2013 IEEE 9th International Conference on Mobile Ad-hoc and Sensor Networks

A Framework of Temporal Data Retrieval for Unreliable WSNs Using Distributed Fountain Codes

Hsien-Tzu Chiu, Chia-Cheng Yen, and Jia-Shung Wang
Department of Computer Science,
National Tsing Hua University, HsinChu, Taiwan 30013
htchiu@vc.cs.nthu.edu.tw, ccyen@vc.cs.nthu.edu.tw, jswang@cs.nthu.edu.tw

***Abstract*—Distributed storage coding has been widely applied on data gathering over unreliable wireless sensor networks (WSNs), where it is essential to ensure the data persistence in case of several sensor failures caused by battery run-out or some physical damage problems surroundings. How to efficiently disseminate and collect the sensing data over WSNs is a key challenge yet. In this study, assumed that there are *K* sensor nodes equipped with sensing apparatus within *N* storage sensors, these *K* numbers of sensors can sense environmental changes and disseminate coded (by Fountain codes) time-series data over WSNs using the simple random walk. In order to perform the Fountain codes over WSNs, the question is to disseminate data in the long range of random walks to preserve the randomness so as to promote the source decoded rate. In this paper, a framework with partial decoding is proposed due to the temporal dependency of time-series sensing data. The reasons are twofold: (a) the complete decoding is not necessary for time-series data since the missing portions can be compensated by that of neighbors; (b) even if the ideal Luby transform (LT) code is optimized in terms of convergence, the complete decoding process is high power-consuming. Furthermore, a mathematical model to estimate the appropriate source decoded rate is given to guarantee the error tolerable level (< 4% normalized root-mean-square error (NRMSE)). Experimental results show that the communication cost is affordable in the real cases.**

Keywords—WSNs; Distributed storage coding; Fountain codes; LT codes; Temporal dependency

I. INTRODUCTION

Wireless sensor networks (WSNs) are composed of hundreds or even thousands of little devices (sensors) that are capable of wireless communication and sensing interested environmental information such as temperature, humidity, pressure, etc. WSNs can be widely deployed in different kind of environments. It is more feasible using mobile collectors collect measured data rather than send data periodically to the sink. The researches in WSNs usually face some challenges because that sensors are vulnerable to failure, unaware of others locations, energy-constrained with limited computation ability, and tiny memory, which means that they are unreliable and fragile and may disappear due to sensors crash. Therefore, the issue about data persistence in WSNs arises: how to acquire all the sensing data from remaining sensors despite of massive sensor failures. The popular solution to this problem is to encode data in a distributed fashion [1]-[5], [6], [7], [8]. Fountain codes [9], [10] are suitable for distributed storage coding in WSNs because of their intriguing properties: rateless and low complexity for both encoding and decoding processes. However, the challenge is how to apply Fountain codes in a distributed manner since they were originally designed for the centralized computing environment. Currently, the common way of disseminating data from multiple sensing nodes to sensor networks is random walk based mechanism [2]-[4]. There are two challenges should be addressed here. First, the dissemination cost is quite large to preserve the randomness so as to boost the source decoded rate. Second, the convergence is sluggish during the ending phase of Fountain codes because the receiving packet is unlikely to be useful, which means that almost of its neighbors had been decoded already.

Inspired by previous work on distributed Fountain coding in WSNs, in this paper, a framework with partial decoding is proposed aiming to largely reduce transmission cost with tolerable errors to fulfill the data persistence requirement in WSNs. We employs two distinguished Fountain codes: LT codes [11] and Repairable Fountain (RF) codes [12] to increase reliability of data retrieval in unreliable WSNs. And apply the concept of partial decoding to reduce transmission cost. Considering that the sensing data collected in a sensor is in the nature of temporal dependency, we can compensate lose (fail to decoded) data using interpolation with accessible neighboring data. So, the complete decoding is not necessary for most of time-series sensing data.

In this paper, we consider data propagation model as hop-by-hop and assume there are *K* sensors with sensing capability among *N* sensors. Where the remaining *N-K* nodes cannot sense data but can collect and encode sensed data for the purpose of storage. Whenever the *K* sensors sense environmental signals at each time slot *t*, generate a source packet and disseminate it over WSNs by the simple random walk [13]. We observed that the length of random walk influence the average decoded rate while the length of random walk longer, the transmission cost larger. In our framework the source packets are disseminated with long hop count at intervals of *t* time slots and the source packets at remainder (*t*-1) time slots with short hop count. Therefore we can recover all the data sensed at every interval of *t* time slots with high probability, and partial data (50% ~ 80%) sensed at remaining time slots, then we can retrieve predicted value of the other unrecovered sensed data using interpolation through temporal dependency. Furthermore, a theoretical model for analyzing the relationship between the average decoded rate and the mean square error (MSE) of compensated data through Wald's

978-0-7695-5159-3/13 $31.00 © 2013 IEEE
DOI 10.1109/MSN.2013.54

Equation [14] is provided. If the MSE of retrieved data could be controlled under a desired value, then we can choose the value of initialized hop count to achieve the corresponding decoded rate to reduce the transmission cost to an affordable level.

The rest of this paper is organized as follows. In Section II, we briefly describe LT codes and Repairable Fountain (RF) codes and reviews related works that increase data persistence of sensed data through distributed coding over WSNs. Also, the Castalia simulator in our WSN simulations is briefly described. In Section III, we introduce the proposed framework. In Section IV, simulation environment settings, network model and simulation results are presented. The conclusion is given finally.

