On Modeling the Reliability of Data Transport in Wireless Sensor Networks*

Faisal Karim Shaikh, Abdelmajid Khelil and Neeraj Suri
Department of Computer Science,
Technische Universität Darmstadt, Germany
{fkarim|khelil|suri}@informatik.tu-darmstadt.de

Abstract

Data transport is a core function for Wireless Sensor Networks (WSNs) with different applications having varied requirements on the reliability and timeliness of data delivery. While node redundancy, inherent in WSNs, increases the fault tolerance, no guarantees on reliability levels can be assured. Furthermore, the frequent failures within WSNs impact the observed reliability over time and make it more challenging to achieve the desired reliability. Unfortunately, a framework for modeling reliability of data transport protocols in WSNs is currently missing. The existence of such a framework would simplify evaluation, comparison and also adaptation of these protocols. We formulate the problem of data transport in a WSN as a set of operations carried out on raw data. The operations aim at filtering the raw data to streamline its reliable transport towards the sink. Based on this formulation we systematically define a reliability framework. This paper argues for the usefulness of the reliability framework by classifying existing transport protocols and comparing their reliability.

1. Introduction

Wireless Sensor Networks (WSNs) constitute a rapidly growing research area covering both a wide variety of devices and applications. Typical applications involve tracking or monitoring as (a) either statically as embedded sensors or (b) dynamically as mobile (semi) autonomous entities. Correspondingly, applications such as monitoring of traffic, disaster scenarios or target detection are seeing increased use of WSNs. Empirically the core function for a WSN is to collect data from the environment and transport it to a gateway node termed as *sink*. The general data collection and dissemination process involves the flow of the *raw data* from source nodes towards the sink.

Typically a WSN comprises of a large number of sensor nodes possessing limited processing and power capabilities, often communicating over unreliable and low bandwidth radio links [3]. Consequently, this resource constrained environment is also subject to frequent node and communication failures. However, the utility of a WSN based application arises from delivering reliable services, necessitating the incorporation of fault tolerance techniques. A common approach to provide fault tolerance in WSNs is using node redundancy. However, this approach is not sufficient to fulfill the requirements of the application. Users are interested in detecting a targeted *phenomenon* (fire detection, tracking) with a certain quality, e.g., they may require no false negatives with or without tolerating false positives. Thus, the desired *responsiveness*, i.e., reliability and timeliness of data transport often varies for different applications. In extreme cases, there are applications that may require limited responsiveness such as habitat monitoring, and others that require high responsiveness such as military applications. Other intermediate responsiveness classes can be identified such as applications that do not require high delivery reliability [6] but require delivery timeliness, i.e., if some data is lost the application performance will not degrade but data should reach within time bounds specified by the application.

One possible solution for reporting the phenomenon is to *flood* the raw data. Flooding of (bursty) raw data causes broadcast storms, which can result in more failures such as collisions, contention and power depletion. Subsequently, timeliness can not be assured. A established solution to this problem is *convergecast* [14], guide the flood of raw data in the direction of the sink. This can substantially increase the responsiveness of the WSN, however the large volume of data increases the probability of failures and consequently, decreases the overall reliability. It is shown that *in-network* processing is an optimization that further reduces the redundancy of data [5], resulting in fewer collisions and less contention, as well as enhanced responsiveness.

Consider the example of fire detection. The application requires the fire to be detected and reported within speci-

*This research is supported in part by MUET, FP6 IP DECOS, NoE ReSIST and DFG GRK 1362 (TUD GK).

fied time bounds. For this scenario we assume that sensor nodes send ”fire” message to all neighbors as soon as the sensed temperature exceeds a given threshold. If a node receives a certain number of ”fire” messages (for instance two), it sends a ”fire” message to the sink using unicast. If in-network processing operation for fire detection is not reliable and some ”fire” messages are lost due to collisions (Figure 1a), application performance degrades due to non-reporting of fire to the sink. Similarly, if the in-network processing is reliable but the unicast is unreliable or the ”fire” message does not reach the sink in time, the performance of the application drops (Figure 1b). Therefore, in order to model the reliability of data transport all the operations on data starting from its generation (dissemination, aggregation, unicast) need to be considered.

