Theoretical Computer Science 608 (2015) 120–134

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

The wake up dominating set problem

Amir Bannoura *, Christian Ortolf, Leonhard Reindl, Christian Schindelhauer

Faculty of Engineering, University of Freiburg, George-Köhler-Allee 106, 79110 Freiburg, Germany

ARTICLE INFO

Article history:
Received 28 February 2014
Accepted 10 January 2015
Available online 15 January 2015

Keywords:
Wake-up receivers
Online algorithm
Connected dominating set
Unit-disk graph
Epidemic algorithms

ABSTRACT

Recently developed wake-up receivers pose a viable alternative for duty-cycling in wireless sensor networks. Here, a special radio signal can wake up close-by nodes. We model the wake-up range by the unit-disk graph. Such wake-up radio signals are very energy expensive and limited in range. Therefore, their number must be minimized.

We revisit the Connected Dominating Set (CDS) problem for unit-disk graphs and consider an online variant, where starting from an initial node all nodes need to be woken up, while the online algorithm knows only the nodes woken up so far and has no information about the number and location of the sleeping nodes.

We show that in general this problem cannot be solved effectively, since a worst-case setting exists where the competitive ratio, i.e. the number of wake-up signals divided by the size of the minimum CDS, is $\Theta(n)$ for n nodes. For dense random uniform placements, this problem can be solved within a constant factor competitive ratio with high probability, i.e. $1-n^{-c}$, when the nodes positions are known or at least some rough distance estimator. For a restricted adversary with a reduced wake-up range of $1-\epsilon$ we present a deterministic wake-up algorithm with a competitive ratio of $O(\epsilon^{-\frac{1}{2}})$ for the general problem in two dimensions.

In the case of random placement without any explicit position information we present an $O(\log n)$-competitive epidemic algorithm to wake up all nodes with high probability. Simulations show that a simplified version of this oblivious online algorithm already produces reasonable results, that allow its application in the real world.

1. Introduction

Energy is the driving problem of wireless sensor networks (WSNs), since sensor nodes usually operate for long periods and the only source of energy is battery cells which are difficult to be exchanged. The functionality of WSNs can be extended through the use of low power microprocessors, sensors, and radio transceivers. The availability of low power hardware components provided a technological break-through of wake-up receivers. These receivers interact only when a special wake-up signal is addressed to them. When a wake-up signal is received, the wake-up receiver triggers an interrupt to wake the sensor node. In addition, any sensor node has the capability of transmitting a wake-up signal to wake up all other close-by sensor nodes. Recent research [1] has decreased the energy consumption of sensor nodes when no activity is required to less than 9 μW, whereas sensor nodes that are not equipped with wake-up receivers that use duty cycle would be spending around 51 mW checking the medium from time to time. Fig. 1 shows a sample board of a designed wake-up receiver integrated with a wireless sensor node.

* Corresponding author.
E-mail address: bannoura@imtek.de (A. Bannoura).

http://dx.doi.org/10.1016/j.tcs.2015.01.006
0304-3975/

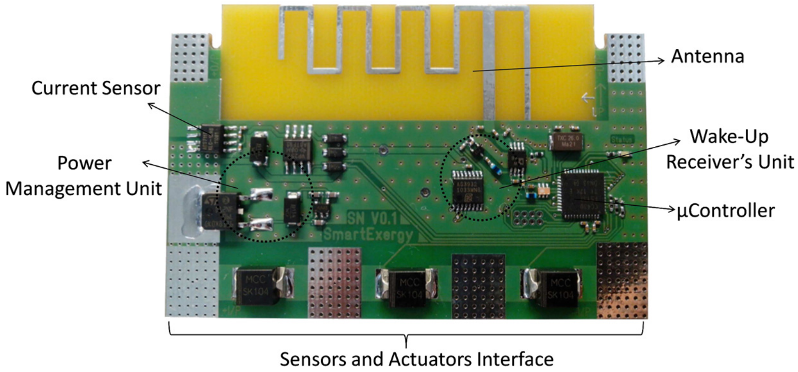

Fig. 1. Wireless sensor node with integrated wake-up receiver.

