

Aggregation-Aware Routing on Wireless Sensor Networks^{*}

Antonio J.G. Pinto, Jorgito M. Stochero, and José F. de Rezende

Grupo de Teleinformática e Automação (GTA)
COPPE/PEE – Universidade Federal do Rio de Janeiro
Caixa Postal 68504,
21945-970 Rio de Janeiro, RJ, Brazil
{antonio,stochero,rezende}@gta.ufrj.br

Abstract. In wireless sensor networks, data aggregation is critical to network lifetime. It implies that data will be processed in an efficient flow from multiple sources to a specific node named sink. However, there are trade-offs for fusing multisensor data and creating a path between source and sink that increase the likelihood of aggregation. In this work, we propose a decentralized mechanism using parametric-based techniques, such as Bayesian Inference and Dempster-Shafer Method, for data aggregation in wireless sensor networks. Moreover, we propose an extension to an existing data-centric routing protocol in order to favor aggregation. Our approach is evaluated by means of simulation.

Keywords: Sensor networks, routing, in-networking aggregation

1 Introduction

Recently, the research interest in Wireless Sensor Networks (WSNs) is considerable because of its wide range of potential applications, from weather data-collection to vehicle tracking and physical environment monitoring [1]. Communicating wirelessly consumes much more power at the nodes than processing or sensing [2]. Thus, it is preferred that data be processed *in-network* as opposed to a centralized processing. Sensors data are transmitted from multiple acquisition sources toward one or more processing points, which may be connected to external networks. Since sensors monitor a common phenomenon, it is likely to appear significant data redundancy, which may be exploited to save transmission energy, throughout in-network filtering and data aggregation procedures. Techniques employed range from suppressing duplicated messages and performing distributed basic functions, such as *max*, *min*, *average*, and *count*, to data fusion.

In this paper we deal with two different issues: aggregation tree construction in data-centric routing protocols and aggregation mechanism using parametric techniques, in order to save energy with a low cost to the network. The main

^{*} This work was supported by CNPq, CAPES, SUFRAMA, and FAPERJ/Brazil.

motivation behind this work is to exploit the collaborative characteristics of sensor nodes to jointly estimate about a certain common sensed phenomena. The remainder of the paper is organized as follows. In section 2 we present a review of related work in routing for aggregation and aggregation techniques themselves. Section 3 describes the routing protocol and the multisensor aggregation techniques proposed. Simulation results are presented in Section 4. Finally, the paper is concluded in Section 5.

2 Data-Centric Routing and Aggregation

Energy is a factor of utmost importance in WSNs. To increase network lifetime, energy must be saved in every hardware and software solution composing the network architecture. According to the radio model proposed in [2], data communication is responsible for the greatest weight in the energy budget when compared with data sensing and processing. Therefore, to save energy it is better to increase data processing in order to avoid data transmission. Energy-aware routing protocols can exploit two particularities of WSNs: data redundancy and a many-to-one (sources to sink) association.

In order to *in-network aggregation* to be effective, routing protocols should use the most suitable addressing and forwarding schemes. In data-centric routing protocols [3,4], intermediate nodes build forwarding tables based on interests sent by the sink and on the information provided by the sources. Source nodes are recognized based on the sort of data content they can provide (temperature, smoke presence, etc.) and in their current geographical location (if available). Thus, instead of assigning a Global Unique ID to sensor nodes, network protocols may use *attributed-based addressing*.

Other routing protocols aim at building aggregation trees from multiple sources to the sink. The NP-hard Minimum Steiner Tree gives the optimal aggregation path in terms of number of transmissions [5]. Authors in [6] analyze three suboptimal aggregation schemes as function of two positioning models and an increasing number of source nodes. Hierarchical cluster-based routing favors aggregation in the cluster-heads and is analyzed in [2]. Building aggregation paths creates a trade-off between delay and energy saving.

