This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

Distributed Database Management Techniques for Wireless Sensor Networks

Ousmane Diallo, Joel J. P. C. Rodrigues, *SeniorMember, IEEE*,
Mbaye Sene, and Jaime Lloret, *SeniorMember, IEEE*

***Abstract*—In sensor networks, the large amount of data generated by sensors greatly influences the lifetime of the network. In order to manage this amount of sensed data in an energy-efficient way, new methods of storage and data query are needed. In this way, the distributed database approach for sensor networks is proved as one of the most energy-efficient data storage and query techniques. This paper surveys the state of the art of the techniques used to manage data and queries in wireless sensor networks based on the distributed paradigm. A classification of these techniques is also proposed. The goal of this work is not only to present how data and query management techniques have advanced nowadays, but also show their benefits and drawbacks, and to identify open issues providing guidelines for further contributions in this type of distributed architectures.**

***Index Terms*—Distributed database management; wireless sensor networks; distributed storage; query techniques, data reduction techniques; query optimization.**

I. Introduction

Wireless Sensor Networks (WSNs) are composed of a large number of devices, called sensor nodes, which are able to sense, process, and transmit information about the environment on which they are deployed. These devices are usually distributed in a geographical area in order to collect information for users interested in monitoring and controlling a given phenomenon. This information is transferred to a sink node in order to be accessible by remote users through generally application-level gateway, e.g. global sensor network (*GSN*) [1],[2], [3]. To obtain the data, these applications should also provide supports of efficient queries, which allow communication with the network [4],[5],[6] (see Fig. 1 for an illustration of a WSN).

In wireless sensor networks, the sensor nodes are battery powered and are considered intelligent with acquisitional, processing, storage, and communication capacities [7],[8]. However, these resources are generally very limited, especially in terms of storage and energy, and the sensor nodes activities are sometimes not negligible in energy consumption [9], [10].One of the most used techniques to save power is to activate only necessary nodes and to put other nodes to sleep [11]. Some authors have studied how a 3 dimensional sensor field can be efficiently partitioned into cells in order to save energy [12].

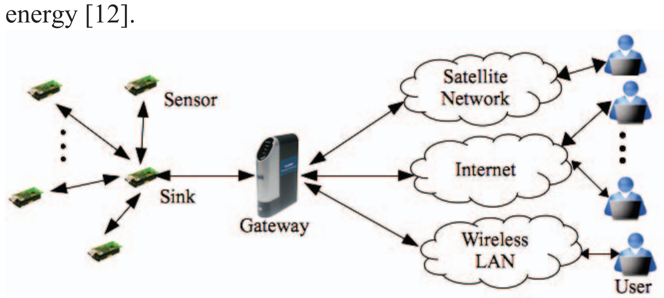

Fig.1. Illustration of a wireless sensor network architecture.

Sensors can be placed anywhere there is data that should be collected, what makes information omnipresent. Consequently, systems based on sensor networks are increasingly common in many areas of the knowledge, giving rise to several flavors of WSNs [13], [14]. These numerous WSNs have allowed the development of many applications [15], [16], [17]. In addition to data gathering [18] and data replication issues [19], in such applications, a database-oriented approach of WSNs has proven to be useful in order to manage the large amount of data generated by the sensors. According to this approach, a WSN is viewed as a distributed database where sensor nodes are considered as data sources with sensed data stored in the form of rows of a relation distributed across a set of nodes in the network [20], [21]. This database-oriented approach has motivated the design of WSN data acquisition with two fundamental objectives [22]: similarly to traditional database systems, a WSN database should provide SQL-like abstractions so that nodes can be easily programmed for simple data sensing and collection. In addition, the data collection process should minimize the energy consumption in the network.

The main goal of distributed database management on WSNs is to support the management of the huge amount of sensed data in an energy-efficient manner [23]. In fact, research into sensor hardware has shown that the energy depletion in the network is mainly due to the data communication tasks among the nodes [24]. To deal with this problem, various data reduction techniques exist[25], [26], including *data aggregation* [27], [28], *packet merging*, *data compression techniques* [29], [30], *data fusion*, and

Manuscript received ……
Ousmane Diallo is with the Instituto de Telecomunicações, University of Beira Interior Covilhã, Portugal and with the Department of Mathematics and Informatics, UCAD, Senegal. E-mail: Ousmane.diallo@it.ubi.pt.
Joel J. P. C. Rodrigues is with the Instituto de Telecomunicações, University of Beira Interior, Covilhã, Portugal. E-mail: joeljr@ieee.org.
Mabye Sene is with the Department of Mathematics and Informatics, UCAD, Senegal. E-mail: mbaye.sene@ucad.sn.
Jaime Lloret is with Integrated Management Coastal Research Institute, Universidad Politécnica de Valencia, Spain. E-mail:jlloret@dcom.upv.es.

approximation based techniques [31], [32]. The data aggregation techniques consist to perform data aggregations (e.g., MAX, AVG etc.) at intermediate nodes between the source nodes and the sink node. The packet merging combines multiple small packets into a big one, without considering the semantics and the correlation between the packets. The data compression techniques are also used to reduce the amount of data transmitted between the nodes, but they involve data encoding at the source nodes, data decoding at the sink node. The data fusion techniques refer to more complex operations on a data set and are usually used in multimedia data processing [33]. The approximation based techniques use statistical techniques to approximate the queries results. These techniques provide, among other advantages, the reduction of the size of the transmitted data, the communication tasks, the network load, and the data transmission time.

The aim of this paper is to show how distributed database techniques are adapted to wireless sensor networks in order to improve the management of the great amount of sensed data in an energy-efficient way by presenting and classifying the most recent and relevant proposals of distributed database management on WSNs. Moreover, a discussion and open issues on distributed database management techniques for wireless sensor networks are identified in order to facilitate further contributions.

The remainder of this paper is organized as follows. Section II presents the essential conceptual features of distributed data storage and querying in WSNs, while protocols and techniques used on the studied proposals on distributed data management in WSNs are exposed in Section III. Section IV discusses the techniques used on the studied approaches and proposes some open research issues. Finally, Section V concludes the paper and pinpoints further research works.

II. Background

As in traditional database systems, the sensor databases try to create an abstraction between the end-users and the sensor nodes. This abstraction aims to permit the users to only concentrate on the needed data to be collected rather than bothering with the complexities of mechanisms deciding how to extract data from a network [27], [34]. As such, the sensor databases have been subject to two main approaches to data storage and query in WSNs [35]: the *warehousing* approach and the *distributed* approach.

1. In the *warehousing* approach, the sensors act as collectors. The data gathered by sensors are periodically sent to a central database where user queries are processed. This model is the most used one in data storage and query processing. However, it has some drawbacks, such as eventually wasting resources and creating a bottleneck with an immense amount of transmitted data. This approach is unsuitable for real-time processing.
2. The *distributed* approach is the alternative, where each sensor node is considered as a data source, and then the WSN forms a distributed database where the sensed data are in the form of rows with columns representing sensor attributes [20], [21]. In this second approach, the sensed data are not periodically sent to the database server. They remain in the sensor nodes and some queries are injected in the network through the base station. These queries are disseminated into the network according to the routing techniques as per [36], [37], and the sensors, thanks to their processing and storage capabilities, process them. The sensors send their data to their parent nodes whenever they correspond to the query requirements. The parent nodes combine this coming data with their own data and transmit to their parent nodes and so on until the data reaches the gateway. This approach that consists to process the data inside the sensor nodes themselves is called *in-network processing* and it reduces the amount and size of transmitted data and the latency [38]. According to the scope of this work, an illustration of a distributed database on WSN may be seen in Fig. 2.

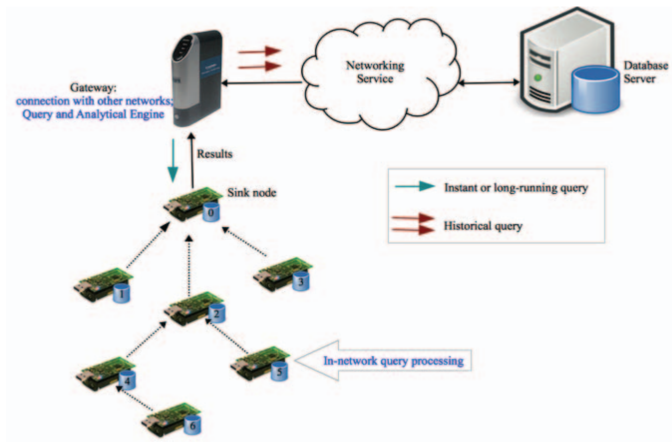

Fig. 2. Illustration of a distributed database architecture for wireless sensor networks.

As largely detailed in [39], there are four essential methods to design a distributed data management system: *in-network processing*, *acquisitional query processing*, *cross-layer optimization*, and *data-centric data/query dissemination*.

A. In-network Processing

The in-network processing technique [38], [40], [41] generally includes the different types of operation that are traditionally done on the server, for instance, aggregations to inside the sensor nodes themselves. It is generally used, as its name indicated, to process sensing values inside the network nodes so as to filter and reduce the huge and needless data.

B. Acquisitional Query Processing

The acquisitional query processing [20] permits to minimize energy consumption in the network by reducing the number of sensor nodes participating in the query processing. This reduction is done by expressing in the query when or what sensors to sample.

C. Cross-layer Optimization

Unlike the traditional computer networks in which layers in the conventional OSI model are separated and isolated, the cross-layer optimization [42], [43] permits to combine information available on these different layers and profit from this information sharing. For instance, in wireless sensor networks the routing takes care of, among others, the quality of service (QoS) parameters of the network, network connectivity, the power available on the node, and the network lifetime [44]. In traditional computer networks the routing is done by the network layer only considering the destination address of a packet.

D. *Data-centric data/query dissemination*

In contrast with traditional networks, nodes in WSNs usually do not have a single identifier because of data-centric [45], [46], [47]nature of sensor applications as well as the large number of sensors deployed. Generally, applications are not interested in specific sensors, but rather in data, which they generate. For example, a query as *"which is the temperature measurement of the sensor with the ID XXXX"* does not have much interest for a sensor application, but a query like *"in which region, sensors measure fewer than 7ºC"* is more significant. Routing protocols must take these characteristics into account.

In the next section, the most relevant proposals of distributed data storage and query management techniques designed for WSNs are going to be discussed. This discussion will be done by classifying each proposal into the four above-mentioned approaches.

III. Distributed Data Storage and Query Management Techniques for WSNs

There is a lot of research addressing various aspects of data management in sensor networks. However, generally, the main goals are the transparent access of the end-users to the sensor nodes for retrieving required relevant sensor data as well as the improvement of the network's lifetime.

A. *In-Network Processing*

After receiving one query, the query processor analyses it and generates an optimized query execution plan that consists of the best possible execution of the query inside the sensor network. While the different solutions to perform in-network processing may differ, the objective is to save energy by reducing communication. The in-network processing technique can be divided into two sub-categories, *aggregation based techniques* and *approximation based techniques* [48].

1. *Aggregation based Techniques*

Since wireless communication involves more energy consumption than the sensing and processing operations, aggregating the data through intermediate nodes results in less message transmissions. Therefore, the network lifetime is improved with less energy consumption.

As part of the pioneering researches on distributed database on WSNs, TinyDB [20] and COUGAR [21], [34], [49]are the first to adopt a database query optimization technique for WSNs based on the data acquisition declarative approach.

The TinyDB project [50] was developed for networks based on the TinyOS operating system [51]. It is a distributed query processor for sensor networks that incorporates acquisitional techniques. Through an interface, the user chooses what data he wishes to acquire. The query is decomposed by a query processor and distributed across the network. The sensor nodes collect, filter and aggregate the data and respond to the user query.

