On the Effects of Aggregation on Reliability in Sensor Networks

Jonathan P. Benson, Utz Roedig, Andre Barroso, Cormac J. Sreenan.
Mobile and Internet Systems Laboratory (MISL), University College Cork (UCC), Ireland

Abstract— **Data collected in a sensor network is transported hop-by-hop to a sink for further analysis. The quality of the analysis depends on the amount of data reaching the sink. Hence, data transport reliability influences the quality of the analysis. Data aggregation is a common method used in sensor networks to reduce the amount of messages transported. By aggregating, the data contained in several messages is fused into one single message. Therefore, data aggregation significantly influences the overall data transport reliability observed at the sink. This influence is analyzed and described analytically and by experiment within this paper. Furthermore it is shown how the influence of data aggregation on data transport reliability can be controlled for a particular class of data gathering application.**

I. Introduction[1]

Many wireless sensor network (WSN) applications collect periodically generated sensor data at a central point - the data sink or base station - where the data is subsequently analyzed. This class of applications is considered within this paper.

In a realistic deployment scenario, messages are lost in transport while traveling hop-by-hop through the network towards the sink. These packet losses happen due to the natural lossy characteristics of the wireless links between the sensor nodes. An application analyzing the data might be able to deal with some of these losses. More specifically, the application might be able to infer the correct conclusions even if a (small) portion of the sensor readings is not available for the analysis (due to the redundancy in sensor networks).

Using data aggregation, several messages transported along the same path can be combined into one single message. Aggregation techniques reduce the amount of messages and thus reduce energy expensive transceiver operation and help to preserve scarce bandwidth. As aggregation increases the amount of data concentrated in a single message, the data reliability at the sink is altered. Losing a message containing a single data reading has surely a different impact on the overall data reliability than losing a message containing the information of several sensor readings. Thus, the goal of this paper is to *describe* and *control* the influence of data aggregation on data reliability. A predictable and controllable interaction between data aggregation and data reliability is necessary for sensor network applications that need assurances of performance.

The remainder of the paper is organized as follows. Section III describes formally the interdependency between data aggregation and data reliability. Section IV shows how the influence of data aggregation on data reliability can be controlled. This control mechanism is verified by simulation in Section V. Section II describes related work and Section VI concludes the paper.

II. Related Work

The related work section is split into two parts. First, related work on data aggregation in sensor networks is discussed. Second, existing work that describes methods to control the end-to-end reliability is presented. End-to-end reliability control is the method proposed in Section IV to counter the problem of variable link reliability and path length; thus it is important to show that appropriate technical implementations exist.

Aggregation: Several papers address the issue of aggregation in sensor networks. These papers vary in their approaches and emphasis.

A common approach is to abstract aggregation from the underlying network operation by implementing a SQL like query layer which a programmer or end user can use to pose queries to the sensor network [1], [2], [3], [4]. This form of aggregation is not related to the problem discussed within the paper.

Reliability Control: Ensuring reliable delivery in sensor networks has been the focus of a number of research papers.

Several papers advocate the use of acknowledgments (ACKs) or negative acknowledgments (NACKs) and the subsequent retransmission of a lost message [7], [10]. Another approach is to forward a message more than once so that its reliability is increased [9], [10], [11]. A more complex method involves forwarding multiple packets along multiple disjoint paths [9], [10], [11].

[12] is closely related to this work and describes, in general, some methods that may be used to evaluate the informational value of sensor data. Various informational values are then mapped to various protection measures, FECs in this case. The principal difference between [12] and this paper is that this paper presents a formal link between data and the reliability needed for a given application scenario. [12] does not calculate the required reliability for an aggregate and does not take into account the number of hops to the data sink.

III. Aggregation - Reliability Interdependency

This Section defines the terms aggregation and data transport reliability. Subsequently, the interdependency between data transport reliability and data aggregation is investigated.

[1] The support of the Informatics Commercialisation Initiative of Enterprise Ireland is gratefully acknowledged.

