Computer Communications 000 (2016) 1–16

Contents lists available at ScienceDirect

Computer Communications

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

Revisiting XOR-based network coding for energy efficient broadcasting in mobile ad hoc networks

Nikolaos Papanikos, Evangelos Papapetrou*

Department of Computer Science and Engineering, University of Ioannina, GR 45110, Greece

ARTICLE INFO

Article history:
Received 7 June 2016
Revised 16 July 2016
Accepted 31 July 2016
Available online xxx

Keywords:
Energy-efficient broadcasting
Network coding
Mobile ad hoc network
XOR-based coding
Connected dominating set
Deterministic broadcasting

ABSTRACT

Network coding is commonly used to improve the energy efficiency of network-wide broadcasting in wireless multi-hop networks. In this work, we focus on XOR-based broadcasting in mobile ad hoc networks with multiple sources. We make the observation that the common approach, which is to benefit from the synergy of XOR network coding with a CDS-based broadcast algorithm, suffers performance breakdowns. After delving into the details of this synergy, we attribute this behavior to an important mechanism of the underlying broadcast algorithm, known as the “termination criterion”. To tackle the problem, we propose a termination criterion that is fully compatible with XOR coding. In addition to that, we revisit the internals of XOR coding. We first enhance the synergy of XOR coding with the underlying broadcast algorithm by allowing each mechanism to benefit from information available by the other. In this way, we manage to improve the pruning efficiency of the CDS-based algorithm while at the same time we come up with a method for detecting coding opportunities that has minimal storage and processing requirements compared to current approaches. Then, for the first time, we use XOR coding as a mechanism not only for enhancing energy efficiency but also for reducing the end-to-end-delay. We validate the effectiveness of our proposed algorithm through extensive simulations on a diverse set of scenarios.

1. Introduction

Network-layer broadcasting is fundamental for mobile ad hoc networks because it provides the means to disseminate information throughout the network. Besides application data, broadcast protocols are also used to distribute control information to every network node [1,2]. In this way, each node maintains a view of the network structure that is a basic element for many networking mechanisms. The seminal work by Ahlswede et al. [3] introduced network coding, a concept that significantly enhances the performance of networking protocols in both wired and wireless networks. As a result, over the last years, many researchers focused on incorporating network coding into broadcasting in wireless ad hoc networks [4–25]. Based on their main objective, coding-based approaches can be classified into: (i) energy-efficient, and (ii) delivery guarantee schemes. The schemes of the first class [4–13] utilize network coding towards energy efficiency aiming to strike the best possible balance between delivery and cost (as expressed by the number of transmissions). On the other hand, the schemes of the second class [14–25] use network coding to guarantee the delivery of broadcast packets to all network nodes, treating the minimization of the related costs as a secondary objective.

In this work we focus on energy-efficient broadcasting in mobile ad hoc networks. Moreover, we are interested in the scenario of multiple broadcasting sources, i.e. we examine the many-to-all and all-to-all communication paradigms. Such scenarios appear rather frequently when multiple nodes in parallel and independently engage in discovery phases. Some representative examples include discovering routes in on-demand routing protocols [1], locating resources in service discovery applications [2,26] and retrieving volatile data from peer databases [27,28]. In all of the aforementioned examples the focus is on energy efficiency rather than on guaranteeing delivery.

In the field of energy efficient broadcasting the most popular design choice is to adopt XOR-based network coding [29]. Algorithms that follow this approach [4–9] encode packets on a hop-by-hop basis using bitwise XOR and then forward them using a Connected Dominating Set (CDS) based broadcasting scheme [30–33]. Although this strategy has been proved successful, we bring to light several occasions where its performance severely degrades and the coding gain becomes negligible. Motivated by this finding, we examine in depth the synergy of network coding

* Corresponding author. Fax: +302651008997.
E-mail addresses: npapanik@cse.uoi.gr (N. Papanikos), epap@cse.uoi.gr (E. Papapetrou).

http://dx.doi.org/10.1016/j.comcom.2016.07.016

and the underlying CDS-based broadcast algorithm. We conclude that the weak link is the mechanism of the broadcasting algorithm known as "the termination criterion". Therefore, as a first step, we explore the use of alternative termination criteria proposed in the literature of traditional broadcasting. Unfortunately, we find that none of them is compatible with XOR network coding. To address the problem, we propose the *Network Coding Broadcast with Coded Redundancy* (NOB-CR) algorithm which incorporates a novel termination criterion that is fully compatible with XOR coding. Moreover, NOB-CR revisits the coding internals in order to enhance the overall performance in terms of energy efficiency, delivery delay and utilization of network resources. In summary, our main contributions are:

- We unveil the shortcomings in the synergy between XOR coding and CDS-based broadcasting (Section 3). Then, after analyzing the reasons of this finding, we propose a coding-friendly termination criterion for the CDS-based algorithm and illustrate its efficiency (Section 4).
- We delineate a novel method for detecting coding opportunities (Section 5.2). The method is lightweight in the sense that, in contrast to current approaches, requires each node to maintain minimum state while it mostly utilizes information that is already available through the underlying broadcast mechanism.
- We enhance the pruning efficiency of the underlying CDS-based algorithm by exploiting information available from the coding mechanism, i.e. we establish a bidirectional synergy between network coding and CDS-based forwarding (Section 5.3).
- We address the problem of increased end-to-end delay in network coding broadcasting (Section 5.4). This problem is a direct consequence of Random Assessment Delay (RAD), a mechanism used by XOR coding in an effort to increase coding opportunities. The proposed solution, called *Coded Redundancy*, takes a novel approach and uses XOR coding to achieve a cost-free increase of packet redundancy across the network in order to reduce end-to-end delay.

