ISSN 1726-5479

SENSORS & TRANSDUCERS

3/12
vol. 14-2
Special

Physical and Chemical Sensors & Wireless Sensor Networks

International Frequency Sensor Association Publishing

IFSA

Sensors & Transducers

Volume 14-2, Special Issue
March 2012

www.sensorsportal.com

ISSN 1726-5479

Sensors & Transducers Journal (ISSN 1726-5479) is a peer review international journal published monthly online by International Frequency Sensor Association (IFSA).
Available in electronic and on CD.

Sensors & Transducers Journal

Contents

Volume 14-2
Special Issue
March 2012

www.sensorsportal.com

ISSN 1726-5479

Research Articles

Authors are encouraged to submit article in MS Word (doc) and Acrobat (pdf) formats by e-mail: editor@sensorsportal.com
Please visit journal’s webpage with preparation instructions: http://www.sensorsportal.com/HTML/DIGEST/Submition.htm

International Frequency Sensor Association (IFSA).

The 6th International Conference on Sensor Technologies and Applications

SENSORCOMM 2012

19 - 24 August 2012 - Rome, Italy

Deadline for papers: 5 April 2012

Tracks: Architectures, protocols and algorithms of sensor networks - Energy, management and control of sensor networks - Resource allocation, services, QoS and fault tolerance in sensor networks - Performance, simulation and modelling of sensor networks - Security and monitoring of sensor networks - Sensor circuits and sensor devices - Radio issues in wireless sensor networks - Software, applications and programming of sensor networks - Data allocation and information in sensor networks - Deployments and implementations of sensor networks - Under water sensors and systems - Energy optimization in wireless sensor networks

http://www.iaria.org/conferences2012/SENSORCOMM12.html

The 3rd International Conference on Sensor Device Technologies and Applications

SENSORDEVICES 2012

19 - 24 August 2012 - Rome, Italy

Deadline for papers: 5 April 2012

Tracks: Sensor devices - Ultrasonic and Piezosensors - Photonics - Infrared - Geosensors - Sensor device technologies - Sensors signal conditioning and interfacing circuits - Medical devices and sensors applications - Sensors domain-oriented devices, technologies, and applications - Sensor-based localization and tracking technologies

http://www.iaria.org/conferences2012/SENSORDEVICES12.html

The 5th International Conference on Advances in Circuits, Electronics and Micro-electronics

CENICS 2012

19 - 24 August 2012 - Rome, Italy

Deadline for papers: 5 April 2012

Tracks: Semiconductors and applications - Design, models and languages - Signal processing circuits - Arithmetic computational circuits - Microelectronics - Electronics technologies - Special circuits - Consumer electronics - Application-oriented electronics

http://www.iaria.org/conferences2012/CENICS12.html

Sensors & Transducers Journal, Vol. 14-2, Special Issue, February 2012, pp. 1-17

Sensors & Transducers

ISSN 1726-5479

http://www.sensorsportal.com

Information Extraction from Wireless Sensor Networks: System and Approaches

Tariq ALSBOUI, Abdelrahman ABUARQOUB, Mohammad HAMMOUDEH, Zuhair BANDAR, Andy NISBET

School of Computing, Mathematics & Digital Technology, Manchester Metropolitan University, Manchester, UK

E-mail: {tariq.al-sboui, abdelrahman.m.abuarqoub}@stu.mmu.ac.uk, {m.hammoudeh, z.bandar, a.nisbet}@mmu.ac.uk

Received: 27 November 2011 /Accepted: 20 December 2011 /Published: 12 March 2012

Abstract: Recent advances in wireless communication have made it possible to develop low-cost, and low power Wireless Sensor Networks (WSN). The WSN can be used for several application areas (e.g., habitat monitoring, forest fire detection, and health care). WSN Information Extraction (IE) techniques can be classified into four categories depending on the factors that drive data acquisition: event-driven, time-driven, query-based, and hybrid. This paper presents a survey of the state-of-the-art IE techniques in WSNs. The benefits and shortcomings of different IE approaches are presented as motivation for future work into automatic hybridization and adaptation of IE mechanisms.

Keywords: Wireless sensor networks, Information extraction, Event-driven, Time-driven, Query-based.

1. Introduction

The main purpose of a WSN is to provide users with access to the information of interest from data collected by spatially distributed sensors. In real-world applications, sensors are often deployed in high numbers to ensure a full exposure of the monitored physical environment. Consequently, such networks are expected to generate enormous amount of data. The desire to locate and obtain information makes the success of WSNs applications, largely, determined by the quality of the extracted information. The principal concerns when extracting information include the timeliness, accuracy, cost, and reliability of the extracted information and the methods used for its extraction. The process of IE enables unstructured data to be retrieved and filtered from sensor nodes using sophisticated techniques to discover specific patterns [1, 2]. Practical constraints on sensor nodes, such as power consumption, computational capability, and maximum memory storage, make IE a challenging distributed processing task.

In terms of data delivery required by an application, IE in WSNs can be classified into four broad categories: event-driven, time-driven, query-based, and hybrid. In event-driven, data is only generated when an event of interest occurs, while, in the time-driven, data is periodically sent to a sink every constant interval of time. With query-based, the data is collected according to end user's demand. Finally, the hybrid approach is a combination of one or more of the above.

The rest of this paper is organized as follows: Section 2 identifies what types of information need to be reported to end users. Section 3 looks at event-driven IE approaches and presents sample developments. Section 4, describes time-driven IE and recent successful deployments. Section 5, describes query-based IE and present some of the recent approaches. Section 6, describes and identify recent advances in hybrid IE methods. Section 7, describes recent approaches to IE from mobile WSNs. In section 8, a summary on future research direction for IE is discussed. Section 9, presents a model for accommodating and integrating the diverse IE approaches and mechanisms. Section 10 concludes this paper.

2. What Needs to Be Reported?

IE is one of the most vital efforts to utilize the ever burgeoning amount of data returned by WSNs for achieving detailed, often costly task of finding, analyzing and identifying needed information. The process of IE involves the classification of data based on the type of information they hold, and is concerned with identifying the portion of information related to a specific fact. In the context of WSNs, the notion of fact can be defined as a property or characteristic of the monitored phenomenon at a certain point in time or during a time interval. Fact can also refer to an event or action. An event is a pattern or exceptional change that occasionally appears in the observed environment [3]. Events have some distinct features that can be used as thresholds, e.g. temperature > 50, to make a distinction between usual and unusual environmental parameters.

An event may arise in many other forms. It can be a continuous, gradually occurs over time (e.g. temperature does not change instantly), and has obvious limit with normal environment parameters. In [4] complex events are defined as sequences of sensor measurements over a period of time indicating an unusual activity in the monitored environment. In WSNs, the network owners may be unaware in advance what type of events may occur. This is because one of the ultimate goals of such networks is to discover new events and interesting information about the monitored phenomenon. For this reason, threshold-based event detection methods are not always efficient to identify and extract event-based facts. From this deficiency arise the need for periodic, query-based, and hybrid IE approaches.

Events can be further classified into two categories: system events and environmental events. System events are concerned with architectural or topological changes, e.g. a mobile node entered a cluster area. Environmental events are concerned with the occurrences of unusual changes across the monitored environment, e.g. spotting a moving target [4].

