

A Study on Data Aggregation and Reliability in Managing Wireless Sensor Networks

S. Brown

*Department of Computer Science
National University of Ireland, Maynooth
stephen.brown@nuim.ie*

C. J. Sreenan

*Mobile & Internet Systems Laboratory,
University College Cork, Ireland
c.sreenan@cs.ucc.ie*

Abstract

Using data aggregation to reduce power consumption is a well studied area. In this paper we show a new result: data aggregation can also be used to increase reliability. In scenarios where status is being collected from every node (e.g. network management), one measure of reliability is the completeness (nodes included). Using some of the power savings from aggregation to increase the retransmission limit for aggregated data gives significant improvements in completeness. The contribution of this paper is in providing a detailed comparison of the completeness and energy efficiency for aggregated and non-aggregated data. Results, derived by numerical analysis and simulation, show that with data aggregation and increased retransmission limits, reliability can be substantially improved, while still saving power. These results are applicable to designing network management protocols for Wireless Sensor Networks, as well as for other scenarios where high completeness is required for network-wide data collection.

1. Introduction

Determining the collective status of all the nodes in a WSN (wireless sensor network) is a fundamental building block for network management. For large scale networks, the individual management of nodes is impractical (and inefficient). The motivation for the work presented here is the monitoring of node software update status as part of the software update process, as discussed in [1] and [2], but the results presented here are applicable to other network-wide

management and data-collection activities requiring near-100% completeness.

In this paper we investigate the problem of reliably collecting status information from all the nodes in a WSN based on a request from a management station. This form of status collection differs from sensor event collection in several ways: it does not inherently require a flow of data from each node; it is infrequent; it is not triggered by a sensor event; it requires a result that is collected from near-100% of the nodes; and it needs to work on a wide range of WSNs (thus implying the need for scalability).

There is a lack of extensive studies on the relationship between data aggregation and reliability (as measured by completeness) despite the importance of aggregation for collecting data in Wireless Sensor Networks. The contribution of this work is to provide a detailed study, allowing various design tradeoffs to be clearly identified.

2. Related Work

Data Aggregation is a key mechanism for WSNs, and reduces energy use through reduced transmissions by processing raw data en route and delivering just the results [3]. Efficiency issues are addressed in SPIN [4] which uses meta-data communications to eliminate redundant data transmission, and LEACH [5] is a routing protocol for low energy, hierarchical clusters, aggregating data and transmitting it directly to the base station. In [6] a new aggregation quality metric DAQ is proposed, and used to propose two new protocols for aggregation: Enhanced LEACH and Clustered PEGASIS, where data aggregation takes place in the cluster heads.

In [7] it is shown that data distribution and collection tasks can be performed optimally on tree networks. In [8] the performance of a greedy

Supported by SFI/Tyndall National Institute under grant NAP-18

algorithm to build trees in networks of differing densities is evaluated. Our work does not examine the spanning-tree generation or request phases, as supported for example in Directed Diffusion [9], SNMS/Drip [10], and REAR[11].

TAG [12] introduces the idea of aggregation as a high-level, application-independent service to realize these performance benefits. It also introduces the idea of an aggregation function. This is developed further in [13], which extends the classes of data aggregation from simple types such as SUM and COUNT to include MEDIAN, Consensus values, Histograms, and ranges.

Retransmissions can be used to improve reliability: as shown for PSFQ in [14], hop-by-hop error recovery is significantly more effective than end-to-end recovery in a wireless environment. For a WSN environment, data aggregation does not effect reliability in terms of the number of reports received at a root node from a source (the expected value) when compared to the non-data aggregation case [15]. But it does increase the variability.

3. Completeness

We define completeness to be a measure of the reliability of the response to a query made to multiple nodes. It is defined as the ratio of the actual number of responses received to the expected number of responses.

$$C(A, E) = \frac{|A|}{|E|}$$

where:

$$\begin{aligned} A &= \{x \mid \text{a response from node } x \text{ is received}\} \\ E &= \{x \mid \text{a response from node } x \text{ is expected}\} \\ A &\subseteq E \end{aligned}$$

The completeness depends on the effectiveness of two factors: the query propagation, and the response delivery. In this paper we address the impact of data aggregation on the response delivery effectiveness.

