Competition:
Wireless-Transparent Sensing Platform

Chun-Hao Liao, Theerat Sakdejayont, Makoto Suzuki,
Yoshiaki Narusue, and Hiroyuki Morikawa (Team 07)

School of Engineering, The University of Tokyo

東京大学
THE UNIVERSITY OF TOKYO

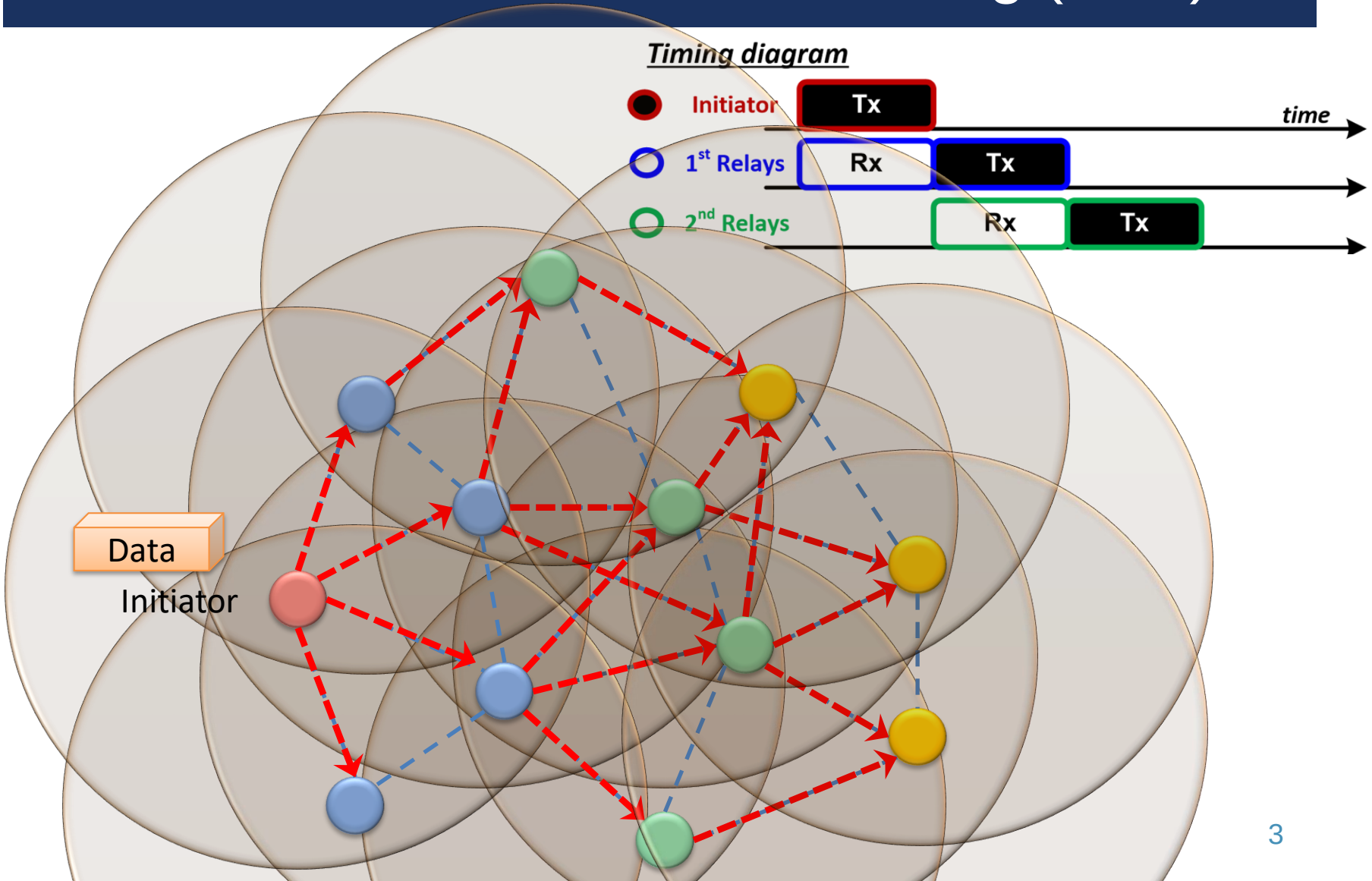
EWSN Dependability Competition 2018
15 Feb 2018

Wireless-Transparent Sensing Platform

- Full paper presented in EWSN 2017 [1]
- TDMA-based protocol adopting the concurrent transmission flooding (CTF)
  - The analysis in PHY layer [2] showed why IEEE 802.15.4 can survive synchronized packet collision.
- Flexible timeslot management
  - On-the-fly scheduling
  - Service-driven scheduling

[1] M. Suzuki, C.H. Liao, S. Ohara, K. Jinno, and H. Morikawa, “Wireless-transparent sensing,” in Proc. EWSN ’17.
[2] C.H. Liao, Y. Katsumata, M. Suzuki, and H. Morikawa, “Revisiting the so-called constructive interference in concurrent transmission,” in Proc. LCN ’16.

