Research Article

Data Gathering Techniques for Wireless Sensor Networks: A Comparison

Giuseppe Campobello, Antonino Segreto, and Salvatore Serrano

Department of Engineering, University of Messina, Contrada Di Dio, Sant’Agata, 98166 Messina, Italy

Correspondence should be addressed to Giuseppe Campobello; gcampobello@unime.it

Received 2 November 2015; Revised 3 February 2016; Accepted 4 February 2016

Academic Editor: Haiping Huang

Copyright © 2016 Giuseppe Campobello et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Westudytheproblemofdatagatheringinwirelesssensornetworksandcompareseveralapproachesbelongingtodifferentresearch
fields; in particular, signal processing, compressive sensing, information theory, and networking related datagatheringtechniques
areinvestigated. Specifically, we derived a simple analytical model able to predict the energy efficiency and reliability of different
datagatheringtechniques. Moreover, we carry out simulations to validate our model and to compare the effectiveness of the above
schemes by systematically sampling the parameter space (i.e., number of nodes, transmission range, and sparsity). Our simulation
andanalyticalresults show that there is no best data gathering technique for all possible applications and that the trade-off between
energy consumptions and reliability could drive the choice of the data gathering technique to be used. In this context, our model
could be a useful tool.

1. Introduction

Wireless sensor networks (WSNs) are composed of a lot of tiny, low power, and cheap wireless sensors, deployed in a geographic area to perform distributed tasks, for example, to monitor a physical phenomenon [1]. In 2004, MIT Technology Review ranked WSNs as the number one emerging technology [2] and today they are effectively employed for many applications, such as surveillance (e.g., real-time area audio or video surveillance), security (e.g., detection of biological agents or toxic chemicals), habit monitoring (e.g., environmental measurement of temperature, pressure, or mechanical vibration), home automation, military systems, and, in general, scientific experiments.

In a typical WSN topology, we can distinguish between ordinary wireless sensor nodes and base stations named sinks. The sink is usually connected to a power supply and it is capable of performing more complex operations than the ordinary nodes. Ordinary wireless sensor nodes, which are capable of transferring processed or raw sensed data to the sink, due to economical reasons, are instead usually powered by small size batteries that in most application scenarios are difficult or even impossible to replace or recharge.

So, in contrast to many other wireless devices (e.g., cellular phones, PDAs, and laptops), usually it is not expected to renew the energy supplied to a wireless sensor node during the life of the WSN. For this reason, each sensor node is required to work under very low power consumption conditions.

In general, to design a highly energy-efficient WSN, it is extraordinarily important to take into account capture, transmission, and routing issues, that is, data gathering techniques that specify how ordinary sensors work for gathering information and delivering them to the sink. As a consequence, data gathering is the main and more critical function provided by a WSN.

The main aim of this paper is to compare some of the state-of-the-art data gathering techniques considering their trade-off between reliability (i.e., packet loss and reconstruction error) and energy consumptions (i.e., network lifetime) by taking into account both compression and networking aspects. To the best of our knowledge, this is the first paper that considers such type of comparison for data gathering techniques belonging to different research fields (i.e., signal processing, compressive sensing, information theory, and networking related techniques are discussed and compared.
in this paper). Specifically, we derived a simple analytical model able to predict the energy efficiency and reliability of several data gathering techniques. Moreover, we carry out simulations to validate our model and to compare the effectiveness of the above schemes by systematically sampling the parameter space (i.e., number of nodes, transmission range, and sparsity).

The rest of the paper is organized as follows. In Section 2, we present a summary of related works. In Section 3, further details about existing data gathering techniques are provided by highlighting their advantages and drawbacks. In Section 4, the simulation scenario used for comparisons is detailed and an analytical model able to predict the energy efficiency and reliability of different data gathering techniques is derived. In Section 5, the metrics used to compare data gathering techniques are introduced. In Section 6, simulation results are provided and the developed analytical model is validated. Finally, in Section 7, some concluding remarks and future works are drawn.

For the sake of clarity, symbols and notations used throughout the paper are reported in Notations section.

2. Related Works

In the past few years, several data gathering techniques have been proposed for WSNs with the main aim of reducing energy consumptions in WSNs by exploiting correlations among sensory data. We can distinguish them into two broad categories: compression-oriented and networking-oriented.

The first category, named compression-oriented, is focused on maximizing network lifetime by taking advantage of data compression techniques [3–10]. In particular, [3, 4] analyze different lossless compression schemes for WSNs exploiting the temporal correlation in the sampled signals; in [5, 6], the authors exploit spatial correlation by using distributed source coding techniques based on the Slepian-Wolf theorem; finally, [7–10] investigate the fundamental limits of data gathering techniques based on the new paradigm of compressive sensing [11, 12]. A comprehensive review of existing data compression approaches in WSNs is provided in [13].

Further details about compression-oriented data gathering techniques will be provided in Section 3. In particular, signal processing, compressive sensing, and information theory related techniques are discussed, respectively, in Sections 3.1, 3.2, and 3.3.

Since radio transmission is the primary source of power consumption in WSNs, a second category of data gathering techniques, named networking-oriented, have dealt with the problem of maximizing network lifetime by taking into account network protocols and, more specifically, forwarding/routing mechanisms [14–17].

In particular, in [14], the authors show how it is possible to maximize the lifetime of a WSN by exploiting routing algorithms. In [15], the authors study the problem of forest construction for maximizing the network lifetime and adopt a simple data aggregation model where an intermediate sensor can aggregate multiple incoming messages into a single outgoing message. A different approach is proposed in [16] where a smart splitting technique is used in order to achieve different trade-offs between reliability and energy saving. Finally, in [17], the authors address the problem of maximizing network lifetime by taking into account also latency and reliability.

However, with the exception of [15], the above papers do not perform any kind of data aggregation with the aim of not introducing extra delay.

As shown in [18–21], by combining data aggregation and routing mechanisms, efficient data gathering schemes can be obtained.

In particular in [18], the problem of jointly optimized routing and data aggregation is investigated; in [19], the authors combine data compression and multipath routing techniques to obtain a reliable and low-latency data aggregation scheme; in [20], an energy-balanced data gathering and aggregating scheme is proposed which integrates a clustering hierarchical structure with the compressive sensing to optimize and balance the amount of data transmitted; finally in [21] a data gathering technique based on the network coding paradigm is proposed.

Further details about the above works will be provided in Section 3.4.

Several other papers exist which compare signal processing techniques in WSNs from energy efficiency and network lifetime perspectives and several works highlight the effect of using different routing mechanisms for data gathering.

Nevertheless, techniques belonging to different research fields such as compressive sensing, information theory, and networking are seldom evaluated against one another and this is the main goal of this paper.

Only recently, such comparisons have started, for instance, [22] where lossy data aggregation techniques are evaluated and compared in terms of reconstruction errors and energy consumptions. However, authors do not consider the impact of network reliability (i.e., packet loss) nor compare networking-based data gathering techniques such as those based on the network coding paradigm.

The aim of this paper is to fill this gap by investigating the effectiveness of all the above data gathering techniques also in terms of reliability. In particular, an analytical model able to predict the energy efficiency and reliability of different data gathering techniques is derived.

3. Data Gathering Techniques

We can classify data gathering schemes on the basis of the research field from which the technique used to exploit correlation among sensor nodes is drawn, that is,

(i) signal processing;
(ii) compressive sensing;
(iii) information theory;
(iv) networking.

Techniques belonging to the above fields are discussed in the next subsections by highlighting their advantages and drawbacks.
3.1. Signal Processing Techniques. Frequently high correlations (spatial and/or temporal) among sensor readings exist. In this case, it is inefficient to deliver the entire raw data to the destination [8, 9] and signal processing, in particular Transforms and Encoding Compression (TEC) techniques, can be exploited in order to reduce the amount of data to send.

In the case of local TEC techniques, node collects measurements following the Shannon-Nyquist sampling theorem; these measurements are transformed and properly encoded and the output of such transformation is stored in the payload of one or more packets and sent to the sink. In particular, either lossy or lossless techniques can be used depending on the particular application scenario.

With lossy techniques [3], the original data is compressed discarding some of the original information; this allows achieving higher compression ratios but at the receiver side one can only reconstruct the data with a certain accuracy.

However, in some types of monitoring, the accuracy of observations is critical for understanding the underlying physical processes. In other cases, it is not possible to have an a priori knowledge about the magnitude of observational errors that are tolerable without affecting a correct data gathering. Moreover, some application domains (e.g., body area networks BANs in which sensor nodes permanently monitor and log vital signs) demand sensors with high accuracy and cannot tolerate measurements corrupted by lossy compression processes.

In all these kinds of WSNs, lossless data gathering is essential and desirable. Examples of local lossless compression schemes have been proposed in [4, 23, 24].

Lossy compression techniques have been evaluated and compared in terms of reconstruction errors and energy consumptions in [22]; therefore, in this paper we concentrate our attention on lossless techniques.

For the sake of space, we did not consider distributed TEC techniques in this paper but we refer the reader to [13]. Nevertheless, the major distributed approaches of signal processing applied to WSNs will be discussed in the next subsections.

