Demo Abstract: DHV - An Efficient Code Consistency Management Protocol for Wireless Sensor Networks

Thanh Dang and Nirupama Bulusu and Wu-chi Feng and SeungWeon Park
Department of Computer Science
Portland State University
Portland, OR, USA, 97201
Email: {dangtx,nbulusu,wuchi,swpark}@cs.pdx.edu

***Abstract*—Ensuring that every sensor node has the same code version is challenging in dynamic, unreliable, multi-hop sensor networks. When nodes have different code versions, the network may not behave as intended, wasting time and energy. We demonstrate DHV, a code consistency maintenance protocol to ensure that every node in a network will eventually have the same code. DHV is based on the simple observation that if two code versions are different, their corresponding version numbers often differ in only a few least significant bits of their binary representation. DHV allows nodes to carefully select and transmit only necessary bit level information to detect a newer code version in the network. DHV can detect and identify version differences in $O(1)$ compared to the logarithmic of current protocols in terms of messages and latency. Simulations and experiments on a real MicaZ testbed show that DHV reduces the number of messages by 50%, converges in half the time, and reduces the number of bits transmitted by 40-60% compared to DIP, the state-of-the-art protocol.**

I. Demonstration

The demonstration needs one table with power supply.

We will demonstrate that the DHV protocol helps reduce network reprogramming latency and conserve energy consumption in communication. The demonstration is on a testbed of 16 MicaZ nodes. A code item can be updated through the default setting at boot time or by dynamically injecting packets from a laptop. A node will act as a base station to broadcast the packets injected from the laptop and sniff the traffic of the network. Based on the sniffed packets and the debugging LEDs on each node, we can determine the state of the network. We will demonstrate the protocol with a testing program for different scenarios. Readers are encouraged to look at the complete paper [1] for detailed results.

Network programming varying the number of total code items. In this scenario, we vary the total number of code items in the network. Users can choose to update one or more arbitrary code items. We will show that the DHV protocol helps discover if nodes need to update code in O(1) latency with respect to the total number of code items.

Network programming varying the number of new code items. In this scenario, we let users vary the number of new code items to be updated. We will show that the DHV protocol helps the network converge in $O(N)$ where N is the total number of new code items.

Network programming varying the number of nodes in a network. We will vary the number of nodes in the network and observe the total number of transmitted messages and the network convergence time. We will show that the total number of transmitted messages and the convergence time increase slightly with the total number of nodes in the network.

Network programming with nodes joining and leaving the network dynamically. We will also show that the network can quickly discover if there is a newly added node with older code versions and reprogram it accordingly. This is an important scenario showing that a CCMP and specifically the DHV protocol is necessary in real situations.

II. Conclusion

We demonstrate the DHV protocol to maintain code consistency in wireless sensor networks. The key innovation in DHV is that it reduces the number of transmitted bytes in the network by carefully selecting and transmitting only absolutely necessary information at the bit level to detect and identify which code items need updates. Theoretically, DHV can identify differences with $O(1)$ complexity in the total number of items instead of logarithmically compared to DIP. Simulations and real-world experiments validate that DHV performs at least twice better than the state-of-the-art DIP protocol. We believe that DHV can not only be used in wireless sensor networks but also in other distributed applications that require data consistency. The preliminary version of the DHV source code can be downloaded at http://sys.cs.pdx.edu/home/dhv.

Acknowledgments — This research was supported in part by National Science Foundation grants 0514818 and 0722063. Any opinions, findings, conclusions or recommendations it contains are those of the authors, and do not necessarily reflect the views of the Foundation.

References

[1] T. Dang, N. Bulusu, W. chi Feng, and S. Park, "Dhv: A code consistency maintenance protocol for multi-hop wireless sensor networks," in *Proceedings of the sixth the European conference on Wireless Sensor Networks, EWSN 09*, Cork, Ireland, February 11–13, 2009.