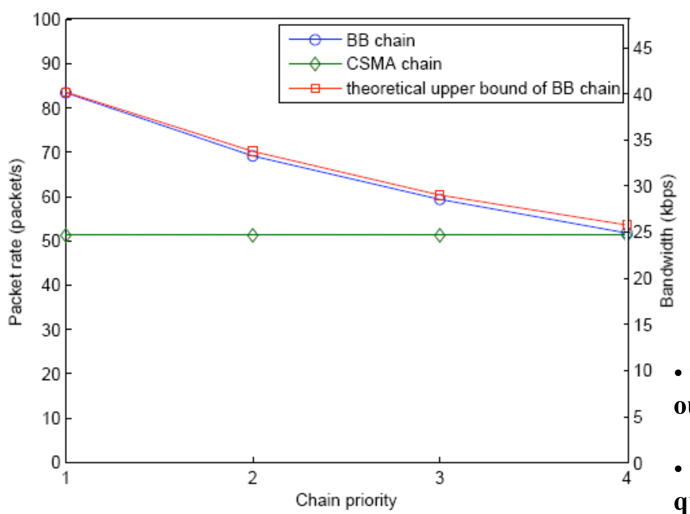
$$\rho_k \leq \left(1 - \sum_{l \in I \wedge l > k} \rho_l / \rho_l^{\max}\right) \cdot \rho_k^{\max}$$

  - Solve the system by individually maximizing each flow rate starting from the highest priority ρ_m to the lowest priority ρ_1 subject to all constraints.

35

Real-time chains for multi-hop ad-hoc wireless

- **Some experimental results:**



Bandwidth for a single chain

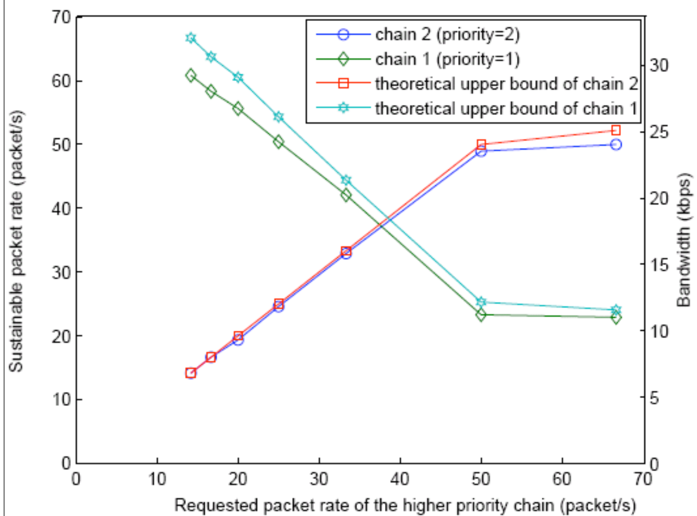
• we tested upto 20 hops outdoor

• we were able to carry low quality audio sampled at about 4KHz

36

Real-time chains for multi-hop ad-hoc wireless

- **Some experimental results:**



Bandwidth of two crossing chains

37

Conclusions

- RT-WSNs will support audio/video streaming and enhance existing sensor network applications such as surveillance and environmental monitoring.
- RI-EDF (single-hop) allows for high throughput, soft real-time guarantee and power awareness in spite of node failures and in absence of clock synchronization
- Real-time chains are prioritized real-time data flows (subject to soft real-time guarantee) that can coexist with non-realtime (CSMA/CA like) traffic.

Future work:

- Apply this research to other classes of devices (IEEE 802.11a compatible?)
- Define a notion of real-time capacity for the multi-hop case

38