A. Aggregation

The term data aggregation, sometimes also referred as message aggregation, can be applied to a range of different operations taking place inside a network. For the purposes of this study, a valid aggregation function ϕ is defined as follows:

Definition 1: An aggregation function ϕ maps several messages to a single message. Formally, if M is the set of all possible messages transmitted, this can be expressed as: $\phi : M^a \rightarrow M \; \forall a \geq 2$.

Data aggregation is used in sensor networks for several reasons. The main objective of data aggregation is the reduction of energy consumption. Energy and bandwidth is saved as less messages, normally containing a smaller payload than the unaggregated messages together, have to be forwarded.

Aggregation can be performed on a packet level by combing the payload of several messages in a single message or on an application level by applying operators such as *COUNT, AVG, SUM* on the data carried by two or more messages. The particular aggregation operation used in the system is irrelevant to the results presented in this paper, provided the properties stated above are satisfied.

B. Reliability

In this paper, it is assumed that sensor data readings are transported towards a sink. It is also assumed that sensors send data readings periodically. Thus, the amount of data readings generated per time interval in the sensor field is known. It is assumed that all sensor samples are considered to be equally valuable. Additionally it is assumed that packet losses are independent of additional factors such as message size and traffic density. Increases in message size will increase the probability of bit errors within a message, but, note that in many forms of aggregation, packet size remains unchanged or does not change significantly. Also, while aggregation changes the traffic and thus reduces the likelihood of MAC layer collisions, note that reductions in traffic are countered by data converging on the data sink having the opposite effect. We believe that these assumptions still result in a reasonably accurate model that can be used for the study described in the paper. Using the assumptions, the reliability on the different abstraction levels is given by the following three definitions:

Definition 2: The *hop-by-hop message transport reliability* (short: hop-by-hop reliability), $r_{i,j}$, describes the probability that a message is delivered successfully between two neighbouring sensor nodes i and j.

Definition 3: The *end-to-end message transport reliability* (short: end-to-end reliability), r, is described by the product of the message transport reliabilities $r_{i,j}$ on the path from source to sink.

Definition 4: The *data transport reliability* (short: data reliability) is described by the expected amount of sensor readings $E(X)$ per unit time reaching the sink and also by the variance σ^2. The variance describes fluctuations about the expected value.

C. Interdependency

The data reliability, characterised by $E(X)$ and σ^2, is influenced by the amount of data lost in transit. These losses are characterised by the hop-by-hop reliability of each link and the degree of aggregation. The degree of aggregation, a, influences how many data readings are lost by losing a single message.

Consider a line of nodes where the topmost node is the data sink and the bottommost node has a number of N data readings to send. The readings can now either be sent unaggregated as N messages, each containing a single sensor reading, or aggregated in $n \leq N$ messages depending on the selected aggregation degree. The value $1 \leq a \leq N$ describes how many readings are combined in each message. Thus it is assumed that all messages carry the same number of a sensor readings (homogeneous aggregation). Note that the assumption of homogeneous aggregation has no net effect on the expected value calculations and gives a worst case variance calculation for a maximum aggregation level a. As a result of the aggregation, the following number of messages are sent to the sink:

$$n = N/a \tag{1}$$

1) Expected Values : The question here is how aggregation influences the expected value $E(X)$. The expected value can be calculated by:

$$E(X) = \sum^{n} a \cdot r = n \cdot a \cdot r \tag{2}$$

If we use Equation (1) and substitute the value of a with N/n we obtain:

$$E(X) = Nr \tag{3}$$

Thus, the expected value is a function of the number of sensor data N and the end-to-end reliability r. The degree of aggregation a has no effect on the expected value. It seems therefore logical to aggregate as much as possible as no cost regarding data transport reliability, in terms of the expected value, has to be paid. In the literature [12] it is assumed that aggregated packets have to be handled with greater care than non aggregated ones. As shown, this is not true regarding the expected value of the amount of data readings.

