Ad Hoc Networks 16 (2014) 142–164

Contents lists available at ScienceDirect

Ad Hoc Networks

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

# A comprehensive analysis on the use of schedule-based asynchronous duty cycling in wireless sensor networks

Ricardo C. Carrano *[,1], Diego Passos [2], Luiz C.S. Magalhães [1], Célio V.N. Albuquerque [2]

*Laboratório MídiaCom, Universidade Federal Fluminense, R. Passo da Pátria 156, bl.E sala 408, Niterói, RJ 24210-240, Brazil*

ARTICLE INFO

*Article history:*
Received 11 October 2012
Received in revised form 15 November 2013
Accepted 23 December 2013
Available online 1 January 2014

*Keywords:*
Asynchronous duty cycling
Discovery latency
Energy efficiency

ABSTRACT

Duty cycling is a fundamental mechanism for battery-operated ad hoc networks, such as Wireless Sensor Networks, Delay Tolerant Networks, and solar-powered Wireless Mesh Networks. Because of its utter importance, it has been proposed in a wide variety of flavors, one of the most prominent being that of the asynchronous mechanisms. In particular, schedule-based duty cycling has earned attention due to its low requirements and simplicity of implementation.

Despite its potential, a comprehensive and realistic study on the neighbor discovery latency that results from schedule-based asynchronous duty cycling is still missing. This paper fills in this gap, by providing accurate models for major schedule-based mechanisms: Block Designs, Quorum systems and Disco. The provided models consider message loss probability and yield more precise estimations than traditional models. Based on this improved accuracy, the relative latency, a new metric for studying the trade off between latency and power, is proposed as a substitute to the power-latency product. Finally, a practical mapping of which schedule is more adequate for given requirements of latency, energy savings and link reliability is presented.

## 1. Introduction

Energy efficiency is fundamental to most Ad Hoc Networks. In Wireless Sensor Networks (WSNs), Delay Tolerant Networks (DTNs) and also in some Wireless Mesh Networks (WMNs), for example, nodes are battery-powered and often operate under severe energy constraints. In such applications, the duty cycling of the radio interface is a necessity, since the radio is usually responsible for a significant amount of a node's power drain [1,2].

However, duty cycling demands coordination. There must be ways to guarantee that a node will find an active neighbor for relaying its data, and do it timely. Such coordination may be achieved with synchronous mechanisms, in which nodes keep a common time reference, or with asynchronous mechanisms, where a common clock is unnecessary. There are also hybrid mechanisms in which the network is divided into clusters that are synchronized internally, while the communication between the clusters remains asynchronous.

In comparison to synchronous mechanisms, asynchronous contenders have the advantage of not relying on additional hardware, such as GPS or extra-precision clocks, and of generating less control traffic. Because of that, a plethora of asynchronous schemes has been proposed during the last decade. One of the most prolific categories of asynchronous duty cycling schemes is based on the use of special *wakeup schedules*. These schedules are alternations of active and inactive time slots selected in a way that guarantees that nodes will have a minimum overlapping active time, independent of their synchronization. During these periods, nodes may exchange messages and go back to

---

* Corresponding author. Tel.: +55 21 2629 5486.

*E-mail addresses:* carrano@midiacom.uff.br (R.C. Carrano), dpassos@midiacom.uff.br (D. Passos), schara@midiacom.uff.br (L.C.S. Magalhães), celio@ic.uff.br (C.V.N. Albuquerque).

[1] Department of Telecommunications Engineering.

[2] Institute of Computing.

1570-8705/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.adhoc.2013.12.009

sleep afterwards. We will refer to these overlapping active slots as *opportunities of discovery*.

As it has been pointed out [3,4], asynchronous mechanisms may result in long delays that accumulate over multihop paths. The *sleep waiting problem*, or *data forwarding interruption problem* [4], for instance, refers to the time a node has to wait until the next-hop neighbor wakes up. In fact, as duty cycling is reduced, neighbor discovery time (NDT) tends to increase, which is the fundamental trade off of the asynchronous approach and a core issue to our analysis.

