Sensor Stream Reduction For Clustered Wireless Sensor Networks

André L. L. de Aquino¹, Carlos M. S. Figueiredo², Eduardo F. Nakamura², Alejandro C. Frery³, Antonio A. F. Loureiro¹, Antônio Otávio Fernandes¹.

¹ Department of Computer Science – Federal University of Minas Gerais
Belo Horizonte, MG, Brazil
{alla, loureiro, otavio}@dcc.ufmg.br

² FUCAPI – Research and Technological Innovation Center
Manaus, AM, Brazil
{mauricio.figueiredo, eduardo.nakamura}@fucapi.com.br

³ Instituto de Computação – Universidade Federal de Alagoas
Maceió, AL, Brazil
acfrery@pesquisador.cnpq.br

ABSTRACT

This work presents the use of sensor stream reduction algorithms in clustered wireless sensor networks (WSNs), where the cluster head node is responsible to reduce the amount of data generated by the nodes in its cluster. Moreover, a formal description to sensor stream reduction problem is proposed and an analytical model is adapted to show that when our solutions are adopted in cluster based organization, they have a superior performance to that observed when they are used in a non-clustered organization.

1. INTRODUCTION

In wireless sensor networks (WSNs) [1], physical variables, such as temperature and luminosity, can be monitored so that the data is continuously generated along the network operation. Data with such features are commonly referred to as data streams [6], and can be delivered in different ways. Particularly, sensor nodes can either send data to the observer by using a multihop routing protocol, or through a cluster-based communication architecture. These delivery models characterize the WSNs as flat or hierarchical, respectively.

Research in data stream algorithms is focused either in improving the algorithms or in applying these algorithms in specific scenarios [6]. Other research considers the WSN as a distributed database where some properties (e.g. maximum, minimum, and average) are computed in-network [3]. Moreover, some proposals employ the database abstraction to extract management information from the WSN, such as energy and node location [5].

Specifically, the work [2], proposes and evaluates two algorithms based on data stream that use sampling and sketching to reduce data traffic in non-clustered WSNs and, therefore, to decrease the delay and energy consumption. The sampling scheme uses \( \log n \) items to represent the original \( n \) elements data, preserving information.

Non-clustered based organization is used in many applications. However, this organization may not be appropriate in specific scenarios, and its solutions can seldom be applied. For that reason, we enhanced the sensor stream reduction algorithms developed for non-clustered networks [2], in order to apply them in clustered networks. Clustered networks offer major advantages over their non-clustered counterparts in these situations, so the question we address is Are the sensor stream algorithms, developed for non-clustered networks, efficient in clustered networks?

Research on clustered network, generally, focuses on the algorithmic aspects of cluster creation or on the aggregation solutions for clustered WSNs [4]. Specifically, the work [10] provides an analytical model to show that clustered networks have better performance than non-clustered ones.

Besides evaluating the data stream reduction in a cluster based organization, our work proposes a formal description of the sensor stream reduction problem, and adapts the analytical model proposed by [10] to show that when sensor stream solutions in clustered networks have a superior performance with respect to a non-clustered organization.

This work is organized as follows. In Section 2, we make a formal description to sensor stream reduction problem in cluster based networks. Next, in Section 3, we present an analytical model to compare the clustered vs. non-clustered networks. Section 4, presents the sensor stream algorithms for clustered WSNs. Experimental results are given in Section 5, and Section 6 concludes this study and suggests future work.
2. PROBLEM DESCRIPTION

WSNs consist of spatially distributed autonomous devices using sensors to cooperatively monitor physical or environmental conditions, such as temperature, sound, vibration, pressure, motion or pollutants, at different locations [8]. Such a system can be represented by the diagram shown in Figure 1, where \( N \) denotes the environment and the process to be measured, \( P \) is the phenomena of interest, with \( V^* \) their space-temporal domain. If true, complete and uncorrupted observation was possible, one could devise a set of ideal rules \((R')\) leading to ideal decisions \((D^*)\). Because of its characteristics, we consider \(V^*\) a data stream.

\[
\begin{array}{ccc}
D^* & \xrightarrow{\psi} & R' \\
N & \xrightarrow{P} & V^* \xrightarrow{S} V \xrightarrow{R} D \\
\quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \text{Figure 1: Representation of a WSN system, where the ideal (N → V* → D*), sensed (N → V* → V → D) and reduced (N → V* → V' → D') behaviors are shown}
\end{array}
\]

Instead of the ideal situation, one has a set of \( s \) sensors \( S = (S_1, \ldots, S_s) \), each making a reading of the phenomena and producing a report in the domain \( V_i \), with \( 1 \leq i \leq s \); all possible domain sets are denoted \( V = (V_1, \ldots, V_c) \). Using all this information, one can conceive the set of rules \((R)\) leading to the set of decisions \((D)\). Therefore, differently from \(V^*\), \( V \) is a data stream.