3.2. Compressive Sensing. Compressive sensing (CS) is a new paradigm introduced by Candes and Tao [11] and Donoho [12] used to capture and to compress signals in WSNs where compression and sampling are merged and carried out at the same time. Basically, CS compresses a signal while acquiring data at its information rate (without relying to the Shannon-Nyquist sampling theorem). CS theory states that if a signal is sparse or compressible in a certain basis, then it can be reconstructed from a small number of linear measurements by solving an l1 based convex optimization problem [7].

More precisely, let us define k-sparse signals \( x = (x_1, \ldots, x_n)^T \) as signals that can be expressed as \( x = \Psi \alpha \), where \( \Psi \) is an orthonormal transform and \( \alpha \) is a vector with at most \( k \ll n \) nonzero entries; CS theory states that \( x \) can be recovered from \( m = O(k \log(n/k)) \) linear combinations of measurements obtained as \( y = \Phi x \), where \( \Phi \) is an \( m \times n \) matrix.

Note that, considering that \( k \) is a small value in comparison to \( n \), it follows that \( m \) can be much smaller than \( n \) and therefore high compression ratios can be achieved using CS (i.e., by transmitting CS measurements \( y \) instead of raw data \( x \)).

Reconstruction is achieved by solving a complex optimization problem of the following form:

\[
\arg \min \| \alpha \|_1 \quad \text{s.t.} \quad y = A \alpha,
\]

where \( A = \Phi \Psi \). Once the above problem is solved, \( x = \Psi \alpha \). Alternatively, when noised measures are considered, the following optimization problem can be considered:

\[
\arg \min \| \alpha \|_1 \quad \text{s.t.} \quad \| y - A \alpha \|_2^2 \leq \epsilon,
\]

where \( \epsilon \) bounds the noise.

Several algorithms exist which are able to solve the above optimization problem (Basis Pursuit [25], OMP [26], and CoSaMP [27], to name just a few) and several theoretical results exist describing when these algorithms recovered sparse solutions. In particular,

(i) as proved in [28], a signal \( x \) can be recovered with high probability if \( \Phi \) satisfies the Restricted Isometric Property (RIP).

Formally, a matrix \( \Phi \) satisfies the RIP if for all \( k \)-sparse signals \( x \) exists a \( \delta_k \in (0, 1) \) such that

\[
(1 - \delta_k) \| x \|_2^2 \leq \| \Phi x \|_2^2 \leq (1 + \delta_k) \| x \|_2^2.
\]

Example of matrices that satisfy the RIP condition are \( \pm 1 \) Bernoulli matrix and Gaussian distribution matrix where \( \phi_{ij} \sim N(0, \sigma_q^2) \) with \( \sigma_q^2 = 1/m \).

(ii) As proved in [29], in the noise-free case, exact recovery with Gaussian matrix can be obtained if

\[
m = m^* = 2k \log \left( \frac{n}{k} \right) + \frac{5}{4}k + 1.
\]

We will exploit the above results to derive a reliability model for CS.

CS can be applied in cluster-based WSNs considering that each sensor node in a cluster sends its reading \( x_i \) to the cluster head which will multiply all received readings by random coefficients \( \phi_{ij} \) by generating \( m \) weighted sums \( y_j = \sum_i \phi_{ij} x_j \) with \( i \in \{1, \ldots, m\} \). Next, values \( y_j \), named CS measurements, are sent to the sink through one or multiple packets.

On the basis of the CS theory, under sparsity condition, by collecting a sufficient number of CS measurements, the sink will be able to reconstruct the original sensor data \( x_i \).

CS can be applied also in tree-based WSNs considering that the source node includes in its packet the sensing information which is the product of its acquired value and a random coefficient and then sends it to its next hop node [30]. In such way, CS compression could be performed with a low complexity at source nodes [9] and data traffic over
the network is reduced [8]. However, in the last case a high number of hops are needed, that is, \( h = O(n/k) \), which may lead to high network latency.

In [7], comparisons of CS-based and conventional signal processing techniques for WSNs have been carried out in terms of energy efficiency and network lifetime.

However, there are several challenges that must be addressed in order to use CS:

(i) Decoding time for reconstruction can be \( O(n^2) \) and therefore prohibitively expensive for large networks. Less expensive algorithms exist (e.g., matching pursuit) but they provide less stable recovery and weaker error bounds in the recovered solution.

(ii) CS assumes that the sensed data has a known constant sparsity, ignoring that the sparsity of real signals varies in temporal and spatial domain. In particular, the sparsifying basis \( \Psi \) is assumed to be given and fixed with time, but this is not the case for a realistic WSN scenario, where the signal of interest is unknown and its statistical characteristics can vary over time.

(iii) CS-based techniques introduce not negligible losses (recovery errors) by reducing reliability and work best for large scale networks (at least a thousand nodes).

(iv) Quantization effects: CS theory has mostly focused on real-valued measurements but in practice measurements must be represented with a finite number of bits. As a consequence, a trade-off exists between the number of measurements \( m \) and the number of bits per measurement \( b_{CS} \).

In this paper, we concentrate on the last two problems, by analyzing the effect of sparsity and quantization on energy saving and reliability. Further details on CS and how to exploit it for WSNs will be given in the next sections.

3.3. Information Theory Related Techniques. In order to exploit the correlation of data concurrently acquired by different sensors, DSC techniques, inspired by the Slepian-Wolf theorem, can be applied [2]. The DSC techniques imply that each sensor node sends its compressed outputs to the sink for joint decoding. This means that the nodes need to cooperate in groups of two or three so that one node provides the side information and another one can compress its information down to the Slepian-Wolf or the Wyner-Ziv limit. Furthermore, DSC approaches are also difficult to be applied in such scenarios since they work with the assumption that the data distribution should be known in advance [9, 31].

The most practical and well-known implementation of DSC is DISCUS [5, 32] where sensor nodes are considered divided into clusters. For each cluster, a node (the cluster head) sends uncompressed data (as side information) while all other nodes transmits encoded (i.e., compressed) data.

To encode data, a sensor node firstly divides all possible measurements into disjoint sets (named bins) so that values in the same bin have a minimum distance \( d \). Each piece of sensory data is then compressed with a code that identifies the unique bin where the sampled value lies.

To better explain how DISCUS works let us consider a simple example.

Let us suppose that (quantized) measurements are in the integer range \([0, 7]\) and that all data sensed from different sensors at (almost) the same time differ by at most \( \pm 1 \). Without DSC compression, three bits are needed for each sensor to represent sensory data. Instead, in the case of DSC only the cluster head sends a three-bit value. The other sensor nodes can split the possible values into four bins \([0, 4] ; [1, 5] ; [2, 6] ; [3, 7] \) so that values in the same bin have a minimum distance \( d = 4 \) and encode them, respectively, with \([00] ; [01] ; [10] ; [11] \). So if the sink receives 01 from a sensor node it knows that only two values are possible, that is, \([1, 5] \). Now let us suppose that the sink receives also the value 6 (properly encoded) from the cluster head; in this case, the value 01 is immediately interpreted as 5 without ambiguity as a consequence of the fact that sensed data can differ by at most 1.

In the above example, only the cluster head transmits 3 bits for each measurement while for all the other nodes 2 bits are enough; therefore, compression is achieved.

However, DSC relies on the assumption that statistical characteristics (i.e., correlation function) of the underlying data should be known a priori, which is difficult to obtain in practical scenario. For instance, the simple DISCUS scheme discussed above works only if the difference between the value sampled by the cluster head and all the other nodes in the same cluster is less than \( d/2 \). Moreover, losing side information (i.e., cluster head data) will cause fatal errors to the decoder, that is, low reliability.

A simple manner to improve reliability is achieved by retransmitting cluster head packets more times but this reduces compression efficiency. Therefore, a trade-off exists between energy consumptions and reliability on the basis of the maximum allowed number of retransmissions. In this paper, we investigate such a trade-off.

3.4. Networking Techniques

3.4.1. Routing-Based Techniques. Since radio transmission is the primary source of power consumption at the nodes, the design of energy-efficient routing is another important topic to investigate in the design of data gathering technique. The basic idea is to route the packet through the paths so as to minimize the overall energy consumption for delivering the packet from the source to the destination. The problem focuses on computing the flow and transmission power to maximize the lifetime of the network, which is the time at which the first node in the network runs out of energy [18]. Specifically, the energy consumption rate per unit of information transmission for each node depends on the choice of the next hop, that is, the routing decision. This choice can influence the energy required to reach the sink [14].

One of the most recent works which addresses the problem of maximizing network lifetime taking into account the routing mechanism is [17]. The authors try to achieve both
low latency and high reliability. They construct a data-gathering tree based on a reliability model, schedule data transmissions for the links on the tree, and assign transmitting power to each link accordingly. However, they do not perform any kind of data aggregation or data compression with the aim of not introducing extra delay.

Data aggregation can be performed on top of the routing algorithm. The aggregation function is usually performed by extracting some statistical values (e.g., maximum, minimum, and average) and then by transmitting only these [15]. In such a way, it is possible to reduce the amount of communicating data in the dense sensor networks and reduce the power consumption. However, this technique loses much of the structure of the original acquired data.