Wake-up receivers integrated in sensor nodes constitute a paradigm shift for wireless sensor protocols. In which, sensor nodes interact with the surrounding neighbors only when they are required to receive and send information. The duty cycle process for periodically waking up to check for messages or synchronizing with other sensor nodes is no longer essential.

Despite that this technology provides a new solution for the energy consumption problem, new problems arise. Sensor nodes are required to produce a wake-up signal, these signals are energy expensive compared to the signals that are required for normal data communication. Furthermore, the communication range of a wake-up signal is smaller than the normal data communication range, which requires a multi-hop wake-up signals to wake up sensor nodes that are located in the normal data communication range. Our aim is to reduce the number of wake-up signals transmitted as much as possible in order to increase the covered area and reduce the energy consumption of the sensor nodes.

A straight-forward solution is to establish minimum set of sensors which are able to wake up all sensor nodes in case some data need to be collected or distributed. This is the well known Connected Dominating Set (CDS) problem, where one tries to compute the Minimum CDS (MCDS). This problem plays an important role in wireless networks and it is known to be NP-complete since no efficient way is proposed to find a solution in polynomial time.

For simplicity we assume a unit-disk range model and we are interested in computing the minimum connected dominating set in unit disk graphs, i.e. geometric graphs where an undirected edge exists between nodes, if their distance is at most 1.

However, our problem is somehow different. When the sensor nodes are placed, no positions are known before the first wake-up signal. Also, the nodes may be moving, sensor nodes may fail, and persistent memory might not be available. All these reasons justify to build up a CDS from scratch regularly.

So, we face an online version of the MCDS problem in the context of wake-up receivers. At the beginning, one sensor node wakes up, maybe because of new sensor data. It broadcasts a wake-up signal and receives responses from all the nodes in the neighborhood. Then, a decision needs to be taken regarding the next node that is allowed to broadcast another wake-up signal. Since normal data communication consumes only little energy compared to wake-up signals, we can assume that all active nodes are aware of each other. Furthermore, the information which sensor received a wake-up signal is available to us, even if the sensor has already been woken up. The question is now, can we wake up all nodes with minimal number of wake-up signals. This is what we address as the *wake-up minimum connected dominating set problem in unit disk graphs*. In this variant the positions of the woken up nodes become available as soon as they are awake. For the *wake-up position-aware minimum connected dominating set problem in unit disk graphs* positions are not known as well at the beginning until the nodes are awake.

2. Related work

The new generation of wake-up receivers developed by Gamm et al. in [1] give us an alternative to the concept of duty cycles for awaiting incoming messages in wireless sensor networks.

A perfectly efficient online wake-up would use a minimum connected dominating set of nodes to wake up all the nodes. Finding such a MCDS was already shown to be NP-hard for general graphs, as well as for unit disk graphs [2,3]. For the general (non-unit-disk graph) problem no polynomial time approximation exists unless $NP \subseteq DTIME[n^{O(\log\log n)}]$ [4], yet for MCDS with unit disk graphs a PTAS has been presented in [5].

Movement of sensor nodes and maintaining an existing MCDS was discussed before by Das et al. in [6]. Of course the wake-up problem is an online version of MCDS, because of the differences to the online version presented by Eidenbenz [7] we are referring to it as the wake-up problem. Eidenbenz models the online problem by node added every round by an adversary, while the online algorithm has to present a CDS, but may never remove nodes once added to the CDS. He shows a competitive ratio of $\Theta(n)$ for the CDS size.

Another online MCDS problem closer related to the wake-up problem is the reactive routing problem [8] leading to a lower bound of $\Omega(n)$. This motivates why comparing to an adversary with the same radio range is pointless, as asymptotically no online algorithm can beat trivial flooding, i.e. using the whole graph as CDS.