An energy-aware spatio-temporal correlation mechanism to perform efficient data collection in wireless sensor networks

Leandro A. Villas\textsuperscript{a,b,e,*}, Azzedine Boukerche\textsuperscript{a}, Daniel L. Guidoni\textsuperscript{c}, Horacio A.B.F. de Oliveira\textsuperscript{d}, Regina Borges de Araújo\textsuperscript{e}, Antonio A.F. Loureiro\textsuperscript{b}

\textsuperscript{a}Paradise Research Laboratory – SITE, University of Ottawa, Ottawa, Canada
\textsuperscript{b}UWL – Research Laboratory, Federal University of Minas Gerais, Belo Horizonte, Brazil
\textsuperscript{c}Department of Computer Science, Federal University of São João del-Rei, São João del-Rei, Brazil
\textsuperscript{d}LAMP – Research Laboratory, Federal University of Amazonas, Manaus, Brazil
\textsuperscript{e}WINDIS – Research Laboratory, Federal University of São Carlos, São Carlos, Brazil

\textbf{A B S T R A C T}

Large scale dense wireless sensor networks (WSNs) will be increasingly deployed in different classes of applications for accurate monitoring. Due to their high density of nodes, it is very likely that information that is both spatially and temporally correlated can be detected by several nodes what can be exploited to save energy, a key aspect on these networks. Furthermore, it is important to take advantage of these correlations to decrease communication and data exchange. However, current proposals usually result in high delays and outdated data arriving at the sink node. In this work, we go further and propose a new algorithm, called Efficient Data Collection Aware of Spatio-Temporal Correlation (EAST), which uses shortest routes for forwarding the gathered data toward the sink node and fully exploit both spatial and temporal correlations to perform near real-time data collection in WSNs. Simulation results clearly indicate that our proposal can sense an event with a high accuracy of more than 99.7\% while still saving the residual energy of the nodes in more than 14 times when compared to the accurate data collection strategy reported in the literature.

Crown Copyright © 2012 Published by Elsevier B.V. All rights reserved.

1. Introduction

Wireless sensor networks (WSNs) [1–4] can be defined as a cooperative network of small, battery-operated, wireless sensor nodes whose main goal is twofold: to monitor their surroundings for local data and to forward the gathered data toward a sink node using typically multipath communication. This sink node will then be responsible for processing all of the received data from several source nodes and reporting them to a monitoring facility. This network architecture allows a number of novel monitoring-based applications in several areas such as environmental, medical, industrial, and military.

One of the main limitations of the WSNs is the battery-operated nature of their sensor nodes, which makes this kind of network highly energy-constrained. Thus, saving energy should be one of the main concerns of protocols and applications in WSNs. Since communication among the nodes is one of the main sources of energy consumption, most protocols in WSNs try to avoid or delay communication until it is really required [5–11]. However, by doing so, outdated and/or incomplete information is usually obtained by the sink node making the underlying application neither reliable nor useful.

In our work we go further from only delaying or avoiding communication by proposing a solution that gets the most out of each required communication. We do so by exploiting both the high density of WSN nodes and the observed similarity of nearby gathered data. This not only extends the lifetime of a WSN but also provides near real-time information about the monitored area. Our proposal extends some current published studies [12–14], which show that in several WSN applications, nearby nodes data tend to be correlated in both time and space:

- \textit{Temporal correlation}: the change pattern of current sensor readings is equal or similar to the readings observed at previous times.
- \textit{Spatial correlation}: the change pattern of the data sensed by nearby nodes is expected to be the same or similar.

These correlations have been exploited by some current techniques such as spatial suppression [15–19] and temporal...
that detect an event exists when those nodes are geographically
lowing, we present some of these studies as well as the benefits of
and/or temporal data correlations algorithms for WSNs. In the fol-
summary of the basic characteristics of the main proposed spatial
relation, and (iii) spatio-temporal correlation. Table 1 presents a
data correlation protocols: (i) spatial correlation; (ii) temporal cor-
rification, the energy-aware spatio-temporal correlation mechanism
sents a time series. Due to the nature of the physical phenomenon,
terns not only introduce delays in data transmissions but also lead
t the reception of outdated information by the sink node
[21,22,20].

In our solution, we save energy not by delaying or suppressing
messages, but by combining correlated information to make a bet-
ter use of the data communication. In our proposed algorithm,
named EAST (Efficient Data Collection Aware of Spatio-Temporal
Correlation), sensor nodes are clustered under a spatial correlation
proposed algorithm, the Residual Energy Aware Spatio-Temporal Correla-
tion (REAST) algorithm, creates clusters of nodes with similar
sensings and only a node inside the cluster notifies its
observation of a sensor node and gathered data is usually similar
over a short-time period. Thus, in these cases, sensor nodes do
not need to transmit their readings if the current reading is within
an acceptable error threshold regarding the last reported reading.
The sink node can just assume that any unreported data is un-
changed from the previously received ones. The degree of correla-
tion between consecutive sensor measurements might vary accord-
ing to the characteristics of the phenomenon.

Akyildiz et al. [15] studied the relation between reliability of
event detection and spatial location of the sensor nodes in the event
area. Their solution estimates the number of sensor nodes (representa-
tive nodes) required to send the detected event to the sink in order
to have reliable event information. Each representative node repres-
sents a spatially correlated group of nodes. Although their solution
achieves overall energy gain, it fails to consider the remaining energy
during the selection of the representative nodes – an assumption
that should not be neglected in a WSN because of hardware con-
straints. Thus, if a representative node works in the correlation re-
gion for a long period of time, it will spend more energy due to the
number of transmitted messages compared to the other nodes.

Yoon and Shahabi [18] proposed a new mechanism for spatial
correlation in WSNs. The proposed mechanism, called Clustered
Aggregation Technique (CAG), creates clusters of nodes with similar
sensing values and only a node inside the cluster notifies its
reading to the Sink node whereas the other nodes ignore their
readings. The CAG algorithm is divided into two phases: query and
response. In the query phase, the data-centric clusters are cre-
ated according to a user-specified error threshold $\tau$. Nodes that
have sensed values smaller than this threshold belong to the same
cluster. In the second phase (response phase), just one node per cluster (cluster-head) sends its sensed value to the sink node noti-
ifying the detected event. The authors showed that the proposed
mechanism can reduce significantly the number of transmitted
messages during the data collection. However, during the first
phase, the CAG algorithm uses a flooding-based protocol to dis-
seminate the query to all sensor nodes, which is not needed in
most scenarios. Moreover, the maintenance of the data-centric
clusters remains a difficult problem [23].