Nodes organization plays an important role in IE because it defines, among other factors, the cost (amount of energy required to collect raw data), accuracy (level of coverage), reliability (e.g. timeliness) of extracted information. The organization of nodes can be either centralized or hierarchical. In the centralized approach, data collected by all nodes are sent towards a sink node using single or multi-hop communication [5]. However, this approach does not provide scalability, which is a main design factor for WSN. Also, it causes communication bottlenecks and transmission delays due to congestions especially in areas around the sink [6]. To overcome the problems in the centralized approaches, hierarchical techniques have been proposed as an effective solution for achieving longer network lifetime and better scalability.

Since the number of existing IE approaches is significantly large, it will not be feasible to provide a detailed description of each approach. Instead, we have selected recent approaches that particularly represent directions of future research without focusing on the details of these approaches. However, characteristics of various approaches that are common for the approach they apply will be presented. Table 1 lists the reviewed approaches and some older approaches for the more interested readers. In Sections 3 to 6 the approaches are presented based on the categorization so as related sub-categories are discussed in the common context. To make the analysis of different approaches more logical and to set up a common base for their comparison and connection we consider some qualitative criteria.

Table 1. Overview of the selected approaches to IE.

Event-driven	[2-3, 7-15]
Time-driven	[6, 13-19]
Query-based	[21-34]
Hybrid	[38-42]

3. Event-driven IE

3.1. Description and Operation

In event-driven approaches to IE, the initiative is with the sensor node and the end user is in the position of an observer, waiting for incoming information. Any node may generate a report when a significant event (e.g., a change of state) or an unusual event (e.g., fire) occurs. Event-driven is a valuable tool for detecting events as soon as they occur. In the simplest form, sensor nodes are preconfigured with threshold values that when exceeded indicate an event.

Event-driven approaches incur low power consumption and require low maintenance. Among the benefits of this class of approaches are: they reduce the amount of communication overhead by applying local filtering on collected data to determine whether to send new data or not; they implement local mechanisms to prevent multiple nodes reporting the same event; they exploit redundancy to reduce the number of false alarms; they allow timely responses to detected events; they are easy to implement and configure; they allow distributed processing at the node level or within a group of node to collaboratively detect an event; and they are suitable for time critical applications, e.g. forest fire monitoring or intrusion detection.

However, there are a number of limitations to the event-driven IE. First, it is difficult to capture events of spatio-temporal characteristics. Second, detecting complex event may require non trivial distributed algorithms, which require the involvement of multiple sensor nodes [7]. Third, due to the fact that events occur randomly, some nodes generate higher rates of data than other nodes. This will lead to unbalanced workloads among sensor nodes. Fourth, it is not suitable for continuous monitoring applications, where sensed measurements change gradually and continuously. Finally, due to sensors measurement inaccuracies, event-driven approaches may potentially generate false alarms.

3.2. Event-driven Approaches

In earlier studies, events were detected with a user-defined threshold values [8, 9]. In such approaches, sensor nodes are preconfigured with a static threshold value. When the sensor node reading deviates from the pre-defined thresholds, this indicates an event that triggers the node to convey its data back to the sink. To overcome some of the inherent problems in the threshold-based event-detection, [3] have adopted the infrequent pattern discovery technique and developed a function for detecting events of interest in the monitored environment. The function is split into two phases, learning phase and event detection phase. In the learning phase, the function will learn the frequent changes from the measurement series. In the event detection phase, a new pattern is built using the incoming measurement values in which a decision is made on whether the new pattern is frequent (e.g., system events) or infrequent (e.g., event). Then the infrequent changes will be reported to the fusion centre as potential events. The proposed approach reduces the number of transmission, which extends the network life time. It is also characterized by low computational and time complexities. Due to its distributed nature the approach is also scalable. However, high false alarms will be generated due to sensor measurement inaccuracies. Also, it is incapable of capturing complex events, because there is a spatial and temporal correlation in these types of events that requires more complex rules. Finally, it does not have any mechanism to distinguish between errors and events.

In [10], the authors presented a data fusion tool to increase the resilience of event detection techniques. They introduced two levels for event detection: at the first level, each sensor node will individually decide on detecting event using a classifier (naive bayes). At the second level, fusion technique is placed at higher level (e.g., cluster head) and used to distinguish between outliers. Outliers are measurements that differ from the normal pattern of sensed data occurring at individual nodes and events that more nodes agree upon [11]. This approach reduces the number of false alarms, since cluster heads are able to distinguish between anomalies and event. However, processing data at the cluster head introduces delays in reporting an event. Moreover, the efficiency of the approach depends on the efficiency of cluster formation methods. For instance, many clustering algorithms result in energy-unbalanced clusters.

Another threshold-based approach proposed in [12] introduced a double decision mechanism. A sensor may decide about the presence of an event either directly or by asking for additional data from nearby nodes. This approach minimizes the energy consumption since the latter step is activated only when it is needed. There is no need for fusion centre to process the data as a fixed number of nodes will take the responsibility to make decisions about the occurrences of an event. However, it is always difficult to determine node's neighbours. Although these approaches can reduce communication overhead and report events promptly, however, it is difficult to define the optimal threshold values.

More advanced approaches, such as SAF [13] and Ken [14], exploit the fact that physical environments frequently exhibit predictable stable and strong attribute correlations to improve compression of the data communicated to the sink node. The basic idea is to use replicated dynamic models to reflect the state of the environment being monitored. This is done by maintaining a pair of dynamic probabilistic models over the WSN attributes with one copy distributed in the network and the other at the sink. The sink computes the expected values of the WSN attributes according to the defined prediction model and uses it to extract information. When the sensor nodes detect anomalous data that was not predicted by the model within the required certainty level, they route the data back to the sink. This approach is subject to failure as basic suppression. It does not have any mechanism to distinguish between node failure and the case that the data is always within the error bound. Ken is not robust to message loss; it relies on the Markovian nature of the prediction models to presume that any failures will eventually be corrected with model updates, and the approximation certainty will not be affected by the missed updates. They propose periodic updates to ensure models can not be incorrect indefinitely. This approach is not suitable for raw value reconstruction; for any time-step where the model has suffered from failures and is incorrect, the corresponding raw value samples will be wrong. Finally, as the approach presented in [15], SAF and Ken can only handle static network models.

A decentralized, lightweight, and accurate event detection technique is proposed in [16]. The technique uses decision trees for distributed event detection and a reputation-based voting method for aggregating the detection results of each node. Each sensor node perform event detection using it is own decision tree-based classifier. The classification results, i.e. detected events, from several nodes are aggregated by a higher node, e.g., a cluster head. Each node sends it is detected events, called detection value, to all other nodes in it is neighbourhood. The detection value will be stored in a table. Finally, tables are sent to the voter (e.g. cluster head), which in turns decides to make a final decision among different opinion. The decision tree approach provides accurate event detection and characterized by low computational and time complexities. However, similar to [9], the processing of data at the cluster head will introduce further delays in reporting an event.

4. Time-driven IE

4.1. Description and Operation

In time-driven approaches to IE, a sensor node periodically generates a report from the physical environment to give the end-user its current status. The reporting period may be preconfigured or set by the end-user depending on the nature of the monitored environment and applications requirements.

Time-driven approaches have the ability to enable arbitrary data analysis, they provide continuous monitoring of the sensor network to reflect environmental changes, they scale to handle millions of nodes (through aggregation), they extend network life time by sending nodes to sleep between transmissions, they can reduce congestion and improve system reliability by scheduling nodes to transmit at different times, they explicitly incorporate resource capacity, and highlights unused resources. However, there are a number of limitations to the time-driven approaches. First, they are limited to specific set of applications where consistent changes occur across the network, e.g. agricultural applications. Second, a large portion of the returned data might be redundant and not useful for the end-user thereby resulting in wastage of resources. Third, nodes have to maintain global clock and deal with synchronization issues. Finally, it is extremely difficult to define optimal time intervals.