The completeness of a network-wide management response depends on the end-to-end reliability of each node in the WSN. There are two way to try and ensure maximum end-to-end reliability in a WSN: sending the same datum multiple times in a stream, thus increasing the probability that at least one of the instances will be received [9][10]; and sending the datum once (“one-shot”) with acknowledgements and local re-transmissions [14] to improve the probability that an unduplicated data instance will be received. As we show in our results, a “one-shot” response results in significantly less traffic.

In the examples addressed here, we consider a WSN scenario where all the nodes report “aggregatable” data to a root node. By ‘aggregatable’, we mean data that allows for the aggregation of multiple values from multiple nodes without increasing the space requirements in the packet(s) used to transfer the data: it is thus scalable. Network-wide status information such as SUM, COUNT, minima, and maxima would fall into this category [13]. A network-management example, based on the original motivation for this study, would be in determining the number of nodes that are running a new software version following the activation of a software update [1]. The completeness of an aggregated report measures the number of responses incorporated into that report.

4. Analysis

The parameters used are defined as follows:

- a: aggregation degree in a packet (response count)
- d: spanning-tree depth (in hops from the sink)
- f: spanning-tree fan-out (number of children)
- n: number of reporting nodes
- p: per-link probability of single transmission success
- q: per-link probability of a packet being received successfully with retransmissions
- t: transmit limit (for non-aggregated data) – the maximum number of retransmissions=(t-1)
- u: modified transmit limit for aggregated data (see the policies defined in section 5)
- v: the actual number of transmissions used on a link (including data & acks)

For a set of responses sent up a tree from a node (Fig. 1A) the number of reports received at the sink is the same whether each report is sent individually, or all are aggregated into a single packet [15].

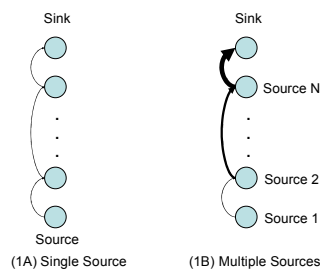


Figure 1. Single & multiple data sources

We now examine the situation where every node in the tree is a source (Fig. 1B). There are two cases: each node sends it own report, or each node

aggregates its own data as the report from Source 1 propagates up the tree.

4.1. Non aggregated case

$$\Pr(\text{receive from source at depth } i) = q^i$$

$$|A| = \sum_{i=1}^n q^i$$

$$\begin{aligned} \text{Packets} &= v + (v + qv) + (v + qv + q^2v) \dots - n \text{ terms} \\ &= v(n + (n-1)q + (n-2)q^2 + \dots) - n \text{ terms} \end{aligned}$$

$$\Pr(100\% \text{ success}) = q^n * q^{n-1} * q^{n-2} \dots * q = q^{n(n+1)/2}$$

4.2. Aggregated case

$$|A| = q(1 + q(1 + q(1 + q(\dots)))) = \sum_{i=1}^n q^i$$

$$\begin{aligned} \text{Packets} &= v + v + \dots + v - n \text{ terms} \\ &= vn \end{aligned}$$

$$\Pr(100\% \text{ success}) = q^n$$

Thus the aggregated case provides the same average completeness, with a higher probability of 100% completion, using fewer packets. Thus we can improve the completeness by using some of the saved packets for extra retransmissions. Due to the complexity of the equations we explore this numerically.

5. Numerical analysis

In the analysis shown below, figures of $p=0.7$ and $t=3$ are used, giving a base per-link packet success rate (q) of 0.973 (with retransmissions).

The results for a balanced spanning-tree of 258 nodes are shown in Table 1 and Fig. 2. These show the improvement in the completeness of the response for five different aggregated transmit limit policies:

- non-aggregated data (transmit limit t)
- aggregated data (modified transmit limit $u=t$)
- aggregated data ($u=2t$)
- aggregated data ($u=at$)
- aggregated data ($u=2at$)
- aggregated data ($u=4at$)
- aggregated data ($u=8at$)

In this analysis we are using packets transmitted to compare power consumption: typical WSN radio receivers (e.g. CC2420) use the same amount of power in receive mode whether actually receiving or not. We also assume contention is avoided by the MAC layer – in practice, reducing the number of transmissions will reduce MAC-layer contention [9] and thus increase the power efficiency.

