Journal of Network and Computer Applications 35 (2012) 818–825

Contents lists available at SciVerse ScienceDirect

Journal of Network and Computer Applications

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

A power-saving data storage scheme for wireless sensor networks

Wen-Hwa Liao*, Hung-Chun Yang

Department of Information Management, Tatung University, No. 40, Section 3, Chungshan North Road, Taipei 104, Taiwan

ARTICLE INFO

Article history:
Received 14 May 2011
Received in revised form
6 October 2011
Accepted 9 November 2011
Available online 23 November 2011

Keywords:
Data storage
Power-saving
Wireless sensor networks

ABSTRACT

In wireless sensor network (WSN), sensors are small, inexpensive, and computable. However, they are limited in power, memory, and computational capacities. A large number of tiny sensors are usually deployed randomly to monitor one or more phenomena to collect and process the sensed data, and to send the data back to the sink. Many literatures focus on developing power-saving protocols. In addition, many papers present data storage schemes but they do not take power saving into consideration. Hence, these data storage schemes cannot perform well based on power-saving protocols. Therefore, it is very critical to propose a data storage scheme to support power-saving mechanism. In this paper, we propose a power-saving data storage scheme for WSN. Our scheme adopts grid-based architecture, in which each grid guarantees that two sensors will stay in active mode while the others stay in sleep mode to save energy. Simulation results show that our power-saving data storage scheme can reduce energy consumption.

1. Introduction

Wireless sensor network (WSN) is composed of large number of small sized, inexpensive and computable sensors, which are limited in power, memory, and computation. Normally, large numbers of tiny sensors are deployed randomly to monitor one or more phenomena, to collect and process the sensed data, and to send the data back to the sink. The important applications of WSN include environmental monitoring, personal healthcare, enemy monitoring, etc. In such applications, users are interested in events instead of the identity of a particular sensor. Energy consumption is the primary consideration in the design of a sensor network, with most related research focus on reducing energy consumption. The use of a sleeping mechanism is one of the most effective energy saving methods, but it is important to arrange the sleeping schedule without affecting the normal operation of the sensor network, especially the functions of the sensors. That is, the sensing coverage of the network cannot be reduced too much; otherwise, the network will not function normally.

Since a large amount of data is often distributed in entire sensor networks and is generated over a period of time, collection of data across the sensor networks is a critical problem when dealing with data storage. Recently, an efficient scheme for data storing and processing has been proposed, called the "data-centric storage (DCS) scheme (Le et al., 2006; Ratnasamy et al., 2003)". This scheme outperforms other approaches such as external storage (ES) (Wu et al., 2006) and local storage (LS) (Xia et al., 2006) in certain conditions. In ES, sensor nodes send relative data to a sink outside the sensing field while they still detect data. This may consume a lot of energy since each node has to deliver its sensing data to the sink. In LS, each node stores its sensing data in itself. As the sink requests the specified information, it floods queries throughout the sensor network. Although this storage scheme can reduce energy consumption when sending data back to the sink, it also increases the communication cost of the flooding queries. In the data-centric scheme, an event is stored in nodes within the network, each with a "name". Therefore, all data with the same names are stored in the same nodes. Queries for a particular kind of events are routed to the appropriate network node, where the relevant data (or pointers to that data) can be found. Storing the data by their name provides a logical rendezvous mechanism between data and queries so that the queries need not be flooded.

Many literatures focus on developing power-saving protocols (Dam and Langendoen, 2003; Jiang et al., 2003; Liao and Wang, 2008; Ye et al., 2002; Zheng et al., 2005). In addition, many papers present data storage schemes (Albano et al., 2007; Aly et al., 2008; Demirbas and Lu, 2007; Luo et al., 2007; Liao and Chen, 2010; Liao et al., 2010), but they do not take power saving into consideration. However, these data storage schemes cannot perform well without considering power-saving schemes. It is very critical to propose a data storage scheme to support power-saving mechanism. Hence, we propose an energy-efficient data storage scheme in this paper.

The rest of this paper is organized as follows. Section 2 discusses related work in power saving and data storage for WSNs. Our power-saving data storage scheme for WSNs is

* Corresponding author. Tel.: +886 2 25925252x3608; fax: +886 2 25853966.
E-mail address: whliao@ttu.edu.tw (W.-H. Liao).

1084-8045/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.jnca.2011.11.015

presented in Section 3. Section 4 presents the performance analysis. Section 5 presents the results of performance evaluation. Section 6 concludes this paper.