4.2. Time-driven Approaches to IE

In time-driven IE, most of the published work in the literature is based on probabilistic models that attempt to predict the next value that the sensor is expected to acquire. For example, Ken's [14] model exploits the spatio-temporal data correlations while guaranteeing correctness. It involves placing a dynamic probabilistic model on the sensor node and on the sink, and these models are always kept in synchronization for periodic updates. An approach similar to Ken was proposed in [17]. In contrast to Ken, the approach exploits only the temporal correlation of sensed data and is based on the Auto Regressive Integrated Moving Average prediction model. It places the model on the sensor nodes and the aggregator nodes (cluster head) to predict the next values. These models are always kept in synchronization for periodical updates. The approach is energy efficient since the number of transmitted massages is reduced. However, the forecasting can be badly distorted by outliers leading to wrong prediction.

Similar approaches to [14, 17] have been suggested in [7, 18]. These approach use dynamically changing subset of the nodes as samplers where the sensor readings of the sampler nodes are directly collected, while, the values of non sampler nodes are predicted through probabilistic models that are locally and periodically constructed. All approaches in [7, 14, 17, 18] save energy by reducing the number of transmitted messages. However, the additional cost to maintain models synchronized is not negligible.

Another approach called Cascading Data Collection (CDC) is presented in [19]. In CDC only a subset of sensor nodes are selected randomly to periodically transfer data back to the sink node. The mechanism is distributed and only utilizes local information of sensor node. The CDC reduces energy consumption by choosing a subset of sensor nodes to periodically transmit readings back to the sink. However, the CDC uses packet aggregation at an intermediate node, which introduces undesirable communication delays. The work presented in [20] takes CDC one step further by enabling each node to use its local and neighbourhood state information to adapt its routing and MAC layer behaviour.

5. Query-based IE

5.1. Description and Operation

Query-based approaches to IE, typically involve request-response interactions between the end-user or application components and sensor nodes. End users issue queries in an appropriate language, and then each query is disseminated to the network to retrieve the desired data from the sensors based on the description in the query.

Query based approaches provide a high level interface that hides the network topology as well as radio communication from end users. Queries can be sent on demand or at fixed intervals. They provide a solution if the data needs to be retrieved from the entire network.

However, there are a number of limitations to the query-based approaches. First, most of existing query languages do not provide suitable constructs to easily articulate spatio-temporal sense data characteristics. Second, it is difficult to formulate queries using current languages that represent higher level behaviour, or specify a subset of nodes that have significant effect on the query answer. This may result in generating large amount of data of which big portion is not useful for the end user. Third, to the best of our knowledge, there is no published work that fully exploits all the potentials of different heterogeneous resources in WSN applications in a context-aware manner. Forth, approaches that take a database view of the network are inclined more towards the extraction of the reactive behaviour of the WSN and suggestions were made that the active database should be viewed as two end-points of the range of rule-based languages in databases [21]. Finally, though declarative languages came into view in WSNs settings, the trigger that are the fundamental means for specifying the reactive behaviour in a database have not yet been maturely developed.

5.2. Query-based Approaches to IE

Query-based approaches, applies techniques used in traditional database systems to implement IE in WSNs. A query is sent to the network and data is collected according to the description in the query. COUGAR [22] was the first project that attempted to introduce the concept of WSN as a distributed database. It allows the end user to issue a declarative query (SQL) for retrieving information. The authors introduced a query layer between the application layer and the network layer. The query layer comprises a query proxy, which is placed on each sensor node to interact with both the application layer and the networking layer. The goal of the query proxy is to perform in-network processing. In-network processing increases efficiency by reduces the amount of data that needs to be sent to the gateway node. The user does not need to have knowledge about the network, or how the data is retrieved or processed. However, COUGAR is incapable of capturing complex events, e.g. of spatio-temporal nature, or a produce queries that targets only a subset of the network [23].

A similar approach to COUGAR is proposed in [24]. TinyDB is a query processing system, which extracts information from the data collected by the WSN using the TinyOS operating system. TinyDB maintains a virtual database table called SENSORS. It disseminates the queries throughout the network by maintaining a routing tree (spanning tree) rooted at the end point (usually the user's physical location). Every sensor node has its own query processor that processes and aggregates the sensor data and maintains the routing information. TinyDB is extensible and complete framework with effective declarative queries. In-network processing reduces the amount of data that is required to be sent to the sink, thus, energy consumption is reduced. However, data does not include a geo-referencing of sensor nodes for spatial quires and it imposes tight correlation among routing and queries.

In [25], a new IE algorithm is proposed with the aim of reducing energy consumption by focusing on selective aggregate queries. The proposed algorithm, named Pocket Driven Trajectories (PDT), deals with queries that aggregate data only from a subset of all network nodes. PDT is based on the logical assumption that spatial correlation in sensor values coupled with query selectivity gives rise to a subset of participating nodes formed by one or more geographically clustered sets (pockets). The algorithm starts by discovering the set of pockets for a given query. Then, the aggregation tree to the spatially optimal path connecting these pockets is aligned. Targeting only nodes with interesting data makes PDT energy efficient and suitable for large scale WSNs. However, PDT introduces a delay in reporting data to a sink, because data is processed at an intermediate node. Moreover, the formation of pockets is not a trivial task.

The authors in [26], proposed a query processing algorithm, that allows the user to specify a value and time accuracy constraints based on an optimized query plan. Using these optimization constraints, the algorithm can find an optimal sensing and transmission of attribute readings to sink node. Rather than sending sensors readings directly to the sink, the proposed algorithm report only updates. This results in considerable reduction in communication costs. However, the algorithm does not support dynamic adjustment of accuracy constraints.

More recently in [27], the authors designed and implemented a distributed in-network query processing, called Corona. Corona is composed of three components: the query engine that is executed on the sensors; a host system on the client's PC that is connected to the sink; and GUI that is connected to the host system via TCP/IP. The Corona query processing provides multi-tasking capabilities by running multiple queries concurrently, which in turns reduces processing delays and communications cost by applying data aggregation. However, the language can not easily capture spatio-temporal events even though most of the processing is done centrally at the client PC.

5.3. Macroprogramming

Macroprogramming is an application development model for WSNs, where the programmer specifies the global behaviour of the system instead of specifying the behaviour of its individual nodes. Over the past few years, a number of macroprogramming systems have been proposed in the literature [28-34]. We refer the interested reader to the recent survey [35] and the references therein for a comprehensive review of older macroprogramming approaches. A number of abstraction levels have been proposed in the literature ranging from a database or node-level, group, or semantic data streams. The data within the network is accessed through these different abstractions that offer access through logical groups of nodes.

Regiment [33] is one of the early macroprogramming systems where the network is represented as time varying signals. Signals might represent one node sensor readings, the computational state of that node, or aggregate values based on multiple source signals. Streams of data can be grouped into regions. A region is defined as a collection of spatially distributed signals, e.g. the collection of nodes in a geographic area with a specific range of sensor readings. Therefore, Regiment is capable of representing groups of nodes with topological, geographical, and logical relationships. In all cases, the region abstracts away the details of data collection, storage, and dissemination from the programmer.