Concerning the data aggregation mechanisms, the breadth of applications and diversity of applicable techniques [1] make their study very complex. These techniques correspond to a mixture of mathematical and heuristic methods drawn from statistics, artificial intelligence, and decision theory [7]. The most trivial data aggregation function is to suppress duplicated messages. The aggregation techniques involve *how* to aggregate data. The basic functions of aggregation (such as *max* and *min*) are appropriate to aggregate data from multiple inputs of a same type of sensor, while fusing data from multiple kinds of sensors requires more complex methods. Some of these methods have been exploited in diverse mathematical developments, such as those called parametric techniques, like Bayesian and Dempster-Shafer Inferences. These techniques compose the data fusion model created by *Joint Directors of Laboratories Data Fusion Sub-*

panel (JDL DFS) of the American Department of Defense [7]. JDL DFS model has been the target of several research and developments involving traditional sensors (for instance, radar and ELINT sensors). However, WSN literature has been limited only to cite the potential of these techniques. In this work, we apply parametric techniques on a data-centric routing protocol, establishing an aggregation mechanism that minimizes the negative aspects of traditional approaches.

3 Aggregation-Centric Routing and Multisensor Data Aggregation

This work addresses different aspects of aggregation in WSNs. These aspects involve the use of routing protocols and filtering mechanisms implemented in the sensor nodes to save energy. In this paper, we propose an extension to an existing data-centric routing protocol in order to increase the likelihood of aggregation without incurring in the overhead related to aggregation tree building and maintenance. In addition, an aggregation mechanism, using parametric techniques, with the minimum of losses in terms of delay, scalability and robustness.

3.1 Privileged Aggregation Routing (PAR)

The data-centric routing protocol used is the Directed Diffusion proposed in [3], which allows application specific processing within the network. Data is named using attribute-value pairs. The sink performs *interests* dissemination to assign sensing tasks to the sensor nodes. This dissemination sets up *gradients* in order to draw data that match the interest. Events (data) start flowing towards the originators of interest along multiple paths. The sink *reinforces* one particular neighbor from which it receives the first copy of the data message. Intermediate nodes using the same criterion reinforce their respective neighbor. This approach privileges lower delay paths, while it does not necessarily lead to the establishment of better aggregation paths.

We propose a new scheme to reinforce neighbors. The idea is to use the same localized interactions of Directed Diffusion and exploit its filtering architecture, described in [8], to build an empirical aggregation tree in the network. The diffusion filter architecture is a software structure for a distributed event system that allows an external software module, called *filter*, to interact with the Directed Diffusion *core* and modify its routing capabilities to influence how data is moved through the network. In this work, Directed Diffusion is modified to reinforce intermediate nodes with greater potential to combine data from different sources. After these nodes are found and included in the source-to-sink path, distributed estimation algorithms are used to efficiently aggregate network traffic into a reduced number of higher delay data messages.

A sensor node has potential for aggregation if it receives, in a timely manner and from different sources, data messages with the same attributes (*e.g.* temperature), but not necessarily the same values. Data fusion will depend on the

aggregation function used. We consider a path efficient if the number of aggregation nodes is higher and these nodes are closer to the sources. By selecting these paths, we trade an increase in delay and in processing time for a reduced number of transmissions, while limiting measurement accuracy loss. To identify and select these better paths, we added a software module to the Directed Diffusion implementation available on ns-2 [9]. We named this module *PAR filter* (Privileged Aggregation Routing filter). Our filter acts on the data messages used for route discovery and path setup, i.e. *exploratory* data messages.

Instead of immediately forwarding the first received exploratory data message toward the interest message originator, the PAR filter sets up a timer associated to this message and compares it with other messages received before the timer expires. To perform the comparison, three attributes were added to the message header, namely *distance_to_source*, *aggreg_nodes* and *aggreg_ID*. The first contains the number of hops between the source and the first aggregation node, indicating how far is the aggregation point from that specific source of information. The second stores the number of aggregation nodes in the path followed by that exploratory data message. Finally, *aggreg_ID* keeps an indication of which of the sources in the network are included in the aggregation path.