2. Related work

The S-MAC (Ye et al., 2002) is a MAC protocol in WSN. Its main design is to periodically put the sensors into a sleeping mode to replace the problem of idle listening. Even though compared with 802.11 (LAN MAN Standards Committee of the IEEE Computer Society, 1999) S-MAC can save more energy, its fixed duty cycle does not respond very well to the adaptation of network traffic because no matter how the environment changes, the protocol will not change the mode of transfer. The Timeout-MAC (TMAC) (Dam and Langendoen, 2003) protocol adjusts the operation cycle of the sensors appropriately; when a sensor does not have any action after time T_A, the sensor will re-enter the sleeping mode. This design solves the problem for the length of time to select for an operation cycle. Therefore, under the same traffic condition, TMAC can have the same efficacy performance, but consumes less energy than S-MAC. According to the design concept of Pattern-MAC (PMAC) (Zheng et al., 2005), the sensor can generate an appropriate sleeping schedule based on its own data transfer condition. The results of numerous experimental analysis and verification showed that this kind of design, when compared to S-MAC, indeed produces higher data throughput and saves more energy. The AMAC (Liao and Wang, 2008) has good adaptation ability to environmental change. In WSN, the time asynchronous operation mode is closer to the spirit of disperse computation. It not only requires the consideration of the effect of time deviation on the entire network operation, but also eliminates the execution of extra synchronization mechanism for time synchronization.

There are several previous works propose for the data storage in WSN. In Ratnasamy et al. (2002), a data-centric storage scheme GHT is proposed. GHT is a geographic hash table system that hashes an event type into geographic location and stores the data of the event at the sensor node that is closest to the hash location. In Tamishetty et al. (2004), they adapt multiple hash locations to store data that makes each key different by combining its event type and resilient level. It makes the same event type with different locations and uses each hash location as the centroid of the geometric shape based on a mathematical calculation. In Aly et al. (2005), zone sharing (ZS) is a distributed scheme for the decomposition of storage hotspots. ZS presents the solution in the context of the distributed index for distributed index for multi-dimensional data (DIM) scheme (Li et al., 2003). The benefit of DIM's structure is that it stores data with similar attributes nearby and enables energy-efficient query resolution. When a hotspot arises in a set of sensor nodes, those nodes falling on the border of the hotspot will transfer some loads of their zones to some of their less-loaded neighbors. It will send information to notify its neighbors and increase the cost of communication. Although ZS can distribute the heavy loads of a sensor node to other nodes, it will incur extra overhead and destroy the DIM structure. In Le et al. (2006), they propose a dynamic geographic hash table for data-centric storage. A temporal-based hashing function redefines the original basic operations of GHT and combines the timeslot. For each timeslot, each event is mapped into a grid cell. For the next timeslot, each event will be mapped into another grid cell. Thus, their method can achieve load balancing. The concept of node contribution potential is used to avoid situations where events are mapped into locations where surrounding nodes do not have enough resources to service the networks. However, the contribution potential value and the coordinate of the node with the highest potential in a cell must send back to the sink. The sink selects a set of cells with potentials above a certain threshold as the location set. Hence, it may cause a lot of overhead. In addition, as all nodes must be time synchronized, it also needs extra overhead. In Liao and Wu (2008), they propose an effective hotspot storage management mechanism to solve the hotspot storage problem. Their schemes include cover-up and multi-threshold mechanisms. The cover-up mechanism can adjust to another storage node dynamically when a storage node becomes full and the multi-threshold mechanism can spread data into several storage nodes to get load balancing among sensor nodes. There are some papers (Cheung and Maxemchuk, 2007; Wang et al., 2008) that discuss data compression. In Wang et al. (2008), they propose a multi-resolution data compression and storage in WSN. The sensor nodes are arranged into multiple layers. Messages transmitted by nodes in the upper layer are compressed by a certain node in the upper layer through the spatial coding technology. In Cheung and Maxemchuk (2007), they consider a specific application that are characterized by uneven distribution of information content, with useful information concentrated in isolated target regions. They identify the target regions by sampling sparsely at a low energy cost, and then raise the sampling density progressively only in those regions of interests until the desired resolution is reached. These coding processes will consume computation resources. In Madsen et al. (2008), they improve the reliability of data retrival using a full copy of the measurement of a sensor distributed to other sensors in the cluster. The data are split into many parts, and disseminated into a network based on the Reed–Solomon (RS) coding. However, the amount of the data increases using RS coding.

3. The power-saving data storage scheme

In this section, we describe the basic idea behind our proposed power-saving data storage scheme. Firstly, we describe the data query problem in a power-saving environment and then present the basic idea of our scheme. Secondly, we propose a snake-like power-saving scheduling. Thirdly, we describe the procedure of storing data. Finally, we describe our query mechanism.

3.1. Basic idea

Our power-saving data storage scheme is based on a grid-based architecture (Liao et al., 2001; Liao and Wu, 2008). Each grid can communicate with its neighboring grids. When a node detects an event, it uses the event type to compute a geographic location L by hash function. The hash location L of each event is mapped into a grid and the hash location L is transformed into a grid ID. Intuitively, at least one sensor should be awoke to store the data in the grid, which is reasonable in the data storing phase. However, a critical problem will occur in the querying phase. The problem is that the active node may not store the querying data by users. Hence, we design a power-saving mechanism that will guarantee that two sensors will stay in active mode simultaneously in a grid. The two active sensors are responsible for querying data. When querying is not finished completely, one active sensor A will transfer its query data to another active sensor B. Then the active sensor A will go into sleep mode. In the mean time, another sensor C will wake up. The sensor B will pass the querying task to sensor C. Both sensors B and C will continue to perform the querying task. Hence, our power-saving scheme guarantees not only two sensors in the active mode, but also one sensor still can be in active mode, while others go to sleep mode in the next time slot.