The interrogation of the sensors is based on the relational model and queries are specified using an SQL-like query language [20]. The sensor data are form of tuples that conform to a predetermined schema. For example, tuples produced by a temperature sensor may be of the form *<sensorId, location, temperature, timestamp>*.

Physically, the sensor tuples belong to a sensors table, which is partitioned across all of the devices in the network, with each device producing and storing its own readings. Queries are formulated on a virtual table that is logically formed by the sensor tuples horizontally partitioned in the network. Unlike traditional queries that focus on the current state of a database, these queries are often continuous with as function to continually run in order to inform the applications of changes recorded by the sensors. Results of queries stream to the root (base station) of the network by multi-hop topology.

TinyDB includes support for grouped aggregation queries (e.g. *MIN*, *MAX*, *SUM*, *COUNT*, *AVERAGE*, etc.), as sensor readings flow up the communication tree called a semantic routing tree (*SRT*), they are aggregated by intermediate nodes that contain relevant information for the query (See Fig. 3). This in-network aggregation reduces the huge quantity of data that must be transmitted through the network, preventing the bottleneck to the root node and increasing the lifetime of the network.

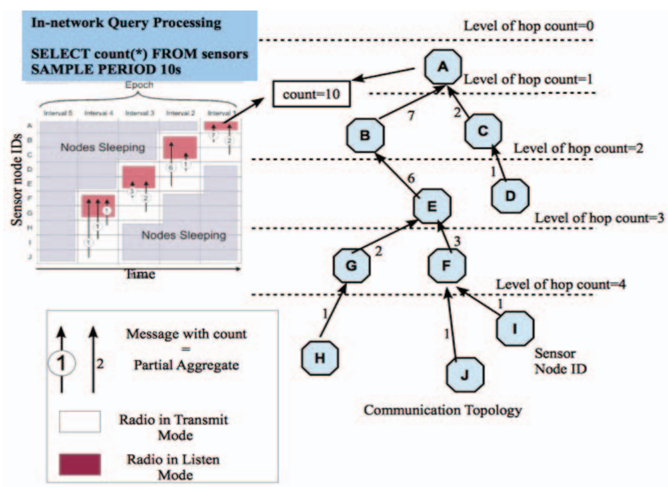

Fig. 3. Aggregation steps of sensor readings during an epoch using interval-based communication [38].

The Cougar project [52] is a platform for distributed query processing. To deal with in-network processing in this platform, they use a clustered approach. A network is composed of several clusters, each of them managed by a cluster head. The child nodes that belong to clusters send periodically their readings to the corresponding cluster head, which then aggregates the received readings and forwards the computed result toward the Front End of the network. This Front End is a query optimizer, located at the gateway node, which generates optimized distributed query processing plans after receiving user queries. Furthermore, in this architecture, each node embeds a query layer. The query layer [21], [49]is a query proxy between the network layer and the application layer, which process queries (See Fig. 4). Additionally,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

Cougar carries out packet merging by aggregating several packets into one. This increases the lifetime of the network, since sending multiple small packets is more expensive than sending one larger packet.

Like TinyDB, Cougar adopts a declarative queries approach [49] to in-networking processing. This approach allows users and application queries a transparence access to sensor nodes. Thus, Cougar uses an efficient catalog management, query optimization, and query processing techniques to abstract the user from the physical details of contacting the relevant sensor nodes, which process the sensor data and send the results to the user. The queries are specified using an SQL-like query language and the sensor data are form of records with several fields included information about the sensor node (e.g., id, location, etc.), a timestamp, the sensor type (e.g., temperature, light, etc.), and the value of the reading. The sensor network is considered as a widely distributed database system consisting of multiple tables of different types of sensors.

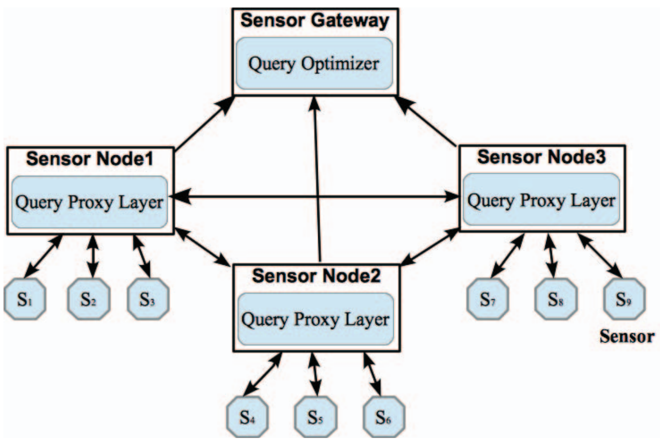

Fig.4. Illustration of Cougar Architecture.

Cougar claims to be designed for WSNs, while it was deployed on PDA-class devices that had significantly larger processing power than WSNs and could even run Windows CE and Linux [20]. So, it does not take into account the power and computational constraints of sensor nodes.

TiNA(Temporal Coherency-Aware In-Network Aggregation) [53], [54], [55] is an improvement over TinyDB. It uses in-network processing to increase the lifetime of the network. It is freshness-aware as it adds a new clause, *VALUES WITHIN tct,* in the specification of the query aggregation syntax of TinyDB, which indicates the temporal relaxation degree allowed by the user or the network. In TinyDB, the readings are transmitted at fixed interval, but TINA transmits the sensor node reading only if that reading differs from the last recorded reading by more than the accepted tolerance *tct* (See Fig. 5).

In [54], TiNA is improved by designing a semantic routing tree for sensor networks with a main objective the reduction of the size of transmitted data. For that, by performing in-network aggregation, the reduction of the number of groups is adopted when a node performs Group-By query by clustering the sensor nodes belong to the same group along the same path. This approach is called group-aware network configuration method. This technique can certainly reduce the size and number of messages circulating in the network but at the expense of some consistency of data transmitted.

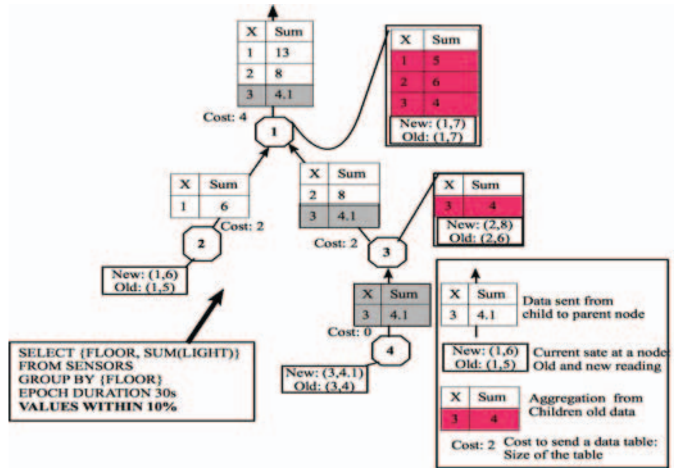

Fig.5. In-network aggregation query using TINA [53].

The authors in [56] proposed a set of algorithms to minimize the overall energy consumption of the sensor nodes by considering a real time scenario where the raw data gathered from the source nodes must be aggregated and transmitted to the sink within a specified latency constraint. This work is particularly important for applications requiring a prompt delivery of the information to the sink. However, it does not address the problem of constructing the underlying aggregation tree.

The main purpose of data management in WSN is to allow transparent access to sensed data, as well as, to increase the network lifetime. To these ends, the authors of [57], [58] propose an adaptive algorithm, called *ADAGA* (ADaptive AGgregation Algorithm for sensor networks), for processing in-network aggregation in WSNs. In fact, in order to reduce the amount of data transmitted between sensor nodes, *ADAGA* performs in-network aggregation. The key idea is to aggregate the sensed data progressively in each node it passed through. This will further reduces the data traffic, the energy consumption, and the memory usage in the network. The algorithm also supports the packets replication in order to reduce the packets losses and allows the approximation of sensed values from collected data. Furthermore, a data model for data streams and a declarative SQL-like query language named *SNQL* (Sensor Network Query Language) for WSNs are provided.

The context aware system paradigm means the capability of a system to adapt according to a rapidly changing context in which it is. That is useful because the system should adapt to context changes in order to react rapidly. In this context, the authors of [59] propose a framework in the context aware architecture (See Fig. 6). This framework exploits a distributed query processor approach for integrating wireless sensor networks. This proposed architecture is based on the *MaD-WiSe* system [60]. This latter includes a set of modules running on the WSN nodes; the *MaD-WiSe* network side and a set modules running on the base station; the MaD-WiSe context information provider. The *MaD-WiSe* network side implements an in-network distributed data stream

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

management system that acts as server; while the *MaD-WiSe* context information provider offers access to the sensor services. The queries are specified using an SQL-like query language named *MW-SQL* that allows users to express queries to manipulate, temporal aggregate, filter, and organize sequences of tuples generated by the sensors. Through the *MaD-WiSe* system interface the user chooses what data he wishes to acquire. An optimized distributed query execution plan is generated and disseminated in the network by the query manager. The latter receives the results of the query stream obtained from in-network query execution in an on-line fashion.

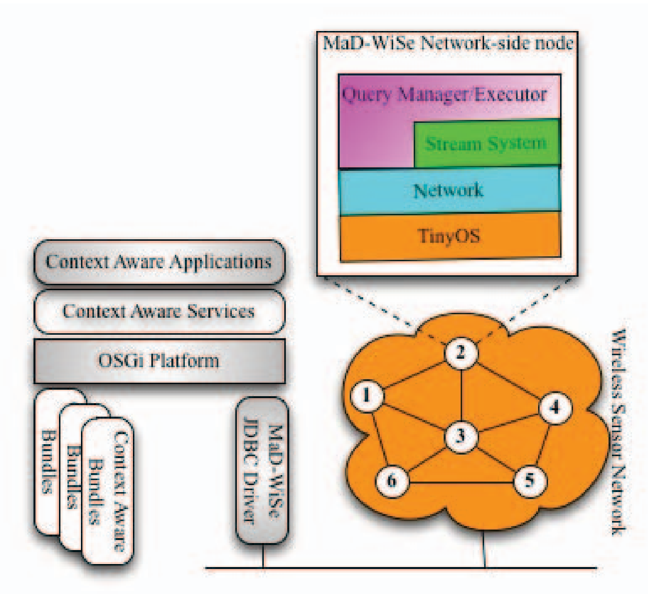

Fig. 6. Context aware architecture using in-network query processing.

Similar to most of the proposed solutions about distributed query processing in WSNs, in [61] the authors propose an in-network aggregate query processing. But rather than using aggregate operators in sensor nodes to reduce the number of transmitted messages, they propose a mechanism that allows queries to share intermediate results together in order to reduce the number of messages transmitted. The main idea assumes that when the sink receives multiple aggregate queries it propagates them via a routing tree. Thus, with this set of query tree, it will decide how to determine a set of backbones and non-backbones and where each non-backbone is allowed to access intermediate results from the backbones in order to reduce the total number of transmitted messages. For that, according to a cost function, they derive a reduction graph and propose two algorithms. First, they propose a heuristic algorithm *BM* (standing for Backbone Mapping), which is based on the reduction graph; it determines the set of backbones and mapping relationships between backbones and non-backbones. Second, they propose an algorithm named *OOB* (standing for Obtaining Optimal Backbones) to obtain the optimal backbone set. Furthermore, they propose a maintenance mechanism for dealing with dynamic scenarios as queries could submit or leave.

Generally, because of wireless sensor network application types (e.g., military battle field, volcano, etc.), sensor nodes are scattered in hostile environments and may relay sensitive data. Moreover like in traditional computer networks, sensor networks are particularly vulnerable to several key types of attacks that can be performed in a variety of ways [62], [63], [64], [65], as Sybil attack [66], traffic analysis attacks [67], node replication attacks [68], and so on. Therefore, security concerns should be addressed efficiently in order to provide reliable communication between nodes and to collect reliable data from the network.