Similar to Regiment, Kairos [36] is another high level abstraction that provides abstractions for manipulating global behaviour. Kairos provide programmers with three levels of abstractions: Node abstraction, where programmers use nodes integer-based identifier to control single nodes or lists of nodes; List of direct neighbours, this is similar to the region abstraction and express the natural construct of radio neighbourhood; Remote data access, gives programmers direct access to variables at a specific node.

Unlike Regiment, Kairos is language independent, which makes it possible to be deployed as a plug-in to existing declarative programming languages. However, the distributed computation in Kairos covers the whole network but uses a centralized method. In attempt to reduce communication overhead, Kairos utilizes inefficient consistency model called "eventual consistency". Compared to Regiment, Kairos provides a small number of constructs for its three levels of abstractions, whereas, Regiments offers a bigger set of operations and data types. These systems are tailored for rather narrow range rang of target applications, e.g. target tracking or composing in-network data-processing services.

Logical Neighborhood [34] is a virtual node programming abstraction. A node is characterized by several exportable attributes that can be static or dynamic. These attributes are used to replace the notion of physical neighbourhood (defined by wireless broadcast) with the notion of logical neighbourhood (defined by application conditions). The span of a logical neighbourhood is defined declaratively based on the state of the node and communications costs. These logical neighbourhoods are implemented using SPIDEY programming language and are supported by a special routing strategy enabled by SPIDEY programming constructs. The authors of Logical Neighbourhood assure that their approach can be on top of any routing protocol. However, they confirm that the utilization of other routing mechanisms to support their abstractions will result with various performance drawbacks. Another limitation of this approach is that the user is expected to manually declaratively specify using the SPIDEY language which nodes to consider as neighbours. This is not always easy to achieve; the configuration of various parameters requires the developers to have deep understanding of the nature of data being collected and good knowledge of the network topology which might change frequently. Therefore, an approach that places the intelligence of defining the span of communication is desired.

Pathak et al. [29] presented a data-driven macroprogramming framework to solve the problem of initial placement of high-level task graph representation onto the nodes of the target network. Their framework adopts the view of data-flow to model this task-mapping function in the context of WSNs macroprogramming. Srijan [31] is another graphical toolkit designed to aid application developers in the compilation of the macroprogram into individual customized runtimes for each constituent node of the target network. These frameworks are centralized in nature because the sink knows the location and initial energy levels of all sensor nodes. Also, adaptation via code updates decisions is made by the developer at a centralized gateway. Moreover, the dissemination of the code from the gateway to the WSN nodes is performed via the gateway, which is an energy-intensive process.

Hnat et al. [30] implemented a macroprogramming compiler for MacroLab [32]. Their compiler separates function decompositions from low-level compiler optimizations, which makes it expandable and modular. MacroLab programming abstractions are based on the notion of macro vector where one dimension in the vector is indexed by node identifiers. The system-wide or global computation scope provided by MacroLab limits the efficiency of the new compiler. However, a logical segmentation of nodes can be used to partition the large macro vector to carry only data of logically-neighboring nodes.

The Watershed Segmentation approach [28] groups nodes sharing some common group state into segments. A segment is described as a collection of spatially distributed nodes with an example being the set of nodes in a geographic area with sensor readings in a specific range. The logical segments of nodes generated by the Watershed algorithm replace the tight physical neighbourhood provided by wireless broadcast with a higher level, application defined abstractions. Segments are created such that the span of a logical neighbourhood is specified dynamically and declaratively based on the attributes of nodes, along with requirements about communication costs (specifically the diameter of the segment). A network segment formed with a logical notion of proximity determined by applicative information is, therefore, capable to return specified information with high confidence. Differently from [30, 31, 34], segments memberships are dynamically updated in a distributed manner without developers intervention, attaining grater robustness and higher efficiency. The advantages of Watershed Segmentation approach include: dynamic construction and updating of segments; no need for compilers or new programming constructs; it cater for node level communication; localized computation; segment setup considers the physical topology provided by wireless broadcast; and the spam of the segment improves response accuracy while reducing the processing cost.

6. Hybrid-based IE

6.1. Description and Operation

A hybrid IE approach is an approach that combines two or more algorithms from different IE categories. Hybrid approaches aim to overcome some of the disadvantages of individual IE categories described above.

6.2. Hybrid Approaches to IE

Many hybrid approaches to IE have been recently proposed in the literature. In [37], the authors proposed a hybrid protocol that adaptively switches between time-driven and event-driven data collection. A sensor node is triggered to detect an event of interest, and from the point when an event detected to the point when the event becomes no longer valid, the protocol switches to behave as a time-driven protocol. During this period sensor nodes continuously report data to the sink. This protocol reduces unnecessary data transmission and minimizes event notification time. However, it is not guaranteed to work well for all applications.

More recently, in [38], the authors proposed a hybrid framework, similar to [7, 14], which deploys both event-driven and query-based approaches to IE. The idea is to process continuous "group-by" aggregate queries, and to allow each sensor node to check whether sensor readings satisfy local predicates based on predefined thresholds. Then, nodes send only data that satisfy local predicates to their cluster heads, which in turns process the data to answer the query as accurate as possible. The proposed hybrid framework is able to target a subset of the network by using the group-by clause. It reduces communication cost by using one dimensional Haar wavelets. However, it introduces a delay in reporting events since the data is processed at the cluster head. Moreover, this approach requires the extension of existing query languages to include the group-by clause and the associated language parser.

The authors in [39] proposed a hybrid framework that deploys both event-driven and time-driven approaches. The idea is to use Complex Event Processing (CEP) to find meaningful events from raw data. As soon as a node detects an event, local data is continuously transferred towards the sink for further processing. This approach provides continues monitoring of the environment, which enables arbitrary data analysis. However, using CEP, data has to be sent to a central location for integration with data from other sources. It does not support in-network processing, which has been identified as an efficient technique for resource utilization [40].

7. IE from Mobile WSNs

Recent applications of WSNs (e.g. in medical care) make use of mobile sensor nodes to improve their performance. However, mobility poses new challenges to IE researchers including: increased data loss and delivery delay due to intermittent connectivity; lower throughput due to low channel utilization; frequent topological changes; amongst others. Few IE solutions have been proposed to deal with some of these challenges. These solutions can be categorized according to the type of the mobile entity as follows:

7.1. Mobile Nodes

Researchers developed approaches specifically designed to extract information from WSNs where sensor nodes are mobile. These approaches can be classified according to the purpose of mobility as follows:

a) Coverage: Some approaches, e.g. [41-43], move nodes to provide better coverage by filling in holes in sensing coverage. They relocate redundant nodes to areas where node density is low to improve the accuracy of extracted information. However, this type of algorithms needs complex relocating models to calculate the moving nodes' trajectories and their new locations.

b) Mobile environments: Approaches such as [44-46] move mobile nodes to monitor moving objects, e.g. wildlife monitoring or offline monitoring of vehicle fleets. Mobile nodes are mounted on the monitored objects to log the sensed information on their memories for later analysis. When the mobile nodes move within radio range of the sink, they upload the logged information to it. However, these algorithms have several drawbacks. They are designed for specific types of applications that are not time critical. Moreover, in most situations animals or vehicles move in groups, the nodes density will not be disseminated in an effective way. Consequently, sinks located in dense areas will be overloaded, leading to increase latency and data loss.

c) Relay nodes: In some approaches, mobile nodes are used as relay nodes besides their sensing duties. Mobile nodes can be used to carry information from the sensing field and deliver it to a fixed sink. In these algorithms, mobile nodes send data over a short range communication (from a sensor to the relay node) that necessitates less transmission power. In [47], the authors proposed energy-efficient hybrid data collection approach. Its aim is to enhance the network performance and reduce the total energy consumption by introducing mobile node entities. A mobile node is moving through the network deployment region to collect data from the static nodes over a single hop radio links. The mobile node visits the sink periodically to drop off the collected data. The proposed solution reduces energy consumption and communication overhead by moving the sink node near to the nodes to collect data. Mobile nodes can move randomly, as in [48-50], on fixed trajectories, as in [51-54] or based on occurrence of an event of interest, as in [55, 56]. However, these approaches introduce considerable delays on data delivery and may potentially miss some important information in case of frequently changing phenomena. To overcome some of these issues, some recent researches have proposed moving the base station itself rather than moving relay nodes.