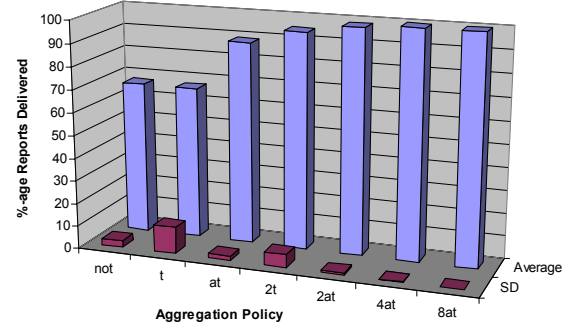


Figure 2.
Completeness vs aggregation policy
($d=3, f=6, t=3, p=0.7, n=258$)

Note that the variability, shown by the standard deviation (SD) for aggregated data ($u=at$) is similar to that for non-aggregated data. Note also, that as the maximum transmit limit increases (from at to $8at$) the average number of reports delivered approaches 100%. The detail in Table 1 shows that the energy used increases slightly as u increases, but is significantly less than for non-aggregated data. The energy per report is also significantly lower.

Data aggregation has the maximum effect for a very deep spanning-tree, and minimum effect for a very shallow one. Fig. 3 shows the effect on the completeness of varying the height and fan-out of the spanning-tree. The total number of reporting nodes is kept at approximately 256 – as a balanced spanning tree is used it is not possible to keep this number exactly constant, and the reports received is normalized in the range (0..1).

Table 1. Completeness for different policies
($d=3, f=6, t=3, p=0.7, n=258$)

Policy	Reports Avg	Reports SD	Total-tx
Not agg	173.172	7.523	1923
$u=t$	172.738	30.771	776
$u=2t$	245.206	14.803	879
$u=at$	229.292	5.023	796
$u=2at$	254.19	1.941	882
$u=4at$	257.932	0.26	895
$u=8at$	258	0.01	895

With the selected parameters ($p=0.7$, transmit-limit=3), the non-aggregated data case performs very badly for a very deep spanning tree, and performs as well as the $u=t$ aggregated data case for a very shallow tree. The $u=2at$ policy provides good completeness; the $u=4at$ policy provides near 100% completeness. In

Fig. 4 the energy cost of each policy is shown (packets transmitted).

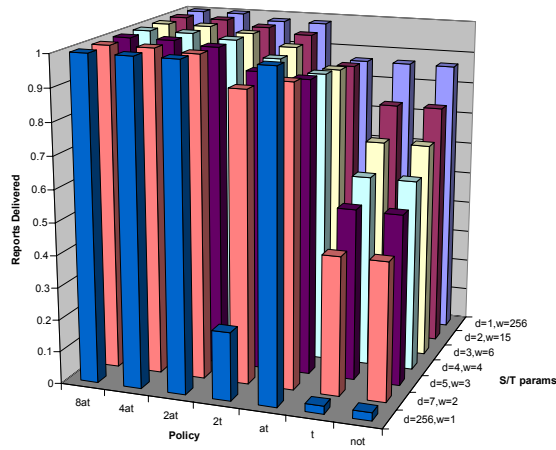


Figure 3. Effect of tree width and depth

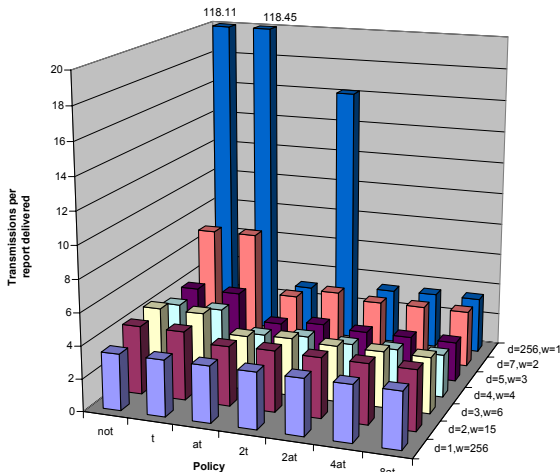


Figure 4. Energy efficiency of different policies (for clarity, the policy axis is inverted wrt Fig. 3)

Note that in all cases, except for the single-hop spanning tree, the non-aggregated cost is significantly greater than the aggregated data cost. As the policy gradually increases the transmit limit, the energy cost rises slightly for improvements in completeness.