But dealing with \( V \) may be too expensive in terms of, for instance, power, bandwidth and computer resources usage. Since the level of redundancy is not negligible in most situations, one is tempted to reduce the volume of this information. Sensor stream reduction techniques are denoted by \( \psi \), and they transform the complete domain \( V \) into the smaller one \( V' \). New rules that use \( V' \) are denoted by \( R' \), and they lead to the set of decisions \( D' \).

The problem addressed in this work can be stated as follows:

Problem statement: What is the impact over the decisions \( D \), when we use a \( \psi \) data stream technique over \( V \) generated by \( S \) with a clustered WSN? Moreover, is it possible to meet the clustered WSN requirements by reducing data traffic (using \( \psi \))? To address this problem, the scope of this work considers the following assumptions:

- **Sensor network topology**: In our simulations, the set of sensors \( S = (S_1, \ldots, S_j) \) is distributed in a squared area \( A = L \times L \), which is partitioned in a set of \( c \) clusters \( C = (C_1, \ldots, C_c) \), \( c \leq s \) and \( c \) is even. Each cluster \( C_j \) is responsible for an area \( A_{C_j} \). The set \( C_j \) is composed by one cluster head \( CH_j \) and all nodes \( SC_j = (S_1, C_1, \ldots, S_k, C_k) \) located in \( A_{C_j} \), where \( s_k \) is the number of sensor per cluster. There is only one sink node located at \((0, 0)\).
- **Data stream processing**: The \( \psi \) reduction, performed by \( CH_j \), is defined by two data stream techniques [2]: sampling and sketch.
- **Data stream decision**: The decision \( D' \) is based upon the analysis of data quality: \( V' \) must follow (approximately) the same distribution and the average error if compared with \( V \). The rules \( R \) and \( R' \) are:
  (i) Kolmogorov-Smirnov (KS) test [9];
  (ii) discrepancy of the values in the reduced stream [2]; and
  (iii) checking if \( V_{C_j} \) has the same arrival order if compared \( V_{C_j} \).
- **Stream generation**: The streams \( V_i \) are generated in each sensor \( S_i \), continuously at regular intervals (periods) of time following a Uniform distribution.

3. CLUSTERED VS. NON-CLUSTERED DATA STREAM REDUCTION

In order to answer the question: Are the sensor stream algorithms, applied in non-clustered networks, efficient in clustered networks?, we adapt the analytical model, proposed by Vlajic et al. [10], to show that clustered WSNs performance is usually better than the non-clustered one.

In the following, the main properties of this model are summarized and we show how to apply it in sensor stream scenarios. Initially, consider the following assumptions, that describe the operational cost of non-clustered WSNs:

- There are \( s = |S| \) sensor nodes in the network.
- The nodes are uniformly distributed on a grid over the area \( A = L \times L \), so each cell of size \( a \times a \) contains only one node, with the sink located at the center.
- Each node communicates with the eight nodes located at the neighboring cells.
- \( h_{avg} \) represents the average hop-distance between the nodes and the sink; it is defined in number of hops as \( h_{avg} = \sqrt{5}/3 \).
- Each node sends an \( n \) bits sample \( V_i \) to the sink.

Based on these assumptions, in the non-clustered WSN, the bit-hop \((BT)_{noncl}\), in bits, is defined as

\[
BT_{noncl} = s h_{avg} n
\]

Now, in order to derive the operational cost of non-clustered WSNs, consider the following assumptions:

- There are \( c = |C| \) clusters in the network.
- Just for this demonstration, the network is divided into fixed, equal, and non overlapping square zones, where each square zone contains one cluster head at the center. The sink is located in the center.
- As \( c \) represents the number of clusters, \( s/c \) is the number of nodes per cluster.
- The average inter-cluster hop distance in hops is \( h_{avg cl} = h_{avg} / \sqrt{c} \).
- The average factor of in-cluster data reduction per each reported reading is \( \alpha \), and the amount of data sent to the sink is \( \alpha ns/c \).
Based on this assumption, in the clustered WSN, the bit-hop ($BT_{cl}$), in bits, is

$$BT_{cl} = c \left( \frac{s}{c} - 1 \right) h_{avg} n + c h_{avg} \left( \frac{a n}{c} \right).$$

