On Efficient Data Reduction for Network Partition Forecasting in WSNs

FAISAL KARIM SHAIKH*, WAJIHA SHAH**, AND SYED ASIF ALI SHAH***

RECEIVED ON 12.02.2011 ACCEPTED ON 22.03.2011

ABSTRACT

WSNs (Wireless Sensor Networks) are generally deployed for long-lived missions. However, they rely on finite energy resources which lead to network partitioning. Network partitioning limits the dependability of WSN by making relevant spatial regions disconnected thus requiring the maintenance of the network. The network maintenance necessitates early warning and consequently forecasting of the network partitioning such that some early action can be taken to mitigate the problem. There exist approaches allowing for detection of network partitioning but none for its forecasting. We present an efficient approach for a proactive network ParFor (Partition Forecasting) based on energy maps. ParFor implements spatial and temporal suppression mechanisms such that from energy weak regions only a few nodes report short alarms to the sink. Using these alarms the forecasting is done centrally at the sink. Using simulations we highlight the efficiency and accuracy of ParFor.

Key Words: **WSNs, Partition Forecasting, Sensor Map, Energy Regions.**

1. INTRODUCTION

WSNs represent a rapidly growing research area in computing and communications domain. The sensor nodes are typically embedded systems with limited power, computational and communication capabilities. The sensor nodes are deployed in physical environment to monitor phenomenon of interest, process the obtained data and forward it towards a gateway node termed as sink. WSN maintenance is a critical operational requirement for continuous monitoring of the physical phenomenon of interest. An important parameter for WSN maintenance is connectivity of sensor nodes. Generally, sensor nodes are deployed in such a way that connectivity is maintained. Since, the sensor nodes rely on a finite energy it is possible that some nodes die out earlier than others, leading to connectivity holes. Accordingly, the network partitioning happens if a set of sensor nodes becomes isolated from the sink, i.e., cannot communicate with it because nodes surrounding them are running out of energy. The occurrence of network partitioning requires maintenance (deployment of supplemental nodes) and/or reconfiguration (waking up of existing nodes, movement or adaptation of transmission range) actions. During these actions the partitioned part of WSN will stay disconnected. Subsequently, the

* Assistant Professor, Department of Telecommunication Engineering, Mehran University of Engineering & Technology, Jamshoro.
** Assistant Professor, Department of Electronic & Biomedical Engineering, Mehran University of Engineering & Technology, Jamshoro.
*** Assistant Professor, Department of Electrical Engineering, Mehran University of Engineering & Technology, Jamshoro.

forecasting network partitioning is a fundamental task for a successful maintenance of WSN. Furthermore, an efficient maintenance procedure requires information about network partitioning such as which geographic area or nodes are affected.

The existing approaches consider only the detection of an existing partition in WSN [1-4]. These approaches are based on network connectivity and therefore they are less suitable for forecasting that requires also the energy level of the communicating nodes. To the best of our knowledge, we are the first to consider forecasting of network partitioning in WSN. A naive approach for WSN partition forecasting is to allow each sensor node to send its energy to the sink. Given the position of each node and the corresponding residual energy values the energy distribution in the network can be easily obtained at the sink. Consequently, the sink determines which part will be partitioned first [5-6]. However, this approach generates a huge number of messages which aggravates the scarcity of both power and bandwidth. Thus, more efficient approaches are required to be developed.

In this paper we have developed an efficient technique for reducing the data to be transferred for network ParFor in WSN. We develop a hybrid approach where regions intelligently communicate minimal information to the sink, which is sufficient to forecast network partitioning.

The remainder of the paper is organized as follows. Related work is discussed in Section 2. In Section 3 we detail the definition of network partitioning and discuss the relevant models. In Section 4 we present our Partition forecasting approach ParFor. Section 5 presents our evaluation results. Section 6 concludes the paper and presents our future work.

2. RELATED WORK

In WSN literature, we identify techniques for detecting network partitioning [1,3-4], others for its avoidance [2] and a few for its repair [7]. We review these techniques and show the need for forecasting techniques.

We identify three centralized approaches [1,3-4] and one decentralized approach [2] to detect WSN partitioning. In [4], a generalized approach is presented to compute digests (e.g. total number of nodes) through continuous collection of appropriate aggregates of network properties. For this the sink should compute the digest and conclude a network partitioning if a sudden drop in the network size occurs. Memento [3] also collect connectivity information regularly for network partition detection. Both approaches create a large communication overhead as periodically collecting data from all network nodes at the sink. The approach for detecting cuts [1] overcomes some of these drawbacks. The approach considers linear partitions of a size at least $\varepsilon * N$ (where N is the number of nodes in the network and ε is a small determined value). A subset of nodes send a heart beat beacon from time to time. When the beacons are not received by the sink it assumes that part of the network is partitioned. The numerous reports to the sink, detection of linear partitions and the poor accuracy are major drawbacks of [1].