The work in [69] is one of the pioneering wireless sensor network security paradigms. It analyzed the resilience of aggregation techniques for cluster based WSN and proposed a mathematical framework for formally evaluating the security for aggregation, allowing them to quantify the robustness of an aggregation operator against malicious data. However, one can argue that the one-level homogeneous aggregation model is simple to represent real sensor network deployments.

In [70], the authors present a secure information aggregation technique (*SIA*) that helps to defend against a type of attack called the *stealthy attack*. In a *stealthy attack*, the attacker tries to provide incorrect aggregation results to the user without he/she knows that the results are incorrect. Therefore in order to prevent from that type of attack, the mechanism in [70] aims to ensure that if a user accepts an aggregate value as correct, then there is a high probability that the value is close to the true aggregation value. Otherwise, if the aggregate value has been altered, the incorrect results should be rejected with high probability.

The proposal in [71], [72] proposed an energy-efficient and secure pattern-based data aggregation (*ESPDA*) protocol for wireless sensor networks. *ESPDA* is intended for hierarchy-based sensor networks, then a cluster-head first requests sensor nodes to send the corresponding pattern code for the sensed data. If multiple sensor nodes send the same pattern code to the cluster-head, only one of them is permitted to send the data to the cluster-head. ESPDA is secure because it does not require encrypted data to be decrypted by cluster-heads to perform data aggregation.

2. *Approximation based Techniques*

Here statistical techniques like approximation, linear regression, probabilities, etc. are used to approximate data or query answers.

BBQ [73] improves upon TinyDB as it includes certain statistical modeling techniques to answer queries about the current state of the sensor network. The BBQ query system creates models that provide approximated results with probabilistic confidence interval, improving thus the lifetime of the network. Through an SQL like query that includes error tolerances and target confidence bounds, the user chooses what data he/she wants to acquire. The query is parsed by the query-processor, an energy-efficient observation plan is generated, and a time-varying multivariate Gaussians model, which includes correlations and statistical relationships between sensor readings on different nodes, is used to estimate its answer. Unlike TinyDB and COUGAR that interrogate all sensors every time a query is injected into the network, in this query processing, sensors are only solicited to update the data to refine the model if the model itself can't satisfy the query with acceptable confidence.

In [74] the authors adopt a replication solution. However, their proposal is based on the hypothesis of the random nature of failures in wireless sensor networks. In fact, in wireless sensor network areas, due to generally harsh environments of deployment and the resource limitation, sensor nodes are always subject to random failures [38]. For these reasons, in [74] the authors propose a distributed energy-efficient replica placement for increasing data availability and prolonging the network lifetime. The basic idea is, taking into account probabilistic node failures, they compute (based on probabilistic equations) sufficient data replica of nodes with minimal communication cost between nodes such that the whole network data could be reached after a failure. The replication can protect from failures and provide data availability but it should be completed by an efficient data update policy in order to ensure the data freshness.

To satisfy as much as possible the end user requirements as well as to improve the network lifetime, the distributed top-k query processing [75] computes, in a quick and efficient manner, the subset of most relevant answers rather than the all answers. This prevents the transmission of irrelevant answers and minimizes the power cost of retrieving the huge amount of values.

Like the previous work, the distributed processing of probabilistic top-k queries in wireless sensor networks [76], [77] proposes, with bounded rounds of communications in cluster-based wireless sensor networks, three suite of energy-efficient algorithms: sufficient set-based (*SSB*), necessary set-based (*NSB*) and boundary-based (*BB*). Instead of transferring the huge amount of sensor data from the network to the end users, these algorithms return the subset of most relevant data answers efficiently with a constant round of data communications according to a probabilistic weight. This permits to minimize the cost of retrieving all huge values and just transfer relevant answers. Moreover, for better minimizing the communication and energy overhead, this solution proposes also an adaptive algorithm that dynamically switches among the three algorithms based on their estimated costs.

The work in [78] improves the top-k query in which the algorithm tries to find the *k* nodes with highest readings among the sensor nodes, by implementing top-k query in duty-cycled WSNs (*DC-WSNs*). For that, this work solves the underlying data accessibility and network connectivity problems in *DC-WSNs* by proposing a mechanism named *DCDC-WSNs,* where data replication (*DR*) is applied into *DC-WSNs* and the whole combined with a sleep scheduling algorithm named connected k-neighborhood (*CKN*). Thus, the implementation of top-k query in *DCDC-WSNs* can achieve very high query data accessibility at the cost of low total energy consumption and top-k query response time.

As much of the WSN research centers around increasing network lifetime [79], *ENERGY** [80] based on *ENERGY* (Energy Efficient Rate Governed Yardstick) provides an approximate but effective solution to minimize the energy consumption in WSNs. Thus, based on the information about the complete network topology and the Euclidean distance as an estimate to the hop count between two nodes, *ENERGY** provides the optimal placement of the data transformation function which impacts on the energy consumption on data transmission. However, considering this solution for virtual nodes, the proposed algorithm uses the sink node that requires to know the locations and bit rates of all the sources to map the virtual nodes to the real nodes in the network.

In [81] an energy-efficient and accurate estimate is proposed. Hence, the proposal based on a distributed algorithm for in-network data processing provides a new cost function, which is robust to node failure and impulsive noise. The main strategy is to pass around the network a parameter estimate, and along the way small adjustments to the estimate are made by each node based on its local measured data.

Like their predecessors, in [82] the authors propose an efficient approximation algorithm, which improves the network lifetime. The main idea is to optimize the communication cost by performing in-network data aggregations of approximate coefficient. Thus, each node transmits the approximate coefficient upward after it compresses and aggregates the child node's coefficient. Finally, each root node of concerned splay tree obtains the approximate coefficient set about its complete covered area. Therefore, the sink node can query any position of interested area through the root nodes. The approximate is performed by using multiple linear regression models.

B. *Acquisitional Query Processing*

Like in traditional database systems, in distributed database systems, where the requested data might be stored in small fragments around the whole network, the complexities of mechanisms used by the query processor to efficiently extract the relevant data from the network are completely transparent to the end-users.

In WSNs, the query processor is charged, among other tasks, to generate an optimized query execution plan that defines how a query should be executed in an energy-efficient way. This optimization is performed by minimizing the activities of sensor nodes, principally by reducing the data transmission and sampling the sensors that participate in a query processing.

TinyDB [20] is a distributed query processor for sensor networks that has first introduced the management of sensor sampling, called as *acquisitional query processing*. It incorporates a metadata management system that supports optimizations of query processing. In fact, to manage sampling of the sensors for a particular query processing, metadata such as information about the costs of processing and delivering data, the necessary time and energy for that sampling, etc. are periodically copied from the nodes to the root and used by the query optimizer. TinyDB also provides extensions to SQL to formulate queries evaluated when the event specified in the request is made. The main objective is thus avoiding the sending of measures that are not relevant. TinyDB also includes the possibility of using time windows applied to a particular sensor. A time window contains the measures recently performed by the sensor. Moreover, a multi-query optimization on event-based queries is taken into

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

account in order to reduce costs due to transmission and sensor sampling.

Replication [83] allows managing multiple copies that differ at a given time, but eventually converging to the same values [84]. The motivations to make a replication are essentially improved performance, increased data availability and eventually prevent from failures [32], [85], [86].

According to the harsh energy constraints of sensor networks, the authors of [41], [87], [88] propose a hierarchical architecture for in-network data acquisition and replication in mobile sensor networks called *SenseSwarm*. It is composed of two levels: the *perimeter nodes* that perform the data acquisition in energy efficient manner, and the *core nodes*, which are physically and logically strong in order to manage the storage and replication of sensed data (See Fig. 7). In order to increase the fault-tolerance and the availability of the system, a data replication algorithm (*DRA*) is proposed. This algorithm is based on a vote that consists of deciding which set of neighbors will participate to the most energy-efficient replication strategy. Additionally, the *DRA* is extended with a spatial-temporal in-network aggregation strategy based on minimum bounding rectangles. This work leads to a hierarchical data replication algorithm (*HDRA*), which allows an approximate answer.

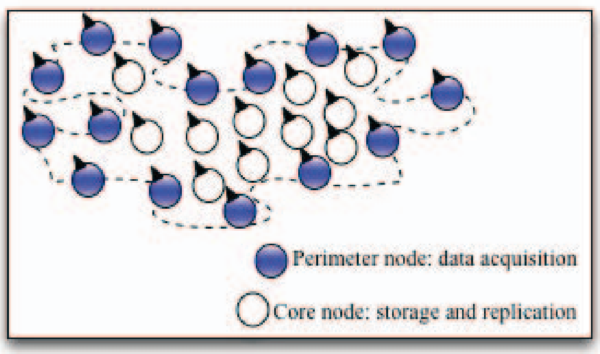

Fig. 7. Framework for the acquisition and storage of spatio-temporal data in mobile sensor networks.

Like most of the above works, in [22] minimizing the energy consumption is a main objective. For that, these works use selectivity awareness technique to optimize the monitoring queries in sensor network. An algorithm called *PDT* (pocket driven trajectories) based on the main features of monitoring queries is used for optimizing the network lifetime. The idea is the use of the data acquisition technique to optimize the communication overhead by sampling sensor nodes (notion of pocketed node participation) for processing a given query, after the setting up of efficient communication paths between the clustered sensor nodes and the base station. This discards the non-selected nodes in the data collection and optimizes the energy consumption. Furthermore, the *PDT* is extended for context aware system. Therefore, it continuously adapts the data collection paths according to changing of node participation and environmental conditions.

In [89] the authors perform a distributed data processing by adapting artificial neural-networks algorithms for wireless sensor networks. For that, three types of cluster based architectures are presented in order to carry out distributed computation and storage, auto-classification of sensor data. The auto-classification of sensor data is performed by using different sensitivity threshold. Thus, depending on the level of details needed at the moment, the corresponding device unit can be queried depending on the level of the sensitivity threshold used to classify the data. The aims of this work are to provide data robustness and improving the network lifetime by optimizing the communication costs and energy savings.

Similarly to TinyDB and Cougar, in [90] the authors have designed a distributed query processing system called *SSDQP*. Through a high-level user interface, the user formulates what data he/she wishes to acquire. The query is decomposed, optimized and distributed by the optimizer across the network. The sensor nodes collect, process the data, and respond to the user query. In this work, each node runs a time-triggered query engine and the queries are scheduled in a distributed manner among the sensor nodes. For improving the network lifetime, the optimizer is parameterized. Thus, according to the user's need, it makes a trade-off between accuracy of time synchronization and consumed energy by choosing an optimized network tree topology. Furthermore, the system performs a synchronized merge operation to minimize the transmitted packet and thus reduce the communication overhead.

Corona [91] system improves upon the *SSDQP* [90] system and the proposal in [92] as it is an in-network distributed query processor that allows sharing sensor readings between several user declarative queries. Through a declarative query interface, several users can choose what sensor data they wish to acquire. With dynamic multi-query execution capability, the query processor can execute concurrently these various queries with different start-times, epochs, and lifetimes. Furthermore, Corona is freshness-aware since it permits to reuse the previous sensor readings from the cache on the sensor node. Hence, if the previous sensor reading differs from the newly required reading by an amount that is less than the query-specified threshold, the query engine will not activate the sensor again. This minimizes the sensor activities. For further improving the network lifetime, instead of sending several small packet messages, corona does in-network clustering of sensor reading results. Furthermore, this clustering operator is resource-aware. Thus, it can dynamically adapt its processing according to the available resources.

Another technique for improving the network lifetime and providing QoS is provided in [93]. In this work, an algorithm that optimizes the execution of continuous queries is provided. Their basic idea, using SQL-like query language, wants to profit from a predefined power and delay weight specified in the query in order to provide an optimal query plan. For that, by taking into account both power and time cost simultaneously, a query plan with minimum cost of computing and data transferring is obtained. Furthermore, to additionally increase the network lifetime, they use reduction techniques like packet merging or data compression to decrease the size of the transferred data into the network.