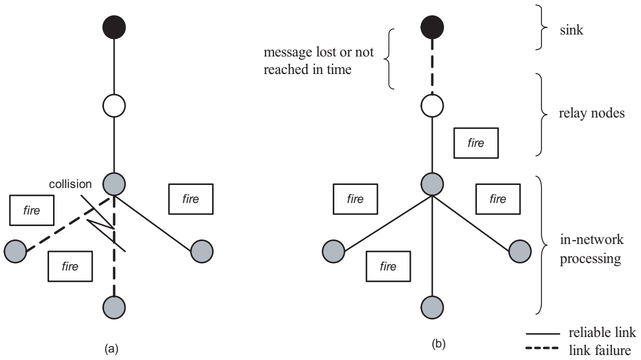

Figure 1. Fire Detection Scenario

This paper targets the following specific objectives. In order to setup the WSN responsiveness requirement inspite of failures, we first develop a WSN fault model along with semantics for data transport and reliability. Next we propose a scheme for the categorization of the existing transport protocols on their operational phases and modules. Following such a classification, we capture these operational phases in the proposed reliable framework that facilitates reliability assessment, relative comparisons and adaptation of data transport protocols. We also show that our reliability model can simplify the online adaptation of data transport protocols.

The paper is organized as follows. Section 2 presents the related work. Section 3 details the proposed generalized system and fault models, along with the existing and our proposed data transport semantic. The proposed reliability framework is presented in Section 4. The reliability comparison across existing transport protocols is conducted in Section 5. Our conclusions and directions for future work appear in Section 6.

2. Related Work

The study of reliable data transport in WSNs has been the subject of extensive research [3, 6, 7, 8, 9, 11, 12, 14, 16]. While the common target is reliability assessment, the use of different transport and reliability semantics exposes a lack of generalized semantics. This also complicates a comparison and development of conformal reliability enhancement strategies across them.

A limited body of work exists for reliability modeling of sensor nodes [4] and especially on the reliability of clustered WSNs [2, 13]. The reliability issues in multimodal fusion sensor networks is discussed in [4]. The authors presented system reliability for different types of sensors. Markov models are used to represent the system reliability. As only sensor reliability is modeled, this work is of complementary value to ours. The authors in [2] proposed an end-to-end reliability measure based solely on the connectivity between sensor nodes, similar to the traditional wired network. In [13] authors have extended this approach by including the reliability for sensing coverage of sensors in the WSN. Both approaches are limited in the sense that they are considering only connectivity and sensing aspects of data transport and are applicable only to clustered topologies. They do not consider other operations on data such as aggregation and routing. From the discussion above we conclude that there is a need for a generic reliability framework for data transport.

3. Models and Semantics: System Model, Fault Model, Data Transport Semantics

On this background, we first present a simple yet comprehensive system and fault model to capture generic WSNs properties. Next we survey the existing data transport semantics and propose our generalized semantic.

3.1. System Model

We consider a WSN consisting of N sensor nodes, $S = \{s_1, s_2, s_3, ...s_N\}$. Typically, each node is equipped with one or more sensing devices, short range transceivers for communication, limited processing, memory buffers and energy capabilities. Overall the WSN consists of a few designated sinks that are adequate in power, ideally up to the entire life of network, and possessing more memory and higher processing speeds as compared to the sensor nodes. In this work we assume, for the sake of simplicity, the existence of a single sink. We assume that all nodes are static in nature but the topology of the network is dynamic due to failures. Sensor nodes are placed in a finite size area and communicate with each other via bi-directional multi-hop wireless links employing a CSMA-based Medium Access Control (MAC) protocol.