In particular, the authors of [15] study the problem of forest construction for maximizing the network lifetime. They adopt a simple data aggregation model and assume that an intermediate sensor can aggregate multiple incoming B-bit messages, together with its own message, into a single outgoing message. Moreover, they provide a polynomial time algorithm to build the tree and demonstrate that it is close to optimal.

In [18], the authors try to jointly optimize routing and data aggregation so that the network lifetime can be extended considering two dimensions. In the first dimension, the traffic across the network is reduced by data aggregation, so that one can reduce the power consumption of the nodes close to the sink node. In the second dimension, the traffic is balanced to avoid overwhelming the bottleneck nodes. A smoothing function is used to approximate an original maximization function by exploiting the special structure of the network. The necessary and sufficient conditions for achieving the optimality of this smoothing function were derived and a distributed gradient algorithm was accordingly designed.

Yang et al. propose in [35–37] a joint design of energy replenishment and data gathering by exploiting mobility. The SenCar, a multifunctional mobile entity, periodically chooses a subset of sensors to visit based on their energy status. It utilizes wireless energy transmissions to deliver energy to the visited sensors and, meanwhile, it collects data from nearby sensors via short-range multihop communications and can convey this data to the sink.

3.4.2. Network Coding Based Techniques. A different approach exploits the network coding (NC) paradigm.

NC is an effective information transmission approach originally introduced by Ahlswede et al. [38] to improve network capacity of multicast networks.

Differently from the classical store and forward network paradigm where nodes simply replicate and forward incoming packets, using NC intermediate nodes in the network have the ability to forward functions of received packets (e.g., linear combinations). In this manner, throughput gain, robustness, and energy saving can be achieved by exploiting the fact that each newly generated packet carries information contained in several original packets [39].

NC has received increasing attention also in WSNs as a promising tool to improve network lifetime and reliability by exploiting the broadcast nature of the wireless channel [40].

However, most of the proposed techniques developed so far [41–43], whilst being useful for data dissemination (e.g., traffic from the sink node to the sensor nodes), cannot be applied for data collection, which is the most important traffic in WSNs.

In fact, to apply NC for data gathering in WSNs, some issues have to be solved.

(i) *Header Overhead*. NC schemes are mostly based on random linear codes [44, 45] which allow implementing them in a distributed manner but introduce large overhead because coefficients used for linear combinations should be specified in packets header. The header size is proportional to the number of aggregated packets that, in the specific case of data collection in WSNs, could be equal to the number of nodes in the network.

(ii) *All-or-Nothing Problem*. When $n$ packets are combined using NC, the sink has to receive at least $n$ packets in order to be able to recover the original information. Thus, even if the sink receives $n−1$ packets, it cannot recover any information. Instead, graceful degradation is desirable in WSNs.

(iii) *Delay*. The delay introduced by NC might be prohibitive for large networks where a large number of packets should be combined and decoded. Instead, many sensor networks applications, for instance, WSNs developed for control/automation or real-time audio/video streaming, require small bounded delays.

(iv) *Duty Cycling*. Most of NC schemes are based on overhearing; that is, nodes should remain in active mode to participate in NC-based routing, which increases the energy consumption of the sensor nodes. So, it is difficult to couple NC paradigm and duty-cycling techniques commonly used in WSNs.

(v) *Reliability*. Full (or at least high) reliability is desirable in sensor networks and is mandatory in several scenarios, for instance, in new scientific experiments, where accuracy of observations is critical, or in the case of biomedical applications, where it is necessary to ensure that important details are not lost causing errors in medical diagnosis. When random codes are used, even in a reliable network, the original messages can be retrieved with “high probability” (though not “certainty”), and high probability is achieved through the use of large finite fields (i.e., large coefficients and therefore large headers).

(vi) *Complexity*. NC techniques should be simple to cope with low computational and memory resources of sensor nodes.
We refer the reader to [46] for further considerations on the applicability of NC to WSNs.

The above issues have been solved in [16] where the authors proposed a new forwarding technique for WSNs based on the Chinese Remainder Theorem (CRT) able to achieve different trade-offs between reliability and energy saving.

Basically, CRT can be seen as a splitting technique able to transform an integer number \( Z \) into a vector of smaller numbers named CRT components, \( \{z_j\} \). CRT components are obtained from number \( Z \) using modular arithmetic as \( z_j = Z \mod p_j \), where \( p_j \) (with \( j \in [1, ..., N_{\text{CRT}}] \)) are prime numbers (or at least pairwise coprime integer numbers).

CRT states that every integer number \( Z \) can be exactly recovered from its CRT components if the product of prime numbers \( P = \prod_{j=1}^{N_{\text{CRT}}} p_j \) satisfies the condition

\[
P > Z
\]  
(henceforward named reconstruction condition). In particular, the CRT always states that \( Z \) can be recovered through a simple linear combination as

\[
Z = \sum_{j=1}^{N_{\text{CRT}}} c_j z_j \mod P.
\]  

Coefficients \( c_j \) are given by \( c_j = Q_j \cdot q_j \), where \( Q_j = P/p_j \) and \( q_j \) is its modular inverse obtained by solving \( q_j Q_j = 1 \mod p_j \).

CRT can be applied in WSNs to split packets produced by sensor nodes. Such smaller packets (i.e., CRT components) can be sent through different paths by exploiting path diversity of WSNs. The fact that relayers nodes forward smaller packets allows reducing energy consumption.

Moreover, CRT has several advantages in comparison to other NC techniques:

(i) The set of prime numbers \( \{p_j\} \) can be chosen so that information produced by the sensor nodes can be reconstructed even if only a fraction of the CRT components are received by the sink, by improving reliability and solving the all-or-nothing problem.

(ii) Differently from coefficients used for NC techniques, the set of prime numbers can be obtained directly by the sink (i.e., CRTs avoid header explosion).

(iii) CRT can be efficiently combined with duty-cycling techniques [48] and distributed compression algorithms [19] to achieve an efficient data aggregation technique.

Considering the above advantages and the fact that this paper is focused on data gathering techniques for WSNs, we will consider CRT as representative of networking-based data gathering techniques.

4. Simulation Scenario

In this section, we will discuss the WSN model used for comparisons and simulations of data gathering techniques.

4.1. Network Model. We assume a WSN where the sink is located in the center of a square sensing area of size \( G \times G \) [m²] and sensor nodes are randomly distributed with density \( \rho \) [nodes/m²]. Each sensor node has a transmission range equal to \( R \) [m] (with \( R \ll G/2 \)) and sends its data to the sink through a multihop scheme.

The network is partitioned into nonoverlapped clusters using the procedure described in [16, 49]. The above-mentioned procedure is mainly based on the exchange of Initial-ization Messages (IMs) and allows organizing the network in clusters minimizing the number of hops needed by a sensor node to reach the sink. The sink is supposed to belong to cluster 1 (denoted as CL_1) and generates a first IM with its own address and a sequence number \( SN = 2 \). Each node, which receives an IM from its neighbors with a sequence number \( SN = h \), will belong to cluster CL_h and will retransmit the IM with an increased SN value together with its own address and the list of the nodes that will be used as forwards (which it knows according to the source addresses specified in the received IMs). On the basis of the received IMs, at the end of the above procedure, each node in the network will know its own next hops and which other nodes will use it as a next-hop. Further details on the initialization procedure are reported in [49].

We assume that the above initialization procedure is carried out only one time so we neglect related energy consumptions.

In the following, nodes along the path from a source to the sink are referred to as relayers and nodes located one hop away from the source along the path to the sink are specifically called one-hop relayers.

We assume that, independently of the specific data gathering technique, relayers transmit packets through a load-balancing shortest-path scheme; that is, a node in cluster CL_{h+1} will select randomly a reachable node in the next cluster toward the sink (CL_h), and this forwarding scheme is repeated until the sink is reached. In this manner, information reaches the sink with the minimum number of hops.

4.2. Data Gathering Model. Until now, we have classified data gathering techniques on the basis of the data aggregation technique used. However, data gathering techniques can be classified also considering the factors that drive data acquisition. In particular, four broad categories can be distinguished [50]: event-driven, time-driven, query-based, and hybrid.

In event-driven category, data are generated when an event of interest occurs, while in the time-driven category data are periodically sent to the sink at constant interval of time; in query-based category, data are collected according to sink requests. Finally, the hybrid approach is a combination of one or more of the above.

For simulation purpose, with the aim of evaluating energy consumptions and reliability, all the above categories can be unified with an abstraction of the concept of event, that is, by simply considering that data must be sent as a consequence of an event.

In particular, in query-based network an event is triggered by the reception of the query message while in time-driven networks the event can be associated with the rising
clock edge of the sampling unit or, more practically, when a sufficient number of measures has been collected and a packet is ready to be transmitted.

Considering the above abstraction, energy consumptions and network reliability can be evaluated in terms of number of events (i.e., packets sent) by not taking into account who drives the event.

Therefore, in our simulation scenarios we will consider that events randomly occur in the sensor network and that, for each event, nodes recognize the event and generate a packet. More precisely, we assume that only nodes inside the circular area of radius \( r \), with center in the location of the event, detect the event and therefore need to send a packet. Henceforward, we call the circular area related to an event a cell.