All approaches [1-4] utilize reactive approach to detect partition based on connectivity and requires large amount of data to be transferred to the sink for necessary action to be taken. For partition forecasting only connectivity knowledge is not sufficient. Combining energy knowledge provides for predictability besides detecting the network partitioning.

In [7], a partition repair algorithm is discussed which assumes a partition detection system such as that of [1]. This work is complementary to ours as our work can significantly improve repair of network partitioning given its forecasting capability.

3. MODELS AND DEFINITIONS

In this section we first elaborate the relevant system and failure model to capture the generic properties of WSN. Next, we define network partitioning and present the requirements for ParFor.

3.1 System Model

A typical WSN system model composed of N static sensor nodes and a sink is considered. The sensor nodes have limited power, processing and storage capacity. The sink has unlimited energy and computational resources. We assume that all the nodes are homogeneous, i.e. having the same transmission range R and batteries with the same initial energy. We also assume that the sensor nodes are always connected and intermittent disconnections are handled by underlying mechanisms such as MAC layer. Furthermore, the sensor nodes know their own geographic position and are able to periodically check their own residual energy. We presume that energy depletion is a long-term process and does not drop suddenly and extremely. After network deployment, the rate of battery depletion at each node is different based on the network operation and the application execution. However, the event-driven nature of WSN and their inherent redundancy leads to spatially correlated energy consumption. In this paper we consider an arbitrary power consumption model to reflect these conditions. However, our solution suits well for other energy consumption patterns.

3.2 Failure Model

In this paper, we focus on network partitioning as one of the major failure that occurs in WSN. The main reasons for network partitioning are either node or link related failures. The node related reasons correspond to node unavailability, i.e. the nodes may be out of battery power or may be crashed. Whereas, link related failures correspond to signal loss due to obstacles or node mobility. We focused on the major reason of network partitioning, i.e. energy depletion in this work. We considered neither node crashes nor deliberate and catastrophic failures. Also, we assume that link level failures are for short time duration and are dealt by underlying layers, i.e. MAC layer.

3.3 Definition of Network Partitioning

Typically network partitioning is said to occur if one or more nodes become isolated from the rest of the network. As the sink presents the main communication partner for all sensor nodes in WSN, the sink plays a major role for defining network partitioning. The isolation of the sink makes the whole WSN unavailable.

Furthermore, in WSN one should account also for coverage. The fact that a part of sensor field becomes uncovered as all nodes there crashed is a special case of network partitioning. In this paper, we extend the conventional definition of network partitioning by involving the loss of coverage as a network partitioning.

In summary network partitioning in WSN is equivalent to a relevant coverage loss. The uncovered parts involve the areas where all nodes have crashed and those where nodes are isolated from the sink.

Definition A WSN is partitioned iff parts of the sensor field are not covered degrading the WSN functionality below application requirements.

3.4 Requirements on ParFor

Now we discuss the main requirements on the forecasting network partition in WSN.

- Network partition forecasting should be achieved with certain accuracy and within a certain time window specified by user/application.
- We require that the solution returns the details of the partitioned area, i.e. location and size of partition.
- The solution should be suitable for a wide range of WSN deployments. Optimally, the solution should be independent of the application and its execution/deployment (e.g. independent of network traffic).
- The solution should not over-drain some nodes with respect to energy and should provide fair load balancing.

4. NETWORK PARTITION FORECASTING

In this section, we develop and evaluate an efficient solution to forecast network partitioning. The sensor nodes require only local knowledge and only a few sensor nodes communicate their information with the sink where the forecasting takes place. In the following, we give an overview of ParFor and detail the steps to forecast the network partition.

4.1 Map-Based Methodology

In [8], it is argued that the region-level abstraction is better than node-level abstraction in WSN. It is also shown that maps present an appropriate technique to address the WSN at a region level while simplifying event specification and detection. As network partitioning can be viewed as an event like any other application event, we rely on the map concept to propose solutions for partitioning forecasting. The occurrence of network partitioning events, as defined in Section 3.3, creates a map of the WSN field at the sink. For network partition forecasting the cMAP (connectivity map) and the eMAP (energy map) are of interest. As stated in the system model, we assume that the WSN is connected therefore, for forecasting we focus only on energy depletion. Subsequently, in the remainder of the paper we consider only the eMAP to define the ParFor approach.