Liu et al. [24] proposed another clustering algorithm, named
Energy-Efficient Data Collection framework (EEDC), to exploit
spatial data correlation. They consider that nodes collect data

2. Related work

In the current literature, we can find three main categories of
data correlation protocols: (i) spatial correlation; (ii) temporal cor-
relation, and (iii) spatio-temporal correlation. Table 1 presents a
summary of the basic characteristics of the main proposed spatial
and/or temporal data correlations algorithms for WSNs. In the fol-
owing, we present some of these studies as well as the benefits of
exploiting spatial/temporal data correlation in WSNs.

2.1. Spatial correlation

The spatial correlation of sensory information among the nodes
that detect an event exists when those nodes are geographically
close, i.e., they have similar information. In this case, instead of
having all sensor nodes reporting the same data, it is more efficient
to choose a few representative nodes to notify the sink node about
the detected event. A representative node reports the event infor-
mation of a given area on behalf of a group of nodes that collects
similar information in the same area.

Sensor readings about the environment are typically periodic;
consequently, the time-ordered sequence of sensed data consti-
tutes a time series. Due to the nature of the physical phenomenon,
tere is a significant temporal correlation among each consecutive
observation of a sensor node and gathered data is usually similar
over a short-time period. Thus, in these cases, sensor nodes do
not need to transmit their readings if the current reading is within
an acceptable error threshold regarding the last reported reading.
The sink node can just assume that any unreported data is un-
changed from the previously received ones. The degree of correla-
tion between consecutive sensor measurements might vary accord-
ing to the characteristics of the phenomenon.

Akyildiz et al. [15] studied the relation between reliability of
event detection and spatial location of the sensor nodes in the event
area. Their solution estimates the number of sensor nodes (representa-
tive nodes) required to send the detected event to the sink in order
to have reliable event information. Each representative node repres-
sents a spatially correlated group of nodes. Although their solution
achieves overall energy gain, it fails to consider the remaining energy
during the selection of the representative nodes – an assumption
that should not be neglected in a WSN because of hardware con-
straints. Thus, if a representative node works in the correlation re-
gion for a long period of time, it will spend more energy due to the
number of transmitted messages compared to the other nodes.

Yoon and Shahabi [18] proposed a new mechanism for spatial
correlation in WSNs. The proposed mechanism, called Clustered
Aggregation Technique (CAG), creates clusters of nodes with similar
sensing values and only a node inside the cluster notifies its
reading to the Sink node whereas the other nodes ignore their
readings. The CAG algorithm is divided into two phases: query and
response. In the query phase, the data-centric clusters are cre-
ated according to a user-specified error threshold $\tau$. Nodes that
have sensed values smaller than this threshold belong to the same
cluster. In the second phase (response phase), just one node per cluster (cluster-head) sends its sensed value to the sink node noti-
ifying the detected event. The authors showed that the proposed
mechanism can reduce significantly the number of transmitted
messages during the data collection. However, during the first
phase, the CAG algorithm uses a flooding-based protocol to dis-
seminate the query to all sensor nodes, which is not needed in
most scenarios. Moreover, the maintenance of the data-centric
clusters remains a difficult problem [23].

Liu et al. [24] proposed another clustering algorithm, named
Energy-Efficient Data Collection framework (EEDC), to exploit
spatial data correlation. They consider that nodes collect data

Table 1
Summary of the basic characteristics of the main proposed spatial and/or temporal data correlations algorithms for WSNs.

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Route structure</th>
<th>Objective</th>
<th>Spatial correlation</th>
<th>Temporal correlation</th>
<th>Overhead</th>
<th>Scalability</th>
<th>Drawback</th>
</tr>
</thead>
<tbody>
<tr>
<td>EEDC</td>
<td>Single hop</td>
<td>Eliminate control overhead</td>
<td>Yes</td>
<td>No</td>
<td>Very Low</td>
<td>Very Low</td>
<td>Centralized and single-hop network</td>
</tr>
<tr>
<td>CAG</td>
<td>Tree-based cluster</td>
<td>Eliminate data redundancy</td>
<td>Yes</td>
<td>No</td>
<td>Very High</td>
<td>Medium</td>
<td>Maintenance data-centric</td>
</tr>
<tr>
<td>GSC</td>
<td>Tree-based cluster</td>
<td>Eliminate data redundancy</td>
<td>Yes</td>
<td>No</td>
<td>High</td>
<td>Low</td>
<td>Is not applied to multi-hop members</td>
</tr>
<tr>
<td>SBR</td>
<td>Tree-based cluster</td>
<td>Eliminate data redundancy</td>
<td>No</td>
<td>Yes</td>
<td>Medium</td>
<td>High</td>
<td>Sink node can receive outdated information</td>
</tr>
<tr>
<td>SCCS</td>
<td>Tree-based cluster</td>
<td>Eliminate data redundancy</td>
<td>Yes</td>
<td>Yes</td>
<td>Medium</td>
<td>High</td>
<td>Sink node can receive outdated information</td>
</tr>
</tbody>
</table>

Please cite this article in press as: L.A. Villas et al., An energy-aware spatio-temporal correlation mechanism to perform efficient data collection in wireless sensor networks, Comput. Commun. (2012), http://dx.doi.org/10.1016/j.comcom.2012.04.007
continuously and are one-hop connected to the sink node or to a center node. The algorithm was designed to be executed at the sink node, since this node has the entire data network information. The algorithm creates clusters of nodes that are spatially correlated. Also, the sink node manages the cluster formation dynamically in order to reflect environmental changes. The primary limitation of that scheme is the assumption of the single-hop communication. This assumption is impractical in a distributed system and difficult to have in large-scale wireless sensor networks. Another disadvantage is the clustering algorithm that is centralized at the sink node. Because of this, all network data needs to be sent to the sink node, which will store and process a great amount of data.

