

Poster Abstract: Capture Effect Based Communication Primitives

Closing the Loop in Wireless Cyber-Physical Systems

Olaf Landsiedel¹, Federico Ferrari², Marco Zimmerling²
olaf@chalmers.se, ferrari@tik.ee.ethz.ch, zimmerling@tik.ee.ethz.ch

¹Computer Science and Engineering, Chalmers University of Technology, Sweden

²Computer Engineering and Networks Laboratory, ETH Zurich, Switzerland

1 Motivation and Principle

Wireless control systems consist of sensing and actuating devices that are commonly driven by a central controller. Wireless communication protocols for Cyber-Physical Systems (CPS) match this design by employing a "sense → collect → process → disseminate → actuate" flow [6], where typically different protocols are employed for collecting sensor data and disseminating actuation signals.

In this paper, we depart from this traditional design and introduce CaptureCom. By relying on capture effects and in-network processing, it omits the need for a central controller and for distinct collection and dissemination phases. In CaptureCom, each node transmits its current data (*e.g.*, temperature reading). Upon receiving, nodes integrate (*e.g.*, aggregate) the received data with previously received data and concurrently forward the result. Due to capture effects, neighboring nodes correctly receive one of the concurrently sent packets with high probability. Repeating this process, the network converges to a stable state where all nodes have received the same data (consensus). The impact of our approach is threefold:

1. CaptureCom closes the loop in CPS: data are processed within the network. Upon completion, all nodes have received with high probability the same data as the base for actuation. Thus, it departs from the widespread architecture of collecting information at a central controller for processing and then disseminating the results.
2. CaptureCom exploits spatial diversity in low-power wireless networks: Consecutive, concurrent transmissions spread out across the network allow for data distribution at very low delays and high energy efficiency.
3. Relying solely on concurrent forwarding and capture effects for communication, CaptureCom simplifies the networking stack by obviating the need for link estimation, neighbor discovery, and routing protocols.

2 CaptureCom: Background and Design

Background and related work. CaptureCom relies on the capture effect, which is a well known phenomenon in wireless communication [4]: If two or more packets are transmitted concurrently, the stronger one is received, assuming it is

sufficiently stronger than the contending signals; that is, the signal-to-noise ratio (SNR) is higher than a certain threshold. For instance, the capture effect has been used for fast flooding in sensor networks: nodes concurrently forward received packets to quickly propagate them in the network [5].

Different from CaptureCom, Glossy [3] and LWB [2] rely also on constructively interfering baseband signals. While this allows packets to be received independently of the SNR threshold, it requires that all simultaneously transmitted packets are the same. Thus, Glossy and LWB cannot exploit spatial diversity and in-network aggregation as CaptureCom.

Basic design and sample execution. To introduce the basic design of CaptureCom and demonstrate its benefits, we use a sample application that aims at determining the maximum sensor reading among all nodes in the network (*e.g.*, occupancy or temperature) and disseminating this maximum to all nodes. We note that aggregation is optional in CaptureCom and that its design is independent of the particular aggregation scheme. In the end, all nodes are aware of the maximum temperature reading in the network and can trigger actuation. To illustrate CaptureCom, we assume a network of three nodes (see Fig. 1(a)). Initially, each node is only aware of its own sensor reading (see Fig. 1(b)). A coordination node, node *A* in our example, triggers the communication round by broadcasting its local temperature reading.

The payload of packets consists of two parts: a *bit field*, indicating from which node data have already been integrated, and a *payload field* (see Fig. 1(b)). In this example, the payload is the maximum temperature reading a node has received so far, including its own sensor reading. Upon receiving, a node merges its local bit field with the received one via a bitwise OR and determines the maximum of the local and the received data (see Fig. 1(c)). After a constant delay, which is sufficient for both processing and I/O operations, receiving nodes transmit the resulting data. These nodes transmit concurrently, enabling neighboring nodes to receive the stronger of multiple, concurrently transmitted packets due to the capture effect. In this way, CaptureCom triggers a chain reaction, where each relay increases the number of participating nodes, eventually covering the complete network with very high probability. To avoid starvation, a node transmits when the local and received bitfields differ, that is, the node learned something new (see Fig. 1(c)).