3.2. Fault Model

WSNs are obviously subject to a wide range of computing and communication level faults. Cheap hardware, limited resources and severe environmental conditions lead to frequent perturbations in WSNs [1]. To achieve the desired responsiveness the proper identification of faults is necessary. Our fault classification is based on the ability of data transport protocols to tolerate the effects of these faults [10]. We categorize all faults encountered during the data transport broadly as intolerable or tolerable faults.

3.2.1 Intolerable Faults

Intolerable faults are those whose effects can not be handled by the data transport protocols. WSNs may be deployed in harsh environments such as for fire detection, tracking of people in catastrophic areas. These environments can permanently destroy the nodes on a large scale or the entire WSN, which obviously can not be handled. Other intolerable faults include crash failure of the sink and network partitioning. The sink plays an important role and acts as a bridge between the user and the WSN. Therefore, if the sink crashes, the network will not be able to communicate with the user resulting in an intolerable fault. Network partitioning is considered as an intolerable fault, since source nodes and the sink may belong to different network partitions. These intolerable faults can be transformed into tolerable ones, if the maintenance of the WSN is possible.

3.2.2 Tolerable Faults

Tolerable faults are those whose effects can be handled by the data transport protocols. We further classify the tolerable faults as communication and node failures.

Communication Failures: Communication failures constitute the most frequent failures in the WSN. Failures relevant to the data transport include message loss and higher message delays. These failures directly impact the responsiveness of the WSN.

- **Message Loss:** Interference, collision and contention constitute the major causes of message loss which effects the reliability and timeliness of data delivery.
- **Message Delay:** Network congestion is the major cause of message delay in the WSN and effect the timeliness of data delivery.

Node Failures: Node failures result in change of network topology and may impact the responsiveness of the WSN. Also nodes may start misbehaving. Due to these failures, the event detection accuracy becomes lower. We subdivide the node failures as follows.

- **Accidental Damage:** During deployment the sensor nodes may get dropped or impacted, damaging the node permanently. Also animals, falling trees or human themselves may accidentally destroy the sensor nodes.
- **Sensing Devices:** As the sensor nodes are interacting with harsh environments, and for extended periods of time the sensors may start misbehaving, i.e., reporting false alarms, resulting in erratic behavior, or the sensed values may be noisy.
- **Energy Depletion:** Sensor nodes may run out of power, which leads to a fail-stop behavior for these nodes.
- **Transient Failures:** These occur from either software or hardware perturbations and typically disappear if the node is rebooted.

3.3. Data Transport Semantics

We first describe the existing data transport semantics and illustrate their limitations. Consequently, we develop and define our generalized data transport semantic.

3.3.1 Existing Semantics

A prominent semantic used for data transport in traditional WSNs is the *end-to-end (e2e)* message delivery. Similar to wired networks, a node has to transport the data towards the sink. Unfortunately, this semantic is less suitable for WSNs, given their data-centric nature [6].

The commonly accepted semantic by the research community is *event-to-sink* [6, 14, 16]. This semantic considers multiple nodes reporting the phenomenon to the sink. Each node that detects the phenomenon is responsible for sending the data to the sink. This semantic is showed to be more suitable than the e2e semantic for WSNs [6], however the event-to-sink semantic does not consider the in-network processing of data.

3.3.2 Our Generalized Semantic

The in-network processing of data is common in WSNs. As soon as the sensed value exceeds a given threshold indicating the existence of the phenomenon, the corresponding sensor nodes generate raw data messages. These messages are disseminated towards the sink. In order to save limited resources such as energy and bandwidth, nodes perform some operations on received raw data and forward this information towards the sink (Figure 2). Operations on raw data are primarily filtering, aggregation and routing of data. Actually, the data transport starts with the generation of the raw data and comprises the different operations until the phenomenon is reported to the sink.