The main idea is to observe the energy levels of the different regions of the eMAP. Once, the energy level approaches a low level (E'_{th}) early warnings should be sent to the user. The trace of low energy regions of eMAP allows forecasting of network partition using regression algorithm. One possible solution is to pro-actively build a complete eMAP at the sink. However, this is less efficient and requires a lot of communication overhead. Therefore, we aim at maximizing the in-network processing in order to reduce global knowledge and increase efficiency by suppressing unnecessary data close to their sources. For eMAP, we utilized the Isolines approach [9] to construct the map and to consider the following global classes of residual energy (Ψ).

$$\Psi_i = \frac{\Psi_{i-1} E_{\max}}{\# Classes}$$

4.2 Overview of the ParFor Approach

In our approach we argue that global knowledge regarding the map is not needed at the sink. Furthermore, it is also not required to collect the maps throughout WSN lifetime for the purpose of forecasting network partition. We present a hybrid solution (Fig. 1) where only energy weak regions will send information to the sink and the sink forecasts the network partitioning (temporal suppression). Furthermore, only a few sensor nodes from each region have to report the information, i.e. its geometry and residual energy value to the sink (spatial suppression).

The ParFor approach relies on the Isoline approach and supposes global classes for the attribute values, i.e. Ψ, known to the sink and all sensor nodes. We denote by $\Psi_i(t)$ the Ψ of node i at time t. For eMAP, the possible Ψ may be 0-9%, 10-19% till 90-100%. The sensor nodes exchange their minimal local view on the eMAP (the values of their residual energy) with their direct neighbors so that nodes can conclude if they are located on the border with neighboring (energy) regions or within a region. If a sensor node has the same as all its neighbors then it is located within a region. On the other hand, if one of its neighbors has a different Ψ then the sensor node is marked as a BN (Border Node). Having the incremental trace of BNs, the sink acquire the relevant Isolines and forecast the partition.

4.3 ParFor Efficiency

The efficiency of ParFor involves spatio-temporal suppression of data to be sent towards sink for forecasting. For spatial suppression, we let only the set of BNs that belong to a energy-weak region report their energy values and their positions to the sink. For temporal suppression, selected BNs should report to the sink only if the energy value of the region reaches E'_{th}. The reporting frequency of selected BNs can be adapted to the user requirements and residual energy of sensor nodes.

4.3.1 Spatial Suppression in ParFor

The spatial correlation in energy consumption is our motivation to correspond energy regions in the WSN field. For the energy weak region we let only the BNs report their values and their position to the sink. Alternatively, one node can acquire the region information, i.e. its perimeter and energy value and send it to the sink. In this case one reporting node is sufficient. In order to provide fault-tolerance, a few further nodes can also report. There is a trade-off between creating the region knowledge at one central region node that reports or all BNs report to the sink (Fig. 2).