Shah et al. [31] proposed a new mechanism for spatial correlation in WSNs, named Gridiron Spatial Correlation (GSC). The GSC is adaptive to achieve the required reliability by dynamically changing the correlation region. The correlation regions are formed as squared rectangles and nodes lying in the rectangle are assumed to be spatially correlated. Cluster-head identifies the redundant and close sources in its vicinity and turns off the activity of nodes by considering their energy level and closeness as criterion. The limitation of GSC is the control mechanism which is not applied to multi-hop members, i.e., it only works well for scenarios where the event radius is smaller than the communication radius of the cluster-head.

2.2. Temporal correlation

Data collected at different time intervals from a specific sensor may be correlated if the set of collected data varies in a similar way. This is also known as temporal correlation. Due to the nature of the physical phenomenon, there is a significant temporal correlation among each consecutive observation of a sensor node. For example, in a daily sampling of temperature performed at each minute, the temperature may not change significantly. In this case, it is not necessary to report the new sampling at each minute since the last reported sampling corresponds to the actual one.

Vuran et al. [25] proposed a new framework to create data centric protocols that explore the nature of the physical phenomenon observed by a WSN. The main goal of the framework is to incorporate temporal correlation among consecutive observations of the phenomenon to reduce communication costs. The authors also explore spatial correlation by showing that nearby nodes tend to have the same observed data. The proposed framework can be used in two ways: (i) to develop efficient protocols, and (ii) to develop reliable sensed information reporting in WSN.

Deligiannakis and Kotidis [21] proposed a framework based on temporal correlation that uses a Self-Based Regression (SBR) algorithm [26] to decrease the number of transmitted messages required to monitor a physical phenomenon. The goal of the SBR algorithm is to process the observed data before sending it to the sink node. The framework stores the sensed information in a buffer and, when it is full, the SBR algorithm processes the data to find representative information. The authors claim that by sending just the representative information, the sink node can reconstruct the observed event without losing accuracy. However, the main drawback of such an approach is the waiting time until the buffer fills up. In this case, the sink node can receive outdated information about the sensed event.

2.3. Spatio-temporal correlation

The spatio-temporal correlation happens when the nature of the collected data has both spatial and temporal correlations, i.e., nodes close geographically have the same reading that is similar to the previous one. In this case, solutions that use both correlations can take advantage of the nature of the detected event to decrease the number of reported data.

Pham et al. [22,20] proposed a spatio-temporal solution, called Spatiotemporal Clustering and Compressing Schemes (SCCS), which uses a buffer to store the monitored data. When the buffer is full, SCCS executes a Divide and Conquer Algorithm (DCA) to find the representative information inside the buffer exploring the temporal correlation. The goal of the DCA is to find the minimum data set to be transmitted to the sink node. Considering all readings inside the buffer, the DCA creates a dividing line between the first element and the last one. For each buffer data, the algorithm calculates the distance between this value and the created line. If the value is smaller than a predefined threshold, the solution indicates that it has already been considered so that it is not necessary to include it again in the packet to be sent to the sink. When the value is greater than the threshold, the algorithm splits the line into two (one line up to this value and another one up to the endpoint of the previous line). When a line is split into two lines, all buffer values are verified again. These steps are repeated until a created line is not split into two anymore. Also, the SCCS solution creates a cluster among nodes that sensed the event in order to perform spatial correlation to reduce the number of transmitted messages. As mentioned before, by using a buffer to perform the temporal correlation, the waiting time to deliver the gathered data can be inappropriate for a number of real-time sensor network applications.

Xu et al. [27] proposed a wavelet-based spatio-temporal data compression algorithm for WSNs. Their algorithm employs a ring topology that explores simultaneously the temporal and spatial correlations among the sensed data. The algorithm considers that the sensor network is divided into clusters and each cluster is controlled by a cluster head. The algorithm also considers a virtual grid where the nodes inside each cell have spatial and temporal correlation. The nodes execute a wavelet transform algorithm to compress the sensed data on the ring in such a way it can be energy efficiently transmitted to its cluster head and then, delivered to the sink node. However, the authors did not investigate the processing task to execute the proposed wavelet transform in sensor nodes with limited capabilities. Also, the created virtual grid is not based on the event characteristics, which can result in inaccurate information.

Most of the current work on spatial and/or temporal correlation algorithms does not consider the energy dissipation and the event characteristics during data collection to better choose the representative nodes. Also, these solutions usually result in high delays and outdated data arriving at the sink node. The proposed algorithm, presented in the next section, exploits both spatial and temporal correlations to perform near real-time data collection in WSNs. In our algorithm nodes that detected the same event are dynamically grouped in correlated regions and a representative node is selected at each correlation region for observing the phenomenon. The entire region of sensors per event is effectively a set of representative nodes performing the task of data collection and spatio-temporal correlation. Our algorithm is introduced in the next Section.

3. A new efficient algorithm for space–time correlation

In this section, we define our spatiotemporal correlation models and propose the EAST algorithm. One of the key aspects of EAST is that the size of the correlation region and error threshold of readings can be changed dynamically according to the event characteristics in order to achieve the application’s accuracy requirements. This results in a better use of the available energy in the nodes that are sensing the event by eliminating redundant notifications as well as by using dynamic routes and low communication overhead.
These and other characteristics of our EAST algorithm are discussed in this section.

3.1. Spatial correlation model

Spatially close nodes tend to detect similar values. However, this closeness (c), i.e., the Euclidean distance between the nodes that detect similar values, depends on both application requirements and event characteristics. Some applications are more critical and less tolerant to discrepancies in the sensed values of the observed phenomenon, requiring that closer nodes notify the sensed data (the correlation region is smaller). Other applications can be more tolerant to discrepancies in the sensed values, not demanding that closer nodes report the sensed data (the correlation region is greater).