(2)

Simplifying and replacing $h_{avg}$ by $h_{avg}/\sqrt{c}$, equation (2) becomes

$$BT_{cl} = (s - c) h_{avg} \sqrt{c} n + c h_{avg} \left( \frac{a}{c} \right) n.$$

(3)

Based on equations (1) and (3), Vlajic et al. [10] show that

$$BT_{cl} < BT_{noncl} \iff s - c \sqrt{c} + \alpha s < s$$

(4)

We make an exact reduction in our sensor stream reduction, so we consider $\beta = \alpha n s / c$ in equation (3). With this,

$$BT_{cl} = (s - c) h_{avg} \sqrt{c} n + c \beta h_{avg}$$

where $\beta$ is the amount of data reduced in the cluster and sent to sink. Moreover, equation (4) is simplified to

$$BT_{cl} < BT_{noncl} \iff s - c \sqrt{c} + \beta c < s.$$

In order to have a more realistic estimate on how the sensor stream algorithms, applied in non-clustered networks, are efficient in clustered networks? Only identifying the size of the sample or sketch which improves the performance of the clustered network. The next step is showing how the sensor data stream is applicable in clustered networks.

4. SENSOR STREAM SOLUTIONS

In order to address the problem stated in Section 2 and to show how the sensor stream data is applicable in clustered networks, we present the solutions based on sampling and sketch techniques [2].

Sampling reduces the amount of data $V_{C_j}$ generated from the $SC_j$ nodes of the cluster $C_j$. The sample size $\beta$ can vary, but it must be representative to preserve relevant information. The $\psi$-sampling algorithm can be divided into the following steps:

- **Step 1:** Build a histogram of the $V_{C_j}$.
- **Step 2:** Create the sample $V_{C_j}^{\beta}$ based on the histogram obtained in Step 1. To create such a sample, we randomly choose the elements of each histogram class, respecting the sample size and the class frequencies of the histogram. Thus, the resulting sample will be represented by the same histogram.
- **Step 3:** Sort the $V_{C_j}^{\beta}$ according to its order in the $V_{C_j}$.

In $\psi$-sketch algorithm, the $V_{C_j}^{\beta}$ is provided by a sketch of the $V_{C_j}$, where the frequency of the data values is kept without losses. Moreover, a constant $\beta$ size is used. However, the sketch solution loses the sequence of the $V_{C_j}$. The $\psi$-sketch algorithm can be divided into the following steps:

- **Step 1:** Sort the data and identify the minimum and maximum values in $V_{C_j}$.
- **Step 2:** Build the data out, only with the histogram frequencies.
- **Step 3:** Mount the $V_{C_j}^{\beta}$ with the data out and the information about the histogram.

The complexity of both sampling and sketch algorithms is $O(n \log n)$, because of the need of the sorting step; the histogram can be built with complexity $O(n)$, since $\beta \leq n$. The space requirement is $O(n + \beta) = O(n)$ because the algorithm stores both $V_{C_j}$ and $V_{C_j}^{\beta}$. Since every source cluster head sends $V_{C_j}^{\beta}$ towards the sink, the communication complexity is $O(\beta h_{large})$, where $h_{large}$ is the largest route (in hops) in the network.

5. SIMULATION RESULTS

In this section, we evaluate the impact of the stream-based solutions on the network. In addition, in order to tackle the problem stated in Section 2, we discuss the impact of the solutions regarding the data quality, which is considered here as our project decision $D$. Initially, we present some general assumptions for the evaluation of the algorithms:

- **Simulation:** We perform our evaluation through simulation and use the NS-2 (Network Simulator 2) version 2.31, where each simulated scenario was executed with 33 random topologies. At the end, for each scenario we plot average values with 95% (symmetric asymptotic) confidence intervals.

- **Network topology:** We adapt and use a tree-based routing algorithm called EF-Tree [7], where one tree is used to route packets from source to cluster-heads and other is used from cluster-heads to sink. The density is kept constant and all nodes have the same hardware configuration. In order to evaluate only the data reduction performance, the trees are built just once before the traffic starts and the network is kept static. Source nodes are randomly distributed in the sensor field.

- **Stream generation - $V_i$:** The stream values $V_i$ follow a uniform distribution in $[0; 1]$. The generation periodicity is 100s, and the size is 20 bytes. The size of the data packet is also 20 bytes. When the data generated in the cluster-head are greater than the packet size, they are fragmented and reassembled at the sink upon reception.