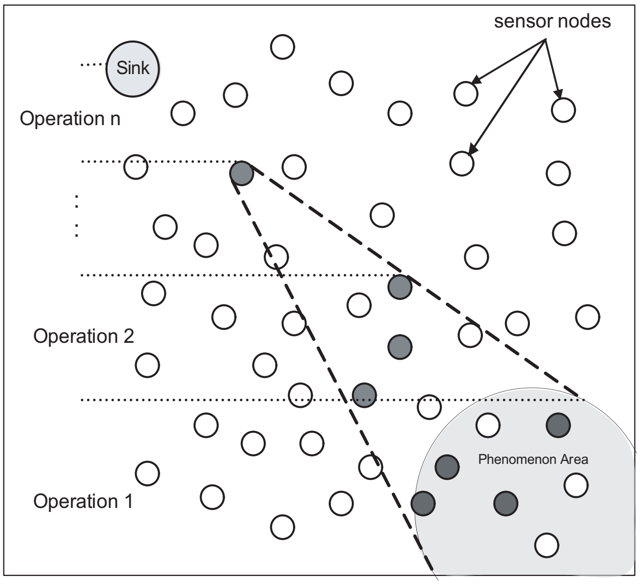

Figure 2. Our Generalized Semantic for Data Transport

Hence, complementing the e2e and event-to-sink, we develop our generalized semantic for data transport:

Definition 1 *Data transport in WSNs is a set of operations carried out on raw data from its generation till the phenomenon is reported to the sink.*

4. The Reliability Framework

As reliability is a major requirement for data transport, we aim at providing a generic framework that allows for a simple investigation of reliability. We first define a generic reliability semantic with an appropriate reliability metric. To simplify the computation of this metric, we then provide a reliability model for data transport.

4.1. Our Generalized Reliability Semantic

In the e2e semantic of data transport, the reliability metric is the probability that the single message generated for the event reaches the sink. In the event-to-sink semantic the reliability metric is the ratio of packets received at the sink to the total number of packets generated for the event. Neither semantics consider timeliness, nor do they address the in-network processing. Consequently, we need to develop a new reliability semantic that explicitly considers timeliness and our generalized semantic for data transport, in order to achieve the desired responsiveness.

Accordingly, the reliability of data transport is a function of the reliability of all the operations carried out on raw data. Furthermore, we define the reliability metric as follows:

Definition 2 *The reliability of data transport is the probability that the sink detects the phenomenon of interest within an application specified time bound.*

The decomposition of the data transport into operations simplifies the computation of the overall reliability, provided that the dependencies between the reliabilities of the different operations are given. This shows the need for a reliability model that simplifies the calculation of overall reliability of data transport.

4.2. The Data Transport Reliability Model

Prior to developing the reliability model, we specifically note that our emphasis is on setting up the reliability model for the operational phases of the WSN rather than modifying standardized reliability evaluation schemes.

There are various popular graphical formalisms to express system reliability such as Fault Trees, Markov Models and Reliability Block Diagrams (RBD). We use the RBD approach for its simplicity. The reliability of data transport depends on the reliability of each operation. If one of the operation fails, then the overall data transport fails. According to the RBD theory, this leads to a series representation of the data transport in the WSN. Figure 3 depicts the resulting RBD, which outlines the dependencies of the data transport reliability versus the different stages for data operations reliability.

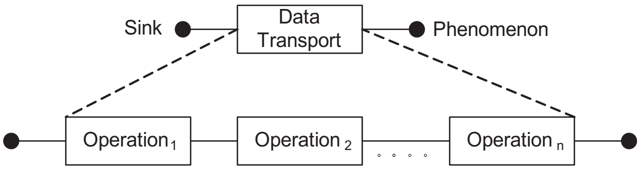

Figure 3. Reliability Block Diagram for Data Transport

We calculate the reliability of data transport R_d as follows:

$$R_d = R_{op_1} \cdot R_{op_2} \cdot \ldots \cdot R_{op_n}$$

$$R_d = \prod_{i=1}^{n} R_{op_i} \tag{1}$$

where R_{op_i} is the reliability of i^{th} operation and $R_d \epsilon [0, 1]$.

Using Equation (1), the data transport reliability is calculated provided the number of operations and their reliabilities are known. The reliability of each operation can be calculated either analytically or by simulations. Equation (1) assumes that the time specified by the application for data delivery is met by all the operations and all the deliveries are in time. We defer the task of assigning time to different operations to achieve overall timeliness as future work.

