Computer Communications 55 (2015) 41–50

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

An efficient network reprogramming protocol for wireless sensor networks ☆

Chuchu Dong *, Fengqi Yu

Department of Integrated Electronics, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China
Shenzhen College of Advanced Technology, University of Chinese Academy of Sciences, Shenzhen, China
The Chinese University of Hong Kong, Hong Kong, China
University of Chinese Academy of Sciences, Beijing, China

ARTICLE INFO

Article history:
Received 3 December 2013
Received in revised form 28 August 2014
Accepted 31 August 2014
Available online 15 September 2014

Keywords:
Wireless sensor network
Code dissemination
Network coding
Load balancing

ABSTRACT

Sensor nodes are often deployed in remote area. It is inevitable to update their codes for introducing new functionality or fixing bugs after the deployment. Network reprogramming provides an ultimate solution to this problem through efficient dissemination that relies upon wireless broadcast. Yet, existing code dissemination protocols for reprogramming Wireless Sensor Network (WSN) become inefficient, in terms of power dissipation or delay, in unreliable broadcast environments. In this paper, we design an Adaptive Code Dissemination Protocol (ACDP) for reprogramming wireless sensor network. The proposed ACDP reduces communication cost without introducing intensive computation or complicated transmission control. More importantly, its load balancing feature is capable of extending the lifetime of the entire sensor network, as well as that of individual sensor. The ACDP is implemented on the TelosB platform in TinyOS. The comparisons are made between ACDP and some existing reprogramming protocols which also use network coding scheme. The results show that ACDP improves the reliability of code dissemination, achieves much better load balance, and reduces the number of packets sent per node. It can save the energy of sensor nodes to prolong the network lifetime.

1. Introduction

Wireless Sensor Networks (WSNs) are distributed, self-organized solution to provide sensing, computing, and communication in remote, isolated, and often harsh environments. After the deployment of the sensor nodes, it is necessary to update network software for a variety of reasons, such as fixing bugs in the previous program, changing a single node or even the entire sensor network tasks, and repairing security vulnerabilities. Therefore it is needed to update the code of the sensor nodes. As so far, several protocols have been specifically designed for this purpose. During the code dissemination process in network reprogramming, the base station distributes the new program image file to the target node via wireless multi-hop. So an efficient protocol must be reliable, fast, and taking minimum network bandwidth. Furthermore, the load on the network must be balanced in terms of energy consumption.

Network coding [1,2] has been recently introduced to reduce traffic in data networks. The main idea of network coding is to operate data by XOR or linear combination at intermediate nodes [3,4]. The intermediate nodes can appropriately encode the incoming received packets before transmission and the data traffic is reduced by encoding packets [5–7]. It does not increase delay because communication is much slower compared to the encoding. In this work, the network coding technique has been used to improve the energy efficiency in code dissemination process in WSNs. Network coding technique needs only a few linear operations and the storage of several bytes. In addition, the broadcast nature of WSNs increases the benefits of network coding.

Although network coding has many advantages and has been applied to several networks, there are few publications on network coding for reprogramming WSNs. In this paper, we propose an Adaptive Code Dissemination Protocol (ACDP). It uses random linear coding which is simple and easily implemented at sensor node. Before disseminating packets into the network, a node randomly generates N coefficients and computes the linear combination of N packets. Specially, we define N as the size of coding window.

☆ The authors are with Department of Integrated Electronics at Shenzhen Institutes of Advanced Technology.

* Corresponding author at: Shenzhen Institutes of Advanced Technology, Chinese Academy of Science, Shenzhen, China. Tel.: +86 755 86392417; fax: +86 755 86392299.

E-mail address: cc.dong@siat.ac.cn (C. Dong).

http://dx.doi.org/10.1016/j.comcom.2014.08.017
0140-3664/

Using this coding methodology, the network node needs only receive a sufficient number (which is equal to or greater than N) of distinct encoded packets. Then Gaussian elimination method is used to decode the original packets. In time-varying channel, ACDP uses the adaptive coding window to reduce traffic. As one encoded packet presents a linear combination of N original packets, the larger the coding window is, the less traffic. The optimal coding window N is determined by the network density. Thus a node dynamically chooses the size of coding window based on the number of its neighbors. How the coding window N affects the protocol reliability will be discussed in Section 3. Since a new program must be received by all nodes, ACDP uses NACK for retransmission to guarantee reliability.

Our ACDP has been implemented in TinyOS on TelosB node. The experimental results show that ACDP works reliably. We compare ACDP with some existing reprogramming protocols which also use network coding scheme and the results show that ACDP produces less number of packets for disseminating the same size of code image and needs shorter time for completing the dissemination. In addition, ACDP is more effective for load balancing. In ACDP, the total power consumption of code dissemination is more equally distributed among all sensor nodes in the network.