2) Variance: The variance gives an impression of the fluctuations of the amount of data readings reaching the sink. The variance σ^2 is given by the formula:

$$\sigma^2 = E(X^2) - [E(X)]^2 \tag{4}$$

The variance can now be calculated and using Equation (1) we obtain:

$$\sigma^2 = \sum^{n} (a^2 \cdot r) - (a^2 \cdot r^2) = N \cdot a \cdot r \cdot (1 - r) \tag{5}$$

Here, the variance depends linearly on the degree of aggregation and linearly on the number of samples. Now both extremes can be compared; no aggregation with $a = 1$ and total aggregation with $a = N$. In the first case, the variance depends linearly on the amount of sensor readings. In the second case, the variance depends quadratically on the amount of sensor readings sent. It can be concluded that the variance of amount of data readings per time unit reaching the sink depends heavily on the degree of aggregation. Regarding the variance it is therefore useful to handle aggregated packets with greater care than non aggregated ones. More specifically, in applications with a strong time correlation, control of the variance σ^2 is of critical importance whereas in applications that can deal with losses as long as the long term average is above a certain point need only be concerned with $E(X)$.

IV. Aggregation - Reliability Control

In this Section, the control goal is formulated along application requirements. Thereafter the control mechanism and its implementation is presented.

A. Application Requirements

An application requires a data transport reliability above a given value to function correctly. Mathematically expressed, it is required that $E(X) \geq N \cdot R$. Here, R is the reliability level desired by the application, N is the total number of sensor data. Additionally, it has now to be taken into account that the amount of actual data delivered will fluctuate about the expected value, which is described by the variance. We thus define our control goal as:

Definition 5: The application should achieve a transport reliability such that expected value minus some multiple of the standard deviation equals to or is greater than the minimum reliability level desired by the application. This can be expressed as follows: $E(X) - z\sigma = NR$.

For example, if we assume a normal distribution of the incoming sensor readings and $z = 1.96$ is selected, in 97.5% of cases the application requirements can be met.

B. Control Mechanism

As it was shown by Equations (3) and (5), the expected value and variance depend on the aggregation degree a and the end-to-end message reliability r. Thus, aggregation degree a and end-to-end transport reliability r have to be balanced, such that the needs of the application can be met.

Using the application requirements given in Definition 5, we can derive equations that allow us to compute the maximum aggregation degree and/or the necessary transport reliability:

$$E(X) - NR = z\sigma \tag{6}$$

Using Equation (2) and (5) we obtain:

$$n \cdot a \cdot r - N \cdot R = z \cdot \sqrt{N \cdot a \cdot r \cdot (1 - r)} \tag{7}$$

Squaring both sides of Equation (7) gives us:

$$N^2 \cdot (r^2 - 2 \cdot r \cdot R + R^2) = z^2 \cdot N \cdot a \cdot r \cdot (1 - r) \tag{8}$$

To calculate the maximum aggregation degree a if r is already known, Equation (8) can be modified as:

$$a = \frac{N \cdot (r^2 - 2 \cdot r \cdot R + R^2)}{z^2 \cdot r \cdot (1 - r)} \tag{9}$$

Finally we can generate the following equation to compute the end-to-end transport reliability r needed for a given a using Equation (8):

$$(N + z^2 \cdot a) \cdot r^2 - (2 \cdot N \cdot R + z^2 \cdot a) \cdot r + N \cdot R^2 = 0 \tag{10}$$

Equation (9) gives the maximum aggregation degree that can be used in the network if the end-to-end reliability r is known. Equation (10) gives the necessary end-to-end reliability for messages if the aggregation degree is known. Of course, both equations can be used together to balance these values.

C. Reliability Control

Equations (9) and (10), assume that the end-to-end reliability, r, for messages transported in the network is constant for all messages regardless of their distance to the sink. This assumption is difficult to implement in reality as messages will have varying travel distances (hop-count) to the sink. For example, if a constant hop-by-hop reliability is assumed, messages will have a different end-to-end reliability. To deal with this issue, two principle methods exist.