This reliability model can be used to measure and estimate the reliability of data transport in presence of the failures outlined in Section 3.2. In this work we mimic the failures by tuning the reliability of each operation. For example during the aggregation operation, loss of some raw data may reduce the accuracy of aggregated value, which can be considered as a decrease in the degree of reliability of the aggregation operation. Similarly, the routing of aggregated values can fail due to the message loss or delay and can be considered as a decrease in the degree of reliability of that operation.

5. Reliability Comparison of Existing Data Transport Protocols

In order to compare the reliability of existing data transport protocols we first classify and investigate the operations performed by the protocols to develop appropriate RBDs. Using these RBDs we calculate the reliability of these protocols in order to compare them.

5.1. Protocol Classification

According to the considered transport semantic we classify the existing protocols in two classes as: e2e and event-to-sink.

Several e2e protocols are available in literature [7, 8, 9, 11]. The basic technique to increase the reliability of e2e data transport is retransmission. These protocols differ mainly in the policies for retransmissions. Each method proposes a strategy to detect message loss and fix the nodes that can retransmit lost messages. There are multiple message loss detection techniques available such as Acknowledgment (ACK), Negative ACK (NACK), Selective NACK and timers. Modeling the reliability of these protocols is therefore similar. For this reason we select only one representative of this class, the Reliable Multi-Segment Transport (RMST) protocol [7].

For the event-to-sink class some protocols are available such as the Event to Sink Reliable Transport (ESRT) [6], Reliable Bursty Convergecast (RBC) [14] and Price-Oriented Reliable Transport (PORT) [16] protocols. Since this class is more suitable for WSNs we select the main representatives of this class: ESRT and RBC (PORT is based on ESRT with focus on energy efficiency) for our comparative study.

5.2. Reliability Models for the Protocols

After classifying the existing protocols and selecting representatives of each class we aim at developing the RBD for the selected protocols in order to evaluate and compare their reliability. The task of the existing data transport protocols in both classes is to report a message, either raw data or the message containing the result of previous operations on the raw data, to the sink. These protocols do not consider previous operations on the raw data. The operations not considered by the protocols are presented as grayed blocks in Figures 4-6 for completeness.

5.2.1 The RMST Protocol

The RMST [7] protocol is Selective NACK-based. RMST places responsibility for loss detection at the sink. Missing message requests are unicast from the sink to the source. RMST does not consider the timeliness of delivery.

Operations on data transport from the source to the sink in RMST are divided as follows:

Routing: This operation is used to identify potential routes for data transport.

Message Loss Detection (MLD): MLD is an essential operation for reliable data delivery. MLD is used for retransmission of missing data.

For reliable delivery in RMST, missing data is detected by Selective NACK and retransmitted. The failure of one retransmission does not result in the failure of data transport. This effect is shown as parallel RBD blocks for RMST in Figure 4.

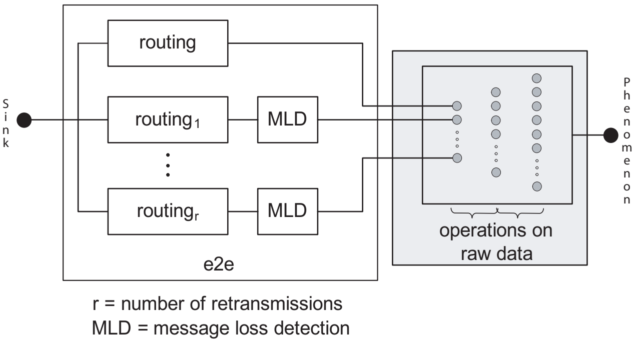

Figure 4. Reliability Block Diagram for RMST

The number of retransmissions play an important role in the reliability of the data transport. The designer can use this model to determine, what is the expected number of retransmissions to achieve the desired reliability.