7.2. Mobile Data-sinks

The largest set of IE approaches that has been proposed in the literature suggests using a mobile sink for data collection and analysis. The mobile sink moves towards isolated nodes according to a particular trajectory to collect their data. Based on the nature of the trajectory, the sink mobility approaches to IE can be further classified into three classes:

a) Fixed trajectory, e.g. [57-60] assumes that the trajectory is fixed such as in roads. In these approaches data or information is conveyed to rendezvous nodes, which are closer to the trajectory, which is then cached until the mobile sink passes by and picks it up. The mobile sink can perform further processing on the received information or data. Hence, IE is achieved periodically.

b) Dynamic trajectory, e.g. [61], assumes that the trajectory is dynamic, different algorithms are used to calculate the trajectory. In these approaches IE is performed according to a pre-computed schedule, e.g. [62], or according to event occurrence, e.g. [63]. Approaches such as [63], propose mobility models that moves the sink node according to the evolution of the current events.

c) Random trajectories, e.g. [64-66] assume that mobile sink moves randomly in the sensor field. Mobile sinks are mounted on people, vehicles, or animals moving chaotically to collect information of interest around the network.

Although mobile sink strategies are desirable due to their simplicity, they suffer from some drawbacks. As the sink moves through the sensor field, it causes high control overhead to find a route to the sink and send packets to it. This may possibly dissipate the energy saved by using the mobile sink strategy. Moreover, constantly relocating the sink introduces significant delays on data or information delivery. Despite the extended coverage, these approaches lack scalability; as the network grows in size, the nodes located close to the mobile sink's trajectory get overloaded leading to energy depletion in the network, disconnections, and bandwidth bottlenecks. Finally, the trajectory calculation is a complex problem.

8. Discussion and Possible Future Directions

Before concluding this paper, this section provides a discussion about research issues, and future directions in the area of IE in WSNs. This survey revealed that most of the existing approaches to IE suffer from inherent problems that limit their applications including: they are application specific; characterized by poor spatio-temporal IE capabilities; consume high power; many approaches trade the amount and quality of returned information by energy consumption; they lack appropriate high-level interfaces that allow the user to set thresholds and issue queries; and the tight coupling between IE algorithms, applications, and hardware stacks leads to lack of code reuse. The lack of development frameworks means each new application has to be tackled from the ground up. These issues limit the usefulness of the developed IE approaches, making it hard to use them on anything other than the application it was designed for.

The problems and limitations presented above are the opportunities we intend to follow in our future work. Possible solutions that we are currently investigating for the integration of the three IE approaches will be achieved through the use of coordination rules [67] and mobile agents [68].

Coordination rules are a set of modelling primitives, design principles, and patterns that deal with enabling and controlling the collaboration among a group of software distributed agents performing a common task. If each algorithm in each IE category is viewed as a service, then the composition of these services will result in a complete IE framework. Service composition provides new services by combining existing services. The coordination rules specifies the order in which services are invoked and the conditions under which a certain service may or may not be invoked.

The mobile agent paradigm will be adopted to facilitate cooperation among services on different nodes. Mobile agent is a piece of software that performs data processing autonomously while migrating from node to node [68]. The agent can collect local data and perform any necessary data aggregation. Mobile agents can make decision autonomously without user input. They provide flexibility in terms of decision making, and reliability in terms of node failure [69].

Fig. 1 shows an illustration of the described hybrid framework. It shows how services on one node are connected and how a service can access other services on remote node.

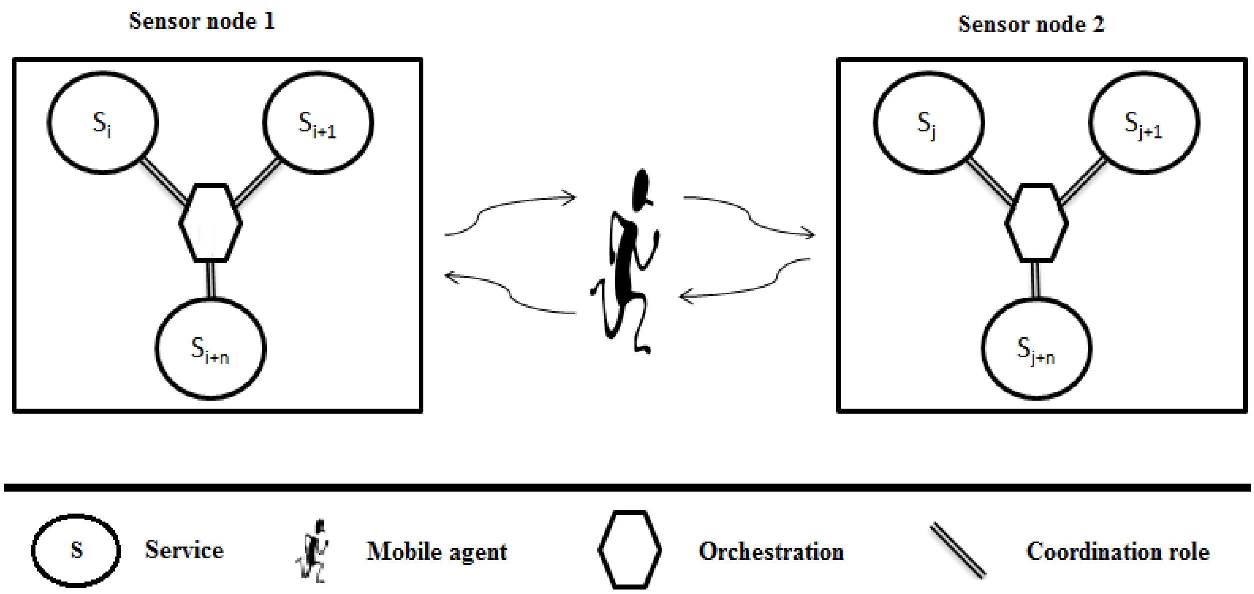

Fig. 1. An integrated IE framework for WSNs.

9. IE Framework: An Application Scenario

This section presents an application scenario that illustrates how different IE services can be coordinated. The coordination is to provide a means of integrating two of the IE services together by interfacing with each service to form a single service that can execute on a distributed WSN system. We consider events that are caused by multiple elements targets, e.g. a herd of animals in a habitat monitoring application. In this WSN system, an event-driven service is used to inform the time-driven service to increase the extracted information accuracy. Nodes send their sensor readings periodically to the end user to indicate the location of the herd. Besides periodic transmission of information, some nodes run an event-driven service. When a node detects an event in the form of movement in the environment, it modifies its time-driven service to increase information transmission frequency. The node will then notify its neighbours about the event in their area to adjust their time-driven services accordingly.