In event-driven networks, usually small packets are sent to specify that an event has been detected (a single \( w \)-bit word could be sufficient in most cases). Instead in the case of time-driven or query-driven networks, packets represent \( M \) measures collected in the time interval between two events. Both cases can be taken into account considering that for each event raw information of \( Mw \) bits has to be sent for each node by fixing \( M = 1 \) for event-driven networks and \( M \geq 1 \) in the case of time-driven or query-driven networks.

With the aim of reducing energy consumptions (i.e., the overall number of bits sent), raw data are not directly transmitted; instead, data are processed according to the chosen data gathering technique.

More precisely, we have the following.

(i) TEC. Nodes using TEC techniques exploit temporal correlation to reduce the number of bits.

Here we do not consider a specific TEC technique but assume that the compression factor, \( F_{\text{TEC}} \), of the TEC technique used is known. As a consequence, we can state that using a TEC technique each piece of raw data of \( w \)-bits will be represented after compression with \( b_{\text{TEC}} = w/F_{\text{TEC}} \) bits and that for each event a node must transmit \( L_{\text{TEC}} = Mw/F_{\text{TEC}} \) bits. So considering that \( N_m \) nodes sense the event, the overall number of bits transmitted for each event when TEC is used is

\[
B_{\text{TEC}} = N_m L_{\text{TEC}} = \frac{N_m Mw}{F_{\text{TEC}}}. \tag{7}
\]

As already stated, we assume that packets are transmitted through a load-balancing shortest-path scheme; that is, a node in cluster \( \text{CL}_{b_{\text{TEC}}-1} \) will select randomly a reachable node in the next cluster toward the sink (\( \text{CL}_b \)), and this forwarding scheme is repeated until the sink is reached. In this manner, information reaches the sink with the minimum number of hops.

(ii) DSC. According to DISCUS, we assume that only one node for each event (henceforward named the cell head) sends uncompressed measures (i.e., side information) into a packet of \( Mw \) bits while all the other nodes send compressed packets of \( Mb_{\text{DSC}} = Mw/F_{\text{DSC}} \) bits. Also in this case we assume that all packets are transmitted through a load-balancing shortest-path scheme. However, to improve reliability we assume that the cell head transmits its packets \( N_{r_{\text{DSC}}} \) times.

So considering that \( N_m \) nodes sense the event, the overall number of bits transmitted for each event when DSC is used is

\[
B_{\text{DSC}} = N_{r_{\text{DSC}}} Mw + (N_m - 1) \frac{Mw}{F_{\text{DSC}}}. \tag{8}
\]

(iii) CS. We assume that the cell header collects the packets of the other nodes in the same cell and sends them by applying CS. More precisely, the collected measures can be represented by a matrix \( X \) of \( N_m \times M \) values of \( w \)-bits each where the \( i \)th column \( x_i \) represents the measures taken by \( N_m \) nodes almost at the same time. Considering that such values are highly correlated in both space and time by taking a proper transform, we obtain with high probability a sparse vector. For instance, we can assume that DCT is applied to each column vector \( x_i \) and that only \( k \) DCT coefficients will be nonzero. In this case the cell head needs to send \( m = O(k \log (N_m/k)) \) measurements for each column, that is, \( M \cdot m \) measurements for each event.

For simulation purpose, we suppose that CS measurements are represented by \( b_{\text{CS}} \) bits and that those measurements are sent through \( m \) packets of \( L_{\text{CS}} = M \cdot b_{\text{CS}} \) bits each through a load-balancing shortest-path scheme.

So the overall number of bits transmitted by the cell head for each event when CS is used is

\[
B_{\text{CS}} = mL_{\text{CS}} = mMb_{\text{CS}}. \tag{9}
\]

Other choices are possible without altering the overall number of transmitted bits \( B_{\text{CS}} \); for instance, we could have considered \( M \) packets of \( m \cdot b_{\text{CS}} \) bits, but the previous choice will simplify comparisons. In particular, we will show that with the above choice comparison results will be independent of \( M \) so our results will be valid for both event-driven (\( M = 1 \)) and time/query-driven (\( M \gg 1 \)) techniques.

(iv) CRT. CRT is exploited as shown in [16].

In particular, we suppose that for each event \( N_m \) source nodes send their packets to a common set of one-hop relayers named CRT relayers. Henceforward, we indicate by \( N_{\text{CRT}} \) the number of CRT relayers.

As in the case of CS, the collected measures can be represented by a matrix \( X \) of \( N_m \times M \) values of \( w \)-bits each where the \( i \)th column \( x_i \) represents the measures taken by \( N_m \) nodes almost at the same time.

CRT relayers process the data of each column \( x_i \) in two steps:

1. In the first step, received data are compressed with a compression algorithm by obtaining a binary sequence \( S \).
2. In the second step, CRT is applied to improve reliability by splitting the binary sequence \( S \) so that each CRT relayer forwards a CRT component.

It is worth noting that each CRT relayer will independently compress the received packets by obtaining the same sequence \( S \). This is possible mainly because as they receive the
same data set and apply the same compression algorithm, the compressed sequence $S$ obtained is the same for all relayers.

Henceforward, we indicate by $w_S$ the length of the compressed sequence $S$.

CRT relayers split the binary sequence $S$ they have constructed and forward it. Specifically, the sequence $S$ is interpreted as an integer $Z_S = \sum_{i=0}^{w_S-1} s_i \cdot 2^i$ (where $s_i$ are bits of $Z_S$) and by properly choosing the set of prime numbers $\{p_j\}$ each CRT relayer calculates and forwards the corresponding CRT component $z_j = Z_S \mod p_j$.

Note that $\lfloor \log_2(p_j) \rfloor$ is the number of bits needed to represent $z_j$, so the overall number of bits transmitted by the CRT relayers is

$$B_{\text{CRT}} = M \sum_{j=1}^{N_{\text{CRT}}} \lfloor \log_2 (p_j) \rfloor. \quad (10)$$

From the theory of CRT, the sink will be able to reconstruct all raw measurements from the CRT components provided that the reconstruction condition is satisfied (i.e., $\prod_{j=1}^{N_{\text{CRT}}} p_j \geq 2^{w_S}$).

Note that the reconstruction condition can be satisfied by multiple sets of prime numbers; however, to reduce the number of bits needed to represent values $z_j$, and therefore the overall number of bits sent, it is preferable to choose the smallest possible set of primes, which we refer to as the Minimum Primes Set (MPS).

For instance, if $N_{\text{CRT}} = 4$ and $w_S = 40$, the MPS will be $\{1019, 1021, 1031, 1033\}$. In fact, this is the set of the smallest four consecutive primes that satisfy the relationship $\prod_{j=1}^{N_{\text{CRT}}} p_j \geq 2^{40}$.

However, when the set of primes is chosen as above, the message can be reconstructed if and only if all the CRT components are correctly received by the sink. So, to take into account the possible losses due to the wireless medium unreliability, we use the MPS with $f$ admissible failures (MPS$-f$), that is, the set of the smallest consecutive primes that satisfy the reconstruction condition even if $f$ CRT components are lost. As shown in [16], when $w_S, N_{\text{CRT}}$, and $f$ are fixed, the MPS$-f$ set is unique so CRT relayers can obtain the MPS$-f$ in a distributed manner.

### 4.3. Source Model

As shown in [13] and references therein, differences among two consecutive samples of several real-world data (temperature, humidity, solar radiation, etc.) fit well with Gaussian distributions. So, in this paper we consider that sensed data $x_i$ are approximated by a Gaussian distribution and that they are correlated in both space and time.

This choice is motivated also by the fact that several analytical results are well known for Gaussian distribution and can be readily exploited to obtain the maximum lossless compression factor for correlated Gaussian sources.

As is well known when compression of discrete sources is considered, Shannon’s entropy $H$ gives the lossless compression limit.