Fig. 5 shows the effect of varying the probability of packet loss between nodes “p” on the completeness.

When the packet loss rate is small, then the completeness ratio becomes independent of the policy. But for larger packet loss rates (smaller values of p), typically associated with wireless communication, the quality of the response is significantly improved using aggregation with increased transmission limits.

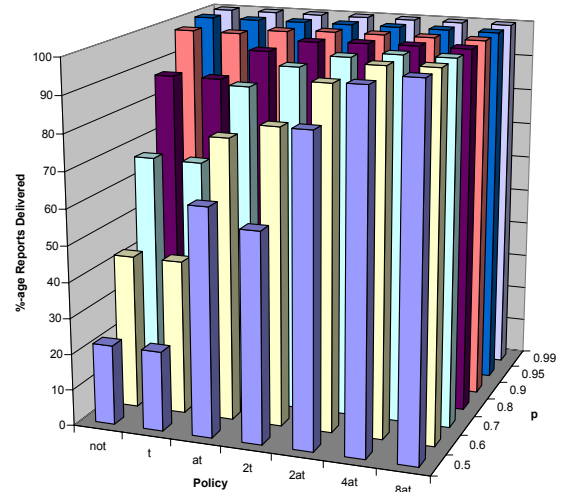


Figure 5. Completeness for varying “p”

The significantly reduced energy usage, as measured by the number of packets transmissions, allows the transmission limit to be increased in the aggregated data case without a severe power cost (see Figure 6). This results in improved reliability (completeness), and reduced variability. The results of using different policies to modify the transmission limit for aggregated data are shown below.

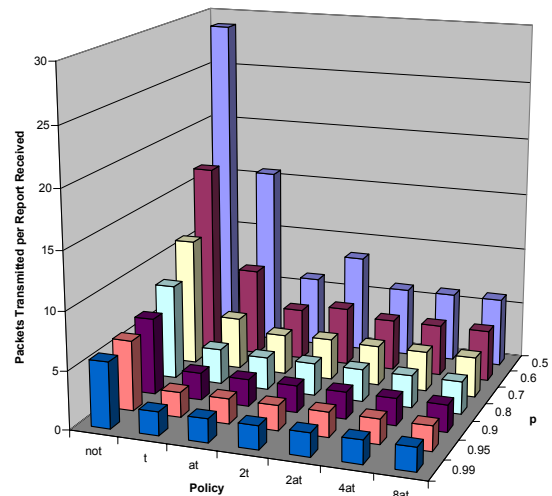


Figure 6. Energy efficiency for varying “p” (note: inverted ‘p’ axis wrt Fig. 3)

The results show that increasing the transmission limit for aggregated data leads to increased reliability, while losing little of the energy gains. The analysis shows that near 100% completeness can be achieved using data aggregation and a weighted transmission limit. For a wide range of network configurations and values of p_{ij} , the policy u=4at seems to be the best balance between energy use and reliability.

5.1. Large numbers of retransmissions

Large numbers of retransmissions lead to:

1. increased latency of responses, and
2. increased time to respond to a node failure.

These problems associated with this can be addressed in a number of ways:

- o limiting the maximum value of retransmissions,
- o accepting the increased latency as the cost of high-completeness, or
- o running spanning-tree maintenance in parallel with re-transmissions (using a threshold less than the transmit limit to initiate S/T repair).

In practice, there is little point in increasing the retransmit limit too far: for example, if $p=0.7$, a transmit limit of 8 will provide a packet success rate with retransmissions (q) of 99.99%. Fig. 7 shows the effect of limiting the maximum number of transmissions to partially address this issue. A maximum transmit limit of 8 was found to be effective for a 250 node network with $p=70\%$ - this result is valid except for very deep networks (>32 hops). The transmit limit was set to a maximum of 64 (the largest value actually attained was 28).