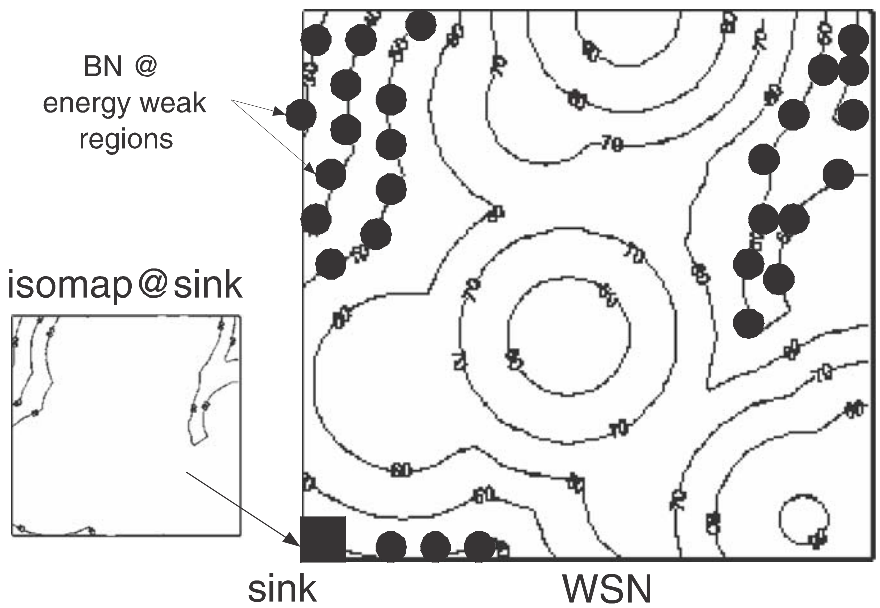


FIG. 1. NETWORK PARTITION FORECASTING USING PARFOR

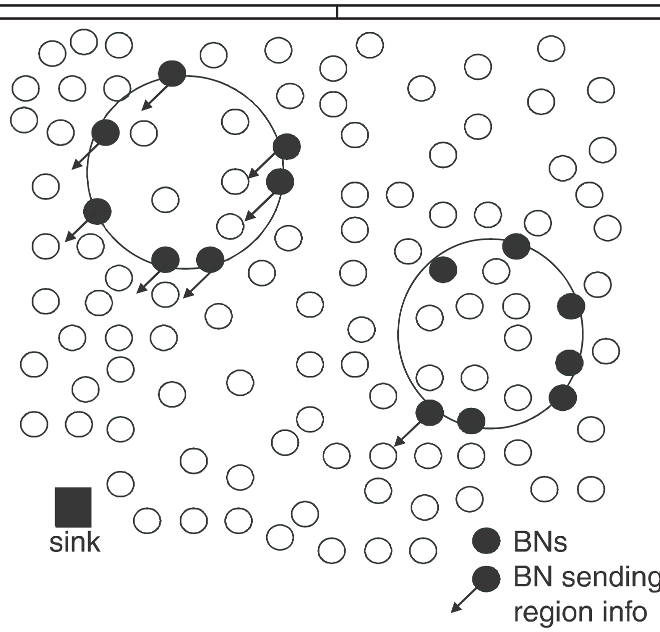


FIG. 2. SPATIAL SUPPRESSION USING PARFOR

4.3.2 Temporal Suppression in ParFor

The temporal suppression targets following core questions, i.e. when to start reporting, how frequent should nodes report and when to stop reporting. The reporting of energy weak region is dependent upon selection of E'_{th}, provided by the user as a forecasting window. If the energy consumption is accurately forecasted then the optimal time to report can be accurately calculated, however this is not typical for WSN. Accordingly, in order to fulfill the requirements on the forecasting window, we incrementally increase the frequency of reporting till we reach a highest reporting rate and then decrease the rate. The time of highest rate should correspond to the optimal time to fulfill the requirements on the forecasting window. Clearly, this is a trade-off between efficiency and accuracy of forecasting. As the nodes are getting energy-critical (E''_{th}), we should reduce the frequency of reporting and eventually stop reporting leading to partition occurrence.

4.4 Sensor Node Algorithm

We describe the sensor node Algorithm-1 in details and to elaborate it an example in Fig. 3(a-b) is presented.

4.4.1 Regioning

As soon as the first sensor node reaches the E'_{th}, the regioning will start. The starting node initiates regioning through broadcasting a regioning beacon message rBEACON {nodeID, SN_{pos}, SN_E, region-ID} to all its neighboring nodes. The receiving nodes behave similar to flooding (further optimization is possible, adaptive flooding, like adaptive gossiping [10]), i.e. they relay the rBEACON after a short random time to all neighbors. In order to restrict flooding to the region of weak-energy, we stop the flooding at the border of the region. We achieve this by identifying the set of nodes that are candidates for BN role and letting only these nodes and the nodes in energy-weak region execute flooding. From the candidates for BN, only nodes belonging to the energy-weak regions will send ALERT messages to the sink.

In Fig. 3(a), we consider a simple example for regioning with E'_{th}=25%. Let Node F be the first node to reach E'_{th}. Node F sends a rBEACON message to its neighbors. Upon receiving rBEACON Nodes A, B, G, K and L observe that they have the same Ψ as Node F (thus, they are not BNs). They also broadcast rBEACON with the similar region-ID as of Node F. Relaying node should keep for short time a history of all regioning rounds, where they were involved ($HIST_{rID}$). Eventually, node P also receives rBEACON (e.g. from node K). As $\Psi_P(t) > \Psi_F(t)$, Node P marks itself as BN. After P sends its own rBEACON, Node F also marks itself BN. As Node K receives a beacon with region-ID already seen, it will be prohibited to unnecessary send rBEACON. Similarly, Nodes L, M, N, I, D, Q, R, S, T, O, J and E mark themselves as BNs.

In order to unnecessarily avoid flooding the whole network, after receiving a beacon message from Node P or Q, node U will not broadcast. Node U observes that it has same Ψ as the sender node, but higher than E'_{th}, therefore it ignores the message and doesn't forward it. Similarly, Nodes U, V, W and Y will stop flooding. From all BNs, only those belonging to the energy weak region should report to the sink, i.e. Nodes D, K, L, M, N and I.

A node becomes initiator, only if it is not already belonging to an energy-weak region. Therefore, a flag can-init on each node is required. Nodes within the threatened region should set the can-init flag to false. Therefore, flooding the whole region is not unnecessary. The can-init flag should be set to true, if a sensor node changes its such that it can initiate regioning if needed for the new Ψ.