C. Cross-layer Optimization

The traditional layer approach leads to independent design of different layers and results in strict boundaries between layers. Cross-layer optimization exploits interactions between different layers and can significantly improve energy efficiency as well as adaptability to service, traffic, and environment dynamics. For example, having knowledge of the current physical state will help a channel allocation scheme at the MAC layer in optimizing tradeoffs and achieving throughput maximization.

In WSNs, the sensor network lifetime depends intrinsically on the available energy in the nodes composing the network. This available energy is consumed by the sensing activity, the communication (sending and receiving packets) activity, which is essential to form a WSN, and the data processing [7]. However, the communication activity is more costly in energy than the sensing and processing activities. Hence, current cross-layer optimization techniques use a variety of methods to schedule tasks in an energy-efficient way.

Among these techniques, one can notice the synchronization mechanism used in TinyDB [20] for data transmission between nodes forming the network. In fact, queries in TinyDB are flooded throughout the network. An interval-based communication scheduling protocol is used to collect the query answers via a semantic routing tree, with the root node being the endpoint of the query. Every other node maintains a parent node one step closer to the root from where it is, along with other routing information. The synchronization of the data transmission between nodes is performed by making a parent node in wait for a certain interval of time before reporting its own reading. Specifically, in TinyDB every epoch is subdivided into shorter fixed intervals, with the number of intervals equal to the maximum depth of the routing tree (see figure 3). During its own interval, a parent node will be active and collecting results from its child nodes. In the next interval, the children nodes will be idle, while the parent is still active transmitting the partial aggregate result. The parent node will become idle when it finished receiving and transmitting the partial aggregates in its sub-tree.

Besides performing in-network processing, to increase the network lifetime and the accuracy of data and queries, the *ADAGA*[57], [58]system adapts the collecting and sending activities of the devices according respectively to the remaining memory and energy of sensors. The main idea is to dynamically adapt values for *Sense interval* clause that specifies the interval between consecutive data collections and values for *Send interval* clause that specifies the interval between consecutive sending of packets. The values of these two parameters are updated according respectively to the available memory and power in the sensor nodes. This makes the sensor nodes self-configurable and leads to reduce the power consumption and improve the memory availability.

In wireless sensor networks communicating, the huge amount of sensed raw data between the nodes within the network leads to a lot of problems (including energy wasting, useless data transferring, etc.) because of the limited sensor resources. To deal with these problems, in [94] the authors proposed a distributed and self-organizing scheduling algorithm (*DOSA*) that allows an in-network data aggregation, which is based on spatial and temporal correlations between automatic code generation, the operator placement, and the deployment of an efficient query execution plan. Therefore, sensor readings of neighboring nodes. This will permit to avoid the transmission of redundant data, thus improving the network lifetime. The first function of the *DOSA* algorithm is to decide when a particular node should perform this correlating function. Moreover, According to the eventual changes in the network topology, *DOSA* uses cross-layer information from the underlying MAC layer to detect these changes and autonomously reassigns schedules of nodes.

In [92], the authors present a resource-awareness framework for in-network data processing in wireless sensor networks. This proposal is an enhancement over the distributed query-processing engine, *SSDQP* [90]. This approach is a two-phase approach. Hence, in addition to in-network data processing features from the *SSDQP* system, this proposal adapts to changing resource levels such as battery power or available memory. The main idea is to use a publish/subscribe pattern to distinguish the monitoring of the resource from adaptive algorithms that subscribe to receive resource availability updates. Hence, the processing techniques can subscribe to act according to the remaining resources (battery, memory, and CPU utilization). The published phase is performed whenever the total change in resource level is greater than a given threshold. The second phase consists of integrating the resource-awareness algorithm into the *SSDQP* system. Moreover, to reduce the communication cost, the proposal benefits from the advantage of the multi-tasking of the *SSDQP* to de-couple the in-network data processing and communication.

Like TinyDB [20] and Cougar [21], [49], in [95] the authors propose a complete distributed query processing framework for wireless sensor networks. This framework includes principally two components: a compiler/optimizer, named *SNEE* (Streaming NEtwork Engine) [95], [96], [97], and a continuous declarative query language over sensed data streams, named *SNEEql* [98], [99]. This latter is an expressive and SQL-like query language, convenient to query data stream from wireless sensor networks. After receiving the *SNEEql* queries, the compiler/optimizer engine (*SNEE*) optimizes them by taking into account metadata such as information about the network topology, the required energy and time, the cost of nodes sampling among other relevant parameters. After the optimization phase, the compiler/optimizer engine creates, compiles and deploys an executable code, the energy-efficient query evaluation plan, which will run into the participating nodes. This work fully describes also the *SNEE* query compilation/optimization architecture (See Fig. 8) [95] and the difference steps that it takes to optimize the *SNEEql* queries. Moreover, for more flexibility, the query can be coupled with user QoS requirements, such as the energy consumption of nodes, the tolerated response time, etc.

The work in [100] is an enhancement over *SNEE*. The basic idea is based on declarative query approach to specify, optimize, and deploy automatic data analysis techniques in a sensor network. Thus, at first, the *SNEEql* query language is refactored with in-network data analysis capabilities, so that it can be well optimized by the query optimization engine *SNEE*. Second, the new in-network declarative query is deployed by using the basic functionalities of *SNEE*, which are the this platform can help, in mobile environments, to adjust changes in the network topology efficiently.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

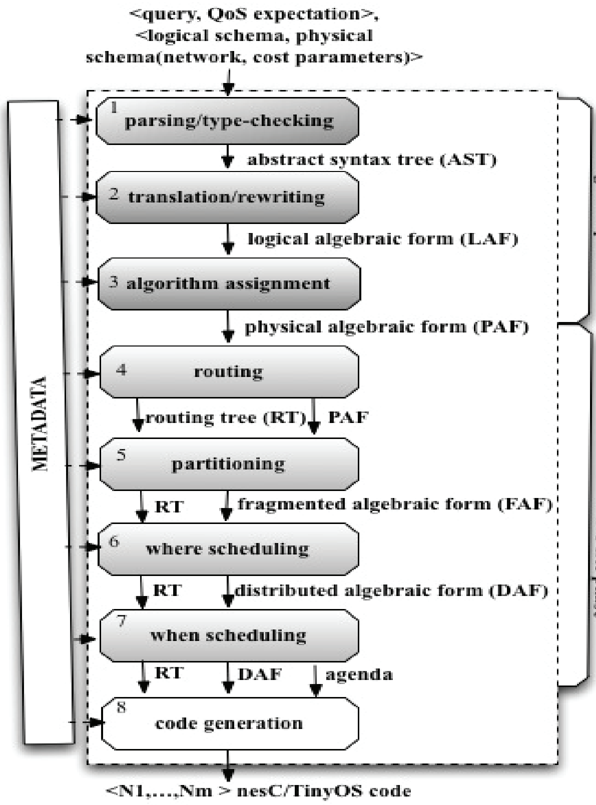

Fig. 8. SNEEql Compiler/Optimizer architecture.

D. Data-centric data/query Dissemination

In data-centric addressing scheme, the collected data is stored by attributes or types (e.g., geographic location and event type) at nodes within the network. Queries for data with a particular attribute will be sent directly to the relevant nodes instead of performing flooding throughout the network; therefore, data-centric approach enables efficient data or query dissemination.

Directed diffusion [101] is one of the pioneering data-centric data dissemination protocols designed specifically for WSNs. The data generated by the sensors are called *events* and queries *interests*. Interests, events and responses of interest are represented by lists of *"attribute = value"*. The interests are injected into the network through one sink node arbitrarily chosen. This sink node broadcasts the interest to its neighbors, which in turn broadcast it to their neighbors, and so on. A node which detects an event compares it with the interests it received, and if there is a match, it sends the description of the event (response to the query) towards the nodes from which it received the interest. Therefore, the event is broadcasted up to the sink node.

In WSNs, the huge amount of information transferring wastes resources. The processing of instructions is much less expensive than the wireless data transmission. Thus, to minimize the communication overhead, in [102] the authors keep, as much as possible, the sensed data in the network and do transmission on demand. To reach their objectives, an API, called *miniDB*, is used to store and extract required data in sensor nodes. In fact, names are associated as identifiers to the data items in order to simplify access. This meta information increases the necessary memory and helps to locate and to retrieve data from arbitrary nodes, even from dynamically placed new nodes, in a simplified manner. Moreover, a tiny query management system, named *miniSQL*, is designed for data identification and interrogation anywhere in the network. An SQL-like query is used to interact with the network. This solution allows to access historic data stored in the sensor network but in order to prevent from the memory saturation it should include an efficient policy of memory saturation management.

Generally, according to natural conditions of environment, the sensor nodes are subjected to failures. Thus, in addition to take resource constraints of WSNs into account, proposed solutions should be aware of risk of sensor failures. For that, the *DISC* (Distributed Information Storage and Collection for WSNs) is proposed in [103]. *DISC* is a protocol for distributed information storage and collection. In *DISC* protocol the network is logically divided into clusters, where each of them is managed by a cluster head. There are two specifics cluster heads; *the primary cluster head (PCH*) and *the backup cluster heads* (*BCH*). The sensor data are periodically reported to the *PCH*, which in turn after eventual aggregation transmits the data to the *BCHs* chosen in a probabilistic way. The *DISC* protocol, compared with the other protocols in this area that make deterministic choice of the backup nodes, performs a random selection strategy. This can prevent from destruction of the exact nodes keeping important information. Moreover, the *DISC* protocol uses the *Bloom filters* method for the data description and the routing mechanism. The data descriptor is unique and it includes the time epoch of the data aggregation, the region identifier of the aggregator and the type of data. Although this protocol is interesting for the robustness of the network, it should be revised in order to take the severe resource constraints of WSNs into account. In fact, the complexity of the algorithms with various backup nodes can be negative factors for the network lifetime and the low memory resource.

Similarly to previous works that investigate the improvement of the network lifetime, the *DDCRS* [104] (dynamic data-centric routing and storage) mechanism minimizes the communication overhead by dynamically determining the locations of the data-centric nodes according to the multiple sink nodes' location and data collecting rate. Moreover, this proposed scheme automatically constructs shared routes from data-centric nodes to multiple sinks, reducing thus duplicate packets transmission. The *DDCRS* scheme consists of two phases: a static phase which consists of defining a data-centric node, called home data-centric responsible for storing sensed data and future replying to sink nodes, by a hash table. In fact, sensor nodes transmit sensed data to neighboring data-centric nodes through *GPRS* routing algorithms. The dynamic phase applies to handle data storage and delivery when the locations of the data-centric node change according to the locations and the reply frequencies of multiple sink nodes.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

In [105] a distributed spatial-temporal Similarity Data Storage (*SDS*) scheme is provided. For that, the monitoring area is partitioned into zones based on the geographical layout. Nodes are deployed according to zones and each clustered nodes of a particular zone has a head, which acts as a server for all the remaining nodes. According to a hash function the sensed data are mapped and stored in certain nodes where similarity of data in the zones is preserved. Thus, a data search in a particular zone could find similar data in the neighboring zones. In addition to a search based on data similarity, *SDS* also allows spatial-temporal data searching as data stored in the neighborhood is also based on time and location. This further optimizes the data querying. Moreover, *SDS* uses geographical information of nodes for routing, instead of based on *GPS*. It also provides a carpooling routing algorithm, which can combine the messages belonging to the same destination in routing for minimizing the communication overhead.