The rest of the paper is organized as follows. In Section 2, we review the basic principles of CDS-based broadcasting and XOR network coding. In Section 6, we present an extensive evaluation of the proposed algorithm. In Section 7, we review the existing work related to coding-based broadcasting in wireless ad hoc networks. Finally, we summarize our findings in Section 8.

2. Preliminaries

Before continuing, we first briefly review the basic principles of CDS based broadcasting as well as XOR-based coding.

2.1. CDS-based broadcast principles

Energy efficient broadcast algorithms aim to minimize the number of transmissions required for delivering a packet to all network nodes [34,35]. The most effective algorithms follow the CDS-based broadcasting approach. According to this, the algorithm constructs a connected dominating set of the network [30–32]. The nodes constituting the CDS are the *forwarders*, i.e. those elected to forward the broadcast packets, while all other nodes just act as passive receivers. Since computing the forwarders should be performed in a distributed fashion, the common approach is to approximate them locally at each node v using its 1-hop neighbor set ($\mathcal{N}(v)$), i.e. the set that consists of v's one hop neighbors, and the 2-hop neighbor set ($\mathcal{N}(\mathcal{N}(v))$), i.e. the set consisting of all nodes that lie at maximum two hops away from v.

Even though transmitting packets only through forwarders successfully reduces packet duplicates, a significant number of them still exists across the network. This is because the selection of forwarders is made in a distributed manner and with limited information. As a result, special attention should be given to these duplicates as they could lead to additional transmissions and degrade energy efficiency. Therefore, the reception of a packet duplicate in a forwarder node leads to a dilemma whether to forward it or not. Forwarding the duplicate could increase redundant transmissions while dropping it could potentially impact the delivery efficiency. The mechanism that is responsible to handle such situations is the *termination criterion*. Multiple criteria have been proposed in the literature [33,36–38]. In the rest of this paper we will use the terminology proposed in [33] and [36] to refer to these criteria:

Termination Criterion 1 (Marked/unmarked (M/U))**.** Each node keeps track of the packets received by each of its 1-hop neighbors. Then, in the case of a duplicate reception, a forwarder transmits the received duplicate if at least one of the neighbors is not marked to have received the packet.

This is the most well-known approach. However, having all nodes to store the reception status for all of their 1-hop neighbors and for all packets could be a daunting challenge in terms of both memory usage and processing overhead [33].

Termination Criterion 2 (Relayed/unrelayed (R/U))**.** A forwarder transmits a duplicate only if no other duplicate of the same packet has been relayed by the same forwarder in the past.

Termination Criterion 3 (Covered/uncovered (C/U))**.** A node acting as a forwarder relays packets seen for the first time while it drops already seen packets including the ones not relayed in the past because the node was not elected as a forwarder at that time.

In contrast to M/U, the latter two approaches are more realistic due to the limited storage and processing requirements.

The algorithms proposed in the literature follow two major strategies for building the CDS, i.e. calculating the forwarders. The first is to build a CDS that is common to every network node using local information [31,38–49] while the second is to build a source-specific CDS [32,33,50,51]. In the first category the nodes of the CDS are used for any packet regardless of its source and updated whenever topology changes are detected. Most efficient studies in this line of research also use information related to the broadcast process, e.g. packet reception status, in order to further prune transmissions and/or enhance reliability [31,38,43–49]. On the other hand, in the second category, a node that relays a packet calculates the list of forwarders by considering the previous hop of the packet and piggybacks the corresponding list on it. In this way, a source-based CDS is formed for each packet. More specifically, when a node v receives a packet from u checks whether it is selected as a forwarder. If so, a common approach is to elect forwarders so as to deliver the packet to (or "cover") the set $\mathcal{U}(v)$ of nodes that lie exactly 2-hops away from v, i.e. $\mathcal{U}(v) = \mathcal{N}(\mathcal{N}(v)) - \mathcal{N}(v)$. The set of candidate forwarders $\mathcal{C}(v)$ is in general a subset of v's neighbors, i.e. $\mathcal{C}(v) \subseteq \mathcal{N}(v)$. Note that $\mathcal{U}(v) \subseteq \bigcup_{\forall u \in \mathcal{C}(v)} \mathcal{N}(u)$ and that $\mathcal{C}(v)$ can be seen as a set of sets if each node $u \in \mathcal{C}(v)$ is replaced by $\mathcal{N}(u)$, thus the election of forwarders is modeled as a set cover problem. The solution is usually given by the well-known greedy set cover (GSC) algorithm [52], however other more efficient approximation algorithms exist [31,53–55]. Furthermore, node v takes advantage of u's neighborhood to reduce both the set of candidate forwarders, i.e. $\mathcal{C}(v) = \mathcal{N}(v) - \mathcal{N}(u)$, and the set of nodes $\mathcal{U}(v)$ that should receive the packet. Algorithms in the sourced-based CDS category vary in the approach taken to minimize the set $\mathcal{U}(v)$ and therefore the number of forwarders. TDP and PDP [33] exploit u's two-hop neighborhood and further minimize the $\mathcal{U}(v)$ set. For example, node v in PDP elects forwarders