Ad Hoc Networks 11 (2013) 490–507

Contents lists available at SciVerse ScienceDirect

Ad Hoc Networks

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

On combining network coding with duty-cycling in flood-based wireless sensor networks ☆

Roja Chandanala, Wei Zhang, Radu Stoleru *, Myounggyu Won

Department of Computer Science and Engineering, Texas A&M University, United States

ARTICLE INFO

Article history:
Received 2 July 2011
Received in revised form 16 July 2012
Accepted 23 July 2012
Available online 4 August 2012

Keywords:
Wireless sensor networks
Energy efficiency
Duty cycling
Network coding

ABSTRACT

Network coding and duty-cycling are two major techniques for saving energy in wireless sensor networks. To the best of our knowledge, the idea to combine these two techniques for even more aggressive energy savings, has not been explored. This is not unusual, since these two techniques achieve energy efficiency through conflicting means, e.g., network coding saves energy by exploiting overhearing (i.e., nodes are awake), whereas duty-cycling saves energy by reducing idle listening (i.e., nodes sleep). In this article, we thoroughly investigate if network coding and duty cycling can be used together for more aggressive energy savings in flood-based wireless sensor networks.

Our main idea is to exploit the redundancy sometimes present in flooding applications that use network coding, and put a node to sleep (i.e., duty cycle) when a redundant transmission takes place (i.e., the node has already received and successfully decoded a sequence of network-coded packets). We propose a scheme, called DutyCode, in which a multiple access control (MAC) protocol implements packet streaming and allows the network coding-aware application to decide when a node can sleep. We also present an algorithm for deciding the optimal coding scheme for a node to further reduce energy consumption by minimizing redundant packet transmissions. Finally, we propose an adaptive switching technique between DutyCode and an existing duty-cycling MAC protocol. We investigate our proposed solutions analytically and implement them on mote hardware. Our performance evaluation results, obtained from a 42-node indoor testbed, show that our scheme saves 30–46% more energy than network coding-based solutions.

1. Introduction

Energy is a scarce resource in wireless sensor networks (WSNs) and its conservation has been the subject of extensive research. While a variety of solutions have been proposed for saving energy in WSN, duty cycling and network coding have proven to be two of the most successful techniques.

Network coding is a technique that increases energy efficiency and reduces network congestion by combining packets destined for distinct users. Since the initial proposal by Ahlswede et al. [2], many applications have incorporated this technique. Network coding is particularly well-suited for WSN due to the broadcast nature of their communications. Overhearing is effortless, propagation is usually symmetric, and energy efficiency is a priority. Network coding can also be found in applications including multi-cast, content distribution, delay tolerant networks, underwater sensing suites, code dissemination, storage, and security. As diverse as these applications are, they all share a common assumption: *nodes in a network are always awake.*

Duty cycling is a technique that increases energy efficiency by allowing *a node to turn off part or all of its systems*for periods of time. Encompassing a range of

☆ A preliminary version of this article was presented at the IEEE International Conference on Networked Sensing Systems (INSS), 2010 [1].

* Corresponding author.
E-mail addresses: roja@cse.tamu.edu (R. Chandanala), wzhang@cse.tamu.edu (W. Zhang), stoleru@cse.tamu.edu (R. Stoleru), mgwon@cse.tamu.edu (M. Won).

1570-8705/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.adhoc.2012.07.008

techniques from peripheral device management to almost complete system shutdown, duty cycling extends node lifetime and reduces maintenance. It has been shown that duty cycling can extend battery life by an order of magnitude or more. In WSN duty cycling is pervasive, and almost all deployed systems use it. Given the importance of duty cycling to WSN, the assumption that nodes will be awake cannot be made. *Since nodes will be asleep at least part of the time, i.e., the time available for overhearing is reduced, network coding becomes more difficult.*

In this article, we address the challenge faced when aggressive energy savings (i.e., both duty-cycling and network coding) are needed in flooding-based WSN applications. To the best of our knowledge, this is the first work that considers the simultaneous use of duty cycling and network coding. We particularly target code/program dissemination (i.e., distributing a new program/executable image to all sensor nodes), a flood-based application that needs a non-negligible amount of time for execution, e.g., tens to hundreds of minutes for large scale WSN.

Our main idea is derived from the intuition that, due to redundancy in network coding for flooding applications, there are *periods of time* when a node does not benefit from overhearing packets. We seek to precisely determine these periods of time and let nodes that do not benefit from overhearing, to be put to sleep, i.e., to duty-cycle.