Using Figure 4, the reliability of RMST R_{RMST} is calculated as follows:

$$R_{RMST} = 1 - \{(1 - R_R) * (1 - (R_R * R_{MLD}))^r\} \quad (2)$$

where R_R is the routing reliability and R_{MLD} the reliability of message loss detection.

R_R and R_{MLD} vary with respect to the protocols used, the environment where the WSN is deployed and the network conditions. These factors are typically determined during the design stage using simulations.

5.2.2 The ESRT Protocol

The ESRT [6] protocol achieves the optimal operating point by adjusting the reporting rate of sensor nodes depending upon the current network load. In this approach, upon getting information from nodes the sink knows about the network condition and accordingly informs the source nodes to adjust the reporting rates.

In ESRT, each node that detects the phenomenon routes the data towards the sink. If the data from one source node is not delivered, the application can tolerate this and data transport will not fail. Therefore, according to the RBD theory data transport for ESRT consists of n parallel routing blocks as shown in Figure 5.

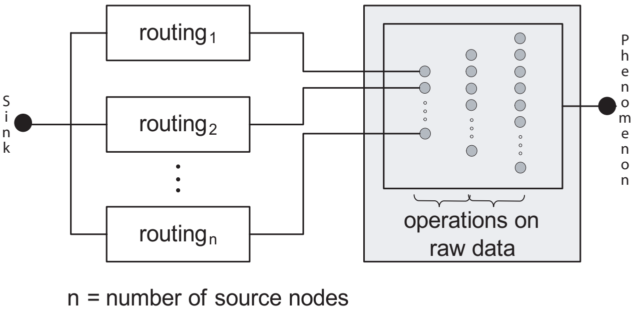

Figure 5. Reliability Block Diagram for ESRT

We calculate the reliability of ESRT R_{ESRT} as follows:

$$R_{ESRT} = 1 - (1 - R_R)^n \quad (3)$$

where R_R is the routing reliability and n is the number of sources reporting the phenomenon to the sink.

5.2.3 The RBC Protocol

RBC [14] focuses on bursty convergecast and uses a window-less block acknowledgment scheme which improves channel utilization and packet delivery delay. Methods are proposed to reduce timer delay and to schedule retransmission.

In RBC all source nodes send data towards the sink. Thus it can be viewed as a special case of RMST where instead of single source node, a set of nodes are transmitting the data, using the e2e semantic. The RBD for the RBC protocol is shown in Figure 6.

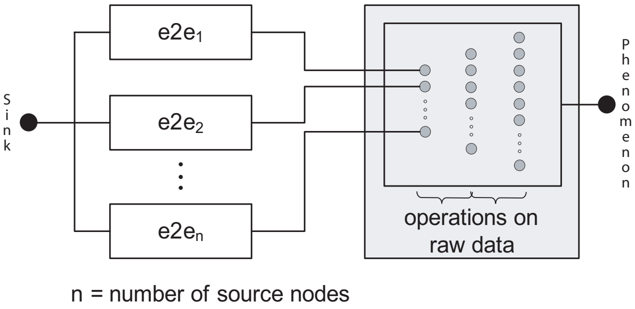

Figure 6. Reliability Block Diagram for RBC

The reliability of RBC R_{RBC} is calculated as follows:

$$R_{RBC} = 1 - (1 - R_{e2e})^n \quad (4)$$

where R_{e2e} is the reliability of the e2e scheme and n is the number of sources reporting the phenomenon to the sink.

Substituting Equation (2) in Equation (4) we obtain,

$$R_{RBC} = 1-(1-\left[1-\{(1-R_R)*(1-(R_R*R_{MLD}))^r\}\right])^n \quad (5)$$

5.3. Analysis

After computing the reliabilities of the selected existing data transport protocols, we explore how failures and important protocol design parameters impact these reliabilities. We investigate the impact of the retransmission strategy and especially the number of retransmissions on the reliability of the e2e protocols. For the event-to-sink protocols, we compare their reliability corresponding to the number of source nodes.