The Coordination Language Facility (CLF) [70] is used in this scenario as a coordination layer on top of the WSN system infrastructure. In CLF, the use of rules to coordinate services is based on a proactive system. A CLF system actively seeks to influence its environment, rather than just responding to external stimuli. This feature respects the autonomy of each participant sensor node. A sensor node waits for events to occur and generates new ones depending on the state of its rules engine.

```
waitUntilNext @ reading(Motion) <>- submit(Motion) @ check activity(Motion)
activity(normal) @ interval(normal) <>- #b
activity(normal) @ interval(high) <>- change_interval(normal)
activity(high)   <>-   change_interval(fast)   @   notify_neighbour(Motion,
Location, Interval)
received(Motion, Location, Interval) @ interval(normal) <>- reading(Motion)
@ submit(Motion) @ change_interval(fast)
received(Motion, Location, Interval) @interval(high) <>- #b
```

In implementing the scenario described above, the rules are distributed amongst and maintained by the nodes themselves so multiple nodes can collaborate independently, i.e. these rules apply to individual nodes. The `waitUntilNext` token is read-only and returns a value at the beginning of each sensing interval; it acts as a trigger to start the search for an instance of the rule. The `reading` token holds the current motion measurement from the motion sensor. The `submit` token takes the sensor measurement and sends it to the higher level nodes (e.g. cluster head or sink). The `check_activity` token checks the level of animal activity or movement within the node's coverage area. When the level of activity, `activity(normal)`, is normal and the transmission interval is also normal, `interval(normal)`, then the node will do nothing. When the node's current level of activity is normal and the transmission interval is high, `interval(high)`, then the node will change it to normal to save its energy, `change_interval(normal)`. If the node senses a high activity within its area of coverage, then it will change its time-driven service interval to fast and notify its neighbours about the event in their area `notify_neighbour(Motion, Location, Interval)`. The node will include in its notification message its sensor measurement, location, and a new interval value. When a node receives the notification message, `received()`, it changes its time-driven service interval to fast and it submits its current sensor measurement.

This example shows that coordination can provide an intuitive method for combining services belonging to a mixture of IE categories and manage the resources required for implementing them. What precisely is being coordinated, how the coordination is accomplished, and what are the relevant metaphors that must be implemented and used, are all questions that we intend to tackle in the future.

10. Conclusion

The main objective of this paper is to provide an understanding of the current issues in this area for better future academic research and industrial practice of WSNs IE. We have presented a review of the state of the art for IE approaches in WSNs. We discussed various approaches to IE. We also discussed the challenges as well as future research directions in developing a complete integrated WSNs IE framework. Finally, an application scenario that demonstrates how coordination can be used to deal with the cooperation among very large number of nodes with several active services that comprise a single IE system.

References

[1]. T. Alsbouí, M. Hammoudeh, Z. Bandar, and A. Nisbet, An Overview and Classification of Approaches to Information Extraction in Wireless Sensor Networks, in *Proceedings of the 5th International Conference on Sensor Technologies and Applications(SENSORCOMM' 2011),* France, 2011.

[2]. S. Sarawagi, Information Extraction, *Found. Trends databases,* Vol. 1, March 2008, pp. 261-377.

[3]. C. Zhang, C. Wang, D. Li, X. Zhou, and C. Gao, Unspecific Event Detection in Wireless Sensor Networks, in P*roceedings of the International Conference on Communication Software and Networks,* Vo l. 0, 2009, pp. 243-246.

[4]. R. Bhargavi, V. Vaidehi, P. T. V. Bhuvaneswari, P. Balamurali, and G. Chandra, Complex Event Processing for Object Tracking in Wireless Sensor Networks, *Web Intelligence and Intelligent Agent Technology (WI-IAT), 2010 IEEE/WIC/ACM International Conference,* Vol. 3, 2010, pp. 211 -214.

[5]. A. Iranli, M. Maleki, and M. Pedram, Energy efficient strategies for deployment of a two-level wireless sensor network, in *Proceedings of the 2005 International Symposium on Low Power Electronics and Design*, 2005*,* pp. 233-238.

[6]. K. Akkaya and M. Younis, A survey on routing protocols for wireless sensor networks, *Ad Hoc Networks,* Vol. 3, 2005, pp. 325 - 349.

[7]. B. Gedik, L. Liu, and P. S. Yu, ASAP: An Adaptive Sampling Approach to Data Collection in Sensor Networks, *IEEE Transactions on Parallel and Distributed Systems,* Vol. 18, pp. 1766-1783, 2007.

[8]. M. L. Segal, F. P. Antonio, S. Elam, J. Erlenbach, and K. R. De Paolo, Method and apparatus for automatic event detection in a wireless communication system, United States Patent No. 6124810, September 2000.

[9]. G. Werner-Allen, K. Lorincz, M. Welsh, O. Marcillo, J. Johnson, M. Ruiz, and J. Lees, Deploying a Wireless Sensor Network on an Active Volcano, *IEEE Internet Computing,* Vol. 10, 2006, pp. 18-25.

[10].M. Bahrepour, M. Meratnia, and N. Havinga, Sensor Fusion-based Event Detection in Wireless Sensor Networks, in *Proceedings of the 6th Annual International Conference on Mobile and Ubiquitous Systems: Networking Services (MobiQuitous' 09),* 2009*,* pp. 1-8.

[11].Y. Zhang, N. Meratnia, and P. Havinga, Outlier Detection Techniques for Wireless Sensor Networks: A Survey, *Communications Surveys Tutorials, IEEE,* Vol. 12, 2010, pp. 159 -170.

[12].S. Sardellitti, S. Barbarossa, and L. Pezzolo, Distributed double threshold spatial detection algorithms in wireless sensor networks, in Proceedings of the *10th IEEE Workshop on Signal Processing Advances in Wireless Communications (SPAWC '09,)* 2009*,* pp. 51 -55.

[13].D. Tulone and S. Madden, An energy-efficient querying framework in sensor networks for detecting node similarities, in *Proceedings of the 9th ACM International Symposium on Modeling Analysis and Simulation of Wireless and Mobile Systems,* 2006 pp. 191-300.

[14].D. Chu, A. Deshpande, J. M. Hellerstein, and W. Hong, Approximate Data Collection in Sensor Networks using Probabilistic Models, in *Proceedings of the 22nd International Conference on Data Engineering*, 2006*,* pp. 48.

[15].A. Deshpande, C. Guestrin, S. R. Madden, J. M. Hellerstein, and W. Hong, Model-driven data acquisition in sensor networks, in *Proceedings of the Thirtieth International Conference on Very Large Data Bases,* Vol. 30, 2004, pp. 588-599.

[16].M. Bahrepour, N. Meratnia, M. Poel, Z. Taghikhaki, and P. J. M. Havinga, Distributed Event Detection in Wireless Sensor Networks for Disaster Management, in *Proceedings of the International Conference on Intelligent Networking and Collaborative Systems (INCoS' 2010)*, November 2010*,* pp. 507-512.

[17].Y. W. Guorui Li, An Efficient Data Aggregation Scheme Leveraging Time Series Prediction in Wireless Sensor Networks, *International Journal of Machine Learning and Computing,* Vol. 1, October 2011, pp. 372-377.

[18].G.-S. Ahn, Information-driven Tracking and Access Control in Wireless Ad hoc and Sensor Networks, *Columbia University*, 2009.

[19].H. Li, H. Yu, L. Li, and A. Liu, On the Cascading Data Collection Mechanism in Wireless Sensor Networks, in Proceedings of the *International Conference on Wireless Communications, Networking and Mobile Computing*, Shanghai, China, 2007, pp. 2479-2482.