1) Method 1: Worst-Case: The worst possible r that can be encountered in the network can be used for the calculation of $E(X)$ and σ. For example the hop-by-hop message transport reliability r_{ij} and the maximum hop distance h in the field might be known and r can be calculated using Equation (10). However, as many paths will in reality have an end-to-end reliability better than r, the resulting $E(X)$ and σ will be better than calculated. In this case, the Equations (9) and (10) can be used for a worst-case dimensioning.

2) Method 2: Adaptation: A node adapts its forwarding mechanism such that the desired end-to-end reliability r for the message is achieved. If adaptive forwarding mechanisms are in place, the Equations (9) and (10) can be used for a more precise dimensioning.

A node, upon receiving a request from the data sink to generate messages and forward these messages with end-to-end reliability r would need to know its local error rate along with the number of hops h to the data sink. The node could then calculate the reliability r_f at which it would need to forward this message over each hop to meet the end-to-end reliability requirements. To calculate r_f the following simple formula is used: $r_f \geq r^{1/h}$. The value of r_f needs to be forwarded in each packet so that any receiving node is able to calculate what steps it needs to take to ensure that the packet is again forwarded with reliability r_f. Methods to achieve the desired r_f are discussed in the II section. In particular, [9], [10], [11] discuss this in detail.

D. Aggregation Control

Equations (9) and (10), assume that a constant aggregation degree a is used within the network. It is simple to use a as an upper bound for the aggregation degree. However, it might be difficult or impossible to assure that all sensor readings are delivered as messages with a readings. In a realistic operation scenario, some messages delivered to the sink will contain less than the maximum allowed a data readings. As in reality the maximum allowed aggregation degree is not always used, the resulting $E(X)$ and σ will be better than calculated. Here again, the Equations (9) and (10) must be seen as a tool for a worst-case dimensioning.

V. Experimental Evaluation

The first experiment shows that the expected value $E(X)$ does not depend on the aggregation degree and that the variance σ depends on the aggregation degree as it is described analytically in Section III.

The second experiment shows the effect of a dynamic adapted end-to-end reliability on the expected value $E(X)$ and the variance σ. As shown in Section IV, this method narrows the gap between analytical calculation (worst-case) and real-world observation (average-case).

A. General Setup

A simulation was conducted using a purpose built simulator. The results from this simulator were identical to simulations performed on the ns-2 simulator but the simulation speed was several times faster than ns-2.

1) Topology: 100 nodes are placed in a grid. The transmission range is set such that each node can only communicate to their adjacent neighbours in the grid. The topmost central node is designated as the data sink. An interest is flooded by the sink into the network and a routing tree is formed along reverse path. For the purposes of the experiment the routing tree is considered to be stable.

2) Aggregation: A waiting period T_i at node n_i in the routing tree is calculated for each message in order to facilitate a cascading aggregation system using the following formula:

$$T_i = \frac{T_{max}}{h_{max}} \cdot (h_{max} - h_i)$$

When the maximum aggregation degree a is reached, a packet is immediately forwarded to the data sink without further delays en route.

3) Traffic: Every node periodically generates a sensor reading (1 per sensing period) and sends it to the data sink. Before the next period all the data generated is forwarded to the sink and recorded. Each node generates 1000 data readings per simulation. After each period the amount of sensor readings delivered to the sink is recorded. Finally the standard deviation is calculated for the 1000 data gathering rounds. This process was repeated 10 times to account for any variations in the topology caused by the random formation of the routing tree. The standard deviation is calculated based on the 10 experiments and used to generate error bars shown in the resulting graphs.

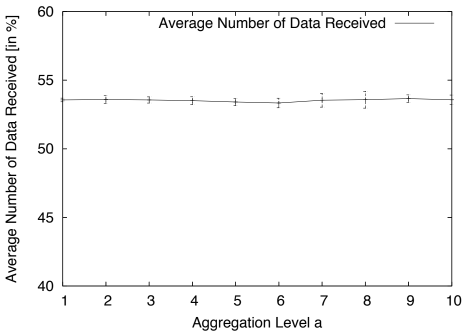

Fig. 1. Average number of data received in percent with hop-by-hop reliability $r_{ij} = 0.9$.