Figure 7 shows the impact of the number of retransmissions on the reliability of RMST using Equation (2). We investigated the number of retransmissions by fixing the routing and MLD reliability at different levels. Our purpose of tuning the reliability levels is to model the behavior of failures. In the case of high routing and MLD reliabilities we observe that after two retransmissions the reliability remains close to 1.0 and the impact of further retransmissions on the reliability is minimal. In case of low routing and high MLD reliability, after eight retransmissions the reliability of RMST becomes close to 1.0. The high reliability of MLD can be achieved if the MLD technique deploys the timer that shows a low probability of failure. In all scenarios after a certain number of retransmissions the behavior remains same and the retransmissions become useless and waste limited resources. These results are in agreement with the results in [15]. However our study specifically provides a new approach to easily determine the number of retransmissions needed for a given MLD strategy and a given routing reliability.

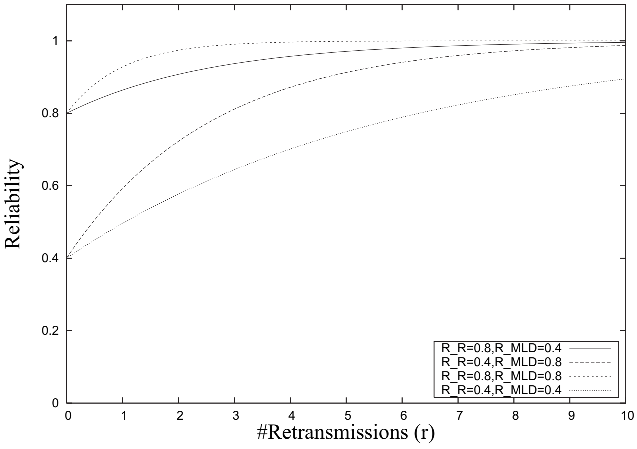

Figure 7. Impact of #Retransmissions on RMST Reliability

Figure 8 plots the reliabilities of ESRT and RBC (using Equations (3) - (4)) for different number of sources and different failure rates. We have fixed the number of retransmissions in RBC equal to three similar to [15]. Also we have fixed the reliability of MLD as 0.8. We observe that if routing reliability is high, then we require less number of sources for reporting the phenomenon. In this case ESRT and RBC performed equally good due to the fact that at a higher routing reliability, less retransmissions are needed. For a routing reliability of 0.5 RBC requires two reporting nodes, whereas ESRT requires six reporting nodes to achieve a reliability close to 1.0. This signifies that RBC requires fewer nodes to send data to the sink, saving precious resources in the network. For routing reliability less than 0.5 we require an higher number of sources.

Online adaptation of protocols can be easily achieved by tuning the protocol parameters according to current network conditions. Considering RBC for instance, R_R and R_{MLD} reflect network conditions and are affected by different failures occurring in the network whereas n and r are protocol parameters which can be tuned depending on the required degree of reliability (Equation (5)). If R_R or R_{MLD} decreases, n or r should be tuned appropriately so that the required degree of reliability is maintained.

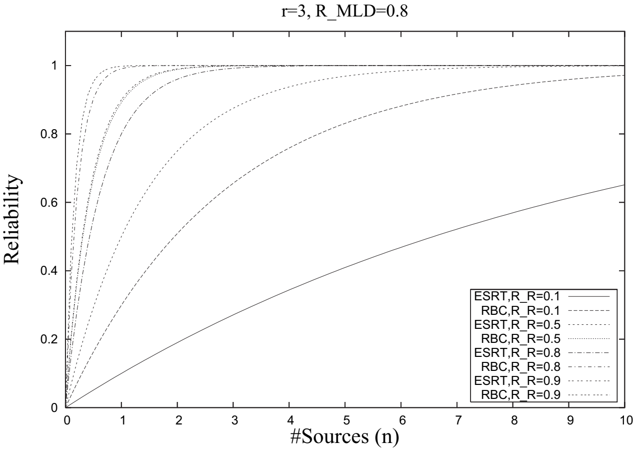

Figure 8. Comparison of the Reliability of ESRT and RBC

The above results are in accordance with the results in the literature, emphasizing the utility of our framework for a simple evaluation, comparison and adaptation of the reliabilities of existing and future data transport schemes.