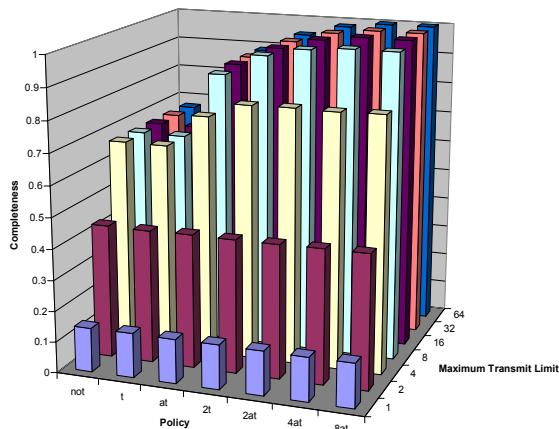


Figure 7. Completeness vs policy for different transmit limit maxima

6. Simulation results

The results achieved here were verified using the TinyOS2/TOSSIM simulator (See TinyOS 2.0 distribution for details, at <http://www.tinyos.net>) with a 225-node network and the standard “15-15-medium-mica2-grid.txt” network configuration file provided. The details are not crucial to these results, as spanning

trees were created from the network configuration by selecting links with the required link qualities (‘ p ’).

Duplicate data is avoided by using the standard Alternative Bit Protocol (ABP). This requires a minimum amount of data to be kept per neighbour. An alternative policy using 3-way handshaking was also examined: at the cost of an extra packet (an ack-ack) per data packet, the memory requirement is reduced to keeping details on a single neighbour conversation at any one time. The results are not shown here, but they provide slightly lower efficiency and completeness results in exchange for increased scalability. The simulation results also include data streaming, where the same data value is sent multiple times to the sink, as used for example in [10]. We include results with & without local acknowledgements and retransmissions.

Space does not allow presentation of the full simulation results, so representative results are shown in figures 9-12. Two different spanning trees, built offline, are used: one is a ‘deep’ spanning tree (max depth=37, root children=1); the other is a ‘broad’ spanning tree, representative of those produced by algorithms such as [d] (max depth=8, root children=13). In creating the spanning trees, ranges of link strengths & noise levels were selected to try and maintain an average receive probability of c. 70%, based on prior measurements of packet reception rates. The base transmission limit is set to 3 (allowing for 2 retransmissions). In the streaming data cases, the number of data values to send is set to 3.

Fig. 8 shows the completion rates for the two spanning trees used.

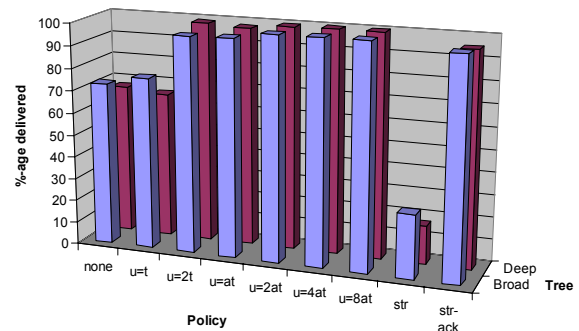


Figure 8. Completion rates for different trees

Note that neither tree is as ‘extreme’ as those shown for Numerical Analysis (Fig. 3), so the results are closer to each other. Fig. 9 shows the energy efficiency. Note the inclusion of non-aggregated, streaming data. The high number of duplicates delivered reduces its efficiency significantly. The aggregated policies all have much higher efficiencies (for an average $p=0.7$).

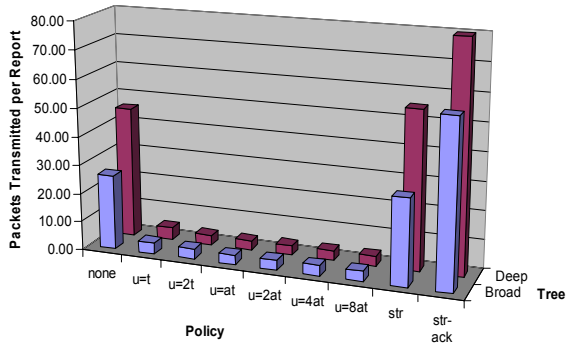


Figure 9. Energy efficiency for different trees ($p=0.7$)

Figs. 10 and 11 show the effect of varying the link quality (p) on the completion rate and energy efficiency of different aggregation policies. Larger u value policies achieve near 100% completion for weaker links.