Messages received from a particular neighbor with the highest *aggreg_nodes* and the lowest *distance_to_source* values are chosen to be forwarded, even if they are received later than other messages, which are discarded. The last field (*aggreg_ID*) is used to avoid computation of the same source twice in the aggregation path, which would give erroneous numbers for *aggreg_nodes* and could jeopardize our mechanism for route selection. Because a node does not have global knowledge of the topology, an exploratory data message from a specific source could have been aggregated two hops away, and the current node has no means of knowing this. The only way to provide the information of which sources are included in the *aggreg_nodes* computation is to pass this information along with the message. If the *aggreg_ID* of the received message matches the messages already in the node, then the current message is not aggregated.

The route selection mechanism works in two phases. The first one occurs when an exploratory data message arrives. It is compared with similar messages already stored in the filter to check if it is a candidate for aggregation or to be discarded. Messages are candidate for aggregation if they respond to the same interest come from different sources and have not been previously aggregated, as shown by their *aggreg_ID*. When two candidates are found, their aggregation parameters are updated to reflect this. To discard a message, the PAR filter checks aggregation parameters of messages that come from the same source, preserving messages with better parameters and discarding the others. The comparison is made using the *aggreg_nodes* (higher is better) and *distance_to_source* (lower is better) values, in this order. If a message is to be kept, a timer is set before transmission. Otherwise, it is deleted. When the timer for a specific message expires, the second phase begins. Now, a double check is performed to identify any modification in the message queue, and then the better message is sent back to the diffusion core.

The Directed Diffusion routing module will act as it normally does, except that now it receives different exploratory data messages than before, with better aggregation parameters, thus different neighbors are reinforced. These neighbors are the ones we chose to privilege aggregation. For instance, suppose a node has two neighbors, a and b , from which it receives similar messages, using the criteria discussed above, and the message from a arrives first. Normally, the a message (low latency) would be forwarded and the b message would be discarded. With the PAR processing, it could be identified that the *aggreg_nodes* value of a is 1, indicating it conveys aggregated information from two distinct sources, where the *aggreg_nodes* value of b is 2, indicating an aggregation of three sources. The PAR filter would then discard a and forward b to the next node.

This process continues in every intermediate node in the path until the sink is reached and the desired (aggregated) routes are reinforced. Empirically, intermediate nodes in the reinforced path receive similar data messages, candidates for aggregation, from different sources. These intermediate nodes are the closest to these sources, which presents the greatest gains in terms of a reduced number of transmissions. The aggregation timer can be used to adjust the desired latency to the number of transmissions ratio. A small timer implies low latency, while a larger timer reduces the number of transmissions with the cost of higher latency. After the desired routes were established, another software module must be attached to the nodes of the sensor field to act on the *data* messages and to aggregate and suppress redundant information. In the remainder of this work, we present two aggregation filters, one using Bayesian Inference (Bayes Filter) and another the Dempster-Shafer Inference (DS Filter).

3.2 Distributed Multisensor Aggregation

In order to reduce the impact of multisensor data aggregation in the WSN, we propose a distributed aggregation mechanism. In such mechanism, aggregation occurs locally at each node on the basis of available information (local observations and information communicated from neighboring nodes). Data does not need to be held at any node in order to be combined, and in this sense it differs from traditional store-and-forward techniques. The result is a distributed processing in the network, which favors scalability and a reduction of transmissions, without occurring in increase of the latency.

The aggregation mechanism starts after the definition of source-to-sink routes, as illustrated in the figure 1. The aggregation procedure consists of keeping a copy of the last evidence received in each node (in the source-to-sink path). This evidence can be used for aggregation for a period T_v , as illustrated in the figure 1(a), which defines the valid time of the evidence. To prevent data aggregation of the same sensor, the parameter T_v must be inferior to the lesser interval of data transmission (I_t) from the sources. Another parameter to be adjusted is the discard time of the messages T_d , illustrated in figure 1(b). This parameter defines the period in which aggregate messages can be considered redundant, and therefore discarded. Finally, in the case of Dempster-Shafer aggregation, the