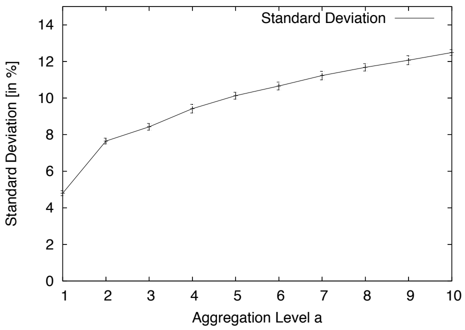

Fig. 2. Standard deviation of the number of data received in percent with hop-by-hop reliability $r_{ij} = 0.9$.

B. Experiment 1

The first experiment demonstrates that aggregation does not affect the expected value but increases the variance and therefore the standard deviation. A hop-by-hop reliability $r_{ij} = 0.9$ was assigned to each link. The maximum possible distance in the routing tree was $h_{max} = 10$. The experiment was conducted for maximum aggregation degrees from $a = 1$ (no aggregation) to $a = 10$. In all cases it is apparent that the average amount of sensor data being delivered does not change and that the variance increases with respect to aggregation (see Fig. 1 and Fig. 2.).

C. Experiment 2

The second experiment makes use of the algorithm described in Section IV to ensure that the end-to-end reliability is the same for each individual message. The following values are used: $R = 0.7$, $z = 1$, $a = 1$,...,$a = 10$. . The desired end-to-end reliability r can be pre-calculated based on Equation (10) and is shown in Fig 4.. Thus, each node can calculate the hop-by-hop forwarding reliability r_f for each message. The calculated r_f is used as hop-by-hop forwarding reliability. The result of the simulation is shown in Fig. 3..

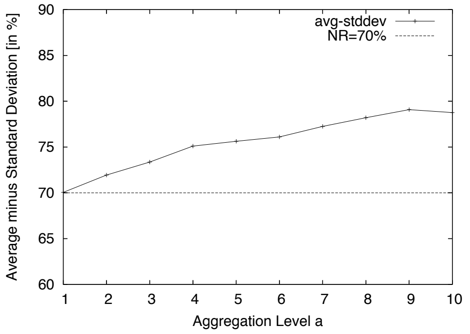

Fig. 3. Calculated $N \cdot R$ compared with the measured $E(X) - z \cdot \sigma$.

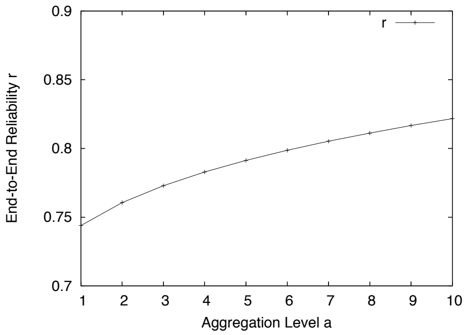

Fig. 4. Calculated end-to-end reliability using Equation (10).

The results in Fig. 3. show that the desired application quality level can be met at all times if the correct $r_f$is achieved. However, the higher the aggregation level selected, the more desired quality level and measured quality level differ.

One cause of the observed difference is founded in the assumption that all packets are sent with the maximum possible aggregation degree a. In reality, messages often contain less sensor readings than allowed and thus the variance of $E(x)$ is not as high as anticipated. For example the highest aggregation degree of $a = 10$ used in the experiment will only be used rarely in a sensor network of 100 nodes.

Another cause of the observed difference is caused by the selection of the hop-by-hop reliability r_f. A sensor reading sent from further away must be forwarded with greater reliability than one sent from nearby. Again we assume a worst case scenario and use the highest forwarding reliability of the constituent sensor readings for the aggregated message.