The rest of the paper is organized as follows. Section 2 introduces the related work. The random linear coding and optimum coding window for different network densities are discussed in Section 3. In Section 4, the implementation of ACDP is presented. Section 5 pays a special focus on the memory and computational overhead for ACDP. Section 6 shows our experimental results, where the performance of ACDP is evaluated. Our conclusions are presented in Section 7.

2. Related work

In this paper, we apply network coding in reprogramming WSNs to reduce the traffic used in disseminating large amount of data. Related research includes three categories: data dissemination, network coding, and network reprogramming.

2.1. Traditional dissemination methods

As so far, a number of code dissemination protocols have been proposed in the last few years. Among them, the Deluge protocol [8] is currently perhaps the most popular one which was recently ported to TinyOS version 2.1.2. In Deluge, each node periodically advertises the most recent version of its program, and nodes request (and receive) program updates based on these advertisements using a NACK-based protocol for reliability. The code image in Deluge is fragmented into pages of fixed size to enable pipelining. This pipelined page transmission can make dissemination faster. MOAP [9] is similar to Deluge, but nodes in MOAP should wait for the complete code image before forwarding. Kulkarni and Wang propose another protocol MNP [10] which, like Deluge, fragments the code image and uses pipelining. Moreover, MNP implements sender selection to limit the number of concurrent transmission in each neighborhood. This sender selection mechanism can reduce collisions and avoid the hidden terminal problem.

These reprogramming protocols disseminate complete binary image of a piece of code. Some of them deliver only the difference between the old and new programs, such as Incremental [11], Trickle [12] and TinyCubus [13]. Similar protocols are proposed in [14,15]. In [14], the authors propose a novel update-conscious compilation (UCC) technique to reduce the amount of transmitted data to remote sensors. The work in [15] makes the same effort to reduce the size of transmitted code significantly by introducing a new mechanism called Elon. Although CSMA has been used in most protocols, TDMA has also been proposed to reduce contention and achieve higher throughput in Infuse [16] and Sprinkler [17]. However, TDMA is much more complex than CSMA to be implemented on a sensor platform and thus currently is not available for most sensor platforms. A detail description and comparison of these methods are provided in [18].

Although these traditional methods have many merits, they suffer from fundamental limitations that can significantly impair their use in some systems. For example, the performance of these protocols quickly degrades when the network size and density get large and thus packet loss becomes high.

2.2. Network coding

Network coding is used to improve throughput or save bandwidth and has been widely applied in wireless networks. The works in [19–21] apply network coding to increase unicast throughput in the face of lossy wireless links. The network coding has also emerged as an energy efficient technique to improve broadcasting capabilities in a network [22–25]. The researchers in [26–30] further develop new coding-aware routing protocols for wireless networks to provide better path bandwidth estimate and identify high throughput paths. Moreover, network coding technique can be used to protect against wireless link failure [31] and path failure [32].

In recent years, network coding has also been introduced to wireless sensor networks for data collection [33–35] and data aggregation [36,37]. In order to reduce the required transmission path hop and redundant data in WSNs, network coding algorithms are combined with multipath routing to deliver the message from source to sink [38–41]. Furthermore, network coding can be proposed to handle the security problem in WSNs. In [42,43], the authors propose LNCS and SBLNC algorithm respectively to provide data availability and identify the false data. Then, in [44], the authors summarize security goals and design challenges in achieving security for network coding systems.

Energy efficiency is one of the most important parameters to prolong the lifetime of WSNs. In comparison to the traditional store and forward mechanism, the network coding technique has been proved to be able to increase the bandwidth efficiency and thus decrease the energy consumption in WSNs.

2.3. Network reprogramming

Since network coding can significantly reduce traffic and improve the transmission performance of multi-hop links, in recent years, some research in code dissemination protocol has been conducted. Crepaldi et al. and Rossi et al. propose two different code dissemination protocols for wireless sensor network reprogramming, named FRP [45] and SYNAPSE [46]. These two protocols both adopt the code methodology called Fountain Code [47–50]. Ha et al. prove that Luby Transform (LT) code [51] is also an implementation of Fountain Code. With this encoding method, the sending node can generate encoded packets from k original packets. While the receiving node can decode successfully after receive any $k(1+\epsilon)$ number of the encoded packets. Well-designed Fountain Code not only has a very small overhead and a simple decode algorithm, but also can obtain fast propagation speed and less packets communication during the code dissemination. However, in the protocols using Fountain Code, a node must broadcast enough encoded packets to make sure the receiving node can decode successfully and thus introduce some redundant packets.

Hagedorn et al. propose Rateless Deluge based on Deluge protocol [52], using random linear network coding RLC (Random Linear Codes) [53]. In this protocol, the sending nodes randomly generate N coefficients and compute the linear combination of the packets.