Since discovery latency may be a hindrance to the use of asynchronous duty cycling, it is crucial to understand it thoroughly. However, to the best of our knowledge, analyses published so far do not apprehend important aspects of latency. Firstly, they tend to adopt an oversimplified model for the latency that assumes, for instance, that the opportunities of discovery occur at the end of the cycle. On top of that, they fail to consider the possibility of message loss. As channels are always imperfect, a *discovery opportunity* does not necessarily lead to *neighbor discovery*, meaning that many opportunities may be necessary until neighbors can communicate. Even when this fact is acknowledged [5] it is not incorporated to an analytical model. The models provided in this paper address these two issues and, as we demonstrate, provide more accurate estimations for the NDT.

The proposed models estimate the NDT for the most cited schedule-based asynchronous mechanisms: Block Designs [6], Quorum systems (Grid and Torus) [5,7] and Disco [8], and were validated with real implementations on sensor motes and with statistical simulations. Moreover, their improved accuracy allowed us to introduce a new metric – the *relative latency* – which provides a level playing field for comparisons between the mechanisms.

The work is concluded with a synoptic table that compares all asynchronous mechanisms presented. As it will be demonstrated, though Block Designs have a substantial advantage over the other proposals in most scenarios, it is not a suitable choice for all deployments. Also, the link quality, and the resulting probability of frame reception, may favor one or another mechanism in non-obvious ways.

The contributions of this paper may be divided into four groups (a more detailed list of contributions is provided in Section 7):

1. General contributions to the understanding of asynchronous duty cycling proposals, with formal definition and survey of the category of schedule-based asynchronous duty cycling mechanisms.
2. Proposal of accurate models for determining the neighbor discovery time (NDT) for the main asynchronous schedule-based duty cycle mechanisms, that incorporate message loss and relative time offsets between the neighbors. Current models are grossly inaccurate, with errors easily amounting to 400% for low quality links (e.g. $p < 0.2$). With the proposed models this error is typically inferior to 10%.
3. Practical comparison between the mechanisms. With improved models and the introduction of a new metric (relative latency) that allows direct and fair comparisons between the mechanisms, we demonstrate that the previous comparison, based on the power-latency metric is misleading. We show that the schedules perform comparatively better or worse depending on the link quality, and we indicate which schedule performs better given certain practical requisites.
4. In-depth analysis and discussion of aspects of each mechanism, that were not provided in current literature.

The rest of this paper is organized as follows. Section 2 defines the category of schedule-based asynchronous duty cycling and quickly reviews the other important asynchronous mechanisms. Sections 3–5 are dedicated to the most important categories of schedule-based asynchronous mechanisms, namely Block Designs (Section 3), Quorum systems (Section 4), and mechanisms based on prime numbers, such as Disco (Section 5). In each of these sections, the main mechanisms are studied and an analytical model for the estimation of the neighbor discovery time (NDT) is presented. Moreover, all models are validated through tests in real sensor motes and statistical simulations. Section 6 provides useful comparisons between the mechanisms, supported by the presented models. Section 7 presents our final remarks and a synoptic table of the most important findings of our analysis.

## 2. Asynchronous duty cycling

It is generally accepted that synchronizing nodes in a multihop wireless network is hard and costly [1,5], requiring extra hardware or processing capacities that may be too demanding for certain nodes, or adding frequent control traffic, which takes airtime and drains precious energy for transmission. In response to that, the asynchronous branch of proposals is prolific and diverse [6–13]. We start this section with a quick review of the main asynchronous mechanisms, and then proceed to a formal definition of the category of our interest: schedule-based mechanisms.

### 2.1. Overview of the asynchronous duty cycling mechanisms

Asynchronous duty cycling mechanisms tend to present similar issues, the most relevant being an increase in latency due to sleep waiting. Another, that may also affect synchronous mechanisms, is *idle listening*, which happens when a node wakes up in vain, i.e. when no traffic is directed to it. Asynchronous mechanisms differ in how they try to reduce latency or idle listening, while achieving low duty cycles.

In *preamble sampling* [9] mechanisms, every node goes to sleep asynchronously and wakes up periodically to check for channel activity. If a preamble is heard during this check-up period, the node remains active, waiting for the incoming frame. If not, the node goes back to sleep. What guarantees that the frame will be detected is the duration of the preamble – longer than the duration of active and sleep times together.

Another category of mechanisms is based on *receiver-initiated transmissions* [10]. In this category, instead of sig-