For Gaussian correlated data, under suitable assumptions and without loss of generality, it can be shown that, considering $X_1, \ldots, X_N$ obtained from quantization of continuous Gaussian variables $Y_1, \ldots, Y_N$, the joint entropy is [51]

$$H(X_1, \ldots, X_N) = h(Y_1, \ldots, Y_N)$$

$$= \frac{1}{2} \log_2 \left( \left(2\pi e\right)^N \cdot |\Sigma| \right). \quad (11)$$

where $|\Sigma|$, known as generalized variance, is the determinant of the covariance matrix $\Sigma$ and $h(\cdot)$ is the differential entropy (rigorously speaking, we have to distinguish between differential entropy $h(Y)$ for a continuous source $Y$ (i.e., before the A/D conversion) and Shannon’s information entropy $H(X)$ for a discrete source $X$ (i.e., after quantization introduced by the A/D conversion); however, it is straightforward to prove that their values coincide when unitary quantization step is considered, as done in this paper). In particular for Gaussian sources with the same correlation coefficient $\rho$ and variance $\sigma^2$, the generalized variance is $|\Sigma| = \sigma^{2N} \cdot (1 + (N-1)\rho) \cdot (1 - \rho)^{N-1}$ and therefore

$$H = N \log_2 \left( \sqrt{2\pi e (1 - \rho_c)} \cdot \sigma \right) + \log_2 \left( \frac{1}{1 - \rho_c} \right). \quad (12)$$

Moreover, considering that for a broad range of values (i.e., $\rho_c \in [0, 0.99]$, $N \geq 8$) the second term is negligible, it follows that

$$H = N \log_2 \left( \sqrt{2\pi e (1 - \rho_c)} \cdot \sigma \right). \quad (13)$$

Therefore, ideal (maximum) lossless compression factor for Gaussian variables considering blocks of $N$ correlated values of $w$-bit each can be obtained as

$$F_{\text{C,ideal}} = \frac{w \cdot N}{H} = \frac{w}{\log_2 \left( \sqrt{2\pi e (1 - \rho_c)} \cdot \sigma \right)}. \quad (14)$$

Throughout the paper, we assume that

$$F_{\text{DSC}} = F_{\text{TEC}} = F_{\text{C,ideal}} \quad (15)$$

(i.e., maximum lossless compression factor). As a consequence, we have

$$b_{\text{DSC}} = b_{\text{TEC}} = \log_2 \left( \sqrt{2\pi e (1 - \rho_c)} \cdot \sigma \right). \quad (16)$$

As regards CS, the actual compression factor is related to the sparsity level $s = k/N_{\text{cs}}$ in fact

$$F_{\text{CS}} = \frac{N_{\text{cs}} Mw}{B_{\text{CS}}} = \frac{w}{b_{\text{CS}}} \cdot \frac{1}{2s \log(1/s) + (5/4)s + 1}. \quad (17)$$

Note that $F_{\text{CS}}$ is a decreasing function of $s$.

So we consider two cases: an ideal sparsity level $s_{\text{ideal}}$ such that $F_{\text{CS}} = F_{\text{C,ideal}}$ and a slightly greater value $s' = 1.2 \cdot s_{\text{ideal}}$.

Finally in the case of CRT we consider that the simple MinDiff algorithm proposed in [4] is used for compression.
Basically, MinDiff encodes a set of uncompressed data \( U = \{ x_i \} \) with another set of compressed data \( C = \{ \mu, d_1, \ldots, d_i \} \), where \( \mu = \min x_i \) is the minimum of the values in \( U \) and \( d_i \) are the differences \( d_i = x_i - \mu \) represented with \( b_i = \lceil \log_2 (\max |d_i| + 1) \rceil \) bits each.

The number of bits \( b_i \) needed to represent the set of differences and the value of \( \mu \) are necessary for proper reconstruction and therefore an overhead of \( w + \log_2(w) \) bits must be considered.

Therefore, its compression factor considering blocks of \( N \) values of \( w \)-bit each can be obtained as

\[
F_{C, \text{MinDiff}} = \frac{w \cdot N}{w + \log_2(w) + N \cdot b_i}. \quad (18)
\]

### 4.4. Energy Model

Similarly to other works (e.g., [16, 19]), we consider a simple energy model where for each bit to be transmitted a node spends an energy equal to \( e_b \). Apparently, it seems that the model neglects the energy needed for computation and for reception but this is not true if we reflect on the fact that in sensor networks the number of bits transmitted, the number of bits received, and the number of processing operations are all proportional to the number of sensed measures. So energy needed for computation and for reception can be easily included in \( e_b \).

For instance, let us suppose that a node for sensing and processing \( M \) measures of \( w \) bits needs an energy equal to \( Mw \cdot e_c \) and that, using a proper compression technique with a compression factor equal to \( F_c \), it reduces the number of bits to be transmitted from \( Mw \) to \( Mw/F_c \). In this case, the overall energy is \( Mw \cdot e_c + (Mw/F_c) \cdot e_{RAW} \) which we can rewrite as \( (Mw/F_c) \cdot e_b \) considering \( e_b = F_c e_c + e_{RX} \).

Finally, if also the energy needed for reception must be included and it differs from the energy used for transmission, considering that for almost all nodes the number of bits received is equal to the number of bits transmitted, it will be sufficient to use \( e_b = F_c e_c + e_{RX} \).

Therefore, the main simplification introduced by our model is that we consider \( e_b \) distance-independent; that is, we do not consider that \( e_{RX} \) could be adaptively changed by the MAC layer on the basis of distance between source and destination node.

### 5. Performance Metrics

In order to estimate the energy efficiency of the above techniques, let us introduce the Energy reduction factor, \( ERF_X \), which represents the percentage reduction of the energy spent using a specific data gathering technique \( X \) as compared to the case when raw measures are directly sent \( (ERF_{RAW}) \). This metric is defined as

\[
ERF_X = 100 \cdot \frac{E_{RAW} - E_X}{E_{RAW}}, \quad (19)
\]

where \( e_b \) is the energy spent by a node to transmit a single bit and \( B_X \) is the overall number of bits transmitted considering the specific data gathering technique derived in Section 4 (see (7)–(10)).

For instance, in the case of TEC-based data gathering is

\[
ERF_{TEC} = 100 \cdot \left( 1 - \frac{1}{F_{TEC}} \right). \quad (21)
\]

However, in the case of lossy network, the number of transmitted and received bits is different and the expected energy reduction factor has to be expressed taking into account the actual number of bits forwarded.

For comparison purpose, we decided to evaluate energies considering nodes belonging to cluster 2 (i.e., \( CL_2 \)).

We restrict our analysis to the nodes of the second cluster for two reasons. Firstly, these nodes are the most critical as they represent the sinks neighbors. In fact, if these nodes run out of energy, the sink remains isolated. Secondly, network lifetime is defined as the time until the first node in the network dies and with high probability, if not certainty, this node belongs to \( CL_2 \) considering that all messages are routed to the sink through these nodes.

Finally, considering that network lifetime is related to the maximum energy consumed by a node in this paper, we investigate also the energy reduction factor related to the maximum energies:

\[
ERF_{X, \text{max}} = \frac{E_{RAW, \text{max}} - E_{X, \text{max}}}{E_{RAW, \text{max}}}. \quad (22)
\]

Concerning reliability, we consider that a node fails to forward a packet with probability \( p_e \) and evaluate the ratio \( P_{R,X} \) between the number of raw measurements that are obtained by the sink and the number of raw measurements generated from source nodes or, equivalently,

\[
P_{R,X} = 1 - \frac{M_{\text{lost,X}}}{M \cdot N_m}, \quad (23)
\]

where \( M_{\text{lost,X}} \) is the number of raw measurements that are lost due to the network and/or reconstruction errors.

\( P_{R,\text{TEC}} \) can be easily estimated by assuming a perfect decoding technique where all received data are correctly decoded by the sink.

In fact if \( h \) is the number of hops needed to reach the sink and \( p_e \) is the probability that a node fails to forward a packet, the probability that a packet is lost is \( p_{\text{lost}} = 1 - (1 - p_e)^h \) and therefore the expected number of lost data is \( M_{\text{lost,TEC}} = N_m M p_e \). As a consequence

\[
P_{R,\text{TEC}} = 1 - \frac{M_{\text{lost,TEC}}}{M \cdot N_m} = 1 - p_{\text{lost}} = 1 - (1 - p_e)^h. \quad (24)
\]

Note that the reliability of TEC techniques is related only to network parameters \( p_e \) and \( h \) and cannot be improved without relying on channel coding techniques (e.g., FEC).

Differently from TEC techniques, for all the other data gathering techniques, reliability can be improved with a
proper settings of design parameters. Nevertheless, a trade-off exists between reliability and energy saving as briefly discussed below.

(i) CRT. In the case of CRT data gathering primes, numbers can be selected so that all raw measures can be reconstructed even if at most $f$ CRT components are lost.

As shown in [16], when $f$ is fixed the reliability can be estimated as

$$
P_{R,CRT} = \sum_{i=0}^{f} \binom{N_{CRT}}{i} p_n^i (1 - p_n)^{N_{CRT} - i}, \quad (25)$$

where $p_n = 1 - (1 - p_e)^h$ is the probability that a CRT component is lost.

As general rule, high reliability can be obtained by fixing $f$ so that $f = N_{CRT} p_n + k_f$ where $k_f$ is a small constant on the order of $\sqrt{N_{CRT} p_n}$.

This result can be justified by considering that $P_{R,CRT}$ (see (25)) can be approximated by the cumulative distribution function of a normal variable with mean $N_{CRT} p_n$ and variance $N_{CRT} p_n (1 - p_n)$.

For instance, in Figure 1 we show the reliability $P_{R,CRT}$ for different values of $N_{CRT}$ and $f$ when $p_n = 0.04$. As it is possible to observe, small values of $f$ (e.g., $f = 6$) are sufficient to achieve high values of reliability (> 0.99).

Higher reliability can be obtained by further increasing the value of the parameter $f$. However, by increasing $f$ energy consumptions are increased too so in the next section we investigated the trade-off between reliability and energy consumptions for different values of $f$.

(ii) DSC. Also in the case of DSC, the probability to lose a packet is $p_n = 1 - (1 - p_e)^h$. However, DSC compressed measures cannot be recovered if side information (i.e., packets generated by the cell head) is not received.

So, in order to improve the reliability we considered that cell head transmits $N_{r,DSC}$ times the side information.