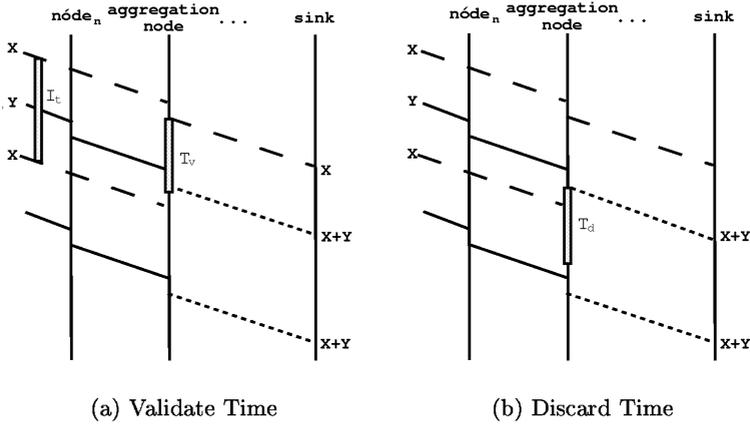


Fig. 1. Parameters of the Aggregation Mechanism.

parameter C defines the admitted maximum variation for the degree of certainty of the messages (in an interval T_d).

When a data type message arrives in the node it is forwarded to the aggregation filter. The first step consists of verifying valid evidences in the cache. This is done comparing the difference between current time and the evidences arrival time with T_v . After that, the message’s content is read to identify which evidences are being sent. After these procedures, the evidences (local and received) are combined, generating a new message. Then, the cache is updated with the most recent evidences. The new message could be forwarded to Gradient filter or simply discarded, if identical (or without significant changes) when compared with the previous message, within interval T_v .

The nodes carry out aggregation by combining data fusion based on parametric techniques with removal of redundant messages. Such techniques allow a direct mapping between the evidences detected by sensors with events of interest, which is simplified by the data centric addressing schema of Directed Diffusion. The aggregation methods have been implemented in the Bayes (Bayesian Inference) and DS (Dempster-Shafer Inference) Filters, which were software modules added to the protocol.

The Bayesian Inference [7] was implemented in the **Bayes Filter**. Bayesian Inference is a statistical-based data fusion algorithm based on Bayes’ rule (equation 1) that provides a method for calculating the conditional or a posteriori probability of a hypothesis being true given supporting evidence detected by sensors.

$$p(H_i|E) = \frac{p(E|H_i)p(H_i)}{\sum_i [p(E|H_i)p(H_i)]} \tag{1}$$

DS Filter implements data aggregation based on Dempster-Shafer Inference [10]. This technique can be considered as a generalization of the Bayesian Infer-

ence, being used when the sensors contribute with information (called propositions) that cannot be associated the one hundred percent of certainty. Knowledge from multiple sensors about events are combined using Dempster' rule (equation 2) to find the intersection or conjunction of the proposition (for instance, e_1) and the associated probability, denoted by m_i . This combining rule can be generalized by iteration if we treat m_j not as sensor S_j 's proposition, but rather as the already combined (using Dempster-Shafer combining rule) observation of sensors.

$$(m_i \oplus m_j)(e_1) = \frac{\sum_{E_k \cap E_l = e_1} m_i(E_k)m_j(E_l)}{1 - \sum_{E_k \cap E_l = \emptyset} m_i(E_k)m_j(E_l)} \quad (2)$$

4 Simulation Results

In this section, we describe the results obtained with our simulation model implemented in the ns-2 simulator [9]. In this evaluation, we used two different scenarios. In the first one, a certain number of sensor nodes are randomly distributed over a sensor field of size 1000m x 1000m. The second scenario uses a grid topology where each node is spaced 150m from its neighbors. In both scenarios, all nodes have a transmission range of 250m and the interval between data messages sent by the source nodes is of 1 second. Due to space constraints, some of more detailed simulations and analysis have been omitted.

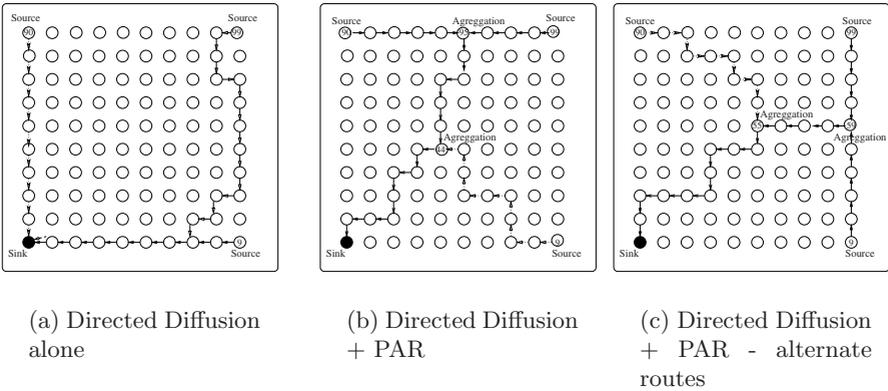


Fig. 2. Preferred and Alternate Routes.