Definition 3.1 (Correlation region). We define a correlation region as an area where the values sensed by the sensor nodes are considered similar (for the application). Therefore, a single reading within this region is sufficient to represent it. The size of the correlation region (c) varies according to both application and event characteristics. For events whose characteristics change significantly over a short range, the sink node can decrease the size of the correlated region to keep high accuracy in the collected data, i.e., the event needs to be notified by closer nodes. For events whose characteristics do not change significantly at short range, the sink node can increase the size of the correlated region to save energy of member nodes. The size of the correlation region can be resized by the sink node, which sends the new size of the correlation region to the event’s coordinator by using the shortest path. The event’s coordinator then disseminates the new size of the correlation region to all the nodes within the event’s area, so the size of the correlation region is reconfigured.

3.2. Temporal correlation model

Sensor readings about the environment are typically periodic; consequently, the time-ordered sequence of sensed data constitutes a time series. Due to the nature of the physical phenomenon, there is a significant temporal correlation among each consecutive observation of a sensor node, and gathered data are usually similar over a short-time period. Thus, in these cases, sensor nodes do not need to transmit their readings if the current reading is within an acceptable error threshold regarding the last reported reading. The sink node can just assume that any unreported data is unchanged from the previously received ones. The degree of correlation between consecutive sensor measurements might vary according to the characteristics of the phenomenon.

Definition 3.2 (Temporal suppression). Each source node keeps the last reported reading. When current reading ($R_{\text{new}}$) is available, $R_{\text{new}}$ is compared to the last reported reading ($R_{\text{old}}$). If the current reading is within an acceptable error threshold regarding the last reported reading, the sink node can just assume that any unreported data is unchanged from the previously received ones. The degree of correlation between consecutive sensor measurements might vary according to the characteristics of the phenomenon.

3.3. Overview of the EAST Algorithm

The main idea of our proposed EAST algorithm is to manage the energy consumption of nodes that detected an event by eliminating redundant notifications. Our algorithm considers the following roles to perform data routing (see Fig. 1):

- **Member node**: A node that is currently detecting one or more events. In the case where its sensed data is redundant, it will not report the gathered data.
- **Representative node**: A node that detects an event and reports the gathered data to a coordinator representing not only itself but all nearby nodes with similar readings while still applying temporal suppression.
- **Coordinator node**: A node that detects the event and is responsible for gathering all event data sent by representative nodes. It processes the received data and sends the result towards the sink node.
- **Relay node**: A node that forwards data towards the sink node.
- **Sink node**: The gateway between the WSN and the monitoring facility.

The EAST algorithm uses shortest routes (in Euclidean distance) in two different levels for forwarding the gathered data towards the sink node. In the first level, representative nodes use shortest routes to forward data toward the coordinator node. In the second level, the coordinator nodes use shortest routes to forward data toward the sink node. Fig. 1 shows two examples of the routing structure obtained by the EAST algorithm (the gray field indicates the event area, the cells represent the regions of correlation and the red dotted line shows the shortest route).

The main objective of the EAST algorithm is to reduce energy consumption in data gathering while preserving both data accuracy and real-time reporting. To achieve this goal, EAST dynamically changes the size of the correlation region and the value of the coherency tolerance according to the event characteristics. For this, an event area is divided into cells, as depicted in Fig. 2. Each cell defines a correlation region and nodes within each cell are assumed to be spatially correlated. Only one node within a cell notifies the sensed information, if and only if, the given relative error threshold is greater than the temporal coherency tolerance. This last node is the representative node of the cell. Cells are independent from each other, so the change of representative nodes in one cell does not require any reconfiguration. The change of a representative node in each cell is performed to balance the energy consumption of spatially correlated nodes, while temporal suppression is applied to reduce the reporting of redundant data. Since correlation regions are independent, their resizing does not require any additional communication among the nodes within the event areas in order to compute the new cell they belong to. Furthermore, each node performs temporal suppression locally without communicating with its neighbors. The proposed spatio-temporal correlation approach is adaptive and scalable regarding events of different intensities as will be shown during its evaluation.

The EAST algorithm is performed in three phases. In Phase 1, presented in Section 3.4, sensor nodes store the sink position as well as their neighbors positions. In Phase 2, presented in Section 3.5, three different actions are performed: cluster formation; coordinators election; and the division of the event area into cells. Finally, in Phase 3, presented in Section 3.6, representative nodes are chosen, the proposed temporal correlation mechanism is applied, and data is then transmitted.

3.4. Node localization

After the deployment of the sensor nodes, the sink node starts by flooding a configuration message that contains four fields: $\text{ID}$, $\text{CoordSender}$, $\text{CoordSink}$, and $\text{Phenomenon_of_Interest}$, where $\text{ID}$ is the node identifier that retransmitted the message, $\text{Phenomenon_of_Interest}$ is the application’s interest (e.g., temperature higher than 25 degrees), $\text{CoordSender}$ is the node’s position ($x_i, y_i$) that relays the configuration message, and $\text{CoordSink}$ is the sink’s position ($x_s, y_s$). In this phase, sensor nodes store the
received information in a table of neighbors neighborhood that will be used in the next two phases.

Algorithm 1. EAST algorithm

> **Variables:**
1. \( tct = \{\text{Temporal Coherent Tolerance.}\} \)
2. \( Rold = \emptyset \{\text{Last Reported Reading.}\} \)
3. \( \text{Start Announcement Interest Message} \)

> **Input:**
4. Announcement Interest Message

> **Action:**
5. Stores the Neighbor’s and Sink’s Positions
6. Stores the Phenomenon of Interest
7. \([\text{Re}]\text{Start Announcement Interest Message}\)

> **Input:**
8. Event Detected

> **Action:**
9. node.Role \( \rightarrow \) Coordinator
10. Send Event Announcement Message
11. Start cellComputation

> **Input:**
12. \( \text{msg}_i = \text{response}(\text{Event Announcement Message}) \)

> **Action:**
13. \( \text{if } \text{EnergyLevel}(\text{node}) < \text{EnergyLevel}(\text{msg}_i) \)
14. node.Role \( \rightarrow \) Member;
15. \( \text{Retransmits (Event Announcement Message)} \)
16. end if

