Journal of Network and Computer Applications 39 (2014) 191–201

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

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

Simulation framework for real-time database on WSNs

Ousmane Diallo [a,b], Joel J.P.C. Rodrigues [a,*], Mbaye Sene [b], Jaime Lloret [c]

[a] Instituto de Telecomunicações, University of Beira Interior, Portugal
[b] Department of Mathematics and Informatics, UCAD, Senegal
[c] Integrated Management Coastal Research Institute, Universidad Politécnica de Valencia, Spain

ARTICLE INFO

Article history:
Received 21 December 2012
Received in revised form
7 July 2013
Accepted 7 July 2013
Available online 18 July 2013

Keywords:
Wireless sensor networks
Real-time database management
Model
Simulator
Performance evaluation

ABSTRACT

Wireless sensor networks (WSNs) have been the focus of many research works. Nowadays, because of the time-critical tasks of several WSN applications, one of the new challenges faced by WSNs is handling real-time storage and querying the data they process. This is the real-time database management on WSN and it deals with time-constrained data, time-constrained transactions, and limited resources of wireless sensors. Developing, testing, and debugging this kind of complex system are time-consuming and hard work. The deployment is also generally very costly in both time and money. Therefore in this context, the use of a simulator for a validation phase before implementation and deploying is proved to be very useful. The aim of this paper is to describe the different specificities of real-time databases on WSN and to present a model for a simulation framework of the real-time databases management on WSN that uses a distributed approach. Then, the model of the simulator is described and developed in Java and a case study with some results demonstrates the validity of the structural model.

1. Introduction

Wireless sensor networks (WSNs) may be defined as a set of smart devices, called sensors, which are able to sense, process and transmit information about the environment on which they are deployed. These devices, distributed in a geographical area, collect information for users interested in monitoring and controlling a given phenomenon and transfer them to a sink node. The latter makes the information available to a gateway where remote users can access. So as to obtain those information, users use applications that communicate with the network through queries (Callaway, 2004; Sacks et al., 2003; Baronti et al., 2007).

Systems based on sensor networks are more and more used in many areas providing various types of WSNs (Mendes and Rodrigues, 2010). These different WSNs involved the development of many applications, which are generally connected to databases treating the amount of data collected from sensors. However, the processing time becomes increasingly critical for certain applications. These applications must query and analyze the data in a bounded time in order to make decisions and to react as soon as possible (Diallo et al., 2011). Some examples of the most popular applications are the following: the control of network traffic (Cranor et al., 2002), transactional analysis (web, banking or telecommunication transactions) (Cortes et al., 2000), human motion tracking application (Chen and Ferreira, 2009), the tracking of actions on dynamic Web pages (Zhu and Shasha, 2002; Chen et al., 2000), monitoring of urban or environmental phenomena (Mainwaring et al., 2002; Ulmer et al., 2003), and the sensors data management (Arasu et al., 2003).

Once the sensors perform their measurement, the problem of data storing and querying arises. Indeed, the sensors have restricted storage capacity (Silva et al., 2004) and the ongoing interaction between network devices and environment results in huge amounts of data.

There are two main approaches to data storage and querying in WSN: distributed and warehousing (Bonnet et al., 2000b). The first approach aims at exploiting the capacities of calculation of sensors. Some queries are distributed and evaluated among the nodes into the network. The objective is to locally calculate in order to limit sending of messages, reducing thus the energy consumption. In the second approach, warehousing, one has a centralized system. Collected data from sensors (in stream) are sent to a central database server, in which user queries are processed. Note that this last technique generates large data flows.

The data collected by the WSN must closely reflect the current state of the targeted environment. However, the environment changes constantly and the data are collected in discreet times. So, the collected data have temporal validity, as time advances they become less and less accurate, until the time where they do not reflect the state of the environment (Idoudi et al., 2009a, 2009b). It is fundamental that responses to application queries

* Corresponding author. Tel.: +351 27 524 2081; fax: +351 27 531 9899.
E-mail addresses: ousmane.diallo@it.ubi.pt (O. Diallo), joeljr@ieee.org (J.J.P.C. Rodrigues), mbaye.sene@ucad.sn (M. Sene), jlloret@dcom.upv.es (J. Lloret).

1084-8045/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.jnca.2013.07.001

ensure that returned data comply with logic and temporal constraints. In this context, real-time data management on WSNs is necessary to deal with those constraints. The main goal of real-time data management is to ensure temporal consistence data and process transactions within real-time constraints.

Developing, testing, and debugging this kind of complex system are time-consuming and hard work. The deployment is also generally very costly in both time and money. Therefore in this context, the use of a simulator for a validation phase before implementation and deploying is proved to be very useful (Sundani et al., 2010; Nasreddine, 2012; Egea-López et al., 2005).

The main contributions of this paper are the following:

- Description of the different characteristics of real-time databases management on WSN.
- Proposition of a model for a simulation framework of the real-time databases management on WSN that uses a distributed approach.
- Implementation of the model and performance evaluation.