[20].R. Jurdak, P. Baldi, and C. Videira Lopes, Adaptive Low Power Listening for Wireless Sensor Networks, *IEEE Transactions on Mobile Computing,* Vol. 6, 2007, pp. 988-1004.

[21].G. Trajcevski and P. Scheuermann, (Reactive /+ Proactive Behavior) Situation Awareness in Sensor Networks, in *Proceedings of the Workshop on Research Directions in Situational-aware Self-managed Proactive Computing in Wireless Adhoc Networks,* 2009.

[22].Y. Yao and J. Gehrke, The cougar approach to in-network query processing in sensor networks, *SIGMOD Rec.,* Vol. 31, 2002, pp. 9-18.

[23].M. Bestehorn, K. Bohm, E. Buchmann, and S. Kessler, Energy-efficient processing of spatio-temporal queries in wireless sensor networks, in *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 2010*,* pp. 340-349.

[24].S. R. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong, TinyDB: an acquisitional query processing system for sensor networks, *ACM Trans. Database Syst.,* Vol. 30, 2005, pp. 122-173.

[25].M. Umer, L. Kulik, and E. Tanin, Optimizing query processing using selectivity-awareness in Wireless Sensor Networks, *Computers, Environment and Urban Systems,* Vol. 33, 2009, pp. 79 - 89.

[26].J.-Z. Sun, An Energy-Efficient Query Processing Algorithm for Wireless Sensor Networks, in *Ubiquitous Intelligence and Computing*. Vol. 5061, 2008, pp. 373-385.

[27].R. Khoury, T. Dawborn, B. Gafurov, G. Pink, E. Tse, Q. Tse, M. Gaber, U. Rohm, and B. Scholz, Corona: Energy-Efficient Multi-query Processing in Wireless Sensor Networks, in *Database Systems for Advanced Applications*. Vol. 5982, 2010, pp. 416-419.

[28].M. Hammoudeh and T. Alsboui, Building Programming Abstractions for Wireless Sensor Networks Using Watershed Segmentation, in *Smart Spaces and Next Generation Wired/Wireless Networking*. Vol. 6869: Springer Berlin / Heidelberg, 2011, pp. 587-597.

[29].A. Pathak and V. K. Prasanna, Energy-Efficient Task Mapping for Data-Driven Sensor Network Macroprogramming, *IEEE Trans. Comput.,* Vol. 59, July 2010, pp. 955-968.

[30].T. W. Hnat, T. I. Sookoor, P. Hooimeijer, W. Weimer, and K. Whitehouse, A modular and extensible macroprogramming compiler, in *Proceedings of the Workshop on Software Engineering for Sensor Network Applications (ICSE' 10*), 2010, pp. 49-54.

[31].A. Pathak and M. K. Gowda, Srijan: a graphical toolkit for sensor network macroprogramming, in *Proceedings of the 7th Joint Meeting of the European Software Engineering Conference and the ACM SIGSOFT Symposium on The Foundations of Software Engineering*, 2009, pp. 301-302.

[32].T. I. Sookoor, T. W. Hnat, and K. Whitehouse, Programming cyber-physical systems with MacroLab, in *Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems*, 2008, pp. 363-364.

[33].R. Newton, G. Morrisett, and M. Welsh, The regiment macroprogramming system, in *Proceedings of the 6th International Conference on Information Processing in Sensor Networks*, 2007, pp. 489-498.

[34].L. Mottola and G. P. Picco, Using logical neighborhoods to enable scoping in wireless sensor networks, in *Proceedings of the 3rd International Middleware Doctoral Symposium*, 2006, pp. 6.

[35].L. Mottola and G. P. Picco, Programming wireless sensor networks: Fundamental concepts and state of the art, *ACM Comput. Surv.,* Vol. 43, April 2011, pp. 19:1-19:51.

[36].R. Gummadi, N. Kothari, R. Govindan, and T. Millstein, Kairos: a macro-programming system for wireless sensor networks, in *Proceedings of the Twentieth ACM Symposium on Operating Systems Principles*, 2005, pp. 1-2.

[37].B.-D. Lee, Adaptive Data Dissemination Protocol for Wireless Sensor Networks, in *Security-Enriched Urban Computing and Smart Grid*. Vol. 78, 2010, pp. 188-195.

[38].C.-H. Lee, C.-W. Chung, and S.-J. Chun, Effective processing of continuous group-by aggregate queries in sensor networks, *J. Syst. Softw.,* Vol. 83, pp. 2627-2641, December 2010.

[39].R. Bhargavi, V. Vaidehi, P. T. V. Bhuvaneswari, P. Balamuralidhar, and M. G. Chandra, Complex Event Processing for object tracking and intrusion detection in Wireless Sensor Networks, in *Proceedings of the 11th International Conference on Control Automation Robotics Vision (ICARCV' 2010),* December 2010, pp. 848 -853.

[40].S. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong, The design of an acquisitional query processor for sensor networks, in *Proceedings of the International Conference on Management of Data (ACM SIGMOD' 03)*, 2003, pp. 491-502.

[41].V. Coskun, Relocating Sensor Nodes to Maximize Cumulative Connected Coverage in Wireless Sensor Networks, *Sensors,* Vol. 8, 2008, pp. 2792-2817.

[42].G. Wang, G. Cao, and T. L. Porta, Movement-assisted sensor deployment, *IEEE Transactions on Mobile Computing,* Vol. 5, 2006, pp. 640 -652.

[43].P. Guo, G. Zhu, and L. Fang, An Adaptive Coverage Algorithm for Large-Scale Mobile Sensor Networks, in Ubiquitous Intelligence and Computing, Vol. 4159, *Springer Berlin / Heidelberg*, 2006, pp. 468-477.

[44].N. Gajaweera and D. Dias, FAMA/TDMA Hybrid MAC for Wireless Sensor Networks, in *Proceedings of the Information and Automation for Sustainability (ICIAFS' 08)*, 2008, pp. 67-72.

[45].S. Sivaramakrishnan and A. Al-Anbuky, Analysis of network connectivity: Wildlife and Sensor Network, *Telecommunication Networks and Applications Conference (ATNAC), 2009 Australasian*, 2009, pp. 1-6.

[46].T. Wark, P. Corke, P. Sikka, L. Klingbeil, Y. Guo, C. Crossman, P. Valencia, D. Swain, and G. Bishop-Hurley, Transforming Agriculture through Pervasive Wireless Sensor Networks, *Pervasive Computing, IEEE,* Vol. 6, 2007, pp. 50 -57.