- **Evaluated parameters and stream size:** We varied the number of nodes in each cluster and the number of clusters. In the sampling case, for each evaluated parameter we used $\beta \in \{ \log n, n/2 \}$. The parameters used in the simulation are presented in Table 1.

In order to assess several topologies, the number of nodes* in the network are $\{100, 200, 300, 400\}$, keeping 4 clusters and the $V_{C_j}$ size in 256. Moreover, to evaluate the sampling algorithm performance, we use $\beta \in \{ \log n, n/2, n \}$, whereas for the sketch algorithm $\beta = 10$, that represents the range of histogram used.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network size</td>
<td>Varied with density</td>
</tr>
<tr>
<td>Queue size</td>
<td>Varied with stream</td>
</tr>
<tr>
<td>Simulation time (s)</td>
<td>5000</td>
</tr>
<tr>
<td>Traffic rate (s)</td>
<td>[2000, 4000]</td>
</tr>
<tr>
<td>Stream periodicity</td>
<td>100</td>
</tr>
<tr>
<td>Sink location</td>
<td>0, 0</td>
</tr>
<tr>
<td>Radio range (m)</td>
<td>50</td>
</tr>
<tr>
<td>Bandwidth (kbps)</td>
<td>250</td>
</tr>
<tr>
<td>Initial energy (J)</td>
<td>100</td>
</tr>
</tbody>
</table>

*http://nsnam.isi.edu/nsnam/index.php/Main_Page

**The limited number of nodes (400) is due to NS limitations. There is not enough memory to support all nodes generating around 10000 packets for each data generation.
In Figure 2, we show energy consumption. We observe that whenever the sample size decreases in the sampling solution, the energy consumption decreases accordingly. The sketch solution produces similar results to sampling-log \( n \), because the packet size is constant and close to the sample, i.e., a \( \log n \) packet size. In agreement with the analysis in Section 3, this behavior occurs because the higher the \( \beta \) reduction the better the network performance.

![Figure 2: Different number of nodes in cluster](image)

In summary, when we analyze the rules in hierarchical networks, we have the following conclusions:

- The sketch solution reduces the energy consumption and the packet delay by keeping a constant data transmission rate. Since the data can be generated artificially by the sink from the sketch, the data quality test is not affected in \( R_{dist} \) and \( R_{value} \). In this solution, the \( D_{order} \) decision is false, but in fact this decision may not be important to a large range of sensor applications when the network restrictions are strong. In the cases where \( D_{order} \) is important, the sampling solution must be considered.
• The $\beta = \log n$ reduces the energy consumption and the packet delay by reducing the transmitted data. In the presented scenarios, the $D_{dist}$ and $D_{value}$ are affected, because $KS = 20\%$ and $data\text{-}error = 10\%$ respectively. However, these results can be acceptable by a large range of applications when the network restrictions are strong.

• The $\beta = n/2$ is interesting when the application priority is the $D_{value}$ (near zero).

• It is not interesting to use our algorithms ($\beta = n$) when preserving $D_{dist}$ is required and when clustered WSN restrictions are not considered.

• Finally, if $D_{order}$ is important the sampling algorithm should be used. In this case, one should analyze the application and network requirements to decide about the best sample size. However, if $D_{order}$ is not important the sketch solution could be used because it always provides the best network performance generating artificially all data. The advantage of the sketch over the sampling solution is that the former can be modified for on-line processing of the sensor stream without keeping the original data.

We have therefore presented the applicability of our solution to the problem addressed in Section 2, and these results answer the question: Are sensor stream algorithms, applied in non-clustered networks, efficient in clustered networks?

6. CONCLUSIONS AND FUTURE WORK

In this work, we evaluated two algorithms based on sensor stream that use sampling and sketch techniques to reduce data traffic and, consequently, decrease the delay and energy consumption in clustered WSNs. Moreover, a formal description to sensor stream reduction problem is presented, and an adapted analytical model is used to show that when our solutions can be successfully used in clustered networks; they have a superior performance compared to non-clustered networks. The results show the efficiency of the proposed methods by extending the network lifetime and by reducing the delay without losing data representativeness.

As future work, we will apply the proposed method to process sensor streams along the routing task. Thus, not only the data from a source can be reduced, but similar data from different sources can be subjected to a similar reduction, resulting in more energy efficiency.

7. REFERENCES