The first step in the simulation involves route discovery. We tested a 100-nodes grid topology with and without the PAR filter. We found, as expected, that routes discovered by directed diffusion, without PAR, followed independent paths, where preferred and alternate routes can be observed. Figure 2(a) shows

the preferred routes for each source. The alternate paths can be explained as a way to keep the energy balance in the network, avoiding early depletion of the nodes involved in the preferred path. When the PAR filter was added, we found new preferred routes. These routes alternate always looking for aggregation nodes, as we can observe in Figures 2(b) (node 95) and 2(c) (node 55).

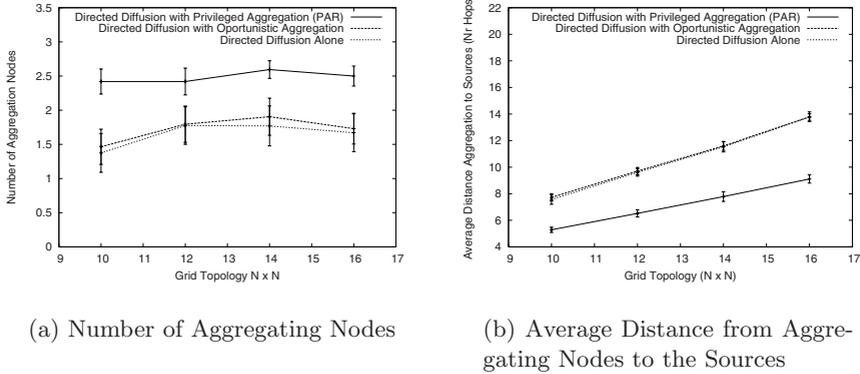


Fig. 3. Aggregation in the N x N grid.

Figure 3 shows the impact of these new routes in the aggregation parameters. Figure 3(a) shows the number of aggregating nodes for different grid sizes, and Figure 3(b) their average distance to the sources. We may observe that the PAR filter increased the number of aggregating nodes for all grid sizes, while reducing their distance to the sources. As discussed in section 3, we expect that the larger number of aggregation nodes will reduce the number of messages received at the sink, while the small distance from the aggregation point to the sources will reduce the number of hops in the network.

To verify the impact of the aggregation timer in the route discovery, we increased its value in successive rounds of simulation. The timer is important for the PAR filter to collect neighborhood information to decide on best aggregation path. However, it increases the route discovery latency, calculated by the difference between the time of arrival of the first positive reinforcement in the source node and the first corresponding exploratory data sent by the same source. Figure 4 shows this metric as function of the aggregation timer. In larger grids, the timer influence is even larger since the route discovery messages pass through a greater number of nodes. A timer comprised around 0,5s proved to be a good balance between efficient aggregation trees and route discovery latency.

Energy savings are compared through the use of Bayes and DS filters in two simulation scenarios. In the first one, we use the Bayes filter in the grid topology and we vary the grid size. The second scenario uses the DS filter and a random topology for a varying number of nodes. In both scenarios, the number of sources

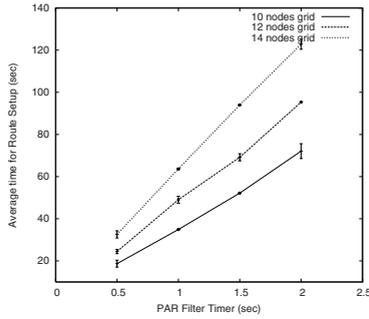


Fig. 4. Route Discovery Latency Versus Timer.