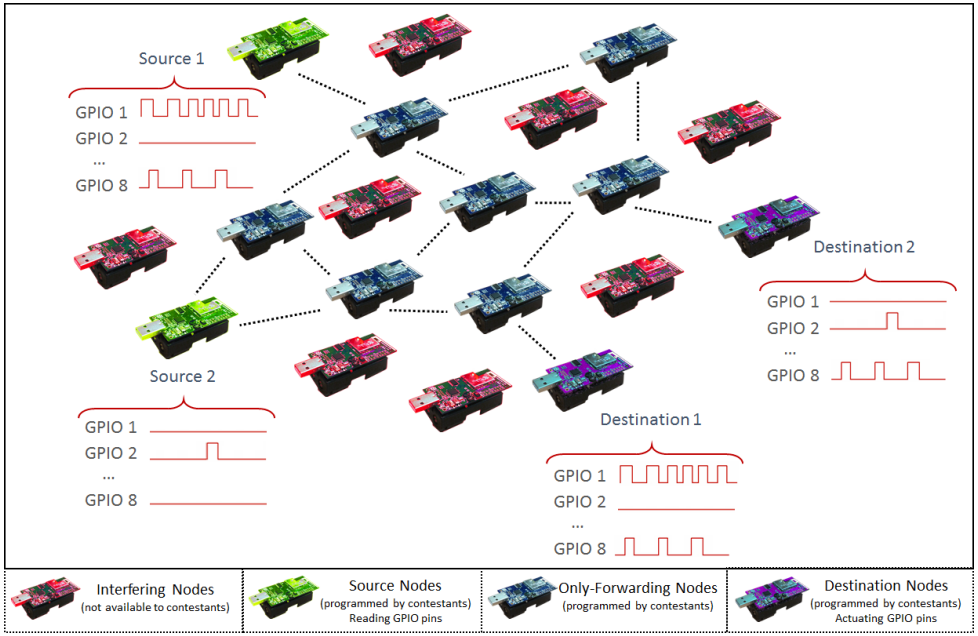
Concurrent Transmission Flooding (CTF)



Toward EWSN Competition 2018

Given multi-source multi-destination traffic pattern and harsh RF environments, several refinements have been made on WTSP including:

1) A dedicated service that realizes UDP-like one-way streaming dissemination
2) Progressive forwarding
3) Source redundancy



UDP-like One-way Streaming (1)

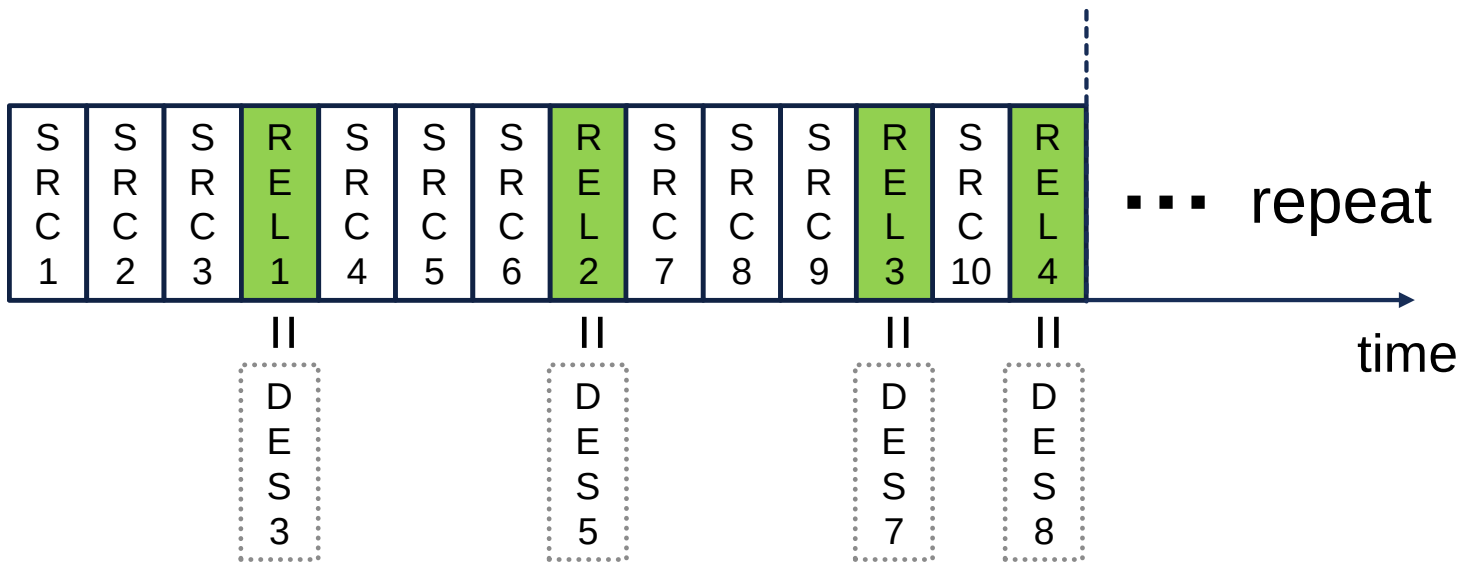
Challenge: Poor reliability even with end-to-end retransmission due to the harsh RF jamming