4.4.2 Regioning Update

The energy region of a sensor node may change if its Ψ changes, i.e. the sensor node leaves one region and joins a neighboring region. Accordingly, the regioning algorithm should update the region information at the sink. This means that the sensor node that joins an energy weak region and it is a BN should report this to the sink, which accordingly updates the eMap. If the sensor node already belongs to a energy weak region then it should be inserted in the new region. The sink should not remove nodes that are not BN any more as this is redundant information. BNs after changing Y can join the neighboring region whereas, the non-BN will sometimes initiate regioning. A BN sends rJOIN message {node ID, SN_{pos}, SN_{E}, region-ID} via

Algorithm-1: Regioning Algorithm for sensor nodes

```
Data: E'_th, E''_th, Ψ, HIST_region-IDs, region − ID_1, region − ID_2, node-ID,
      ENERGY, ENERGY-CLASS, POS, BN, can-init=TRUE
upon change in ENERGY-CLASS:
/* REGIONING INIT */
if (can-init = TRUE) & (ENERGY ≤ E'_th) then
    can-init = FALSE;
    new (region-ID);
    broadcast rBEACON(region-ID, POS, ENERGY);
    insert region-ID to HIST_region-IDs;
/* REGIONING START*/
SN: receive rBEACON
if ENERGY-CLASS = msg.ENERGY-CLASS then
    if msg.region-ID ∌ HIST_region-IDs then
        new rBEACON(msg.region-ID, POS, ENERGY);
        broadcast rBEACON;
    if ENERGY-CLASS = msg.ENERGY-CLASS then
        can-init = FALSE; BN=true
    if BN = TRUE then
        report-to-sink();
/* REGIONING UPDATE */
if (ENERGY ≤ E''_th) & (BN = TRUE) then
    broadcast rJOIN(region-ID, POS, ENERGY);
    insert region-ID to HIST_region-IDs;
    can-init = FALSE;
SN: receive rJOIN
if (BN=TRUE) & (ENERGY-CLASS = rJOIN.ENERGY-CLASS) then
    broadcast rWELCOME(region-ID, ENERGY);
    if random time expired then
        if receive rWELCOME with the same region-ID then
            suppress rWELCOME;
if ENERGY-CLASS ≠ rJOIN.ENERGY-CLASS then
    BN = TRUE;
function report-to-sink():
send ALERT= {region-ID, ENERGY, POS} to the sink using routing
end function
```

broadcast. Nodes that are BN or receive rJOIN with the same Y ignore rJOINs and reply with a rWELCOME message. It is sufficient to receive one rWELCOME beacon such that the sender of rJOIN joins the region. Therefore, nodes that receive rJOIN schedule rWELCOME to a random time and suppress it if they listen another rWELCOME. Other receivers of rJOIN become BNs with region-ID of the received rJOIN message.

Let us consider the same example above (Fig. 3(b)), where after sometime Node R residual energy becomes 29% from 34%. Node R broadcasts rJOIN and Nodes L, M, N, Q and S reply with a rWELCOME as they are either BN or have the same as Node R. Nodes V, W and Y do not have the same Ψ as the sender so they become BN.

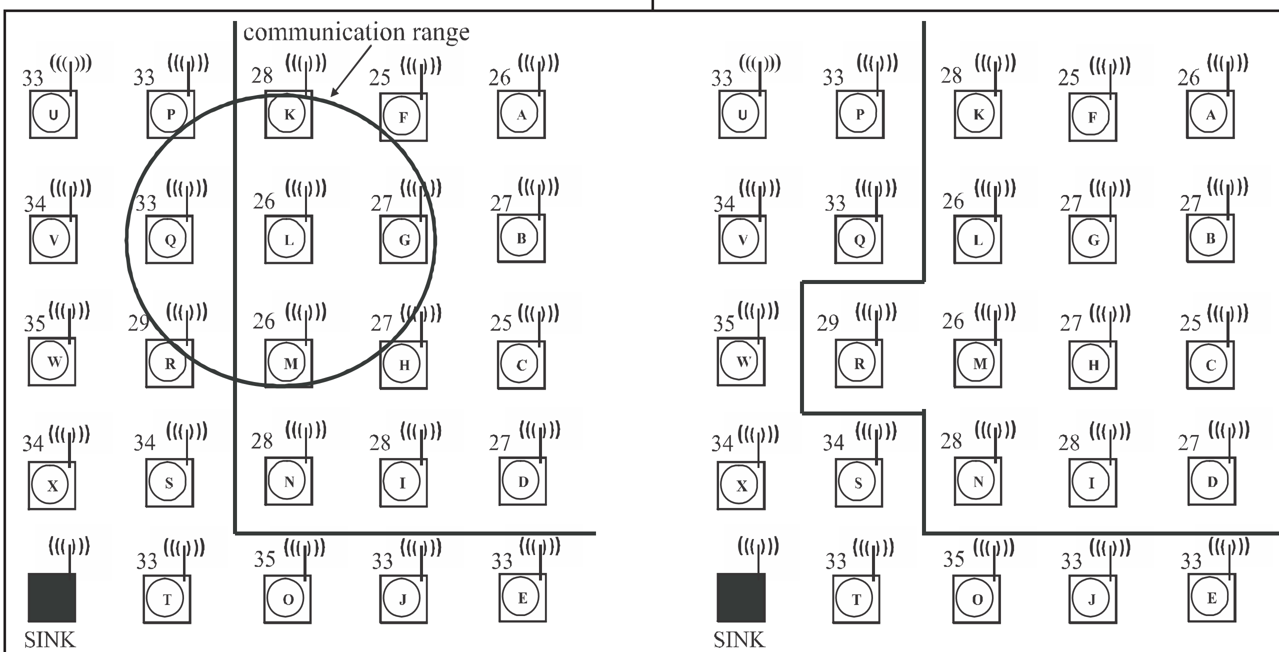
4.3.3 Optimizing the Number of Reporting SNs