Data-centric storage provides energy-efficient data dissemination and organization in wireless sensor networks. Moreover, using metadata is useful for efficient query routing and data access. To support all these requirements, in [106], [107] the authors propose a framework to accelerate query evaluation in content caching networks using XML metadata. This framework includes a concept of content caching networks in which, the collected data are stored by their contents in a distributed way and the data are cached in the network for a certain period of time before they are sent to a database server. Moreover, this framework includes a metadata-guided query evaluation mechanism to improve the query execution. Thus, each cache node maintains a metadata of its data content. Therefore, queries will be based on these metadata to efficiently access the relevant cached data. For the representation of the metadata, they use an XML representation. Additionally to deal with the memory overhead, this work proposes two data clustering and compression algorithms for metadata construction: a clustering, balancing, and compression (*CBC*) algorithm for numerical data, and a clustering, expanding, and compression (*CEC*) algorithm for categorical data.

IV. Discussion and Open Issues

The distributed database approach for sensor networks is adopted when sensor nodes do not need to send periodically the collected data to the base station. The sensed data remains on the sensor nodes and some queries are distributed and processed in the sensor nodes. Thus, in this case, the whole sensor network is viewed as a distributed database. This approach is commonly called *in-network processing*.

The in-network processing is a generic term, which could simply refer to any sort of processing that takes place inside a node. Therefore, the data reduction techniques (data aggregation, packet merging, data compression techniques ,data fusion, and approximation based techniques) can be placed in the class of in-network processing techniques because each of these techniques is done within nodes composing the network.

Moreover, one can notice that the three approaches (acquisitional query processing, cross-layer optimization and data-centric data/query dissemination) can be all classed inside the category of in-network processing, since all these three sub-categories require a sensor node to process data and make decision instead of transmitting all the data to a central server for off-line processing, as it is done in in-network processing [39].

The tables considered above (Table I, Table II, and Table III) provide a summary and offer a comparison between the above-described distributed database management solutions for WSNs. Moreover, these tables highlight the classification of each solution into specific categories.

After the detailed analysis of the most recent distributed data management techniques for sensor networks, the following open issues can be identified and suggested:

- Keeping and processing sensed data among the sensor nodes are very useful, according to the fact that in-network processing is more energy-efficient than transmitting sensed data for off-line processing. However, this approach may be revised and improved to take into account of the memory overload of the sensor nodes according to the huge amount of generated data compared with the hard resource constraints. Moreover, for efficient query execution in terms of saving energy and good latency, an efficient load balancing policy according to the remaining power and the load of the nodes can be adopted.
- This above study shows that most of the sensor data representations are based on the relational model and the queries are specified using an SQL-like query language. According to the frequent changing characteristics of collected data, the sensor data schema should well fit into XML representation. Then, XPATH can be used for flexible and easy queries.
- The distributed technique treats the information within sensor nodes. According to the unstable and generally harsh environment, there may be sudden failures of sensors. This can lead to information loss that greatly influences the result analysis or even the system blocking. To deal with this situation, one can opt for a hybrid approach where, in addition to in-network data processing, some data can be kept in the base station database and updating from time to time. After a certain time, these data may gradually lose their freshness. Thus, one can use a generalized database management system to handle various types of applications and user needs.
- Efficient meta-data management is very useful when one can manage a huge amount of distributed data. Although TinyDB and the work in [22], [107] include meta-data management for query optimization, it is not enough. Thus, proposals may be improved by including energy-efficient meta-data management in order to have fast and accurate routing.
- In multi-users environment, simultaneous multi-query processing is very useful. Systems connected to a database are often subject to numerous queries especially for consultation. Hence, a good policy of multi-query optimization would be a good deal.

Table I – Summary to various kinds of proposals and their features (part 1)

	Characteristics				
Name of Project/ Authors	**Proposal approach**	**Basic conceptual characteristics**	**Queries expression**	**Optimization / Metrics**	**Platform**
Cougar, 2002, 2003	Distributed in-network query processor	• In-networking processing: aggregation based techniques, packet merging; • Clustered approach	Declarative SQL-like query	•Power-aware optimization: Query optimization plan; catalog management; aggregation in cluster heads; Minimizing Communication overhead	Simulation PDA class
TinyDB, 2002, 2005	Distributed Acquisitional Query Processing System	•In-network processing: aggregation based techniques; •Acquisitional query processing: sensor sampling, query optimization based on metadata, multiquery optimization; •Cross-layer optimization: data stream communication scheduling, self-organization using time interval; •Data-centric query/data dissemination: semantic routing tree	Declarative SQL-like query	•Power-aware optimization: Query optimization plan: Metadata management, ordering of sampling, aggregation; Minimizing the Communication overhead	Simulation
TINA, 2003, 2004	Temporal Coherency-Aware In-network Aggregation	•In-network processing: temporal freshness-aware readings; transmit readings if only threshold defined is exceeded: trade-off between the quality of transmitted data and energy consumption	Declarative SQL-like query	•Power-aware optimization: coherence-aware in-network aggregation, optimize the communication cost	Simulation
Yu et al., 2004	Energy-Latency Tradeoffs for data gathering in WSNs	• In-networking processing: aggregation based techniques • Minimize energy dissipation: energy-latency tradeoffs	---	•Power-aware optimization: energy-latency tradeoffs	Simulation
ADAGA, 2007, 2008	Resource-aware in-network aggregation operator	•In-network aggregation, packets replication •Approximation based techniques •Cross-layer optimization: collecting and sending data according respectively to the remaining memory and energy of sensors	SQL-like query: SNQL	•Power-aware optimization: In-Network Aggregation, resource-aware processing; •Optimize data access: packets replication	Simulation
Amato et al., 2007	Context-aware architecture for distributed query processing	•Framework in the context-aware architecture •In-network query processing	SQL-like query	•Power-aware optimization: In-Network Aggregation	---
Hung and Peng, 2011	Mechanism of optimizing in-network aggregate queries	•In-network processing: in-network aggregation, in-network sharing of intermediate results •Acquisitional query processing: sampling by access intermediate results from others sensors.	---	•Power-aware optimization: Minimizing communication overhead; •Multi-query optimization	Simulation
Wagner, 2004	Resilient aggregation techniques for cluster based WSN	• Aggregation based techniques • Security: framework to evaluate the security of data aggregation schemes	---	• Maximize the resilient of data aggregation techniques	Mathemati cal theory
Sia, 2003	Secure aggregation technique against stealthy attack	• Aggregation based techniques • Security: framework for secure data aggregation against stealthy attack; user accepts aggregated result according to a given bound	---	• Ensure the reliability of data aggregation results	---
ESPDA, 2003, 2005	Energy-efficient and secure pattern-based data aggregation for WSNs	• Aggregation based techniques • Cluster-based approach • Security: encrypted data not be decrypted by cluster heads when performing data aggregation	---	•Power-aware optimization: prevent from the transmission of redundant data •Sensor data transmitted to base station in encrypted form	Simulation
BBQ, 2004	Framework to acquire data using statistical modeling techniques	•Approximation based techniques: approximate sensor readings using statistical models; temporal and correlation correlations	SQL-like query with error tolerance and target confidence	•Power-aware optimization: Minimizing the communication cost and the sensor activities, sampling	Simulation

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

Table II – Summary to various kinds of proposals and their features (part 2).

	Characteristics				
Name of Project/ Authors	**Proposal approach**	**Basic conceptual characteristics**	**Queries expression**	**Optimization / Metrics**	**Platform**
Yulong et al., 2011	Distributed replication schemes	•In-network data storage and replication(fault-tolerance); •Probabilistic based techniques	---	•Power-aware optimization: minimize communication overhead; •Maximize fault-tolerance	---
Zeinalipour-Yazti et al., 2008	Distributed top-k query processing	•In-network processing: approximation based technique	Top-k query	•Power-aware optimization: Data Retrieving cost minimizing	---
Ye et al., 2010, 2011	Framework for distributed processing of probabilistic top-k queries	•In-network processing: approximation based technique, aggregation based techniques, algorithms adapted according to the communication cost	Probabilistic Top-k query	•Power-aware optimization: Data Retrieving cost minimizing, minimizing of the communication cost.	Experiment
ENERGY*, 2006	Optimal placement problem of proxy node	•In-network data processing: approximation based technique	---	•Power-aware optimization: minimize the energy consumption on data transmission	Analytic Simulation
Panda *et al.*, 2010	Distributed data processing algorithms that prevent from link failure and noise	•In-network processing •Approximation based techniques	---	•Power-aware optimization: Optimize the communication cost; Cost function optimization	Simulation
ShuKui et al., 2009	Approximation Algorithm for Data Aggregation	•In-network processing: aggregation based techniques; •Approximation based techniques	SQL-like query	•Power-aware optimization: Optimize the communication cost	Simulation
SenseSwarm, 2007, 2009, 2011	Framework for in-network data acquisition and replication in mobile WSN	•Hierarchical voting-based fault-tolerance architecture; •In-network data acquisition and replication(fault-tolerance); •spatio-temporal in-network aggregation; •Approximation based techniques	---	•Power-aware optimization: in-network aggregation; •Maximize fault-tolerance	Simulation
Umer et al., 2009	Selectivity-awareness query optimization	•In-network data processing: in-network aggregation •Acquisitional techniques: queries sample spatially correlated nodes •Data-centric data/query dissemination •Context aware system	SQL-like query	•Power-aware optimization: Optimize energy of the data collection process	Simulation
Kulakov and Davcev, 2005	Distributed data processing based on artificial neural-networks algorithms	•In-network data processing; •Artificial neural-networks algorithms adaptation for WSN	---	•Power-aware optimization: Optimize the communication cost	Simulation
SSDQP, 2007	Time Triggered Query Processing system	•In-network processing: synchronized merge operation, data compression method, aggregation based techniques; •Acquisitional query processing: time-triggered query engine, query optimization based on user's need •Cross-layer optimization: task scheduler, sensing and process before transmitting are synchronized	SQL queries	•Power-aware optimization: Minimizing Communication overhead	Simulation
Corona, 2010	Distributed multi-query processor	•In-network aggregation based techniques, sensor reading clustering; •Acquisitional query processing: freshness-aware data acquisition •Cross-layer optimization: task scheduler	Acquisitional SQL queries	•Power-aware optimization: Optimize the communication cost and the sensor activities	Simulation
Sun and Zhou, 2008	Power-aware query execution plan	•Distributed multi-query engine •In-network processing: packet merging or compression based techniques; •Acquisitional query processing: power-aware data sampling;	SQL-like query	•Power-aware optimization: Optimize the communication and execution cost	Analytic experiment

Table III - Summary to various kinds of proposals and their features (part 3).