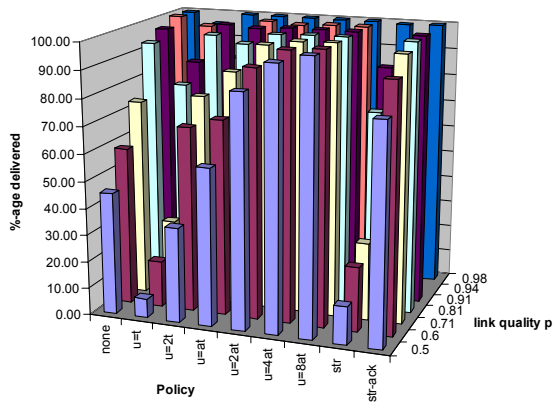


Figure 10. Effect of p on the completion rate (broad spanning tree)

Aggregated data performs much better from an energy viewpoint. Note that the $u=t$ policy loses highly aggregated packets at low link quality values (Fig. 11). Streaming data without acknowledgements leads to a low completion rate at a large cost. Streaming with acknowledgements provides a higher completion rate, but at a cost due to duplication of reports to the root.

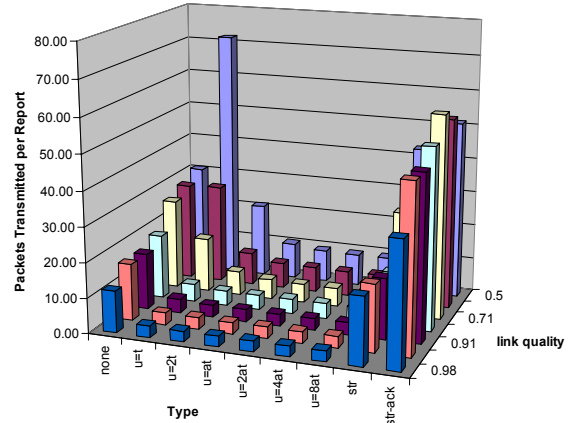


Figure 11. effect of p on the energy efficiency (broad spanning tree, link quality axis inverted)

Fig. 12 shows the impact of asserting a maximum on the transmit limit (u) for the broad spanning tree.

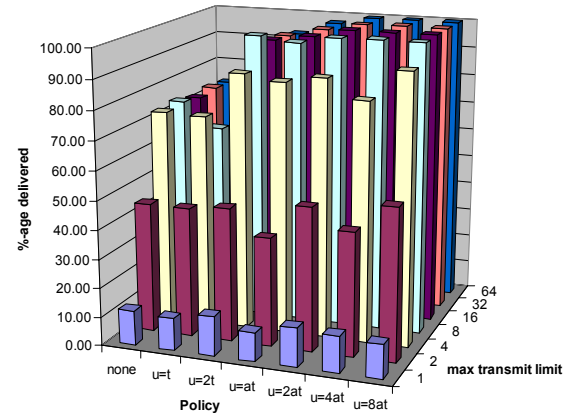


Figure 12. Effect of a maximum transmit limit

In summary, as shown in both the numerical analysis, and the simulation results, data aggregation can significantly reduce the power used in reporting collective, network-wide status to the root of a spanning tree (especially for nodes nearer the root). The reliability, as measured by the completeness, is also significantly improved by using data aggregation and an increased retransmission limit for aggregated data packets. Data streaming (with local acknowledgements) can increase the completion rate, but at a significant energy cost, and resulting in the delivery of many duplicates to the root (not shown here). Increasing the transmission limit for non-aggregated data improves the completion rate somewhat, but again at a very significant energy cost. The improvements provided by aggregation are greater for deeper spanning trees.

12. Conclusion

The completeness of network-wide, management responses in wireless sensor networks can be significantly improved by aggregating all the responses, and using retransmissions with an aggregation-proportional transmission limit. This results in a high probability of near-100% completeness of response, while still realizing significant power savings when compared to the non-aggregated case. Streaming the same data value multiple times without aggregation is particularly inefficient, as the source sends duplicate data values even after successful reception at the root.

Increasing the transmission limit can lead to increased delay in reacting to node failures. This can be limited by setting a maximum value for the transmit limit. Effective values for the limit depend on the reception success rates for each link in the spanning tree; this emphasizes the importance of using link quality metrics when building the tree [10].

Addition benefits can be achieved by using data aggregation and aggregation-dependent transmit limits. The reduction in data traffic means that the network management operation will have a reduced impact on normal WSN data collection activities. Also, though not explored here, the reduction in packets provides an opportunity to reduce the time required to complete the collection of network-wide status data.