The Directed Diffusion paradigm is presented in [5]. Data is cached at each node making it possible to perform aggregation. To achieve this end the use of filters is proposed. [6] discusses the use of low level naming such as sensor type and geographic location to eliminate the need for a name binding service and the subsequent communication overhead associated with using such a service.

VI. Conclusion & Future Work

Our results clearly show that aggregation does not affect the probable amount of data delivered but has an adverse effect on the fluctuations about this value. These fluctuations lead to unstable application level quality and are undesirable. Having quantified this effect we have furthermore presented a methodology to determine the correct end-to-end reliability level to control these effects. The principal contributions of this paper is that it describes how aggregation effects application level quality for a class of applications and how these effects may be controlled by selecting and implementing the correct end-to-end reliability.

Our future work will endeavour to move beyond worst case dimensioning and endeavour to create heuristics to enable an aggregate packet to calculate its reliability such that application level quality constraints are met more closely than at present. It is envisioned that such a heuristic would be applied dynamically within the network, increasing reliability as aggregation occurs. Also a more complex methodology is needed to integrate the increases in the overall end-to-end reliability caused by obeying the constraints of messages generated further away from the sink (i.e. adopting the highest r_f), and increases in the end-to-end reliability necessitated by increases in the variance. This new methodology must also accomodate data types that are not equally valuable. Finally, further analysis is necessary to assess the effects of “data holes” (losses from the same physical area) caused by the loss of aggregate packets.

References

[1] J. Gehrke, Y. Yao. Query Processing for Sensor Networks. IEEE Pervasive Computing 2004, vol 3, number 1, pages 46-55.

[2] P. Bonnet, J. Gehrke, T. Mayr, P. Seshadri. Query Processing in a Device Database System. Tech. Report, number 99-1775, Cornell University, Ithaca, NY, USA, 1999.

[3] S. Madden, M. J. Franklin, J. M. Hellerstein, W. Hong. TAG: a Tiny AGgregation Service for Ad-Hoc Sensor Networks. Proc. of the 5th Annual Symposium on Operating Systems Design and Implementation, 2002.

[4] J. Beaver, M. A. Sharaf, A. Labrinidis, Panos K. Chrysanthis. Power-Aware In-Network Query Processing for Sensor Data. Proc. of the 2nd Hellenic Data Management Symposium, 2003.

[5] C. Intanagonwiwat, R. Govindan, D. Estrin. Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks. Proc. of the 6th Annual Conference on Mobile Computing and Networks, 2000.

[6] J. Heidemann, F. Silva, C. Intanagonwiwat, R. Govindan, D. Estrin, and D. Ganesan. Building Efficient Wireless Sensor Networks with Low-Level Naming. Proc. of the Symposium on Operating Systems Principles, 2001.

[7] F. Stann and J. Heidemann. RMST: Reliable Data Transport in Sensor Networks. Proceedings of the 1st IEEE International Workshop on Sensor Net Protocols and Applications, 2003.

[8] S. Mukhopadhyay, D. Panigrahi and S. Dey. Data Aware, Low Cost Error Correction for Wireless Sensor Networks. Proc. of IEEE Wireless Communications and Networking Conference, 2004.

[9] S. Bhatnagar, B. Deb and B. Nath. Service Differentiation in Sensor Networks. Proc. of the 4th International Symposium on Wireless Personal Multimedia Communications, 2001.

[10] B. Deb, S. Bhatnagar, B. Nath. Information assurance in sensor networks. Proc. of the 2nd ACM International Conference on Wireless Sensor Networks and Applications, 2003.

[11] B. Deb, S. Bhatnagar and B. Nath. ReInForM: Reliable Information Forwarding using Multiple Paths in Sensor Networks. Proc. of the 28th Annual IEEE Conference on Local Computer Networks, 2003.

[12] A. Kopke, H. Karl and M. Lobbers. Using energy where it counts: Protecting important messages in the link layer. Proc. of the IEEE European Workshop on Wireless Sensor Network, 2005.