When the transmission range R is large or the node density is high, the number of BNs increases. The ParFor approach proposes a strategy to keep the number of BN minimal. To achieve this, ParFor probabilistically allow nodes to report ALERT messages to the sink.

Since, nodes exchange their positions they can calculate the distance between each other. Accordingly, BNs know their distance to the BNs in the neighboring region. A BN in the energy weak region calculates its distance to all neighbors, which are BNs in the neighboring region, and fixes the minimal distance to d_{min}. If d_{min} is higher than a certain threshold d''_{th} then the BN will not report its value to the sink. If d_{min} is smaller than a certain threshold d'_{th}, then the node reports to the sink. If $d'_{th} < d_{min} < d''_{th}$, then the node reports with a probability $P(d_{min}<P<d'_{th})$. It should be noted that $d'_{th} < d''_{th} < R$.

4.5 Processing at the Sink

The unique region-ID is used to classify the incoming trace of BNs into a unique region of eMAP. If the ALERT message belong to existing region available at the sink,



(A) REGIONING AT SENSOR NODES *(B) REGIONING UPDATE AT SENSOR NODES*

FIG. 3. LOCAL REGIONING AT SENSOR NODES

the sink will discard the message else the new region is created at the sink. Once the region is available at the sink, it will follow a simple regression approach [5-6] for partition forecasting.

5. EVALUATION

We first present the simulations settings and then describe the evaluation metrics. In last we discuss the results of our simulations.

5.1 Simulation Settings

To evaluate ParFor to forecast network partitioning we use simulations in TOSSIM. The sensor nodes are deployed as a grid (N=m x m). We varied the number of nodes and R in our simulations. First, each node can directly communicate its one hop neighbors (R1). After increasing R(R2=2*R), the sensor nodes can communicate with 2 hop neighbors. By varying R we evaluate our probabilistic approach for minimizing BNs. We assume the presence of a routing algorithm such that the ALERT massages can be sent towards the sink. Following global classes of residual energy are considered, Ψ_1: 0-9%, Ψ_2: 10-19% till Ψ_{10}: 90-99%. When the node reaches fixed energy classes EC4, EC3, EC2 and EC1 (EC4=49-40%, EC3=39-30%, EC2=29-20% and EC1=10-19%), the regioning process starts. Once E'_{th} reaches 45% the nodes start exchanging data and stopwhen E''_{th}=10%. Table 1 summarizes the simulation parameters. For naive approach we allow all nodes to send the energy values periodically to the sink.

5.2 Evaluation Metrics

The following metrics are used to evaluate ParFor approach.

(1) *Accuracy:* The accuracy of the forecasted time of partition is evaluated by comparing the forecasted partition time (T_f) to the expected partition time (T_e). T_e is calculated by leaving the simulation running until the node in a region reaches to energy value 0. Accordingly, the accuracy level (ACC_{Tf}) is as follows:

$$ACC_{Tf} = \frac{T_f\left(REG_i\right) - T_f\left(REG_i\right)}{T_e\left(REG_i\right) - reportingTime\left(REG_i\right)}$$

(2) *Efficiency:* Efficiency corresponds to message overhead and the number of BN selected to report the energy weak regions.

(3) *Message Overhead:* The total number of broadcast messages exchanged to construct the energy-weak region is related to message overhead for ParFor.

(4) *Number of BN:* The ratio of nodes from a region received at the sink to the number of nodes in the region.

$$AR_{reg} = 100\% - \frac{\# n_{\text{Reporting}}}{\# n_{\text{Region}}}$$

TABLE 1. SIMULATION PARAMETERS

Parameter	Value
Simulation Area	100x100 units
Grid Size	[5-6,7,10] units
Nodes	100, 150, 200, 250, 300, 350, 400
Communication Range	R1=5, R2=10 units
Energy Classes	EC4, EC3, EC2, EC1
E'_{th}	45%
E''_{th}	10%

5.3 Simulation Results

We varied our energy threshold from EC4 to EC1 to start regioning process and to send the ALERT messages towards the sink. Fig. 4(a) shows 50080% accuracy for partition forecasting time. For EC1 we observe that the accuracy of forecasted time is higher than EC2, EC3 and EC4. The larger threshold values require regioning to be started earlier and any unsteadiness in energy consumption leads to deviation from accurately forecasting the partition. Accordingly, the lower threshold values lead to more accuracy. The trend remain same for higher number of nodes. For naive approach we observe that the accuracy improves since the nodes periodically send the data to the sink. As the sink has sufficient data available, the accuracy of forecasting is increased.