In this case, DSC reliability can be evaluated as

$$
P_{R,DSC} = 1 - \frac{N_{lost,DSC}}{N_m}, \quad (26)$$

where $N_{lost,DSC}$ is the number of lost packets by taking into account that all packets related to the same event are lost if no one of the cell head packets arrives. Note that the expected value of $N_{lost,DSC}$ is $N_{lost,DSC} = N_m p_n^{N_{r,DSC}} + (N_m - 1)(1 - p_n^{N_{r,DSC}}) p_n$.

In Figure 2, we can compare reliability of TEC and DSC for two different values of the loss probability per hop ($p_e = 0.01$ and $p_e = 0.05$) and a fixed number of hops, $h = 4$, when different values of $N_{r,DSC}$ and different number of source nodes $N_m$ are considered.

As it is possible to observe by fixing $N_{r,DSC} = 3$ we have $P_{R,DSC} \geq P_{R,TEC}$ for a broad range of $p_e$ and $N_m$ (i.e., $p_e \in [0.01, 0.05]$ and $N_m \in [10, 100]$). Note also that further increasing $N_{r,DSC}$ does not improve $P_{R,DSC}$ so much. This can be justified considering that, for high values of, that is, $N_{r,DSC} \rightarrow \infty$, it follows that $P_{R,DSC} \leq 1 - (N_m - 1) p_n / N_m = 1 - p_e + p_n / N_m = P_{R,TEC}$.

Obviously, reliability increases with number of retransmissions $N_{r,DSC}$ but at the cost of reducing energy saving. So in the next section we investigated the trade-off between reliability and energy consumptions for different values of $N_{r,DSC}$.

(iii) CS. By indicating with $m$ the number of packets sent with CS, the probability to receive at least $m^*$ packets is

$$
P_{CS} = \sum_{i=0}^{m^*} \binom{m}{i} p_n^i (1 - p_n)^{m - i}. \quad (27)$$

Therefore, by increasing $m$ it is possible to guarantee that, with the desired probability $P_{CS}$, at least $m^*$ packets are received by the sink. As a general rule, high reliability can be obtained by fixing $m$ so that $m = m^* / (1 - p_e) + k_m$, where $k_m$ is a small constant.

However, differently from all previous techniques, CS reliability is not related only to the number of received packets so that $P_{CS}$, henceforward named network reliability, is not
the actual CS reliability (this justifies why we used notation  \( P_{CS} \) instead of \( P_{CS} \)). In fact, CS techniques are based on reconstruction algorithms that could introduce errors so that reconstructed values \( \hat{x}_i \) could differ from original raw data \( x_i \).

Nevertheless, if we assume that raw data \( x_i \) are quantized values (as usual in WSNs where raw measures come from ADCs), we can state that quantized measures can be exactly recovered if the reconstruction error \( |\hat{x}_i - x_i| \) is smaller than quantization error \( \Delta_q/2 \), where \( \Delta_q \) is the quantization step used for quantizing \( x_i \).

Therefore, in our simulations, the reconstruction error, \( |\hat{x}_i - x_i| \), is evaluated for each measure and raw data is considered lost if this error is greater than \( \Delta_q/2 \).

From simulation point of view, actual CS reliability can be evaluated as

\[
P_{R,CS} = 1 - \frac{M_{lost,CS}}{M \cdot N_m},
\]

where \( M_{lost,CS} \) is the number of lost measures considering both packet loss and reconstruction error.

We show in the next section that by choosing \( b_{CS} = \omega + 3 \) and \( m = m^*/(1 - p_e) + 2 \) we can obtain \( P_{R,CS} \approx 1 \).

Two reconstruction algorithms are considered for CS in this paper: ideal (i.e., oracle-based) reconstruction, where exact positions of nonzero values are assumed to be known at the sink node, and CoSaMP [27]. To distinguish among them, we indicate their reliability as \( P_{R,CSI} \) and \( P_{R,CoSaMP} \), respectively.

### 6. Simulations Results

In this section, we compare data gathering techniques in terms of energy consumptions and reliability.

The results have been obtained through a custom C++ simulator. For each set of parameters mean results are reported considering 20 random topologies where nodes are uniformly distributed in a square area of size 600 \times 600 \([\text{m}^2]\), with density \( \rho \) \([\text{nodes/m}^2]\). We also assume that \( E_r \) events randomly occur in a faraway cluster (e.g., CL so that \( h = 4 \) hops are needed to reach the sink) and that each event is detected by \( N_m = \rho \pi r^2 \) source nodes (where \( r \) is the sense radius).

If not otherwise stated, \( E_r = 300 \) events are considered and raw data are represented with \( \omega = 12 \) bits (which is a typical number for ADCs used in sensor networks).

#### 6.1. ERF with Reliable Networks.

To assess the simulator we first analyzed an ideal (fully reliable) WSN and evaluated the ERF for different values of \( \sigma \) and \( \rho_e \) when different data gathering techniques (TEC, CRT, DSC, and CS) are considered. In particular, for CS two cases have been considered: an ideal case (CSI) where the sparsity level \( s = k/N_m \) is fixed to the minimum value \( s = s_{\text{ideal}} \) such that \( F_{C,CSI} = F_{C,\text{ideal}} \) (see (17) and (14)) and a second case (CS2) where a slightly greater value \( s' = 1.2 \cdot s_{\text{ideal}} \) is used.

As shown in Figures 3 and 4, the results obtained through the analytical model (see (20)) and those reported by the simulator are very close to each other for all the values of \( \sigma \) and \( \rho_e \) considered. These results confirm the validity of our model.

The same results are obtained for different values of \( M \), that is, by changing the number of raw measures per packet. This can be easily justified by the fact that \( E_{RAW} = N_m \cdot M \cdot w_{eb} \) and \( E_X = \rho \cdot N_m \cdot r \cdot p_e \) are both proportional to \( M \) so their ratio and therefore also the ERF (see (19)) are independent of \( M \). So simulation results for different values of \( M \) are not shown for the sake of space.

On the basis of the previous results, we can state that in the case of reliable networks TEC and DSC have a greater ERF and therefore it seems that they should be preferred to the other analyzed data gathering techniques. However, as shown in the next section, this result is not true when reliability is an issue.

Note that in our simulations \( E_{RF,\text{DSC}} \) and \( E_{RF,\text{TEC}} \) are almost the same because we fixed the same values for spatial and temporal correlation coefficients. Obviously, when spatial and temporal correlation are not the same the different results can be obtained.

Finally, note that although CRT appears to be the worst in terms of ERF it is the only data gathering technique where an actual compression algorithm (MinDiff) has been considered (for all the other techniques, ideal compression factors have been assumed). So previous simulations results allow quantifying the penalty in using a simple compression
algorithm (MinDiff) instead of more complex techniques (at least when Gaussian data are considered).

By comparing Figures 3 and 5, we can see that CRT and CS achieve different values of ERF when the sensing radius, \( r \), changes. In particular, ERF\(_{\text{CSI}}\) is no more able to reach the same values of ERF\(_{\text{DSC}}\) for low values of \( r \).

This can be justified considering that when \( r \) decreases, the sparsity level \( s \propto 1/N_m \) increases, and, as a consequence, the compression factor \( F_{\text{CS}} \) decreases too (see (17)).

It is worth noting that our analytical model is able to anticipate this result.

Also in the case of CRT, the ERF slightly decreases for lower values of \( r \) due to the fact that when \( N_m \) decreases the overhead of the MinDiff algorithm is more relevant and \( F_{\text{C},\text{MinDiff}} \) is not able to approach \( F_{\text{C},\text{ideal}} \) (see (18)).

Similar considerations can be made about the node density \( \rho \). In fact, as it is possible to observe by comparing Figures 3 and 6, ERF\(_{\text{CSI}}\) and ERF\(_{\text{CRT}}\) decrease for lower values of \( \rho \) (i.e., lower values of \( N_m \)).

It is worth noting that Figures 6 and 5 report quite similar ERF values despite different values of \( \rho \) and \( r \) being used for simulation. This result can be justified by considering that network density and sensing radius have been changed but without altering the overall number of source nodes (i.e., \( N_m = \pi \cdot \rho \cdot r^2 = 45 \) in both cases).

6.2. ERF with Unreliable Networks. In Figures 7 and 8, reliability and ERF of data gathering techniques for \( p_e = 0.01 \) and different values of \( \sigma \) are reported.

On the basis of the simulation results, we can state that even in the case of unreliable networks TEC and DSC have greater ERF (see Figure 8). However, their reliability is fully determined by the packet loss probability and cannot be improved; instead, by using CRT and CS higher reliability can be achieved by increasing \( f \) and \( m \), respectively.

In particular, as shown in Figure 7, by fixing \( f = 8 \) a reliability higher than 0.975 can be achieved for CRT and even higher values can be obtained using CS when \( m = m^*/(1 - p_n) + 2 \) CS measures are sent.

It is important to note that the reliability plotted for CS is the actual reliability obtained after reconstruction with an ideal (oracle-based) reconstruction algorithm (i.e., \( P_{R,\text{CSI}} \) and not \( \overline{P}_{R,\text{CS}} \)).