[47]. A. Rasheed and R. Mahapatra, An Energy-Efficient Hybrid Data Collection Scheme in Wireless Sensor Networks, in *Proceedings of the 3rd International Conference on Intelligent Sensors, Sensor Networks and Information (ISSNIP' 07)*, 2007, pp. 703 -708.
[48]. S. Jain, R. C. Shah, W. Brunette, G. Borriello, and S. Roy, Exploiting mobility for energy efficient data collection in wireless sensor networks, *Mob. Netw. Appl.,* Vol. 11, June 2006, pp. 327-339.
[49]. R. C. Shah, S. Roy, S. Jain, and W. Brunette, Data MULEs: modeling a three-tier architecture for sparse sensor networks, in *Proceedings of the 1st IEEE International Workshop on Sensor Network Protocols and Applications,* 2003, pp. 30 - 41.
[50]. D. N. Jayasingha, N. P. M. L. Jayawardhane, P. N. Karunanayake, G. D. Karunarathne, and D. Dias, Wireless Sensor Network Testbed for Mobile Data Communication, in Proceedings of the *4th International Conference on Information and Automation for Sustainability (ICIAFS' 08),* 2008, pp. 97 -103.
[51]. G. Anastasi, M. Conti, and M. Di Francesco, Data collection in sensor networks with data mules: An integrated simulation analysis, in Proceedings of the *IEEE Symposium on Computers and Communications, (ISCC' 08)*, July 2008, pp. 1096 -1102.
[52]. A. Chakrabarti, A. Sabharwal, and B. Aazhang, Communication power optimization in a sensor network with a path-constrained mobile observer, *ACM Trans. Sen. Netw.,* Vol. 2, August 2006, pp. 297-324.
[53]. W. Alsalih, H. Hassanein, and S. Akl, Routing to a Mobile Data Collector on a Predefined Trajectory, in Proceedings of the *IEEE International Conference on Communications (ICC '09)*, June 2009, pp. 1 -5.
[54]. D. Jea, A. Somasundara, and M. Srivastava, Multiple controlled mobile elements (data mules) for data collection in sensor networks, *In DCOSS*, 2005, pp. 244-257.
[55]. M. Xi, K. Wu, Y. Qi, J. Zhao, Y. Liu, and M. Li, Run to Potential: Sweep Coverage in Wireless Sensor Networks, in *Proceedings of the International Conference on Parallel Processing (ICPP '09)*, September 2009, pp. 50-57.
[56]. L. He, Y. Zhuang, J. Pan, and J. Xu, Evaluating On-Demand Data Collection with Mobile Elements in Wireless Sensor Networks, in *Proceedings of the IEEE 72nd Conference on Vehicular Technology (VTC 2010-Fall),* September 2010, pp. 1-5.
[57]. G. Pantziou, A. Mpitziopoulos, D. Gavalas, C. Konstantopoulos, and B. Mamalis, Mobile Sinks for Information Retrieval from Cluster-Based WSN Islands, in *Proceedings of the 8th International Conference on Ad-Hoc, Mobile and Wireless Networks*, 2009, pp. 213-226.
[58]. S. Gao, H. Zhang, and S. Das, Efficient data collection in wireless sensor networks with path-constrained mobile sinks, in Proceedings of the *IEEE International Symposium on World of Wireless, Mobile and Multimedia Networks Workshops (WoWMoM' 09)*, June 2009, pp. 1-9.
[59]. L. Song and D. Hatzinakos, Architecture of Wireless Sensor Networks With Mobile Sinks: Sparsely Deployed Sensors, *IEEE Transactions on Vehicular Technology,* Vol. 56, July 2007, pp. 1826-1836.
[60]. L. Cheng, Y. Chen, C. Chen, and J. Ma, Query-based data collection in wireless sensor networks with mobile sinks, in *Proceedings of the 2009 International Conference on Wireless Communications and Mobile Computing: Connecting the World Wirelessly*, 2009, pp. 1157-1162.
[61]. H. Lee, M. Wicke, B. Kusy, O. Gnawali, and L. Guibas, Data stashing: energy-efficient information delivery to mobile sinks through trajectory prediction, in *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks*, 2010, pp. 291-302.
[62]. G. Xing, T. Wang, W. Jia, and M. Li, Rendezvous design algorithms for wireless sensor networks with a mobile base station, in *Proceedings of the 9th ACM international symposium on Mobile ad hoc networking and computing*, 2008, pp. 231-240.
[63]. H. Sabbineni and K. Chakrabarty, Datacollection in Event-Driven Wireless Sensor Networks with Mobile Sinks, *International Journal of Distributed Sensor Networks,* Vol. 2010.
[64]. L. Tzevelekas and I. Stavrakakis, Sink mobility schemes for data extraction in large scale WSNs under single or zero hop data forwarding, in *Proceedings of the European Wireless Conference (EW' 2010)*, April 2010, pp. 896-902.
[65]. I. Chatzigiannakis, A. Kinalis, S. Nikoletseas, and J. Rolim, Fast and energy efficient sensor data collection by multiple mobile sinks, in *Proceedings of the 5th ACM international workshop on Mobility Management and Wireless Access*, 2007, pp. 25-32.
[66]. A. Viana, A. Ziviani, and R. Friedman, Decoupling data dissemination from mobile sink's trajectory in wireless sensor networks, *Communications Letters, IEEE,* Vol. 13, March 2009, pp. 178 -180.
[67]. J. Abreu and J. Fiadeiro, A Coordination Model for Service-Oriented Interactions, in *Coordination Models and Languages*. Vol. 5052, 2008, pp. 1-16.
[68]. M. Chen, T. Kwon, Y. Yuan, and V. Leung, Mobile Agent Based Wireless Sensor Networks, *Journal of Computers,* Vol. 1, 2006.

[69].D. Massaguer, C.-L. Fok, N. Venkatasubramanian, G.-C. Roman, and C. Lu, Exploring sensor networks using mobile agents, in *Proceedings of the 5th International Joint Conference on Autonomous Agents and Multiagent Systems*, 2006, pp. 323-325.
[70].J.-M. Andreoli, S. Freeman, and R. Pareschi, The Coordination Language Facility: coordination of distributed objects, *Journal of Theory and Practice of Object Systems (TAPOS),* Vol. 2, 1996, pp. 1-18.

Guide for Contributors

Aims and Scope

Sensors & Transducers Journal (ISSN 1726-5479) provides an advanced forum for the science and technology of physical, chemical sensors and biosensors. It publishes state-of-the-art reviews, regular research and application specific papers, short notes, letters to Editor and sensors related books reviews as well as academic, practical and commercial information of interest to its readership. Because of it is a peer reviewed international journal, papers rapidly published in *Sensors & Transducers Journal* will receive a very high publicity. The journal is published monthly as twelve issues per year by International Frequency Sensor Association (IFSA). In additional, some special sponsored and conference issues published annually. *Sensors & Transducers Journal* is indexed and abstracted very quickly by Chemical Abstracts, IndexCopernicus Journals Master List, Open J-Gate, Google Scholar, etc. Since 2011 the journal is covered and indexed (including a Scopus, Embase, Engineering Village and Reaxys) in Elsevier products.

Topics Covered

Contributions are invited on all aspects of research, development and application of the science and technology of sensors, transducers and sensor instrumentations. Topics include, but are not restricted to:

- Physical, chemical and biosensors;
- Digital, frequency, period, duty-cycle, time interval, PWM, pulse number output sensors and transducers;
- Theory, principles, effects, design, standardization and modeling;
- Smart sensors and systems;
- Sensor instrumentation;
- Virtual instruments;
- Sensors interfaces, buses and networks;
- Signal processing;
- Frequency (period, duty-cycle)-to-digital converters, ADC;
- Technologies and materials;
- Nanosensors;
- Microsystems;
- Applications.

Submission of papers

Articles should be written in English. Authors are invited to submit by e-mail editor@sensorsportal.com 8-14 pages article (including abstract, illustrations (color or grayscale), photos and references) in both: MS Word (doc) and Acrobat (pdf) formats. Detailed preparation instructions, paper example and template of manuscript are available from the journal's webpage: http://www.sensorsportal.com/HTML/DIGEST/Submition.htm Authors must follow the instructions strictly when submitting their manuscripts.

Advertising Information

Advertising orders and enquires may be sent to sales@sensorsportal.com Please download also our media kit: http://www.sensorsportal.com/DOWNLOADS/Media_Kit_2012.pdf