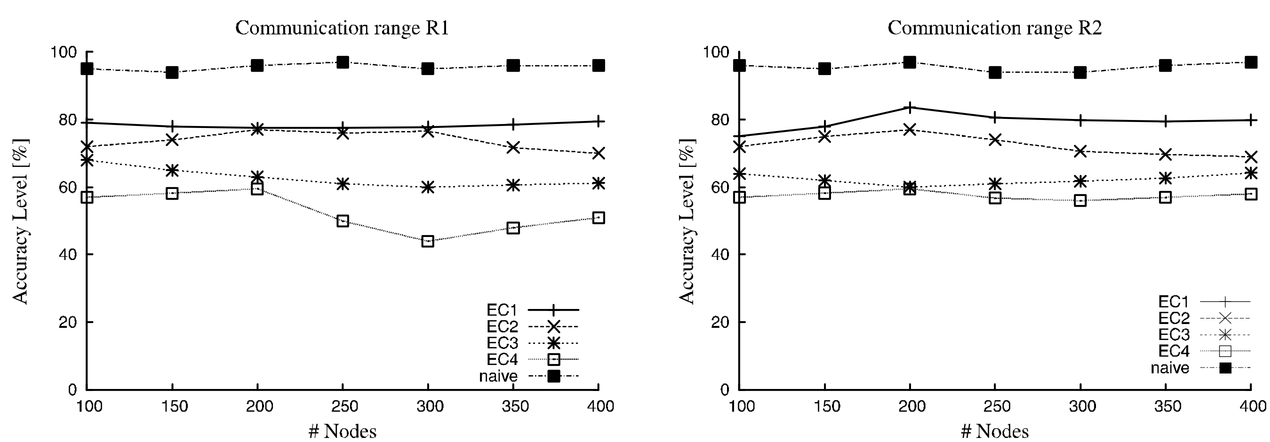
By increasing communication range to R2, the forecasted time slightly improves (Fig. 4(b)). It is also observed that when the energy is close to E''_{th}, the forecasted time is more accurate. It is due to the fact that the sink has more energy dissipation information. The accuracy remains similar for naive approach.

Fig. 5 shows that the message overhead is proportionally growing with the number of nodes in the network. By comparing Fig. 5(a-b), it is observed that the message overhead shows similar trends for R1 and R2. This proves the viability of our proposed probabilistic reporting approach for keeping minimal number of messages. Since, the naive approach requires all the nodes to send their energy value to the sink, the number of transmissions is very high.

Fig. 6 depicts the improved efficiency of ParFor for various number of nodes due to good energy correlation of regions in WSN. It is also identified that around 30% of nodes (BN) in a region forward ALERT messages towards the sink. This shows that ParFor is very much suitable for large deployments as well.

6. CONCLUSIONS

In this paper we proposed and evaluated an efficient approach to reduce the data to forecast network partition. The core idea is to utilize the energy correlation to forecast the time of partitioning. The results show that using the residual energy of only BN leads to a good approximation for energy consumption in the whole WSN. In future, we would like to investigate other forecasting models to improve the accuracy of partition forecasting.



