SmarTiSCH: An Interference-Aware Engine for IEEE 802.15.4e-based Networks

Zihao Yu†, Xin Na†, Carlo Alberto Boano‡, Yuan He†, Xiuzhen Guo†, Pengyu Li†, and Meng Jin §

†School of Software & BNRist, Tsinghua University, China
‡Institute of Technical Informatics, Graz University of Technology, Austria
§ Shanghai Jiao Tong University, Shanghai, China
Industrial IoT: Mission-critical applications

Stringent requirement on reliability
IEEE 802.15.4e

Mission-critical IoT applications

IEEE 802.15.4e standard: TSCH (time-slotted channel hopping)

Upper layers
6LoWPAN
6TiSCH (6top)
IEEE 802.15.4e (TSCH)
802.15.4 PHY

wireless medium

asynchronous vs. synchronous

single-channel vs. channel hopping

frequency vs. time
TSCH Protocol

Slotframe: 2D table schedule

Scheduler: how to allocate the cells with minimum energy expenditure?
Problem: interference

- **Internal interference**: internal collisions between links
- **External interference**: impact from external devices like WiFi devices
Previous works: centralized/distributed scheduler

Both the schedulers rely on extra control traffic to detect and handle interference!
Recent works: autonomous scheduler

Autonomous scheduler

RPL neighbor information $\rightarrow$ Schedule

Vulnerable to interference due to unawareness!
Dilemma: overhead or interference?

Awareness of interference

Centralized/distributed scheduler

With extra traffic

No control traffic

Autonomous scheduler

Vulnerable to interference

Both?
SmarTiSCH: an interference-aware engine

Network node (Transmitter) | Network node (Receiver)

Transmission result | RSS strength | Duplicate packet

Passive observation

Reaction

Reaction Strategy

Internal interference

External interference

Inference of interference

Reaction Strategy
Challenges

Challenge 1 – To **obtain the awareness** of interference based on data transmission

- How to infer the type of interference from **limited information**?
- How to solve the **asymmetry** between the observation of the transmitter and the receiver?
- How to handle the **ambiguity** of the inference process?

Challenge 2 – To **share and utilize the awareness** under interference without extra control traffic

- How to reliably share the information **under interference**?
- How to build consensus between the transmitter and the receiver **without control packets**?
- How to **react** to interference by updating the schedule?
SmarTiSCH: passive observation

**Presence of interference**
- Noise floor
- No interference
- Packet from another link
- Internal interference
- Packets from external devices
- External interference

**Receiver status**
- Packet & ACK
- Overheard packet
- Packet (corrupted)
- Invalid signal

**Infer**

**Lead to**
### Asymmetry between the observations of the transmitter and the receiver

<table>
<thead>
<tr>
<th>Receiver status</th>
<th>Transmitter status</th>
<th>No traffic</th>
<th>CCA busy</th>
<th>ACK received</th>
<th>No ACK</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Clear channel</strong></td>
<td>No traffic</td>
<td>No internal</td>
<td>No external</td>
<td>No ACK</td>
<td></td>
</tr>
<tr>
<td><strong>Overheard packets</strong></td>
<td>CCA busy</td>
<td>Internal</td>
<td>No ACK</td>
<td>No ACK</td>
<td></td>
</tr>
<tr>
<td><strong>ACK sent</strong></td>
<td>ACK received</td>
<td>No internal</td>
<td>No ACK</td>
<td>No ACK</td>
<td></td>
</tr>
<tr>
<td><strong>Invalid signals</strong></td>
<td>No ACK</td>
<td>No internal</td>
<td>No ACK</td>
<td>No ACK</td>
<td></td>
</tr>
</tbody>
</table>

A receiver-dominant design principle to infer the presence of interference.

Ambiguity?
SmarTiSCH: inference of interference

Receiver status:
- Packet received & ACK sent
- Clear channel

Transmitter status:
- ACK received
- Next cell

Transmitter status:
- No ACK
- Next cell

External interference:
- Duplicate packet

The receiver utilizes extra information (duplicate packet) to solve the ambiguity.
Challenges

Challenge 1 – To obtain the awareness of interference based on data transmission

• How to infer the type of interference from limited information?
• How to solve the asymmetry between the observation of the transmitter and the receiver?
• How to handle the ambiguity of the inference process?

Challenge 2 – To share and utilize the awareness under interference without extra control traffic

• How to reliably share the information under interference?
• How to build consensus between the transmitter and the receiver without control packets?
• How to react to interference by updating the schedule?
SmarTiSCH: reaction to interference

The transmitter and the receiver enter the control channel for reliable information exchange.

Reaction strategy:
- Cell in data channel
  - CCA busy
  - No ACK
- ACK received
- Cell in control channel
- Data retransmission
- Cell in data channel
- ACK sent
- Clear channel
- Data retransmission
- Cell in data channel
- Invalid signal
- Overheard packet
- Cell in control channel
- Data retransmission
- Cell in data channel

The transmitter and the receiver enter the control channel for reliable information exchange.
Observation: data exchange provide clues!

Data exchange

Transmission result  RSS strength  Duplicate packet

Clues

Awareness of interference

Reaction to interference

Network node

Data exchange

DATA  ACK

Network node
SmarTiSCH: reaction to interference

How to transfer the information from receiver to the transmitter?

Information embedded in the time of ACK

No extra control packet

Strategy embedded in data exchange

No interference → Normal ACK

Internal interference → Early ACK

External interference → Late ACK

Cell in control channel:

- DATA
- ACK
SmarTiSCH: reaction to interference

How to react to interference in three different conditions?

- Normal ACK: No interference
- Early ACK: Internal interference
- Late ACK: External interference

Strategy to handle interference:

1. Normal ACK: No interference
2. Early ACK: Internal interference
3. Late ACK: External interference
SmarTiSCH: Implementation

- SmarTiSCH: link-based Orchestra with our engine
- Hardware: nRF52840 nodes
- Software: ContikiNG, and use RPL with storing mode on top
- Testbed: 20 nodes in a 50m² area
- External interference: JamLab-NG
- Application: a data collection application at the root note

nRF52840 node

Testbed
SmarTiSCH: Evaluation

Link loss in different channels

Packet distribution in different channels
Conclusion

• SmarTiSCH increases the awareness and robustness to interference without extra cost of scheduling.

• Design: passive observation, inference of interference, reaction

• Evaluation: SmarTiSCH expands the channel resources, increases the network capacity, and improves the network performance without extra control traffic.
Thank you