Our solution to the aforementioned challenge is *DutyCode*, a cross layer scheme in which Random Low Power Listening (RLPL) – a new MAC protocol – facilitates streaming, elastic random sleeping and transmission arbitration, while the Network Coding-aware Application determines the time to sleep and the sleep duration. We also propose an *Enhanced Coding Scheme (ECS)* algorithm, which eliminates redundant packet transmissions by selecting appropriate network coding schemes for nodes. Finally, a novel technique, called *LPL/RLPL Mode Transition*, ensures the smooth transition between our RLPL protocol and Low Power Listening (LPL), a typical duty-cycling MAC protocol which is more energy efficient for non-flooding WSN applications. The contributions of this article are as follows:

- DutyCode – a cross layer scheme that supports packet streaming and a mechanism for randomizing sleep cycles using elastic intervals. These allow nodes to intelligently select sleep periods.
- ECS – an algorithm for deciding an efficient coding scheme in static networks. ECS assigns coding schemes to minimize the number of transmissions, thus allowing for more energy savings.
- LPL/RLPL Mode Transition – a completely adaptive solution allowing the application to smoothly switch between LPL and RLPL, without packet loss.
- Theoretical analysis of our proposed DutyCode and ECS schemes and extensive simulations demonstrating their energy efficiency and high throughput.
- An implementation of our schemes on mote hardware, and performance evaluation in a 42-node testbed where actual energy consumption is measured.

This article is organized as follows. Section 2 provides background on network coding and duty cycling, and the motivation for our work. Sections 3–5 present the design of our DutyCode protocol, ECS algorithm and LPL/RLPL transition technique, respectively. Section 6 presents theoretical analysis of DutyCode and ECS algorithm. Section 7 describes the implementation of our solutions, and Section 8 presents performance evaluation results. We review the state of art in Section 9 and conclude in Section 10.

2. Background and motivation

Network coding enhances energy efficiency by reducing the number of packet transmissions. The basic concept of network coding, as applied to a flood-based application, can be explained using a simple scenario shown in Fig. 1. Sender s wants to flood two packets x_1 and x_2. As shown in Fig. 1a, when network coding is not used, six packet transmissions are required to deliver the two packets to all nodes in the network, i.e., r_1, r_2, d_1, d_2. As shown in Fig. 1b, however, when network coding is used, only 4 transmissions are needed. This is because each of the two relays transmits only one coded packet. For network coding to work, receivers d_1, d_2 must be able to receive both coded packets, i.e., $(x_1 + x_2)$ and $(x_1 + 2x_2)$. Otherwise, they will be unable to decode the other packet received.

It is important to note that, unlike normal broadcast/flooding packets, one missing coded packet can render a sequence of coded packets "useless" (i.e., they do not convey any information). Consider a scenario where a node receives the independent coded packets $(a_1x_1 + a_2x_2 + a_3x_3 + a_4x_4)$, $(b_1x_1 + b_2x_2 + b_3x_3 + b_4x_4)$, and $(c_1x_1 + c_2x_2 + c_3x_3 + c_4x_4)$. For decoding these packets, it becomes critical to receive another coded packet, say $(d_1x_1 + d_2x_2 + d_3x_3 + d_4x_4)$. Otherwise all 3 received packets are useless. As the coding scheme increases (i.e., *coding scheme is defined as the number of different packets coded into a single packet*) the penalty for losing a single packet increases linearly.

2.1. AdapCode design

In this subsection we present AdapCode [3], a flooding application which uses network coding and employs CSMA as its MAC protocol. Fig. 2 describes the protocol. In this figure, a sender node s transmits a sequence of packets

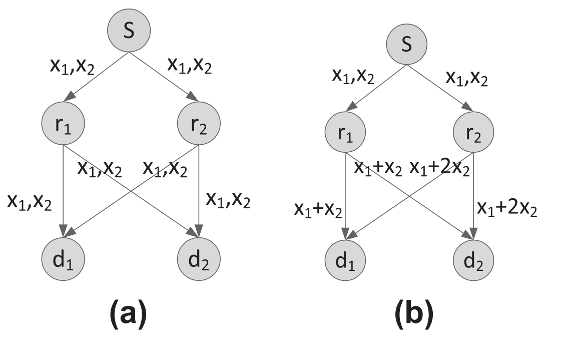

Fig. 1. A flood-based application in which node s floods packets x_1 and x_2 in the entire network: (a) transmissions when network coding is not used (a total of 6 packet transmissions); (b) transmissions when network coding is used (4 packet transmissions).