(A) ACC_{Tf} WHEN R1=5 UNITS

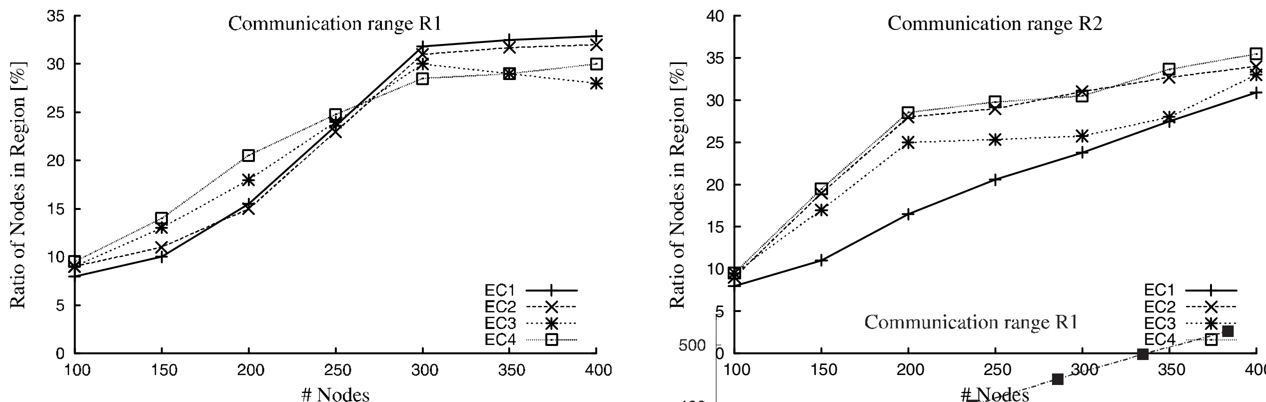
(B) ACC_{Tf} WHEN R1=10 UNITS

FIG. 4. ACCURACY OF FORECASTED TIME OF PARTITION

(A) MESSAGE OVERHEAD WHEN R1=5 UNITS

(B) MESSAGE OVERHEAD WHEN R1=10 UNITS

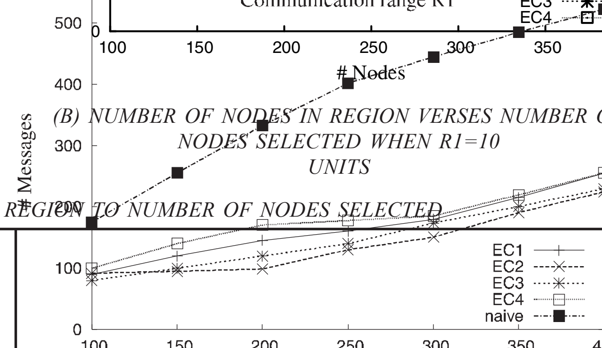
FIG. 5. PARFOR'S MESSAGE OVERHEAD

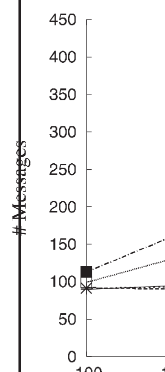


(A) NUMBER OF NODES IN REGION VERSES NUMBER OF NODES SELECTED WHEN R1=5 UNITS

(B) NUMBER OF NODES IN REGION VERSES NUMBER OF NODES SELECTED WHEN R1=10 UNITS

FIG. 6. RATIO OF NUMBER OF NODES IN REGION TO NUMBER OF NODES SELECTED





ACKNOWLEDGMENTS

This research is supported in parts by Mehran University of Engineering & Technology, Jamshoro, Pakistan, and DEEDS, TUD Germany. We are also thankful to Dr. Khelil and Momo Simon, for their help and support for finalizing the paper.

REFERENCES

[1] Shrivastava, N., Suri, S., and T´oth, C.D., "Detecting Cuts in Sensor Networks", International Symposium on Information Processing in Sensor Networks, 2005.

[2] Shih, K.P., Chen, H.C., Tsai, J.K., and Li, C.C., "Palm: A Partition Avoidance Lazy Movement Protocol for Mobile Sensor Networks", Wireless Communications and Networking Conference, pp. 2484-2489, 2007.

[3] Rost, S., and Balakrishnan, H., "Memento: A Health Monitoring System for Wireless Sensor Networks", Sensor and Ad Hoc Communications and Networks, pp. 575-584, 2006.

[4] Zhao, J., Govindan, R., and Estrin, D., "Computing Aggregates for Monitoring Wireless Sensor Networks", International Workshop on Sensor Network Protocols and Applications, 2003.

[5] Ali, A., Khelil, A., Shaikh, F.K., and Suri, N., "MPM: Map Based Predictive Monitoring for Wireless Sensor Networks", International ICST Conference on Autonomic Computing and Communication Systems (Autonomics), pp. 79-95, 2009.

[6] Ali, A., Khelil, A., Shaikh, F.K., and Suri, N., "Efficient Predictive Monitoring of Wireless Sensor Networks", International Journal of Autonomous and Adaptive Communications Systems, 2010.

[7] Dini, G., Pelagatti, M., and Savino, I., "An Algorithm for Reconnecting Wireless Sensor Network Partitions", Wireless Sensor Networks, pp. 253-267, 2008.

[8] Khelil, A., Shaikh, F.K., Ayari, B., and Suri, N., "Mwm: A Map-Based World Model for Event-Driven Wireless Sensor Networks", International Conference on Autonomic Computing and Communication Systems, 2008.

[9] Solis, I., and Obraczka, K., "Isolines: Energy-Efficient Mapping in Sensor Networks", Symposium on Computers and Communications, pp. 379-385, 2005.

[10] Khelil, A., and Suri, N., "Gossiping: Adaptive and Reliable Broadcasting in Manets", Latin-American Symposium on Dependable Computing, pp. 123-141, 2007.