> **Input:**
17. cellComputation timeout

> **Action:**
18. \( x_c = 0 \)
19. \( y_c = 0 \)
20. \( \text{if } \frac{(x_c-x_n)}{c} > 1 \text{ then} \)
21. \( x_c = \frac{(x_c-x_n)-(\frac{c}{2})}{c} + 1 \)
22. \( y_c = \frac{(y_c-y_n)-(\frac{c}{2})}{c} + 1 \)
23. end if
24. \( \text{if } \frac{(x_c-x_n)}{c} < -1 \text{ then} \)
25. \( x_c = \frac{(x_c-x_n)+(\frac{c}{2})}{c} - 1 \)
26. \( y_c = \frac{(y_c-y_n)+(\frac{c}{2})}{c} - 1 \)
27. end if

> **Input:**
28. Data Transmissions

> **Action:**
29. \( R_{new} \leftarrow \text{sensed value} \)
30. \( \text{if node.Role = Representative and } (\frac{(R_{new} - R_{old})}{R_{old}}) \times 100 > tct \text{ then} \)
31. Send \( R_{new} \) to Coordinator
32. \( R_{new} \leftarrow R_{new} \)
33. end if
34. \( \text{if node.Role = Member} \)
35. Forwards \( R_{new} \) to Coordinator
36. end if
37. \( \text{if node.Role = Coordinator} \)
38. Processing received \( R_{new} \)
39. Forwards the \( \text{result to Sink} \)
40. \( \text{if node.Role = Relay} \)
41. Forwards \( R_{new} \) to Sink
42. end if
43. end if

3.5. Cluster formation, leader election, and division of the event area into cells

The second phase of the EAST algorithm starts whenever an event happens. Thus, when an event is detected by one or more nodes, the leader election algorithm starts with the sensing nodes running for leadership (group coordinator) – this process is described in Algorithm 1. For this election, all detecting nodes are eligible (Lines 8 and 9 of Algorithm 1) and the group leader (Coordinator node) will be the node with the higher residual energy. (Lines 13 and 14 of Algorithm 1). At the end of the election algorithm only one leader node exists in the group. In the case of a tie, the ID parameter is used as a tie breaker. The remaining nodes that detect the same event become member nodes. At each notification, a subset of the member nodes will be representative nodes, as explained later in this section. The coordinator gathers the information collected by the representative nodes, processes the information, and sends it toward the sink node.

After the clustering process, the proposed spatial correlation mechanism is executed. Fig. 2(a) illustrates the proposed spatial correlation mechanism. For the sake of simplification, the shape of the considered event is a circle, but any shape can work for the proposed solution. The event region is decomposed into \( (\frac{\pi}{2})^2 \) cells, where \( r_c \) is the events maximum radius and \( c \) is the cells size (correlation region). Fig. 2(a) shows an example in which the event region is decomposed into 25 cells. Each cell is represented by an ordered pair \((x_c, y_c)\). If \( c = 0 \), then there is no spatial correlation between nodes and all nodes in the group are representative nodes. Otherwise, each node computes the coordinates \( x_c \) and \( y_c \) of the cell to which it belongs to. For this computation, the node position \((x_0, y_0)\), the central position \((x_c, y_c)\) of the event, and the cell size \( c \) are required. Lines (17–27 of Algorithm 1) show this computation.

If the sink needs to dynamically resize the correlation region or the coherency tolerance value to meet any application requirement, it sends a new value of \( c \) and \( tct \) to the nodes of the group so that they can recalculate their cells size. Thus, the parameters of our algorithm can be dynamically controlled by the sink node, which has a complete view of the phenomenon.

It is important to point out that the maximum size of a cell \( c \) can be the length of the triangle’s leg in a right triangle, since \( r_c \) is the hypotenuse \((c = r_c \cos 45^\circ)\) where \( r_c \) is the communication radius of sensor nodes. This consideration is important to ensure that all nodes in the same cell communicate with each other. The cell size can vary to control the tradeoff between precision of the sensed data and energy consumption. In this case, the correlation region may vary between 0 \( \leq c \leq r_c \cos 45^\circ \). When \( c = 0 \), all nodes report the sensed data (an optimal solution in terms of accuracy in the information). For \( c > 0 \), only the representative node at each cell reports the sensed data.

The EAST algorithm selects a single representative node at each cell of dimensions \( c^2 \) for each notification. Figs. 2 show representative nodes at different times in the event region. The representative nodes in the set of member nodes are the nodes that have higher energy residual among nodes belonging to the same cell. This ensures the energy consumption distribution in the network.

3.6. Data transmissions

After computing the cell that a node belongs to, the node checks whether it is a representative node and also if the relative error threshold is greater than the temporal coherency tolerance (Line 30). If both conditions are satisfied, then the sensed value \( R_{new} \) is sent towards the group coordinator, which in turn processes and sends the collected information towards the sink using the shortest path. Based on its position and the sink position, the Coordi-
inator creates a straight line segment that connects itself to the sink. When data transmission is performed, the closest nodes to both its straight line segment and the endpoint of this straight line segment will be chosen to forward the data. Fig. 1 shows the straight line between the coordinator and the sink node as well as the relay nodes. The evaluation of our algorithm is presented in the next section.

4. Performance evaluation

In this section, we evaluate the performance of the spatio-temporal correlation mechanism of our proposed EAST algorithm. We also compare its performance with two other known routing protocols:

- Spatiotemporal Clustering and Compressing Schemes – SCCS (briefly described in Section 2.3).
- Accurate data collection strategy, which is the optimal solution in terms of accuracy. In this solution, all nodes send their sensed information to the sink node.

4.1. Methodology

The evaluation is performed through simulations by using the SinaGo version v.0.75.3 [28] simulator. In all results, curves represent average values, while error bars represent confidence intervals for 95% of confidence from 33 different instances (seeds). The simulation parameters are presented in Table 2. The event occurs in random positions. We consider the area of the sensor field as the relation \( \sqrt{\pi r^2 / d} \), where \( n \) is the number of nodes, \( r_c \) is the communication radius, and \( d \) is the average degree of neighbors. Sensor nodes are randomly deployed.

4.2. Event model

For our event model, we used a set of one-week environmental temperature data (degree Celsius) from the Amazon rainforest in Brazil [29] collected at intervals of 1 min. The samples are shown in Fig. 3.