	Characteristics				
Name of Project/ Authors	**Proposal approach**	**Basic conceptual characteristics**	**Queries expression**	**Optimization / Metrics**	**Platform**
DOSA, 2008	A distributed and self-organizing scheduling algorithm for in-network data aggregation aware of correlated data	•In-network data processing: in-network aggregation, approximation based technique •Cross-layer optimization: self-organization scheduling based on the MAC layer	---	•Power-aware optimization: avoid sending redundant data by in-network aggregation aware of correlated data	Theoretical Simulation results
Rohm, 2008	Resource-awareness framework for in-network data processing	•Resource-awareness framework; •In-network data processing: on-line data clustering, merge operation; •Acquisitional query processing: time-triggered query engine, query optimization based on user's need •Cross-layer optimization: task scheduler, sensing and communication are synchronized	SQL queries	•Power-aware optimization: Optimize the communication cost	Simulation
SNEE, 2008, 2009, 2011	Distributed query processing framework for WSNs	•In-network aggregation based techniques; •Acquisitional query processing: sensor sampling, query optimization based on metadata; •Cross-layer optimization: tasks scheduling based on available resources, communication scheduling for data flow •Data-centric query/data dissemination: choice of routing tree nodes based on semantic constraints	SNEEql queries	•Power-aware optimization: Query optimization plan: Metadata management, sampling, aggregation; energy-efficient communication strategy, tasks scheduling based on remaining energy on nodes	Empirical evaluation, Simulation
Valkanas et al., 2011	In-network query processing based on SNEE	•In-network processing: adjust of network topology changes; Data analysis techniques	Data analysis queries based on SNEEql	•Optimization based on SNEE	---
Directed Diffusion, 2002	Reactive data-centric protocol	•In-network processing: filters, suppression of duplicate messages Data analysis techniques •Data-centric query/data dissemination: interest dissemination based on named data, gradient setup used to route data back to the sink node	---	•Optimization based on filter, suppression of duplicate messages	---
Awad et al., 2008	Distributed data management system	•In-network data processing: data transmission on demand; •Utilization of an API: miniDB/miniSQL; •Data-centric query/data dissemination	SQL-like query	•Power-aware optimization: Minimizing the communication overhead	Simulation
DISC, 2007	Distributed data storage and collecting protocol	•backup cluster heads, in-network aggregation; •Randomly selection strategy of backup nodes: lead network robustness; •Data descriptor: Bloom filters technique; •Data-centric data/query dissemination	SQL-like query	---	Formal analysis Simulation
DDCRS, 2010	Frequency-aware data-centric routing and storage	•In-network processing •Data-centric Storage	---	•Power-aware optimization: Minimizing Communication overhead	Simulation
SDS, 2011	Distributed spatial-temporal similarity data storage scheme	•Distributed data-centric storage: semantic routing, spatial-temporal and similarity data searching	---	•Power-aware optimization: Minimizing communication overhead;	Simulation
Liu et al., 2009, 2011	Framework to accelerate query evaluation in content caching networks using XML metadata.	•In-network aggregation; •Acquisitional query processing: sampling nodes by metadata guided; •Distributed data-centric storage	XPath queries	•Power-aware optimization: in-network aggregation; •Improving query execution	Simulation

- Another important research issue is the database security. Although research efforts have been made on security issues in wireless sensor network in general and particularly in security on existing data gathering protocols, some challenges can still be addressed. First, WSNs are application-specific and the choice of the appropriate cryptographic methods depends on the processing capability of sensor nodes, then there is no unified solution for all sensor networks. Therefore, the design of a security mechanism adaptable to various WSN applications could be interesting. Second, WSN are resource constraints, so the design of security services in WSNs must be aware of these constraints. Third, most of security protocols are designed for fixed topologies. Whereas with applications based on mobile sensors, the mobility of the devices influences the sensor network topology, therefore leads to many problems in secure routing protocols.

V. Conclusion

Given the great limited sensor device resources, storing and exploiting the large amount of data generated by sensors is a very big problem. The data management techniques used in traditional databases to manage a huge amount of data are not generally suitable for sensor networks taking their specificities into account. Then, the research community has provided a new data storage and querying method, named the distributed database approach for sensor networks. This latter is viewed as the most energy-efficient method to manage the large amount of data generated by the sensor nodes.

Many different techniques have been proposed to manage data and queries in a distributed architecture, but all these techniques can be categorized by their way of processing data into the network (in-network aggregation processing, in-network approximation processing, acquisitional query processing, cross-layer optimization, data-centric data/query dissemination). Thus, this work presented and discussed from the oldest to the most recent proposed techniques that have been performed and it is particularly interesting in various stages of distributed data storage, distributed query processing and optimization for sensor networks. The processing techniques aim, generally, to optimize the energy consumption in the network and to retrieve more accurate information. Moreover, the queries used in these techniques are generally SQL-like.

Systems based on sensor networks are increasingly common in many areas of the knowledge, giving rise to several flavors of WSN applications. These applications are generally specific. For instance, there are critical applications that have temporal constraint and need more accurate information in order to take efficient decisions. New query processing optimization technique, while ensuring data availability in case of failure is very important in this context. So, one processing technique may not be efficient for the different applications. To deal with these challenges one can opt for a hybrid approach. Furthermore, regarding the Tables I, II, and III, few proposals have been based on metadata management, which is very useful when one manages a huge amount of distributed data. The design of a framework that takes into account the various particular characteristics of WSN applications can well meet the requirements to a generalized database management system to handle various types of applications and user needs.

Acknowledgments

This work was partially supported by the *Instituto de Telcomunicações*, Next Generation Networks and Applications Group (NetGNA), Portugal, by the *Ministerio de Ciencia e Innovación*, through the *Plan Nacional de I+D+i 2008–2011* in the *Subprograma de Proyectos de Investigación Fundamental*, project TEC2011-27516, by the Polytechnic University of Valencia, though the PAID-05-12 multidisciplinary projects, and by National Funding from the FCT – *Fundação para a Ciência e a Tecnologia* through the Pest-OE/EEI/LA0008/2011 Project.

References

[1] L. Shu, M. Hauswirth, L. Cheng, J. Ma, V. Reynolds, and L. Zhang, “Sharing Worldwide Sensor Network,” International Symposium on Applications and the Internet, IEEE, pp. 189-192, 2008, DOI 10.1109/SAINT.2008.22.

[2] K. Aberer, M. Hauswirth, and A. Salehi, “Global Sensor Networks,” TechnicalreportLSIR-REPORT-2006-001, http://lsir.epfl.ch/research/current/gsn/, 2006.

[3] K. Aberer, M. Hauswirth, and A. Salehi, “Infrastructure for Data Processing in Large-Scale Interconnected Sensor Networks,” in Proceedings of the 8th International Conference on Mobile Data Management, Mannheim, Germany, May 7-11, 2007.

[4] E. H. Callaway, “Wireless Sensor Networks: Architectures and Protocols,” CRC Press, 2004.

[5] P. Baronti, P. Pillai, C: V. Chook, S. Chessa, A. Gotta, and F. Y. Hu, “Wireless sensor networks: A survey on the state of the art and the 802.15.4 and ZigBee standards,” Journal of Computer communications, 30 (7), 1655–1695, 2007.

[6] L. M. L. Oliveira and J. J. P. C. Rodrigues, “Wireless Sensor Networks: a Survey on Environmental Monitoring,” Journal of Communications (JCM), Vol 6, No 2, pp. 143-151, April 2011, DOI: 10.4304/jcm.6.2.143-151

[7] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “Wireless sensor networks: a survey,” Computer networks, 38, 393-422, 2002.

[8] A. Nayak, I. Stojmenovic, “Wireless sensor and actuator networks: algorithms and protocols for scalable coordination and data communication,”Wiley-Interscience,2010.

[9] K. Lin, M. Chen, S. Zeadally, and J. J. P. C. Rodrigues, “Balancing Energy Consumption with Mobile Agents in Wireless Sensor Networks,” Future Generation Computer Systems, Vol. 28, Issue 2, pp. 446-456, Feb. 2012, DOI: 10.1016/j.future.2011.03.001,

[10] K. Lin, J. J. P. C. Rodrigues, H. Ge, N. Xiong, and X. Liang, “Energy Efficiency QoS Assurance Routing in Wireless Multimedia Sensor Networks,” IEEE Systems Journal, Vol. 5, Issue 4, pp. 495-505, December 2011, DOI: 10.1109/JSYST.2011.2165599

[11] S. Sendra, Jaime Lloret, Miguel Garcia and Jose F. Toledo, “Power saving and energy optimization techniques for Wireless Sensor Networks,” Journal of Communications, Vol. 6 Issue 6, Pp. 439-459. August 2011. doi:10.4304/jcm.6.6.439-459

[12] S. Matsumae, “Energy-Efficient Cell Partition of 3D Space for Sensor Networks with Location Information,” Network Protocols and Algorithms, Vol 1, No 2 (2009), Pp. 85-98

[13] P. A. Neves, J. J. P. C. Rodrigues, M. Chen, and A. V. Vasilakos, “A Multi- channel Architecture for IPv6-enabled Wireless Sensor and Actuator Networks Featuring PnP Support,” Journal of Network and Computer Applications, 2011, (in press), DOI: 10.1016/j.jnca.2011.03.033

[14] L. M. L. Oliveira, J. J. P. C. Rodrigues, A. F. de Sousa, and J. Lloret, “Network Access Control Framework for 6LoWPAN Networks,” Sensors, Vol. 13, Issue 1, pp. 1210-1230, 2013. DOI: 10.3390/s130101210.

[15] J. M. L. P. Caldeira, J. J. P. C. Rodrigues, and P. Lorenz, “Towards Ubiquitous Mobility Solutions for Body Sensor Networks on HealthCare,” IEEE Communications Magazine, Vol. 50, No. 5, pp. 108-115, May 2012.

[16] A. Campos, E. Souza, F. Nakamura, J. J. P. C. Rodrigues, and E. Nakamura, “On the Impact of Localization and Density Control Algorithms in Target Tracking Applications for Wireless Sensor Networks,”, Sensors, Vol. 12, No. 6, pp. 6930-6952, May 2012, DOI: 10.3390/s120606930

[17] L. Sacks, M. Britton, I. Wokoma, A. Marbini, T. Adebutu, I. Marshall, C. Roadknight, J. Tateson, D. Robinson, A. G. Velazquez, “The development of a robust, autonomous sensor network platform for environmental monitoring,” Sensors and their Applications XXII, Limerick, Ireland, 2003.

[18] L. Shu, Y. Zhang, G. Min, Y. Wang, M. Hauswirth, "Cross-Layer Optimization on Data Gathering in Wireless Multimedia Sensor Networks within Expected Network Lifetime", accepted in Springer Journal of Universal Computer Science (JUCS), 2009.

[19] C. Zhu, L. Shu, T. Hara, L. Wang, S. Nishio, L.T. Yang, “A survey on communication and data management issues in mobile sensor networks,” Wireless Communications and Mobile Computing, 10 Nov. 2011. DOI: 10.1002/wcm.1219

[20] S. R. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong, “TinyDB: An Aquisitional Query Processing System for Sensor Networks,” ACM Transactions on Database Systems (TODS), 30 (1), 122-173, 2005.

[21] Y. Yao and J. Gehrke, “Query Processing in Sensor Networks,” In First Biennial Conference on Innovative Data Systems Research (CIDR 2003), Asilomar, California, January 5-8, 2003.

[22] M. Umer, L. Kulik, and E. Tanin, “Optimizing query processing using selectivity-awareness in Wireless Sensor Networks,” Journal of Computers, Environment and Urban Systems, 33 (2), 79-89, 2009.

[23] L. Cheng, Y. Chen, C. Chen, J. Ma, L. Shu, A.V. Vasilakos, N. Xiong, “Efficient Query-Based Data Collection for Mobile Wireless Monitoring Applications,” The Computer Journal, 53 (10), 1643-1657. 2010.

[24] J. G. Pottie and J. W. Kaiser, “Wireless integrated network sensors,” Communication of the ACM, 43(5), pp. 51-58, 2000.

[25] R. Kacimi,“Techniques de conservation d'énergie pour les réseaux de capteurs sans fil,” Ph.D. dissertation, Dept. Math., Info. and Telecom., INPT Univ., Toulouse, France, September 2009.

[26] C. Dini,“Les Réseaux Capteurs Sans Fil Avec Access Sporadique Au Noeud-puits,” Ph.D. dissertation, Info. Eng., Haute Alsace Univ., France, December 2010.

[27] S. Madden, M. Franklin, J. Hellerstein, and W. Hong, “TAG: a tiny aggregation service for ad-hoc sensor networks,”ACM Symposium on Operating System Design and Implementation, OSDI 2002, Boston, MA, USA, pp.131-146, 2002.

[28] E. Fasolo, M. Rossi, J. Widmer, and Michele Zorzi, “In-network aggregation techniques for wireless sensor networks: a survey,” IEEE Wireless Communications, 14(2), pp. 70-87, April 2007.

