Information Fusion 15 (2014) 32–43

Contents lists available at SciVerse ScienceDirect

Information Fusion

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

Spatial query processing in wireless sensor networks – A survey

Rone Ilídio da Silva [a,b,*], Daniel Fernandes Macedo [b], José Marcos S. Nogueira [b]

[a] Universidade Federal de São João del-Rei, Campus Alto Paraopeba, Ouro Branco, MG, Brazil
[b] Universidade Federal de Minas Gerais, Belo Horizonte, MG, Brazil

ARTICLE INFO

Article history:
Received 25 April 2010
Received in revised form 8 May 2012
Accepted 23 August 2012
Available online 18 September 2012

Keywords:
Spatial query
Wireless sensor network
KNN
Window query

ABSTRACT

Wireless sensor networks (WSN) are particularly useful for obtaining data concerning events limited to a well-defined geographic region, such as a disaster site or a malfunctioning subsection of a factory plant. Such applications typically use spatial queries, which are SQL-like queries where location constraints are imposed on the collected data. Further, spatial queries allow changing the set of nodes (the region of interest) at runtime. This work surveys spatial queries in WSN. Due to the particular energy and resource constraints of WSN, spatial queries are performed by mechanisms having several stages, each of them implemented using localized distributed algorithms. This article categorizes the existing strategies for each stage, in order to ease the understanding of the state of the art. Finally, we analyze the most recent works on spatial query processing, identifying which classes of algorithms are used on each stage.

1. Introduction

Wireless sensor networks (WSN) are formed by many small devices that sense, process, store and transmit environmental data. These devices have limited power supply, and usually it is not feasible to replace their batteries because WSN may be deployed to monitor inhospitable environments [1]. As the technology matured, WSN became more than periodic data gathering tools. Now, they are data gathering tools able to answer users' queries [2]. Several works model WSN as distributed databases and define energy aware mechanisms for in-network query processing, such as TinyDB [3]. They process declarative SQL-Like queries, which simplify the way that the users obtain data from the WSN.

Recently, several researchers have worked on node location for WSN [4]. In-network query processing mechanisms can use location data to answer a special type of query called *spatial query*. In these queries the users' interests are expressed by geographical predicates, such as "the temperature collected by nodes in a region" or "the humidity collected by nodes closest to a point".

Spatial queries are database queries supported by geodatabases [5]. These queries differ from traditional queries in two main points. First, they incorporate geometry data types, such as points and polygons. Second, these queries consider the spatial relationship between the defined geometries, such as a point *inside* a polygon or a polygon that *overlaps* another. WSN can be modeled as distributed geodatabases in order to process queries containing spatial information. Those queries would request data collected inside a region (called *region of interest*) or nearest to a point defined by the user (called *query point*).

Thus, spatial queries in WSN usually rely on distributed and localized algorithms to reduce their resource usage and energy consumption: queries are forwarded towards the region of interest based on the neighborhood information of a few hops; the selection of the nodes to be activated for sensing is performed by the nodes inside the region of interest; finally, the sensed data is aggregated during the return path from the region of interest to the sink, hop by hop. Thus, information fusion is one of the steps performed in spatial query processing.

Due to the energy constraints of the sensor nodes, the collected data are processed during the forwarding to the user, using information fusion techniques to summarize them and to eliminate redundancy, in a process known as *in-network query processing* [6]. In-network processing techniques potentially reduce the energy consumption and traffic load on the network, however they place a heavy burden on the spatial query mechanisms, because they increase the penalties of packet losses and late deliveries. Thus, spatial queries must attempt to reduce the amount of packet losses, even though the network is prone to failures due to bad wireless links, node failures, energy depletion, among others.

This work presents a survey of spatial query processing in wireless sensor networks. We divide spatial query processing in six stages and describe the existing solutions for each stage. This division simplifies the understanding of spatial queries and facilitates the proposal of new solutions. We also survey the literature, categorizing the algorithms for each stage according to their most

* Corresponding author at: Campus Alto Paraopeba, Universidade Federal de são João del-Rei, Rodovia MG 443, KM 7, Ouro Branco, MG 36400-000, Brazil. Tel.: +55 31 37413280.

E-mail addresses: rone@ufsj.edu.br (R.I. da Silva), damacedo@dcc.ufmg.br (D.F. Macedo), jmarcos@dcc.ufmg.br (José Marcos S. Nogueira).

1566-2535/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.inffus.2012.08.006

important properties. Further, we present how each spatial query processing mechanism in the literature implements these stages.

This article is organized into five sections. Section 2 describes the six stages for spatial query processing. Section 3 presents how the spatial query processing mechanisms in the literature perform each stage. Section 4 presents the main contributions and drawbacks of spatial query processing mechanisms found in the literature. Section 7 discusses the importance of information fusion on spatial query processing. Section 6 presents the open research challenges in spatial query processing. Finally, Section 7 shows our conclusions.