In our application, the temperature in a region of interest is monitored. All nodes in this region will then send the data according to our proposed algorithm. To use the real-world data in our simulations, we consider the temperature at coordinate \((x, y)\) in the event area given by Eq. 1, where \( T_e \) is the temperature at the event center, which was obtained from the real-world data set presented in Fig. 3, \( D_e \) is the Euclidean distance (meters) to the event center.

![Fig. 1](image1.png)

(a) Routing structure at instant \( i \). (b) Routing structure at instant \( i + 1 \). Fig. 1. Examples of routing structure used by the EAST algorithm.

![Fig. 2](image2.png)

(a) Control of Correlation Region (b) Instant of time 1 (c) Instant of time 2 (d) Instant of time 3 Fig. 2. Spatial correlation mechanism applied to the event area.

<table>
<thead>
<tr>
<th>Table 2</th>
<th>Simulation parameters.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameters</td>
<td>Values</td>
</tr>
<tr>
<td>Sink node</td>
<td>1 (top left)</td>
</tr>
<tr>
<td># of nodes</td>
<td>1024</td>
</tr>
<tr>
<td># of events</td>
<td>1</td>
</tr>
<tr>
<td>Density (avg. neigh. number)</td>
<td>(20, 25, 30)</td>
</tr>
<tr>
<td>Event diameter (m)</td>
<td>(50, 100, 150, 200)</td>
</tr>
<tr>
<td>Correlation region (( \phi ))</td>
<td>(0, 10, 20, 30, 40, 50)</td>
</tr>
<tr>
<td>Event duration (h)</td>
<td>(1 to 10)</td>
</tr>
<tr>
<td>Notification rate (per minute)</td>
<td>1</td>
</tr>
<tr>
<td>Communication radius (m)</td>
<td>80</td>
</tr>
<tr>
<td>Simulation duration (days)</td>
<td>7</td>
</tr>
</tbody>
</table>

Please cite this article in press as: L.A. Villas et al., An energy-aware spatio-temporal correlation mechanism to perform efficient data collection in wireless sensor networks, Comput. Commun. (2012), http://dx.doi.org/10.1016/j.comcom.2012.04.007
spatial correlation, are the same as the results obtained with algorithms, as well as the other solutions that do not exploitative nodes and, therefore, a lower reporting rate. The region expected, if the correlation region remains fixed, the number of rep-
correlation region, density, and event diameter were varied. As ex-
presented in Table 2) were all varied to evaluate their impact on the
diameter, the density, and the size of the correlation region (pre-
4.3.1. Number of representative nodes

In this section, the proposed spatial correlation mechanism is
evaluated and compared to the accurate data collection strategy,
which is the optimal solution in terms of accuracy. In the accurate
data collection strategy, every sensor is requested to report its reading to the sink node at each round of data gathering. The main purpose of this subsection is to compare the performance of our proposed spatial correlation mechanism to the accurate data collection strategy considering the following metrics:

- **Number of representative nodes**: The number of nodes that report data about the phenomenon.
- **Energy consumption in data collection**: The amount of energy consumed by sensors that detected the event. This metric indicates how much of a sensor node energy is possible to save when the spatial correlation technique is exploited.
- **Data accuracy**: The accuracy of the data on the observed phenomenon regarding the original information.

When \( c = 0 \) the spatial correlation is not exploited, i.e., all nodes report the sensed data, which is the optimal solution in terms of data accuracy. The results for the DST \([5]\), DAARP \([9]\), and InFra \([30]\) algorithms, as well as the other solutions that do not exploit spatial correlation, are the same as the results obtained with \( c = 0 \).

4.3.2. Energy consumption

As depicted in Fig. 4, when the size of the correlation region increases, the number of representative nodes decreases. Consequently, the energy consumption also decreases, since fewer sensor nodes report data. In this scenario, it is possible to save up to 75% of the residual energy of the nodes within the observed phenomena area when compared to the classical approach for data collection (accurate data collection strategy) while maintaining an information accuracy greater than 97%, as shown below.

![Fig. 3. Data collected from Amazon rainforest.](image)

<table>
<thead>
<tr>
<th>Samples (day)</th>
<th>Current temperature (°C)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>32</td>
</tr>
<tr>
<td>2</td>
<td>30</td>
</tr>
<tr>
<td>3</td>
<td>28</td>
</tr>
<tr>
<td>4</td>
<td>26</td>
</tr>
<tr>
<td>5</td>
<td>24</td>
</tr>
<tr>
<td>6</td>
<td>26</td>
</tr>
<tr>
<td>7</td>
<td>32</td>
</tr>
</tbody>
</table>

4.3.3. Data accuracy

When the spatial correlation is exploited, the level of accuracy in information about the observed phenomenon tends to reduce. In this simulation scenario, the event diameter, density and size of the correlation region were varied to evaluate the data accuracy. As mentioned before, when \( c = 0 \), the classical approach for data collection is performed (100% accurate data collection strategy).

As depicted in Fig. 4, when the size of the correlation region increases, the number of representative nodes decreases. Consequently, the energy consumption also decreases, since fewer sensor nodes report data. In this scenario, it is possible to save up to 75% of the residual energy of the nodes within the observed phenomena area when compared to the classical approach for data collection (accurate data collection strategy) while maintaining an information accuracy greater than 97%, as shown below.

![Fig. 4 presents the number of representative nodes when the correlation region, density, and event diameter were varied. As expected, if the correlation region remains fixed, the number of representative nodes increases when the event diameter increases. It is also easy to see that for larger values of \( c \), there are less representative nodes and, therefore, a lower reporting rate.](image)

\[
temperature = T_c - (D_e \times T_D)
\]

4.3.3. Data accuracy

As we can see in Fig. 6, when the node density increases, the data accuracy decreases slightly, which is an unexpected result in most algorithms. It happens because the number of nodes within each cell increases, but the number of representative nodes remains the same. The difference between the combined readings of all nodes and the combined readings of only representative nodes increases.

On the one hand, it can be noted that the worst accuracies are obtained by the maximum value readings. This happens because nodes that detect the maximum value are in the central cell (see Fig. 2). With a greater number of nodes within the central cell, there will be a greater number of nodes that notifies data with values closer to the maximum value. However, for the evaluated scenarios, the smallest observed accuracy was of 97% while the energy consumption was reduced by more than 75%, which indicates the advantages of using the proposed spatial correlation technique of the EAST algorithm.