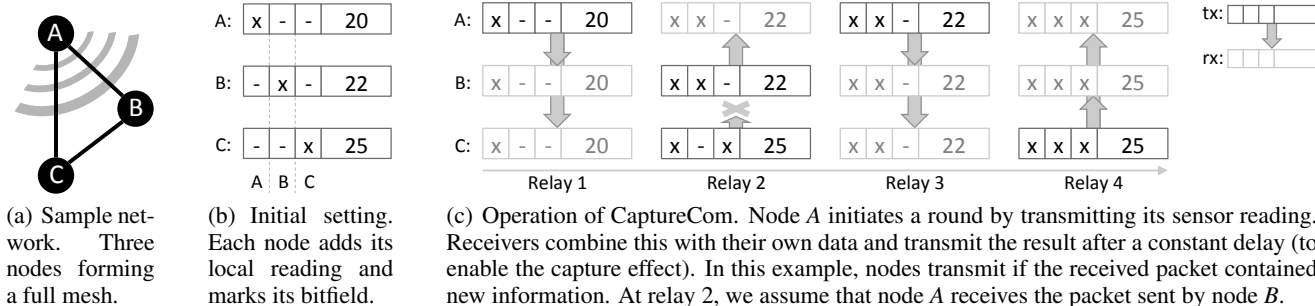


Figure 1. Basic idea of CaptureCom. By exploiting the capture effect, CaptureCom performs data collection, aggregation, and dissemination within the network. As a result, it closes the control loop in wireless cyber-physical systems.

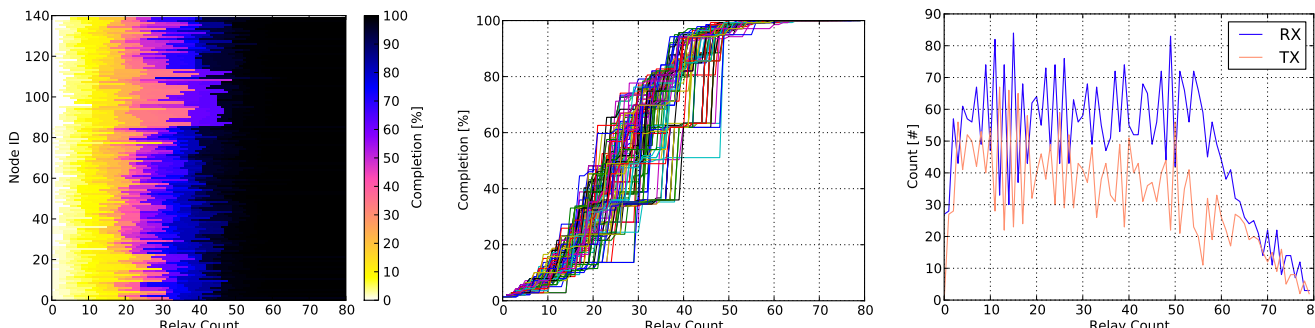


Figure 2. In-network aggregation of a maximum temperature reading (preliminary results for one representative run).

In our example in Fig. 1(c), we reach completion after four relays. At this point, the temperature readings from all three nodes have been integrated because the corresponding flags in the bitfield are set. Thus, all nodes are aware of the maximum sensor reading. Note that we achieve this without any prior transmission schedules or slot assignments. We merely rely on the capture effect to enable nodes to successfully receive data during concurrent forwarding.

3 Preliminary Evaluation

In this section, we discuss preliminary evaluation results with our prototype implementation of CaptureCom. The experiments are performed on Indriya [1], a testbed of 139 nodes deployed on three floors in an office building.

Fig. 2 shows that CaptureCom computes the maximum temperature reading in a network of 139 nodes within about 60 relays at high reliability, achieving an average delay and radio on-time of about 125 ms per round and node. When requesting a temperature reading, say, once per minute, this would result in an average radio duty cycle of about 0.2%.

4 Summary and Future Work

We argue in this paper that CaptureCom closes the control loop in wireless CPS: it provides efficient collection, processing (in our example aggregation), and dissemination within the network, making it suitable for widespread control applications. Our initial evaluation shows that CaptureCom efficiently exploits spacial diversity. It converges within 125 ms at high reliability, leading to low communication de-

lays and high energy efficiency. Motivated by these promising initial results, we are currently investigating the theoretical and experimental foundations of CaptureCom as a novel primitive for wireless CPS.

Acknowledgments

This work was supported by Nano-Tera, NCCR-MICS under SNSF grant number 5005-67322, and partially by the EC, through project FP7-STREP-288195, KARYON.

5 References

- [1] M. Doddavenkatappa, M. C. Chan, and A. Ananda. Indriya: A low-cost, 3D wireless sensor network testbed. In *ICST TridentCom*, 2011.
- [2] F. Ferrari, M. Zimmerling, L. Mottola, and L. Thiele. Low-power wireless bus. In *ACM SenSys*, 2012.
- [3] F. Ferrari, M. Zimmerling, L. Thiele, and O. Saukh. Efficient network flooding and time synchronization with Glossy. In *ACM/IEEE IPSN*, 2011.
- [4] K. Leentvaar and J. Flint. The capture effect in FM receivers. *IEEE Trans. Commun.*, 24(5), 1976.
- [5] J. Lu and K. Whitehouse. Flash flooding: Exploiting the capture effect for rapid flooding in wireless sensor networks. In *IEEE INFOCOM*, 2009.
- [6] M. Pajic, S. Sundaram, J. L. Ny, G. J. Pappas, and R. Mangharam. Closing the loop: A simple distributed method for control over wireless networks. In *ACM/IEEE IPSN*, 2012.