[29] C. Tang and C. S. Raghavendra, “Compression techniques for wireless sensor networks,” Wireless Sensor Networks, pages 207-231, 2004.

[30] C. Dini and P. Lorenz, “Primitive Operations for Prioritized Data Reduction in Wireless Sensor Network Nodes,” 4th International Conference on Systems and Networks Communications, ICSNC 2009, Porto, Portugal, pp. 274-280, 2009.

[31] J. Xu, X. Tang, and W. C. Lee, “A New Storage Scheme for Approximate Location Queries in Object Tracking Sensor Networks,” IEEE Transactions on Parallel and Distributed Systems (TPDS), 19(2): 262-275, Feb. 2008.

[32] C. J. Debono, and N. P. Borg, “The Implementation of an Adaptive Data Reduction Technique for Wireless Sensor Networks,” IEEE International Symposium on Signal Processing and Information Technology (ISSPIT), Sarajevo, 16-19 Dec. 2008.

[33] L. Shu, J. Lloret, J. J. P. C. Rodrigues, M. Chen, “Distributed Intelligence and Data Fusion for Sensor Systems,” IET Communications, Vol. 5, Issue 12. Pp. 1633-1636. October 2011.

[34] P. Bonnet, J. Gehrke, and P. Seshadri, “Towards sensor database systems,” 2nd International Conference on Mobile Data Management, Hong Kong, China, January 8-10, 2001.

[35] O. Diallo, J. J. Rodrigues, and M. Sene, “Real-time data management on wireless sensor networks: A survey,” Journal of Network and Computer Applications, 35 (3), pp. 1013-1021, May 2012.

[36] C. Li, H. Zhang, B. Hao, and J. Li, “A Survey on Routing Protocols for Large-Scale Wireless Sensor Networks,” Sensors, 11(4), 3498-3526, 2011.

[37] J. N. Al-karaki and A. E. Kamal, “Routing techniques in wireless sensor networks: a survey,” IEEE Wireless Communications , 11 (6), 6-28, 2004.

[38] M. Demirbas, X. Lu, and P. Singla, "An in-network querying framework for wireless sensor networks," IEEE Trans. on Parallel and Distributed Systems (TPDS), vol. 20, no. 8, pp., 1202-1215, 2009.

[39] S. Chatterjea and P. Havinga, “A Taxonomy of Distributed Query Management Techniques for Wireless Sensor Networks,” International Journal of Communication Systems, 20 (7), 889-908, 2007.

[40] A. Girighar and P. R. Kumar, “Toward a theory of in-network computation in wireless sensor networks,” IEEE Communications Magazine, 44 (4), 98-107, 2006.

[41] P. Andreou, D. Zeinalipour-Yazti, P. K. Chrysanthis, and G. Samaras, “In-network data acquisition and replication in mobile sensor networks,” Distributed and Parallel Databases, 29, 87–112, 2011.

[42] B. Radunovic, “A Cross-Layer Design of Wireless Ad-Hoc Networks,” Thesis, 2005.

[43] L. D. Mendes and J. J. Rodrigues, “A survey on cross-layer solutions for wireless sensor networks,” Journal of Network and Computer Applications. Volume 34, Issue 2, March 2011, Pp. 523–534.

[44] L. M. L. Oliveira, A. F. de Sousa, and J. J. P. C. Rodrigues, “Routing and Mobility Approaches in IPv6 over LoWPAN Mesh Networks,” International Journal of Communication Systems, Vol. 24, Issue 11, pp. 1445-1466, November 2011. DOI: 10.1002/dac.1228.

[45] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, and F. Silva, “Directed diffusion for wireless sensor networking,”IEEE/ACM Transactions Networking, 11 (1), 2-16, 2003.

[46] S. Shenker, S. Ratnasamy, B. Karp, R. Govindan, and D. Estrin, “Data-Centric Storage in Sensornets,” CM SIGCOMM Computer Communication Review, 33 (1), 137-142, 2003.

[47] Y. Diao, D. Ganesan, G. Mathur, and P. Shenoy, “Rethinking data management for storage-centric sensor networks,” Third Biennial Conference on Innovative Data Systems Research (CIDR’07),Asilomar, CA, USA, Jan. 2007, pp. 22–31.

[48] M. Bin Ahmad, M. Hasan Islam, M. Asif, and Dr. Sadia Aziz, “A Short Survey on Distributed In-Network Query Processing in Wireless Sensor Networks,” Networked Digital Technologies (NDT), IEEE, pp. 541-543, 2009.

[49] Y. Yao and J. Gehrke, “The cougar approach to in-network query processing in sensor network,” ACM SIGMOD Record, 31 (3), 9-18, 2002.

[50] S. Madden, W. Hong, J. Hellerstein, and K. Stanek, “TinyDB: A Declarative Database for Sensor Networks,” accessed 25/01/2013 Available: http://telegraph.cs.berkeley.edu/tinydb/overview.html

[51] Berkeley. TinyOS. Available: http://www.tinyos.net/, 1999.

[52] P. Bonnet, J. Gehrke, and P. Seshadri, "Cougar: The Sensor Network is the Database," accessed 25/01/2013 Available: www.cs.cornelle.edu/database/cougar/, 2001.

[53] M. A. Sharaf, J. Beaver, A. Labrinidis, and P. K. Chrysanthis, “TiNA: A Scheme for Temporal Coherence-Aware in-Network Aggregation,” 3rd ACM international workshop on Data engineering for wireless and mobile access MobiDe '03, pp. 69–76, San Diego, California, USA, 2003.

[54] A. Sharaf, J. Beaver, A. Labrinidis, and K. Chrysanthis, “Balancing energy efficient and quality of aggregate data in sensor networks,” The International Journal on Very Large DataBases (VLDB), 13 (4), 384 – 403, 2004.

[55] J. Beaver, M. A. Sharaf, A. Labrinidis, and P. K. Chrysanthis, “Power-aware in-network query processing for sensor data,” Second hellenic data Management Symposium (HDMS), pp. 1-17. Athens, Greece, 2003.

[56] Y. Yu, B. Krishnamachari, and V. Prasanna, "Energy-Latency Tradeoffs for Data Gathering in Wireless Sensor Networks," in IEEE Infocom 2004, Hong Kong, Mar. 2004.
[57] A. Brayner, A. Lopes, D. Meira, R. Vasconcelos, and R. Menezes, "An adaptive in-network aggregation operator for query processing in wireless sensor networks," The journal of Systems and Software , 81 (3), 328–342, 2008.
[58] A. Brayner, A. Lopes, D. Meira, R. Vasconcelos, and R. Menezes, "Toward adaptive query processing in wireless sensor networks," Signal Processing, 87 (12), 2911–2933, 2007.
[59] G. Amato, F. Furfari, S. Lenzi, and S. Chessa, "Enabling Context Awareness through Distributed Query Processing in Wireless Sensor Networks," 2nd International Workshop on Requirements and Solutions for Pervasive Software infrastructures, Innsbruck, Austria, September 16, 2007.
[60] G. Amato, S. Chessa, A. Caruso, and P. Baronti, "MaD-WiSe: Management of Data in Wireless Sensor networks", 2005. Available: http://mad-wise.isti.cnr.it, accessed 25/01/2013
[61] C. Hung, and W. Peng, "Optimizing in-network aggregation queries in wireless sensor networks for energy saving," Data and Knowledge Engineering, 70 (7), 617-641, 2011.
[62] J. P. Walters, Z. Liang, W. Shi, and V. Chaudhary, "Wireless Sensor Network Security: A Survey," Security in Distributed, Grid, and Pervasive Computing, pp. 1-50, Auerbach Publications, CRC Press, Chapter 17, 2006.
[63] P. Mohanty, S. Panigrahi, N. Sarma, and S. S. Satapathy, "Security Issues In Wireless Sensor Network Data Gathering Protocols: A Survey," Journal of Theoretical and Applied Information Technology, 13(1), pp. 14-27, 2010.
[64] J. Sen, "A Survey on Wireless Sensor Network Security," International Journal of Communication Networks and Information Security (IJCNIS),1(2), August 2009.
[65] S. K. Singh, M. P. Singh, and D. K. Singh, "A Survey on Network Security and Attack Defense Mechanism For Wireless Sensor Networks," International Journal of Computer Trends and Technology (IJCTT), pp. 1-9, May to June Issue 2011
[66] J. Newsome, E. Shi, D. Song, and A. Perrig, "The sybil attack in sensor networks: analysis & defenses," In Proceedings of the third international symposium on Information processing in sensor networks, pp. 259–268, ACM Press, 2004.
[67] J. Deng, R. Han, and S. Mishra, "Counter measuers against traffic analysis in wireless sensor networks," Technical Report CU-CS-987-04, University of Colorado at Boulder, 2004.
[68] B. Parno, A. Perrig, and V. Gligor, "Distributed detection of node replication attacks in sensor networks," In Proceedings of IEEE Symposium on Security and Privacy, May 2005.
[69] D. Wagner, "Resilient aggregation in sensor networks," In Proceedings of the 2nd ACM workshop on Security of Ad hoc and Sensor Networks (SASN '04), pp. 78–87, New York, NY, USA, ACM Press, 2004.
[70] B. Przydatek, D. Song, and A Perrig, "Sia: Secure information aggregation in sensor network," 2003.
[71] H. Cam, D. Muthuavinashiappan, and P. Nair, "ESPDA: Energy-efficient and secure pattern-based data aggregation for wireless sensor networks," In Proceedings of IEEE Sensors, pp. 732-736, Toronto, Canada, October 2003.
[72] H. Cam, D. Muthuavinashiappan, and P. Nair, "Energy-efficient security protocol for wireless sensor networks," In Proceedings of IEEE VTC Conference, pp. 2981- 2984, Orlando, Florida, October 2005.
[73] A. Deshpande, C. Guestrin, S. Madden, J. M. Hellerstein, W. Hong, "Model-driven acquisition in sensor networks," Proceedings of VLDB, Toronto, Canada, 2004.
[74] S. Yulong, X. Ning, P. Qingqi, M. Jianfeng, X. Qijian, and W. Zuoshun, "Distributed Storage Schemes for Controlling Data Availability in wireless Sensor Networks," Seventh International Conference on Computational Intelligence and Security (CIS), pp. 545-549, 3-4 Dec. 2011.Sanya, Hainan, China.
[75] D. Zeinalipour-Yazti and Z. Vagena, "Distributed Top-K Query Processing in Wireless Sensor Networks," 9th International Conference on Mobile Data Management (MDM 2008), Beijing, China, April 27-30, 2008.
[76] M. Ye, X. Liu, W. Lee, and D. L. Lee, "Probabilistic Top-k Query Processing in Distributed Sensor Networks," IEEE 26th International Conference of Data Engineering (ICDE), March 1-6, 2010, Long Beach, California, USA. pp. 585-588.
[77] M. Ye, W. Lee, D. L. Lee, and X. Liu, "Distributed Processing of Probabilistic Top-k Queries in Wireless Sensor Networks," IEEE Transactions on Knowledge and Data Engineering, Vol. 25, Issue 1, June 2013. Pp. 76 - 91. DOI: 10.1109/TKDE.2011.145
[78] C. Zhu, L. T. Yang, L. Shut, T. Hara, and S. Nishio, "Implementing Top-k Query in Duty-cycled Wireless Sensor Networks," 7th Int. Wireless Communications & Mobile Computing Conference (IWCMC 2011), pp. 553 – 558, Istanbul, Turkey, July 4-8, 2011.
[79] Y. Li, C. Ai, C. Vu, Y. Pan, and R. Beyah, "Delay Bounded and Energy Efficient Composite Event Monitoring in Heterogeneous Wireless Sensor Networks," IEEE Transactions on Parallel and Distributed Systems (TPDS), 21(9):1373-1385, 2010.
[80] Y. Chen, H. V. Leong, M. Xu, J. Cao, K. C. Chan, and A. T. Chan, "In-network Data Processing for Wireless Sensor Networks," 7th International Conference on Mobile Data Management (MDM'06), Nara, Japan, May 10-12, 2006.
[81] M. Panda, T. Panigrahi, P. M. Khilar, and G. Panda, "Learning with Distributed Data in Wireless Sensor Networks," First International Conference on Parallel, Distributed and Grid Computing, (PDGC-2010), Waknaghat, Solan (H.P), India, October 2010. Pp. 241-245.
[82] Z. Shukui, C. Zhiming, G. Shengrong, and F. Jianxi, "An Efficient Approximation Algorithm for Data Aggregation in Wireless Sensor Networks," Int. Symposium on Information Processing (ISIP'09), pp. 017-020, Huangshan, P. R. China, August 2009.
[83] C. Plattner and G. Alonso, "Ganymed: Scalable Replication for Transactional Web Application," Journal of Middleware, 3231, 155-174, 2004.
[84] O. Gardarin and G. Gardarin, "Le Client-Server," Edition Eyrolles, Paris, 1997.
[85] I. Sarr, H. Naacke, and S. Gançarski, "Failure-Tolerant Transaction Routing at Large Scale," Second International Conference on Advances in Databases, Knowledge, and Data Application (DBKDA), Menuires, France, 11-16 April 2010.
[86] O. Diallo, M. Sene, and I. Sarr, "Freshness-Aware Metadata Management: Performance Evaluation with SWN models," IEEE/ACS Int. Conference of the Computer Systems and Applications (AICCSA 2010), Hammamet, Tunisia, May 16-19, 2010.
[87] P. Andreou, D. Zeinalipour-Yazti, M. Andreou, P. K. Chrysanthis, and G. Samaras, "Perimeter-Based Data Replication in Mobile Sensor Networks," 10th International Conference on Mobile Data Management: Systems, Services and Middleware (MDM), pp. 244-251, Taipei, Taiwan. 18-20 May 2009.
[88] D. Zeinalipour-Yazti, P. Andreou, P. K. Chrysanthis, and G. Samaras, "SenseSwarm: a perimeter-based data acquisition framework for mobile sensor networks," 4th workshop on Data Management for Sensor Networks: In Conjunction with 33rd International Conference on Very Large Data Bases (DMSN), pp. 13-18. Vienna, Austria, April 2008.
[89] A. Kulakov and D. Davcev, "Distributed Data Processing in Wireless Sensor Networks Based on Artificial Neural-Networks Algorithms," 10th IEEE Symposium on Computers and Communications (ISCC 2005), June 27-30, 2005, Cartagena, Spain.
[90] B. Scholz, M. M. Gaber, T. Dawborn, R. Khoury, and E. Tse, "Efficient time triggered query processing in wireless sensor networks," Int. Conference on Embedded Software and Systems (ICESS 2007), Daegu, Korea, May 14-16, 2007. Pp. 391-402.
[91] R. Khoury, T. Dawborn, B. Gafurov, G. Pink, E. Tse, Q. Tse, et al. "Corona: Energy-Efficient Multi-query Processing in Wireless Sensor Networks," Database Systems for Advanced Application (DASFAA), Lecture Notes on Computer Science 5982/2010, Pp. 416-419, 2010.
[92] U. Roehm, M. M. Gaber, Q. Tse, "Enabling resource-awareness for in-network data processing in wireless sensor networks," Nineteenth Australasian Database Conference (ADC 2008), Wollongong, NSW, Australia, Pp. 107-114, January 22-25, 2008.
[93] J. Sun and J. Zhou, "Power-Aware Data Reduction for Continuous Query In Wireless Sensor Networks," IEEE International Conference on Industrial Technology (ICIT 2008), Pp. 1-6, Chengdu, China, April 21-24, 2008.
[94] S. Chatterjea, T. Nieberg, M. Meratnia, and P. Havinga, "A Distributed and Self-Organizing Scheduling Algorithm for Energy-Efficient Data Aggregation in Wireless Sensor Networks," ACM Transactions on Sensor Networks (TOSN), 4 (4), 2008.