On the other hand, the reading of the minimum value had the highest accuracy since there are more cells with nodes that detect this value (cells in the border of the event). In the case of this event is divided into \((2r_c/c)^2\) correlation regions. In particular, for the event diameter of 200 m, correlation regions of 50 m, and density 30, the number of representative nodes is reduced four times when compared to the accurate data collection strategy \((c = 0)\). Consequently, the amount of energy consumed by nodes within the observed phenomena area is also reduced four times (as shown below).

Note that for larger correlation regions the number of representative nodes is smaller, hence the energy consumption is lower (as shown in the next section); however, the accuracy will be smaller (see Section 4.3.3). In our solution, applications can define the correlation region size by setting the value of \( c \) according to the required accuracy.
Fig. 4. Number of representative nodes.

Fig. 5. Energy consumption of the nodes that are reporting data.

Fig. 6. Accuracy in the readings.

Please cite this article in press as: L.A. Villas et al., An energy-aware spatio-temporal correlation mechanism to perform efficient data collection in wireless sensor networks, Comput. Commun. (2012), http://dx.doi.org/10.1016/j.comcom.2012.04.007
phenomenon, the smallest observed accuracy was greater than 99% while, again, the energy consumption was reduced by 75%.

It is important to note that there is a trade-off between the data accuracy and the energy consumption. For instance, if the application requires the measurement of the maximum value with an accuracy of at least 99.5%, then the value of c will have to be set to 10 (c = 10). In this case, the accuracy in the readings of the maximum value would be more than 99.5% and the reduction in energy consumption would be reduced to 33%.

4.3.4. Data accuracy for each round of data gathering

In this section, we present the analysis of the data accuracy for each notification, complementing the results presented above that analyzed the average accuracy of the measurements. In this simulation scenario, the size of the correlation region was varied to evaluate the data accuracy at each notification. The objective of this analysis is to show that it is possible to ensure the accuracy of insensitive duplication data (such as maximum and minimum) at different times. Consequently, if time and recent readings are taken into account, it is possible to estimate the exact (minimum and maximum) value.

Fig. 7 shows that the minimum accuracy for reading the minimum value at a given time is 92% (when the correlation region is 50 m). However, at least for every four reports, the exact minimum value is reported. Because of this, the exact minimum value can be estimated by representative nodes. Consequently, the accuracy of the readings for a minimum value can be increased very close to the exact value. Note that for a correlation region smaller than 30 m, the minimum accuracy is 98% and at least for every two notifications the exact minimum value is reported.

Similarly, Fig. 8 shows that the minimum accuracy for reading the maximum value at a given time is 93%, but at least for every four reports the exact maximum value is reported. Because of this, the exact maximum value can be estimated by representative nodes and, as a result, increase the accuracy of the reading to a maximum value very close to the exact value.

Different from minimum and maximum values, the exact mean value is not reported at each time interval, as depicted in Fig. 9. This occurs because mean values are sensitive to duplication data. If taken into account the time and recent readings, it is possible to estimate the mean value to increase the accuracy at each reading.

4.4. Performance evaluation of temporal correlation mechanism

In this section, we compared our proposed temporal correlation mechanism for the EAST algorithm to the SCCS algorithm as well as the accurate data collection strategy, which is the optimal solution in terms of accuracy. For our event model, we used a set of one-week environmental temperature data from the Amazon rainforest in Brazil [29] taken at intervals of 1 min. The samples are shown in Fig. 3. For the SCCS algorithm, as mentioned before, each node stores its monitored data in a buffer and, when the buffer is full, the node processes the data in its buffer to consider the temporal correlation among the monitored values and report the result to the sink node. For the accurate data collection strategy, every sensor is requested to report its readings to the sink node at each round of data gathering. The main purpose of this comparison is to evaluate the performance of our proposed algorithm considering the following metrics: (i) notifications, (ii) readings reported, (iii) readings per data packet, (iv) energy consumption, (v) data accuracy, and (vi) delay notification.

The simulation parameters used in this performance evaluation are the same of previous experimentations (shown in Table 2) with the new values presented in Table 3.

The following metrics were used for the evaluation:

- **Number of notifications**: Number of notifications sent by nodes that detect the event.
- **Number of readings reported**: Number of readings reported.
- **Readings per data packet**: Average number of readings within each packet (it is the ratio of Readings reported and notifications).
- **Energy consumption**: The amount of energy consumed by sensors that detected the event.
- **Data accuracy**: The accuracy of the information on the observed phenomenon regarding the original information.
- **Delay notification**: Time to deliver the gathered data.

In all evaluated cases, a number of variations of our proposed EAST algorithm was considered. First, we evaluated our EAST algorithm while exploring only the temporal correlation. We also evaluated our algorithm (EAST-15, EAST-30, and EAST-45) exploiting both temporal correlation and spatial correlation (with correlation regions of size 15, 30, and 45). For the SCCS algorithm, we considered buffers with different storage capacities of 25, 50, 100, and 200 readings.

4.4.1. Notifications and readings reported

For this analysis, the temporal coherency tolerance, buffer size, and correlation region (presented in Table 3) were all varied to evaluate their impact on the number of readings that can be eliminated by exploiting the spatio-temporal correlation.
Fig. 10(a) shows that when the temporal coherency tolerance increases, the number of notifications performed by our EAST algorithm decreases while the number of notifications in the SCCS remains fixed since the readings will only be transmitted when the buffer fills up. Consequently, in the SCCS, the number of notifications depends on the buffer size and not on the temporal coherency tolerance. Moreover, in our EAST algorithm, when the size of the correlation region increases (15, 30, and 45), the number of representative nodes decreases, which also decreases the number of notifications.

In Fig. 10(b), we can see that the number of readings reported by our EAST algorithm is similar to the ones presented in Fig. 10(a). This is because whenever the current reading is above the temporal coherency tolerance, the data is notified. Note that in most cases the EAST algorithm reports fewer readings than the SCCS algorithm, but for small values of the temporal coherency tolerance, the SCCS algorithm presents less notifications by exploring the use of a buffer, in which each notification may contain more than one reading. Since the SCCS algorithm creates a line segment between the first and last reading of the buffer, this technique has good results in terms of energy consumption when the buffer size increases (as depicted in Fig. 11). In this case, it is necessary a few line segments to represent all values inside the buffer.