By choosing \( m \) so that \( m = m^*/(1 - p_n) + 2 \), we have with high probability (i.e., \( \overline{P}_{R,\text{CS}} > 0.99 \)) that the sink receives at least \( m^* \) CS measures, that is, the minimum number of
measures sufficient for reconstruction, and therefore original measurements can be perfectly recovered (so that \( P_{R,CS} = 1 \)).

In the next subsection, we will show that this is true only if \( b_{CS} = w + 3 \), as chosen for our simulations.

Obviously, high reliability is achieved at the cost of a lower ERF but, by comparing Figures 8 and 3, we can state that the impact on the ERF is quite low (both \( \text{ERF}_{\text{CRT}} \) and \( \text{ERF}_{\text{CS}} \) decrease by a few percent).

As a consequence when high reliability is needed even with unreliable networks, CRT and CS should be preferred.

### 6.3. CS Reliability

In all the previous simulations, we have fixed the number of bits \( b_{CS} \) used for representing quantized CS measures equal to \( w + 3 \). A careful reader could observe that this choice is questionable and that by reducing \( b_{CS} \) higher values of \( \text{ERF}_{\text{CS}} \) can be obtained. This consideration is partially true: effectively, \( \text{ERF}_{\text{CS}} \) increases for lower values of \( b_{CS} \), but our simulation results show that using values of \( b_{CS} \) below \( w + 3 \) is not possible to have perfect reconstruction (i.e., \( P_{R,CS} = 1 \)).

To convince the reader in Figure 9, we report simulation results about the actual reliability \( P_{R,CS} \) for different values of \( b_{CS} \) and sparsity levels when \( w = 12 \). Simulation parameters are those reported in Figure 3.

As it is possible to observe, when \( b_{CS} < 15 \) reliability quickly decreases.

Similar results have been obtained for different values of \( w \).

However, in practice, actual reliability depends also on the reconstruction algorithm used. For the sake of completeness, we report in Figure 10 CS reliability when the CoSaMP [27] algorithm is used for reconstruction.

As it is possible to observe also in this case \( b_{CS} = 15 \) is needed for achieving high reliability. Nevertheless, perfect reliability is not obtained with CoSaMP when \( m = m^* \). So in some cases CRT could be preferred to CS because reliability can be better predicted.

### 6.4. \( \text{ERF}_{\text{max}} \) and Network Lifetime

The ERF metric is an expression of mean energy consumptions; instead, network lifetime is more closely related to maximum energy consumptions.

In Figure 11, we report \( \text{ERF}_{\text{max}} \) instead of ERF for different data gathering techniques. \( \text{ERF}_{\text{max}} \) is evaluated on the basis of (22), that is, considering the maximum energy consumptions for nodes belonging to cluster \( CL_2 \). Maximum energies have greater variations in comparison to mean values so in order
to have higher confidence we increased the number of events to $E_v = 3000$.

As it is possible to observe, TEC and DSC have higher $\text{ERF}_{\text{max}}$ and therefore greater network lifetime can be expected when DSC and TEC are used for data gathering.

Finally, note that CRT and CS have similar performance if the actual sparsity degree of CS is 20% more than the minimum (ideal) value. These observations can be extended also to all the previous simulations.
7. Conclusions and Future Works

In this paper, we have compared several data gathering techniques used in WSNs by using both simulation results and analytical models. In particular, the effectiveness of the above techniques has been investigated in terms of reliability (packet loss and reconstruction errors) and energy efficiency (i.e., ERF and network lifetime) by systematically sampling the parameter space (i.e., number of nodes, transmission range, and sparsity). Basically we can summarize our results as follows:

(i) DSC and TEC techniques should be preferred for maximizing network lifetime.

(ii) CS should be preferred when high reliability is needed.

(iii) CRT should be preferred for its inherent low complexity.

As a consequence, we can state that there is no best solution for all possible applications and that only the trade-off between energy consumptions, reliability, and complexity can drive the choice of the data gathering technique to be used for a specific application.

As future work, we plan to refine and improve the model to deal with actual correlated measurements (i.e., not only Gaussian data) and more realistic propagation channels (i.e., by taking into account actual distance between nodes).

Notations

- TEC: Transform and Encoding-Based Compression
- CS: Compressive sensing
- DSC: Distributed source coding
- CRT: Chinese Remainder Theorem
- X: Generic data gathering technique; that is, $X \in \{\text{TEC, CS, DSC, CRT}\}$
- $U = \{x_i\}$: A set of uncompressed values (raw measurements)
- $w$: Number of bits used to represent a raw value (without compression)
- $M$: Number of measurements/words per packet
- $N_m$: Number of source nodes
- $X$: Matrix of $N_m \times M$ raw data of $w$-bits each
- $x_i$: $i$th column of the matrix $X$ (measures taken by $N_m$ nodes almost at the same time)
- $F_X$: Compression factor of the data gathering technique $X \in \{\text{TEC, CS, DSC, CRT}\}$
- $b_X$: Number of bits used to represent a compressed measure with data gathering technique $X$
- $L_X$: Number of bits per packet when data gathering technique $X$ is used
- $B_X$: Overall number of bits transmitted for each event when the data gathering technique $X$ is used
- $\text{ERF}_X$: Energy reduction factor of the data gathering technique $X \in \{\text{TEC, CS, DSC, CRT}\}$
- $P_X$: Reliability of the data gathering technique $X \in \{\text{TEC, CS, DSC, CRT}\}$
- $\mu$: Minimum value within set $\{x_i\}$; that is, $\mu = \min \{x_i\}$
- $d_i$: Difference with the minimum element of $\{x_i\}$; that is, $d_i = x_i - \mu$
- $y_i$: Compressed sensing measurement
- $m$: Number of CS measurements
- $m^*$: Minimum/sufficient number of CS measurements for reconstruction
- $k$: Number of nonzero components of a $k$-sparse signal
- $s$: Sparsity level of CS measurements; that is, $s = k/n$
- $\rho$: Correlation coefficient of raw measurements
- $\sigma$: Standard deviation of raw measurements
- $R$: Maximum transmission distance (coverage radius)
- $r$: Sense radius of a node
- $N_{\text{CRT}}$: Number of CRT relays
- $G$: Grid size (edge in meter)
- $\rho$: Node density
- $e_g$: Energy spent for each transmitted bit
- $H$: Shannon’s entropy
- $N_{r,\text{DSC}}$: Number of retransmissions for the cluster/cell head when DSC is used
- $\text{CL}_h$: Cluster number $h$
- $h$: Number of hops between nodes in clusters $\text{CL}_{h+1}$ and $\text{CL}_h$
- $E_v$: Number of events used for simulations.

Competing Interests

The authors declare that they have no competing interests.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “A survey on sensor networks,” IEEE Communications Magazine, vol. 40, no. 8, pp. 102–105, 2002.
[2] Z. Xiong, A. D. Liveris, and S. Cheng, “Distributed source coding for sensor networks,” IEEE Signal Processing Magazine, vol. 21, no. 5, pp. 80–94, 2004.
[3] D. Zordan, B. Martinez, I. Vilajosana, and M. Rossi, “On the performance of lossy compression schemes for energy constrained sensor networking,” ACM Transactions on Sensor Networks, vol. 11, no. 1, article 15, 2014.
[4] G. Campobello, O. Giordano, A. Segreto, and S. Serrano, “Comparison of local lossless compression algorithms for wireless sensor networks,” Journal of Network and Computer Applications, vol. 47, pp. 23–31, 2015.
[5] J. Chou, D. Petrovic, and K. Ramachandran, “A distributed and adaptive signal processing approach to reducing energy consumption in sensor networks,” in Proceedings of the 22nd Annual Joint Conference of the IEEE Computer and Communications (INFOCOM ’03), vol. 2, pp. 1054–1062, IEEE, San Francisco, Calif, USA, March-April 2003.
[6] X. He, X. Zhou, M. Juntti, and T. Matsumoto, “Data and error rate bounds for binary data gathering wireless sensor networks,” in Proceedings of the IEEE 16th International Workshop on Signal Processing Advances in Wireless Communications (SAPWC ’15), pp. 505–509, Stockholm, Sweden, June 2015.

[7] C. Karakus, A. C. Gurbuz, and B. Tavli, “Analysis of energy efficiency of compressive sensing in wireless sensor networks,” IEEE Sensors Journal, vol. 13, no. 5, pp. 1999–2008, 2013.

[8] H. Zheng, S. Xiao, X. Wang, X. Tian, and M. Guizani, “Capacity and delay analysis for data gathering with compressive sensing in wireless sensor networks,” IEEE Transactions on Wireless Communications, vol. 12, no. 2, pp. 917–927, 2013.

[9] H. Zheng, F. Yang, X. Tian, X. Gan, X. Wang, and S. Xiao, “Data gathering with compressive sensing in wireless sensor networks: a random walk based approach,” IEEE Transactions on Parallel and Distributed Systems, vol. 26, no. 1, pp. 35–44, 2015.

[10] L. Zhu, B. Ci, Y. Liu, and Z. D. Chen, “Data gathering in wireless sensor networks based on reshuffling cluster compressed sensing,” International Journal of Distributed Sensor Networks, vol. 2015, 13 pages, 2015.