[95] I. Galpin, C. Y. Brenninkmeijer, A. J. Gray, F. Jabeen, A. A. Fernandes, N. W. Paton, “SNEE: a query processor for wireless sensor networks,” Distributed and Parallel Databases, 29, 31–85, 2011.
[96] I. Galpin, C. Y. Brenninkmeijer, A. J. Gray, F. Jabeen, A. A. Fernandes, N. W. Paton, “An architecture for query optimization in sensor networks,” IEEE 24th International Conference on Data Engineering (ICDE 2008), Cancun, Mexico, 7-12 April 2008. Pp. 1439 - 1441. DOI: 10.1109/ICDE.2008.4497582
[97] I. Galpin, C. Y. Brenninkmeijer, F. Jabeen, A. A. Fernandes, and N. W. Paton, “Comprehensive optimization of declarative sensor network queries,” 21st International Conference on Scientific and Statistical Database Management (SSDBM 2009), Lecture Notes in Computer Science, Vol. 5566. Pp. 339-360.
[98] C. Y. Brenninkmeijer, I. Galpin, A. A. Fernandes, and N. W. Paton, “A semantics for a query language over sensors, streams and relations,” 25th British National Conference on Databases, BNCOD 25, Cardiff, UK, July 7-10, 2008. Lecture Notes in Computer Science Vol. 5071, Pp. 87–99.
[99] C. Y. Brenninkmeijer, I. Galpin, A. A. Fernandes, and N. W. Paton, “Validated cost models for sensor network queries,” Sixth International Workshop on Data Management for Sensor Networks (DMSN '09), Lyon, FRANCE, August 24, 2009.
[100] G. Valkanas, D. Gunopulos, I. Galpin, A. J. G. Gray, and A. A. A. Fernandes, “Extending query languages for in-network query processing,” 10th ACM International Workshop on Data Engineering for Wireless and Mobile Access (MobiDE '11), Athens, Greece, June 2011. pp. 34-41.
[101] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, F. Silva, “Directed diffusion for wireless sensor networking,” ACM/IEEE Transactions on Networking, 11(1), 2–16, 2002.
[102] A. Awad, W. Xie, E. Rose, R. German, and F. Dressler, “Distributed Data Management in Sensor Networks using miniDB / miniSQL,” 4th IEEE/ACM International Conference on Distributed Computing in Sensor Systems (DCOSS), pp. 27-28. Santorini Island, Greece, June 2008.
[103] C. Jardak, E. Osipov, and P. Mahonen, “Distributed Information Storage and Collection for WSNs,” 4th IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS), Italy-Pista, October 2007.
[104] C. Chang, J. Sheu, S. Chang, and Y. Chen, “A frequency-aware data-centric mechanism for wireless sensor networks,” Wireless Communications and Mobile Computing 2010, 10 (8), 1078–1101, 2010.
[105] H. Shen, L. Zhao, and Z. Li, “A Distributed Spatial-Temporal Similarity Data Storage Scheme in Wireless Sensors Networks,” IEEE Transactions on Mobile Computing, 10 (7), 982-996, 2011.
[107] H. Wang, R. Liu, X. Zheng, Y. Chen, and H. Liu, “To do or not to do: metadata-guided query evaluation in content caching networks,” 28th IEEE Conference on Global Telecommunications, Honolulu, Hawaii, USA, pp. 4524-4529, 30 November - 4 December 2009.
[107] R. Liu, X. Zheng, H. Liu, H. Wang, and Y. Chen, “Metadata-guided evaluation of resource-constrained queries in content caching based wireless networks,” Wireless Networks, 17 (8), 1833–1850, 2011.

AUTHORS’ BIOGRAPHIES

Ousmane Diallo is a PhD student of Computer Science at the University of Beira Interior under supervised by Professor Joel Rodrigues and co-supervised by Dr. Mbaye Sene. He received his 5-years BS degree in Computer Science from the University of Cheikh Anta Diop of Dakar, Senegal, in 2008; and his MSc degree in Computer Science from the University of Cheikh Anta Diop of Dakar, Senegal, in 2009. He is also teacher in the Department of Informatics at the University of Ziguinchor, Senegal. He is a PhD student member of the Institute of Telecommunications, Portugal. His current research interests are distributed databases for wireless sensor networks, real-time databases, wireless sensor networks and performance evaluation. He is author of one journal paper and one international conference paper. He also participated on several reviewed papers for international conferences.

Joel José P. C. Rodrigues is a professor in the Department of Informatics of the University of Beira Interior, Covilhã, Portugal, and researcher at the *Instituto de Telecomunicações*, Portugal. He received a PhD degree in informatics engineering, an MSc degree from the University of Beira Interior, and a five-year BSc degree (licentiate) in informatics engineering from the University of Coimbra, Portugal. His main research interests include sensor networks, e-health, e-learning, vehicular delay-tolerant networks, and mobile and ubiquitous computing. He is the leader of NetGNA Research Group (http://netgna.it.ubi.pt), the Vice-chair of the IEEE ComSoc Technical Committee on Communications Software, the Vice-Chair of the IEEE ComSoc Technical Committee on eHealth, and Member Representative of the IEEE Communications Society on the IEEE Biometrics Council. He is the editor-in-chief of the International Journal on E-Health and Medical Communications, the editor-in-chief of the Recent Patents on Telecommunications, and editorial board member of several journals. He has been general chair and TPC Chair of many international conferences. He is a member of many international TPCs and participated in several international conferences organization. He has authored or coauthored over 250 papers in refereed international journals and conferences, a book, and 2 patents. He had been awarded the Outstanding Leadership Award of IEEE GLOBECOM 2010 as CSSMA Symposium Co-Chair and several best papers awards. Prof. Rodrigues is a licensed professional engineer (as senior member), member of the Internet Society, an IARIA fellow, and a senior member of ACM and IEEE.

Mbaye Sene is a Lecturer in the Department of Mathematics and Informatics of the University of Cheikh Anta Diop of Dakar, Senegal, and researcher at the Faculty of Science and Technology, Senegal. He received a PhD degree in Computer Science from the University of Paris-Dauphine, a MSc degree and a 5-years BS degree in Computer Science from the University of Cheikh Anta Diop of Dakar, Senegal. His main research interests include distributed systems, distributed database systems, design of inter-operable open systems, and performance evaluation of stochastic complex systems. He has authored or co-authored more than 15 international conference papers and journals. He is the coordinator of USCP/ Manager of the project VAE, Ministry of Youth, of the Vocational Training and Employment.

Jaime Lloret (M’07 – SM’10), received his M.Sc. in Physics in 1997, his

M.Sc. in electronic Engineering in 2003 and his Ph.D. in telecommunication engineering (Dr. Ing.) in 2006. He is a CCNP Instructor. He is Associate Professor in the Polytechnic University of Valencia and he is the research line coordinator of the "communications and remote sensing" of the Integrated Management Coastal Research Institute. He is currently the Cognitive Networks Technical Committee (IEEE Communications Society) Vice-chair for the Europe/Africa Region. He more than 210 research and educational papers published in national and international conferences, international journals (most of them with Impact Factor in Journal Citation Report), and books. He has been the co-editor of 15 conference proceedings and guest editor of several international books and journals. He is editor-in-chief of the international journal "Networks Protocols and Algorithms", of "Advances in Network and Communications", IARIA Journals Board Chair, and he is associate editor of several international journals. He has been involved in more than 150 Program committees of international conferences and in several organization and steering committees. He has been the general chair of SENSORCOMM 2007, UBICOMM 2008, ICNS 2009 and ICWMC 2010 and co-chairman of ICAS 2009 and INTERNET 2010. He is the co-chairman of IEEE MASS 2011 and SCPA 2011. He is IEEE Senior Member and IARIA Fellow Member.