Fig. 10(c) shows the average number of readings within each transmitted packet. As we can see, the EAST algorithm, in any situation, sends only one reading within each transmitted packet. But the number of readings per packet in the SCCS algorithm depends both on the capacity of the buffer and the temporal coherency tolerance. When the temporal coherency tolerance is small, the SCCS algorithm needs to split the original line segment into a high number of other line segments to represent the original values. Because of this, more readings will be necessary to represent the monitored values. It is important to point out that when we increase the buffer size, the SCCS algorithm will split the original line segment into more line segments to represent the original values, since more readings are being considered by the algorithm. When we increase

Table 3

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Density (avg. neigh. number)</td>
<td>25</td>
</tr>
<tr>
<td>Correlation region (c)</td>
<td>(15, 30, 45)</td>
</tr>
<tr>
<td>Temporal coherent tolerance</td>
<td>(0.5, 1, 2, 3, 4)</td>
</tr>
<tr>
<td>Buffer size (bytes)</td>
<td>(25, 50, 100, 200)</td>
</tr>
<tr>
<td>Sensor field (m)</td>
<td>900 x 900</td>
</tr>
</tbody>
</table>

Fig. 8. Accuracy in the readings of max value.

Fig. 9. Accuracy in the readings of mean value.

Please cite this article in press as: L.A. Villas et al., An energy-aware spatio-temporal correlation mechanism to perform efficient data collection in wireless sensor networks, Comput. Commun. (2012), http://dx.doi.org/10.1016/j.comcom.2012.04.007
the temporal coherency tolerance, it is not necessary to split the original line segment into a high number of line segments. For instance, when the temporal coherency tolerance is 4%, the SCCS algorithm transmits the same number of readings (compared to our proposal) to represent the sensed event.

4.4.2. Energy

Fig. 11 shows the average energy consumption of the nodes within the observed phenomena area. For this analysis, the temporal coherency tolerance, buffer size, and correlation region (presented in Table 3) were all varied to evaluate their impact on the energy consumption.

As depicted in Fig. 10(a), when the temporal coherency tolerance increases, the number of notifications performed by EAST decreases while the number of notifications in the SCCS remains fixed and depends on the buffer size, which has an impact on the data accuracy (see Fig. 12). Consequently, the energy consumption also decreases in the EAST algorithm (see Fig. 11). Moreover, when the size of the correlation region increases (15, 30, and 45), the number of notifications decreases. Consequently, the energy consumption also decreases, since a smaller number of sensor nodes report their readings.

The results for the accurate data collection strategy were not plotted on the graph because of its very high energy consumption, which is due to the fact that all readings are notified to ensure data accuracy of 100%. On average, the accurate data collection strategy consumes 10 J, i.e., 14 times more than the SCCS and EAST algorithms.

4.4.3. Data accuracy

In this simulation scenario, the temporal coherency tolerance, buffer size, and correlation region were all varied to evaluate the accuracy of the readings. The phenomenon observed was the temperature but, as mentioned before, the proposed mechanism works for any other type of phenomenon with different characteristics. We analyzed the accuracy at the sink node when computing the values for minimum, mean, and maximum temperatures.

As depicted in Fig. 12, when the temporal coherency tolerance increases, the data accuracy in the EAST algorithm slowly decreases. For the SCCS algorithm, when the buffer capacity increases, the data accuracy decreases faster. This happens because when we increase the buffer size, less readings will be necessary to represent the monitored event (see Fig. 10(b)). Thus, considering less values, the SCCS algorithm will not achieve good values of data accuracy.

For the evaluated scenarios, the smallest accuracy observed in our proposed algorithm was of 98.7% and the energy consumption was less than 0.2 J, while the accurate data collection strategy consumes 10 J, which indicates the advantages of using our spatial and temporal correlation techniques.
Because the EAST algorithm considers the last notified reading to exploit the temporal correlation, as soon as the relative error threshold of the actual reading is greater than the determined tolerance of temporal coherency, the algorithm notifies the actual reading to the sink node. As a result, every time the sensed value is beyond the error threshold the sink will be notified in real time what is expected in most WSNs applications.

5. Conclusions and future work

This work proposed EAST, an algorithm for energy-aware data forwarding in WSNs that takes full advantage of both spatial and temporal correlation mechanisms to save energy while still maintaining real-time, accurate data report towards the sink node.

In the current literature of spatial and/or temporal correlation algorithms, most of the proposed studies do not consider the energy dissipation during data collection to better choose the representative nodes. Also, these solutions present a high number of control messages and do not exploit efficiently the spatio-temporal correlation nor their dynamicity. In this work, we went further and proposed an energy-aware spatio-temporal correlation mechanism in which nodes that detected the same event are dynamically grouped in correlated regions and a representative node is selected at each correlation region for observing the phenomenon. The entire region of sensors per event is effectively a set of representative nodes performing the task of data collection and temporal correlation.

We exhaustively simulated our proposed algorithm considering several scenarios and parameters to allow a better understand of its behavior. Simulation results clearly show that by using both spatial and temporal correlation, the information about the event can be sensed with a high accuracy of more than 99.7% while still saving the residual energy of the nodes in more than 14 times when compared to the accurate data collection strategy.

These results are very promising, but some issues still need to be further exploited. As future work we intend to consider not only the last reading, but also the previous readings in the correlation region to improve the accuracy of sensed data about the observed phenomenon. To achieve this goal, representative nodes can estimate the values of their correlation region by taking into account the time and recent readings. In addition, we intend to consider correlation regions of different sizes for the same event to further explore the dynamicity of the event.

Acknowledgments

This work is partially supported by Canada Research Chair program, NSERC, NSERC DIVA and OIT/MRI Research funds. The authors also wish to thank FAPEMIG, CNPq (INCT NAMITEC Project, processes 573738/2008-4 and 575808/2008-0), FAPESP, CAPES (INCT-SEC Project, processes 573963/2008-8 and 08/57870-9).

References