- Contention-based slot scheduling requires ACK to ensure end-to-end reliability.
- In dense traffic and harsh RF environments, this two-way communication may lead to severe network congestion.

One-way communication, Fixed scheduling

UDP-like One-way Streaming (2)

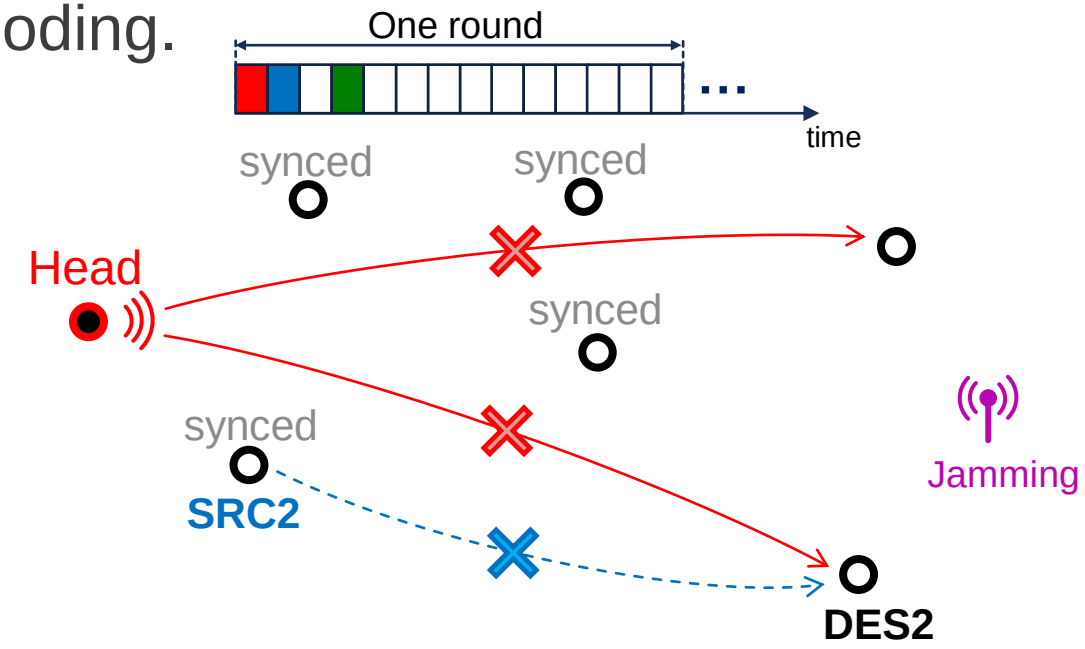
Improvement: Simple but efficient one-way dissemination without ACK



Progressive Forwarding (1)

Challenge: "**Volatile relaying**"

- A packet from an initiator hardly gets through the harsh RF jamming in one flooding.