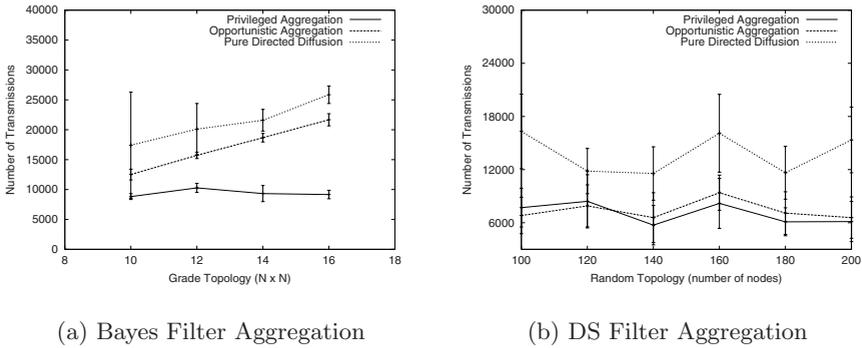


Fig. 5. Total Number of Hop-by-Hop Transmissions.

is constant and equal to five. The total number of transmissions used to send the same amount of information from the source to the sink as function of grid size and the number of nodes is shown in Figure 5. The longer is the route larger is the number of hops even with the same number of messages arriving at the sink. Normally, the routes obtained with the PAR filter are longer than the ones with the pure Directed Diffusion since the former moves away from shorter routes in order to find aggregation nodes. So, it is important to find efficient aggregation trees to reduce the number of hops. The results of Figure 5 show the impact of bringing the aggregation points near to the sources, which produced a considerable gain in this performance metric. Once the sources were far away from the sink, for bigger grids the aggregation benefits increased, because much less hops were necessary to convey information to the sink.

5 Conclusions

In this paper, we propose a multisensor-aggregation-centric routing protocol that increases sensors network lifetime. In such scheme, two filters were added to the directed diffusion protocol. The first filter is called Privileged Aggregation Routing (PAR) and it acts during route setup to identify intermediate nodes with greater potential to combine data from different sources. These nodes are included in the source-to-sink path in order to favor data aggregation, but with the overhead of increasing route discovery time. The second filter is responsible for the aggregation mechanism. In this paper, we explore two techniques, Bayesian Inference (Bayes Filter) and Dempster-Shafer Inference (DS Filter). Both were implemented in order to minimize the aggregation cost in WSNs, since it occurs locally at each node on the basis of the available information (local observations and information received from neighboring nodes). We show by means of simulation that our aggregation mechanism in a privileged fusion path between sources and sink can reduce drastically the total number of transmissions needed to accomplish the task.

References

1. I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirici, "A survey on sensor networks," *Communications of the ACM*, pp. 102–114, Aug. 2002.
2. W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," *Proceedings of the 33rd IEEE Hawaii International Conference on System Sciences (HICSS)*, pp. 1–10, Jan. 2000.
3. C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: A scalable and robust communication paradigm for sensor networks," *Proceedings of ACM MobiCom'00, Boston, MA*, pp. 56–67, Aug. 2000.
4. J. Kulik, W. R. Heinzelman, and H. Balakrishnan, "Negotiation-based protocols for disseminating information in wireless sensor networks," *Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'99)*, 1999.
5. M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman, San Francisco, 1979.
6. B. Krishnamachari, D. Estrin, and S. Wicker, "Modeling data-centric routing in wireless sensor networks," *IEEE INFOCOM*, June 2002.
7. D. L. Hall, *Mathematical Techniques in Multi-Sensor Data Fusion*. Artech House, 1992.
8. J. Heidemann, F. Silva, Y. Yu, D. Estrin, and P. Haldar, *Diffusion Filters as a Flexible Architecture for Event Notification in Wireless Sensor Networks*. USC/ISI Technical Report 2002-556, 2002.
9. <http://www.isi.edu/nsnam/ns>, *The Network Simulator NS-2*.
10. H. Wu, R. S. Mel Siegel, and J. Yang, "Sensor fusion using dempster-shafer theory," *In Proceedings of the IEE Instrumentation and Measurement Technology Conference, Anchorage, AK, USA*, May 2002.