[11] E. J. Candès and T. Tao, “Near-optimal signal recovery from random projections: universal encoding strategies?” IEEE Transactions on Information Theory, vol. 52, no. 12, pp. 5406–5425, 2006.

[12] D. L. Donoho, “Compressed sensing,” IEEE Transactions on Information Theory, vol. 52, no. 4, pp. 1289–1306, 2006.

[13] T. Srisooksai, K. Keamarungsi, P. Lamsrichan, and K. Araki, “Practical data compression in wireless sensor networks: a survey,” Journal of Network and Computer Applications, vol. 35, no. 1, pp. 37–59, 2012.

[14] J.-H. Chang and L. Tassiulas, “Maximum lifetime routing in wireless sensor networks,” IEEE/ACM Transactions on Networking, vol. 12, no. 4, pp. 609–619, 2004.

[15] Y. Wu, Z. Mao, S. Fahmy, and N. B. Shroff, “Constructing maximum-lifetime data-gathering forests in sensor networks,” IEEE/ACM Transactions on Networking, vol. 18, no. 5, pp. 1711–1584, 2010.

[16] G. Campobello, A. Leonardi, and S. Palazzo, “Improving energy saving and reliability in wireless sensor networks using a simple CRT-based packet-forwarding solution,” IEEE/ACM Transactions on Networking, vol. 20, no. 1, pp. 191–205, 2012.

[17] D. Gong and Y. Yang, “Low-latency SINR-based data gathering in wireless sensor networks,” IEEE Transactions on Wireless Communications, vol. 13, no. 6, pp. 3207–3221, 2014.

[18] C. Hua and T.-S. P. Yum, “Optimal routing and data aggregation for maximizing lifetime of wireless sensor networks,” IEEE/ACM Transactions on Networking, vol. 16, no. 4, pp. 892–903, 2008.

[19] G. Campobello, S. Serrano, L. Galluccio, and S. Palazzo, “Applying the Chinese remainder theorem to data aggregation in wireless sensor networks,” IEEE Communications Letters, vol. 17, no. 5, pp. 1000–1003, 2013.

[20] X. Xing, D. Xie, and G. Wang, “Energy-balanced data gathering and aggregating in wsn: a compressed sensing scheme,” International Journal of Distributed Sensor Networks, vol. 2015, Article ID 583191, 10 pages, 2015.

[21] C. Lim, “Network coding for severe packet reordering in multi-hop wireless networks,” International Journal of Distributed Sensor Networks, vol. 2015, Article ID 379108, 9 pages, 2015.

[22] M. Rossi, M. Hooshmand, D. Zordan, and M. Zorzi, “Evaluating the gap between compressive sensing and distributed source coding in WSN,” in Proceedings of the International Conference on Computing, Networking and Communications (ICNC ’15), pp. 911–917, IEEE, Garden Grove, Calif, USA, February 2015.

[23] F. Marcelloni and M. Vecchio, "An efficient lossless compression algorithm for tiny nodes of monitoring wireless sensor networks," The Computer Journal, vol. 52, no. 8, pp. 969–987, 2009.

[24] Y. Liang and W. Peng, “Minimizing energy consumptions in wireless sensor networks via two-modal transmission,” ACM SIGCOMM Computer Communication Review, vol. 40, no. 1, pp. 12–18, 2010.

[25] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," SIAM Review, vol. 43, no. 1, pp. 129–159, 2001.

[26] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," IEEE Transactions on Information Theory, vol. 53, no. 12, pp. 4655–4666, 2007.

[27] J. A. Tropp and D. Needell, “CoSaMP: iterative signal recovery from incomplete and inaccurate samples,” Applied and Computational Harmonic Analysis, vol. 26, no. 3, pp. 301–321, 2009.

[28] E. Candes and M. Wakin, "An introduction to compressive sampling," IEEE Signal Processing Magazine, vol. 25, no. 2, pp. 21–30, 2008.

[29] V. Chandrasekaran, B. Recht, P. A. Parrilo, and A. S. Willsky, "The convex geometry of linear inverse problems," Foundations of Computational Mathematics, vol. 12, no. 6, pp. 805–849, 2012.

[30] X. Wang, Z. Zhao, Y. Xia, and H. Zhang, “Compressed sensing for efficient random routing in multi-hop wireless sensor networks,” in Proceedings of the IEEE GLOBECOM Workshops (GC Wshps ’10), pp. 266–271, IEEE, Miami, Fla, USA, December 2010.

[31] A. Razi, K. Yasami, and A. Abedi, “On minimum number of wireless sensors required for reliable binary source estimation,” in Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC ’11), pp. 1852–1857, IEEE, Cancun, Mexico, March 2011.

[32] S. S. Pradhan and K. Ramchandran, “Distributed source coding using syndromes (discus): design and construction,” in Proceedings of the Data Compression Conference (DCC ’99), pp. 158–167, March 1999.

[33] J.-H. Chang and L. Tassiulas, “Routing for maximum system lifetime in wireless ad-hoc networks,” in Proceedings of the 37th Annual Allerton Conference on Communication Control and Computing, pp. 1191–1200, Monticello, Ill, USA, September 1999.

[34] J.-H. Chang and L. Tassiulas, “Energy conserving routing in wireless ad-hoc networks,” in Proceedings of the 19th IEEE Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM ’00), vol. 1, pp. 22–31, March 2000.

[35] S. Guo, Y. Yang, and C. Wang, “DaGCM: a concurrent data uploading framework for mobile data gathering in wireless sensor networks,” IEEE Transactions on Mobile Computing, vol. 15, no. 3, pp. 610–626, 2016.

[36] S. Guo, C. Wang, and Y. Yang, “Joint mobile data gathering and energy provisioning in wireless rechargeable sensor networks,” IEEE Transactions on Mobile Computing, vol. 13, no. 12, pp. 2836–2852, 2014.
[38] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, “Network information flow,” IEEE Transactions on Information Theory, vol. 46, no. 4, pp. 1204–1216, 2000.

[39] C. Fragouli, J.-Y. Le Boudec, and J. Widmer, “Network coding: an instant primer,” ACM SIGCOMM Computer Communication Review, vol. 36, no. 1, pp. 63–68, 2006.

[40] P. Ostovari, J. Wu, and A. Khreishah, Network Coding Techniques for Wireless and Sensor Networks, Springer, Berlin, Germany, 2013.

[41] S. Chachulski, M. Jennings, S. Katti, and D. Katabi, “Trading structure for randomness in wireless opportunistic routing,” ACM SIGCOMM Computer Communication Review, vol. 37, no. 4, pp. 169–180, 2007.

[42] I.-H. Hou, Y.-E. Tsai, T. F. Abdelzaher, and I. Gupta, “AdapCode: Adaptive network coding for code updates in wireless sensor networks,” in Proceedings of the 27th IEEE Communications Society Conference on Computer Communications (INFOCOM ’08), pp. 2189–2197, Phoenix, Ariz, USA, April 2008.

[43] Z. Yang, M. Li, and W. Lou, “R-code: network coding based reliable broadcast in wireless mesh networks with unreliable links,” in Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM ’09), pp. 1–6, IEEE, Honolulu, Hawaii, USA, November 2009.

[44] T. Ho, R. Koetter, M. Medard, D. R. Karger, T. Ho, and M. Effros, “The benefits of coding over routing in a randomized setting,” in Proceedings of the IEEE International Symposium on Information Theory, Yokohama, Japan, July 2003.

[45] T. Ho, M. Medard, R. Koetter et al., “A random linear network coding approach to multicast,” IEEE Transactions on Information Theory, vol. 52, no. 10, pp. 4413–4430, 2006.

[46] T. Voigt, U. Roedig, O. Landsiedel, K. Samarasinghe, and M. B. Prasad, “On the applicability of network coding in wireless sensor networks,” ACM SIGBED Review, vol. 9, no. 3, pp. 46–48, 2012.

[47] G. Campobello, A. Leonardi, and S. Palazzo, “On the use of Chinese remainder theorem for energy saving in wireless sensor networks,” in Proceedings of the IEEE International Conference on Communications (ICC ’08), pp. 2723–2727, Beijing, China, May 2008.

[48] A. Leonardi, G. Campobello, S. Serrano, and S. Palazzo, “Trade-offs between energy saving and reliability in low duty cycle wireless sensor networks using a packet splitting forwarding technique,” EURASIP Journal on Wireless Communications and Networking, vol. 2010, Article ID 932245, 2010.

[49] G. Campobello, A. Leonardi, and S. Palazzo, “A novel reliable and energy-saving forwarding technique for wireless sensor networks,” in Proceedings of the 10th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc ’09), pp. 269–278, New Orleans, La, USA, May 2009.

[50] T. A. A. Alshou`i, M. Hammoudeh, Z. Bandar, and A. Nisbet, “An overview and classification of approaches to information extraction in wireless sensor networks,” in Proceedings of the 5th International Conference on Sensor Technologies and Applications (SENSORCOMM ’11), pp. 255–260, IARIA, Nice, France, August 2011.

[51] T. M. Cover and J. A. Thomas, Elements of Information Theory, Wiley-Interscience, New York, NY, USA, 1991.
Submit your manuscripts at http://www.hindawi.com