The remainder of this paper is organized as follows: Section 2 presents the Wireless sensor networks. Section 3 discusses the related work, while the characteristics of real-time database management systems are exposed in Section 4. Section 5 briefly presents the data management on WSNs. In Section 6, the description of the real-time database simulation model is provided. In Section 7, some screenshots of the case study is presented to validate the model and Section 8 concludes the paper.

2. Wireless sensor networks

Generally a WSN has a large number of nodes distributed on an interest area and communicating between them so as to measure a physical quantity (e.g., pollution level in a given area) or to do an event monitoring (e.g., vehicles tracking). WSNs are used with different applications in many areas and are very important for applications that should be deployed in places hostile to human interventions (e.g., volcano monitoring). Each network node is considered smart and embeds these units: a sensor unit which provides a measure of environmental data (such as temperature, humidity, pressure, acceleration, sound, etc.), a processing unit, a storage unit, a communication unit, and an energy unit. The communication unit usually performs data transmission by means of radio (Akyildiz et al., 2002, 2007; Oliveira et al., 2011a; Oliveira and Rodrigues Joel, 2011b).

However, these resources are generally very limited, particularly those of storage and energy. The sensor network lifetime depends on the available energy in the nodes composing the network (Lin et al., 2011, 2012). This available energy is consumed by three activities (Akyildiz et al., 2002): sensing activity (data acquisition from the environment), communication (sending and receiving packets), which is essential to form a WSN, and data processing, which consists in some operations applied over data by smart sensors (Cho et al., 2001; Madden et al., 2005). However, the sensing and processing activities are much less expensive in energy consumption than the wireless communication activities (Madden et al., 2002). Therefore, saving energy by optimizing the communication activities is the main point attention in many algorithms designed for sensor networks.

Generally, proposed architectures for WSNs are based on the distribution of sensor nodes in a geographical area in such a way that sensors send collected data to a base station by using the routing techniques like in (Li et al., 2011); Al-karaki and Kamal, 2004).

3. Related work

A lot of simulation tools for WSNs have been already proposed. These simulators are generally designed to meet the development constraints (e.g. communication constraints, constraints on the nodes) related to WSNs and can be divided into classes according to the nature of the specified constraints.

The first class of simulators is the oriented network. They emphasize on the behavior of the wireless network and the protocol stack of the operation. These simulators are based on the computer network simulators designed for computers, such as NS-2 (Zhang et al., 2009), OMNeT (Xian et al., 2008; Varga, 2001), J-Sim (Sobeih et al., 2006), etc. Among these simulators it may be noted: SensorSim (Park et al., 2000), Castalia (Boulis, 2007, 2009), J-Sim sensor simulator (Sobeih et al., 2006), and Prowler (Simon et al., 2003). SensorSim is built on top of a NS-2 802.11 network model and it models a software model of the sensor node and a hardware model. The power models of the hardware components have been implemented and the state of the hardware model changes according to the function performed by the software model. Therefore, the power consumption of the network could be simulated. However, the CPU has not been implemented (Park et al., 2000). Castalia is an application-level simulator for wireless sensor network based on OMNeT++. It is parametric and can be used to evaluate various characteristics for specific applications. In Castalia, multiple simple modules can be linked together and form a compound module, e.g. a sensor node. The simple modules implement the atomic behavior of a model, for instance, the network stack layers of a sensor. Node modules are connected to wireless channel and physical process modules (Boulis, 2007, 2009). J-Sim has been developed at the University of Washington by the National Simulation Resource. It is a general-purpose network simulator modeled after NS-2, but unlike NS-2, J-Sim uses the concept of components, replacing the representation of each node as an object. J-Sim provides support for sensors and physical phenomena. Moreover, the energy modeling, except the radio energy consumption, is provided for sensor networks (Sobeih et al., 2006). Prowler is a probabilistic wireless sensor network simulator running on Matlab environment. It is an event-driven simulator that can be set to operate, either in deterministic mode, or in probabilistic mode in order to simulate the nondeterministic nature of certain component of the network such as the communication channel. Prowler is adapted to the development of algorithms and optimized protocols. However, it does not model the energy consumption of the sensor node (Simon et al., 2003).These simulators are oriented network and they do not deal with the databases management techniques, and particularly the real-time databases management techniques.

The second class of simulators is oriented node. They emphasize on the function of a single node with simple communication models. They are specific to targeted nodes and their operating systems. In addition, these simulators are used to verify the compatibility of a node with a given application. Among these simulators include: TOSSIM (Levis et al., 2003), ATEMU (Karir et al., 2004), and SEN (Sundresh et al., 2004). TOSSIM is a discrete event simulator for TinyOS (TinyOS, 2012) wireless sensor networks. TOSSIM can capture the behavior of the network of several TinyOS nodes at bit granularity. It is designed specifically for TinyOS/NesC applications to be run on MICA Mote platforms. With TOSSIM, designers can easily translate between running an application in the simulation environment and on motes (Levis et al., 2003). ATEMU is an instruction-level cycle-accurate emulator for WSN written in C language. It simulates programs of nodes with an accuracy clock cycle. ATEMU provides support for the AVR processor, and other function units on the MICA2 sensor node platform, such as the transceiver. Moreover, it provides users a