Future work is to explore dynamic spanning tree generation policies and determine their impact on the completeness of the aggregated response. We also plan to implement the data aggregation approaches examined here in a real-world sensor network, and validate the completeness and energy results.

13. References

- [1] S. Brown and C.J. Sreenan, "A New Model for Updating Software in Wireless Sensor Networks", *IEEE Network*, IEEE, Nov-Dec 2006, pp. 42-47.
- [2] C.-C. Han, R. Kumar, R. Shea, M. Shrivastava, "Sensor network software update management: a survey", *International Journal of Network Management*, Wiley, July/Aug. 2005, pp. 283-294.
- [3] J. Heidemann, F. Silva, C. Intanagonwiwat, R. Govindan, D. Estrin, and D. Ganesan, "Building efficient wireless sensor networks with low-level naming", in *Proc. 18th ACM Symposium on Operating Systems Principles*, ACM, 2001, pp. 146-159.
- [4] W.R. Heinzelman, J. Kulik, and H. Balakrishnan, "Adaptive Protocols for Information Dissemination in Wireless Sensor Networks", in *Proc. 5th annual ACM/IEEE Intl. Conf. on Mobile computing and networking*, ACM, 1999, pp. 174-185.
- [5] W.R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks", in *Proc. 33rd Annual Hawaii Intl. Conf. on System Sciences*, IEEE, 2000, 10 pp. vol 2
- [6] T. Pham, E.J. Kim, and M. Moh, "On Data Aggregation Quality and Energy Efficiency of Wireless Sensor Networks Protocols – Extended Summary", in *Proc. 1st Intl. Conf. on Broadband Networks*, IEEE, 2004, pp. 730-732
- [7] C. Florens, and R. McEliece, "Packets distribution algorithms for sensor networks", in *Proc. 22nd Annual Joint Conf. of the IEEE Computer and Communications Societies*, IEEE, 2003, pp. 1063-1072 vol. 2
- [8] C. Intanagonwiwat, D. Estrin, R. Govindan and J. Heidemann, "Impact of Network Density on Data Aggregation in Wireless Sensor Networks", in *Proc. Proceedings of the 22nd Intl. Conf. on Distributed Computing Systems*, IEEE, 2002, pp. 457-458
- [9] C. Intanagonwiwat, R. Govindan and D. Estrin, "Directed diffusion: scalable and robust communication paradigm for sensor networks," in *Proc. 6th Annual Intl. Conf. on Mobile Computing and Networking*, ACM, 2000, pp. 56-67
- [10] G. Tolle and D. Collier, "Design of an Application-Cooperative Management System for Wireless Sensor Networks", in *Proc. 2nd European Conf. On Wireless Sensor Networks*, IEEE, 2005, pp. 121-132
- [11] H. Hassanein and J. Luo, "Reliable Energy Aware Routing in Wireless Sensor Networks", in *Proc. 2nd IEEE Workshop in Dependability and Security in Sensor Networks and Systems*, IEEE, 2006, pp. 54-64
- [12] S. Madden, M.J. Franklin, and M.J. Hellerstein, "TAG: a Tiny AGgregation Service for Ad-Hoc Sensor Networks", in *Proc. 5th Symposium on Operating Systems Design and Implementation, ACM SIGOPS Operating Systems Review*, ACM, Dec. 2002 (36/SI), pp. 131-146
- [13] N. Shrivastava, C. Buragohain, D. Agrawal and S. Suri, "Medians and Beyond: New Aggregation Techniques for Sensor Networks", in *Proc. 2nd Intl. Conf. on Embedded networked sensor systems*, ACM, 2004, pp. 239-249
- [14] C-Y Wan, A.T. Campbell, L. Krishnamurthy, "PSFQ: A Reliable Transport Protocol for Wireless Sensor Networks", in *Proc. 1st Intl. workshop on Wireless sensor networks and applications*, ACM, 2002, pp. 1-11
- [15] J.P. Benson, U. Roedig, A. Barroso and S.J. Sreenan, "On the Effects of Aggregation on Reliability in Sensor Networks", in *Proc. 65th Vehicular Technology Conference*, IEEE, 2007, pp. 145-149