2. Spatial query processing stages

In order to better study spatial queries, we will analyze them in stages. The related works define different numbers of stages for spatial query processing in WSN [7–10]. However, these stages can be subdivided into simpler stages, as proposed in our previous work [11]. We analyzed spatial queries in six stages: Pre-Processing, Forwarding, Dissemination, Aggregation, Sensing and Return. Fig. 1 illustrates them.

2.1. Pre-Processing stage

In the Pre-Processing stage, queries are formatted in order to be transmitted by the sensor nodes. It is performed in the user's computer, since usually this computer has more resources than the sensor nodes. Normally, queries are written in a declarative language based on SQL [3]. Thus, queries must be transformed into a sequence of bytes to be transmitted to the nodes. Moreover, packets transmitted by traditional WSN using 802.15.4 usually have at most 88 bytes [12]. As a consequence, this sequence must be as small as possible, since a big sequence may need several packets to be transmitted and, consequently, would consume more energy during its processing.

Before the Pre-Processing stage, the user must define the parameters of the spatial query. For example, it selects a region of interest, which collection method it will use (a snapshot of the environment or a continuous sampling) and its parameters (e.g. the periodicity among collections in continuous sampling). Those parameters will depend on the application and its requirements. Hence, the Pre-Processing stage must performs application-independent tasks, such as representing the information in a most suitable manner, in order to make the query more efficient and so it takes up less packets.

2.2. Forwarding stage

In this stage, queries are forwarded from the *Originator* (the first node to receive the query in the network) to the region of interest. This is the main difference between conventional query processing and spatial query processing in WSN. In conventional query processing, queries are disseminated to all nodes in the WSN by Flooding [13]. In spatial query processing, queries are first forwarded and then disseminated only to the nodes within the region of interest. Hence, the task of the Forwarding stage can be performed by adapting one of the routing protocols found in the literature [14].

Another important characteristic to be considered in Forwarding is node location. Spatial query processing mechanisms assume that nodes know their location. Location-based routing protocols use this information to create their routes, consuming less energy than the protocols that do not employ location information [14]. Thus, these protocols are usually the best choice for the Forwarding stage. Further, each protocol is adapted to a specific network mobility pattern. Hence, the spatial queries processing mechanisms need to choose the routing protocol that better adapts to the target network.

2.3. Dissemination stage

In the Dissemination stage, the query is disseminated to all nodes within the region of interest. Dissemination algorithms try to minimize the number of packets transmitted by the WSN in order to reduce the energy consumption. The Dissemination strategy defines the type of the spatial query. We found in the literature two types of spatial queries, called window query and k-nearest Neighbor query (KNN).

Window queries are the most common type of spatial queries. The user defines a region of interest (called *window*) and asks for data collected by the sensor nodes inside this region. Windows are defined by rectangles [15,8,16], circles [9] or polygons with irregular contours [11,17]. These queries are also called range queries[1] [8]. Fig. 2a illustrates a Window query in a WSN.

K-Nearest Neighbor queries (KNN) retrieve data from the K nodes closest to a point defined by the user, called *query point* [18–20]. Fig. 2b illustrates a KNN query. We also found in the literature a type of query called *Nearest Neighbor queries (NN)*. It is a special case of KNN, with $K = 1$ (such as [2]). Hence, we focus on KNN queries.

2.4. Sensing stage

In the Sensing stage, the nodes within the region of interest collect the data required by the query. The operation of this stage is heavily dependent on the application, and as such are not part of spatial query mechanisms. We identified three high level the tasks that are usually performed in this stage. These tasks are *raw data collection*, *local information fusion*, and *compression*.

The first task entails the collection of the data itself. This may involve the use of physical or virtual sensors [21]. In the field of context-aware systems, physical sensors are hardware that collect readings related to the environment, such as humidity and temperature, while virtual sensors may collect data referring to the current node's state, such as its remaining energy and processing capabilities. Physical readings are useful for monitoring and actuation, while virtual readings are often employed for management tasks, in order to configure the network according to its current operational state [22].

After the data has been collected from one or more sources, nodes may perform local information fusion. This is the first stage in spatial queries where information fusion may take place. The node may use the raw data to make inferences based on the readings of multiple sensors. Further, in applications that require a coarse spatial granularity of the data [23], the nodes may employ strategies to reduce the number of nodes that sense the environment without compromising the quality of the reported data [24].

Finally, the node may employ data compression strategies to reduce the size of the information, once the irrelevant data has been suppressed in the previous tasks. This tasks deals with information coding, rather than the content of the information to be transmitted. Techniques such as delta coding, Huffman's method and others could be employed [25].

2.5. Aggregation stage

In the Aggregation stage, the collected data are transmitted from the nodes in the region of interest to the *Aggregator*, the node that calculates the query result. In this stage, nodes are able to perform information fusion in the collected data. There are two possible information fusion strategies: naive and in-network processing. In a naive strategy, all the nodes within the region of interest

[1] In traditional database literature, range queries select data between an upper and lower boundary, however these boundaries may not be related to spatial constraints.