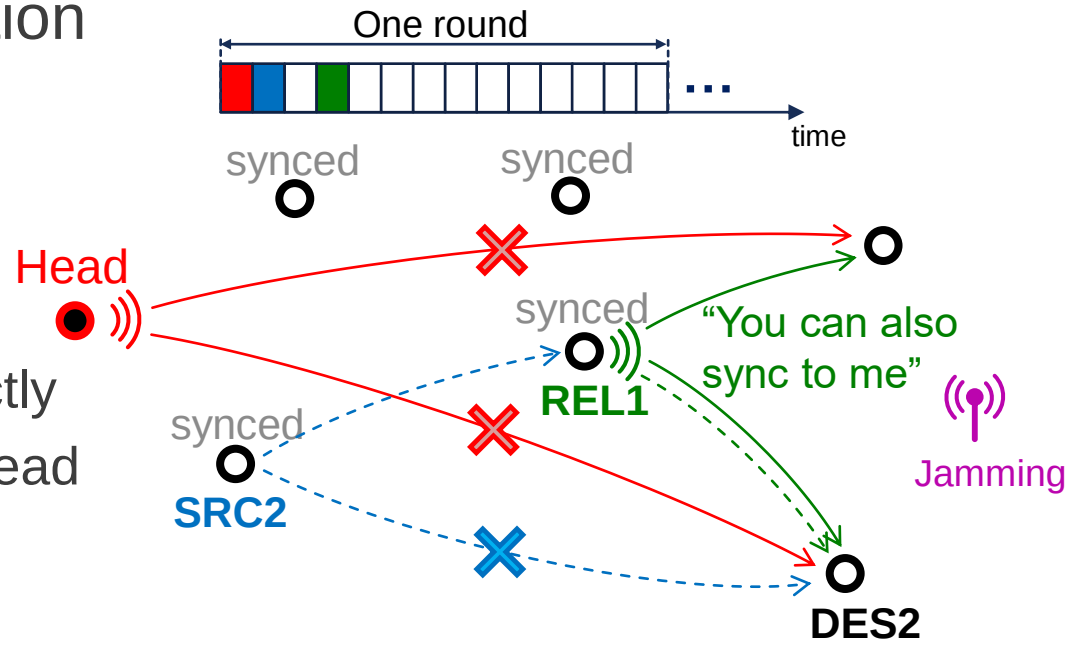
- Poor synchronization
- Poor end-to-end connectivity

Assisting nodes

Progressive Forwarding (2)

Improvement: Slot-owning nodes help synchronization and relaying information in progressive manner.

- Progressive synchronization
  1. "You can sync to me" flag set by a slot-owning node
     → *Quick sync.*
  2. Re-bootstrapping if not directly received from the network head for too long
     → *Unity of the network*



- Progressive forwarding nodes

Source Redundancy (1)

Challenge: Inefficiency of one-to-one packet format

- ***Large*** packet overhead, but ***tiny*** GPIO information

For example,

Field	Size
LEN	1 byte
Service header	1 byte
Slot No.	1 byte
Initiator node ID	1 byte
Relay count	1 byte
CRC	2 bytes

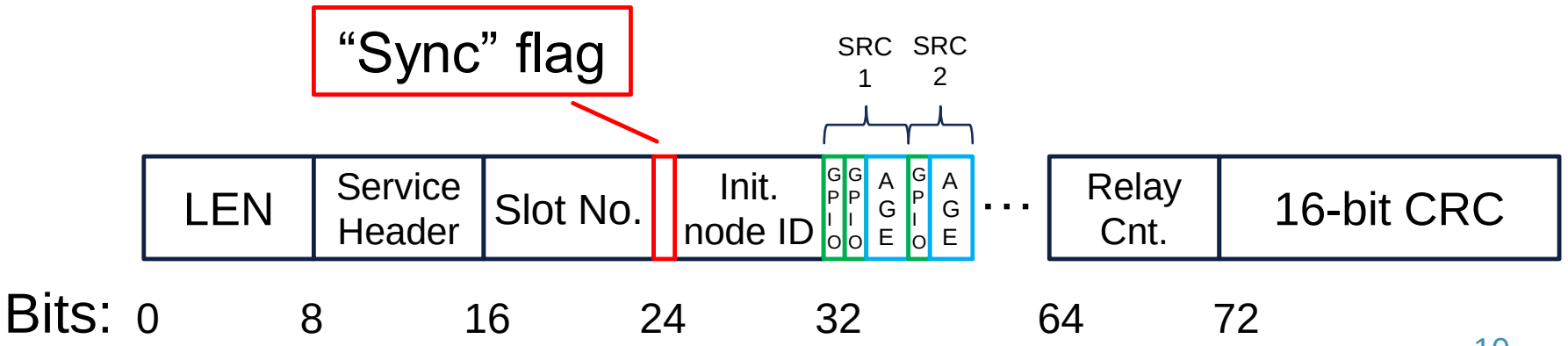
VS

1-bit GPIO

Source Redundancy (2)

Improvement: Transmitting information from multiple sources in a packet

- Highly-efficient packet format:
  1-bit for each GPIO and 2-bit age information for each source



Conclusion

- UDP-like one-way streaming as a dedicated service of WTSP
  - Pre-assigned slot scheduling
- Progressive forwarding
  - Quick synchronization, but still maintaining the unity
  - Arbitrary nodes can be assigned to be forwarding nodes.
- Source Redundancy
  - Efficient packet format
    1-bit for each GPIO, 2-bit age information for each source
- Others: channel hopping

Thank you.