6. Conclusions

In this paper, we developed a reliability framework for data transport based on the different operational phases of the WSN protocols. For this, we established a fault model to capture the possible failures along with generalized data transport and reliability semantics. Consequently we developed a reliability block model based approach that exploits the decomposition of the complex data transport problem into operations and simplifies the investigation of the overall reliability of data transport. Finally, we deployed our framework to study and compare the reliability of existing data transport protocols. This comparative study shows that the developed framework allows a systematic basis for reliability assessment and reliability comparison.

In the future, we plan to expand the framework focusing on other operational phases and the requirement on timeliness. We will also investigate the impact of single and grouped failures on the responsiveness of the WSN.

References

[1] A. Arora et. al. A line in the sand: a wireless sensor network for target detection, classification, and tracking. *Computer Networks*, 46(5):605–634, 2004.

[2] H. M. F. AboElFotoh, S. S. Iyengar, and K. Chakrabarty. Computing reliability and message delay for cooperative wireless distributed sensor networks subject to random failures. *IEEE Transactions on Reliability*, 54(1):145–155, 2005.

[3] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer Networks*, 38(4):393–422, 2002.

[4] D. Bein, V. Jolly, B. Kumar, and S. Latifi. Reliability modeling in wireless sensor networks. *International Journal of Information Technology*, 11(2):1–8, 2005.

[5] S. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong. Tag: A tiny aggregation service for ad-hoc sensor networks. In *Symposium on Operating Systems Design and Implementation (OSDI)*, pages 131–146, 2002.

[6] Y. Sankarasubramaniam, Ö. B. Akan, and I. F. Akyildiz. Esrt: event-to-sink reliable transport in wireless sensor networks. In *Interational Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, pages 177–188, 2003.

[7] F. Stann and J. Heidemann. Rmst: Reliable data transport in sensor networks. In *International Workshop on Sensor Network Protocols and Applications (SNPA)*, pages 102–112, 2003.

[8] T. Stathopoulos, J. Heidemann, and D. Estrin. A remote code update mechanism for wireless sensor networks. In *Technical Report CENS-TR-30, University of California, Los Angeles, Center for Embedded Networked Computing*, 2003.

[9] N. Texan, E. Cayirci, and M. U. Caglayan. End-to-end reliable event transfer in wireless sensor networks. In *Personal Indoor and Mobile Radio Communications (PIMRC)*, volume 2, pages 989–994, 2004.

[10] C. J. Walter and N. Suri. The customizable fault/error model for dependable distributed systems. *Journal of Theoretical Computer Science*, 290(2):1223–1251, 2003.

[11] C. Wan, A. T. Campbell, and L. Krishnamurthy. Psfq: a reliable transport protocol for wireless sensor networks. In *International Workshop on Wireless Sensor Networks and Applications (WSNA)*, pages 1–11, 2002.

[12] C. Wang, M. Daneshmand, B. Li, and K. Sohraby. A survey of transport protocols for wireless sensor networks. *IEEE Network Magazine, Special Issue on Wireless Sensor Networking*, 20(3):34–40, 2006.

[13] L. Xing and A. Shrestha. Qos reliability of hierarchical clustered wireless sensor networks. In *International Performance, Computing, and Communications Conference (IPCCC)*, pages 641–646, 2006.

[14] H. Zhang, A. Arora, Y. Choi, and M. G. Gouda. Reliable bursty convergecast in wireless sensor networks. In *Interational Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, pages 266–276, 2005.

[15] J. Zhao and R. Govindan. Understanding packet delivery performance in dense wireless sensor networks. In *International Conference on Embedded Networked Sensor Systems (SenSys)*, pages 1–13, 2003.

[16] Y. Zhou and M. R. Lyu. Port: A price-oriented reliable transport protocol for wireless sensor networks. In *International Symposium on Software Reliability Engineering (ISSRE)*, pages 117–126, 2005.