II. Related Works

A. Fountain Codes

The basic idea of Fountain codes [9], [10] is that receivers can successfully recover K original source symbols with high probability when they receive enough (little larger than K) encoded packets. Fountain codes are a class of rateless code which means the number of encoded packets that can be generated limitlessly and determined on the fly. Fountain codes are suitable to provide reliable data delivery application over unreliable channels because the encoder can send as many as needed encoding symbols to the receivers until they recover all the source symbols without any feedback. Fountain codes also do not need to know channel conditions while delivering packets. LT codes [11] proposed by Michael Luby in 2002 are the first realization of fountain codes. In LT codes, the decoder is able to recover K source symbols from any subset of $K + O(\sqrt{K}\ln^2(\frac{K}{\delta}))$ encoded symbols with probability 1-δ. The performance of LT codes is dominated by the degree distribution. In [11], *robust Soliton distribution* was given to ensure the expected ripple size large enough so that the ripple never disappears in the process of decoding with high probability.

The decoding process can be divided into three phases. Early phase, the decoded rate is lower than 10% and raising up in a slow speed. Intermediate phase, the decoded rate booms up rapidly. Ending phase, the decoded rate is larger than 95% and the decoding speed slow down again. In particularly, the receiving packet is unlikely to be useful. This long tail effect will cause more overhead to complete the whole decoding process.

In [12], Asteris and Dimakis introduced a new family of fountain codes called Repairable Fountain (RF) codes with systematic form, rateless, near-MDS, and locality properties.

B. Distributed Fountain Codes

In [2], Lin *et al.* proposed a decentralized fountain codes (EDFC) using random walk to distribute source data to a random subset of sensors in the network, where each sensor only encode the data it receive (a random walk stop at that node). They consider a scenario where no sink in the wireless sensor networks because it is not feasible that sensors deployed in a harsh environment transmit information to the sinks periodically, also the neighboring nodes of sinks may occur communicating congestions. They proposed a vision asking the sensors to collaboratively store measured data over a historical period of time on themselves. Then a mobile collector collects such historical data at a later time of convenience. The transmission cost of EDFC is the product of the length of a random walk and the number of random walks.

A random walk corresponds to a time-reversible Markov chain [13]. If the length of the random walk is sufficiently long and the graph is ergodic, the Markov chain has a steady-state distribution. Their work performs well as the (centralized) Fountain codes, however it is not practical. The maximum node degree is not accessible for sensors in real WSNs and too many packets flow into the network, which causes congestion.

In [3], Aly *et al.* proposed a distributed storage algorithm (LTCDS-I), which each node disseminates one source packet throughout the sensor networks by simple random walks without trapping. Each node independently chooses one of neighbors uniformly at random to forward a source packet. The length of random walks is of order $\theta(n\log n)$. LTCDS-I does not maintain a probabilistic forwarding table like that in [2], however, it need the *cover time* of random walks be large enough to achieve the performance as (centralized) LT Codes. The transmission cost is extremely high when LTCDS-I is employed in a large-scale sensor networks.

In [4], Vukobratovic *et al*. proposed a *packet-centric* approach, different from node-centric techniques. This approach uses encoded packets to take responsibility for collecting sufficient source data from different source nodes and performing rateless encoding, which are called *rateless packets*. Initially, each sensor sends b copies of source packet whose packets header attached with a coding degree, a mixing time counter $\tau=\lceil C\log n\rceil$ and ID. The dispersion process, called NRW, continues until the coding degree vanished. Their simulation results [4] demonstrated a better performance of NRW than Maximum-Degree [15] and Metropolis-Hasting algorithm [16].

In [5], Aly *et al*. proposed another control folding mechanism for disseminating packets throughout the sensor network quickly, using a mixing time of the order of $O(n)$. Each sensor broadcasts the received data packet to the set of its neighbors which have not yet received the packet. In their proposed scheme (DSA), each sensor is capable of sensing and storage and has a buffer storing multiple encoded data. The simulation results show it required to query only 20%~30% of total sensors for successfully decoding all sensing data at the expense of large buffer size reaching to 10% of the network size.

In [6], Liang *et al*. proposed a probability broadcast mechanism which enables all nodes to receive the data packet and reduce the redundancy of data transmission. Each sensor has $m \geq 1$ storage units. They analyzed the critical rebroadcast probability which depends on the network topology and the communication model.

In summary, the random walk scheme is unanimously approved; however the range of random walks is still of linear order to preserve the randomness as in the centralized version.

C. Castalia: Simulation Tool for WSNs

Based on the survey [17], we choose Castalia [18] as a simulator for our simulation tool. Castalia is a simulator for WSN, BAN or general networks of low-power embedded devices. In our simulation, we plug-in new application modules for the proposed distributed storage coding.

III. THE PROPOSED FRAMEWORK

We consider a wireless sensor network with K sensor nodes equipped with sensing apparatus among N nodes (N-K storage nodes) capable of limited computation, storage and wireless communication abilities. Sensor nodes are randomly scattered in a plane of size $L \times L$, and unaware of each other's locations. Each node has a same transmission radius r and adopts hop-by-hop data propagation model. That is, if a sensor locates within some one's transmission range, they communicate with each other directly in one hop; otherwise, they need neighboring nodes to help forwarding packets wirelessly. We can formulate such a wireless sensor network as a random geometric graph [19] model, denoted as $G(K, N, r)$. At each time slot the K sensing nodes generate one source packet containing Hello flag, sensing data along with its node ID, and an initial hop count (i.e., the length of random walks) and then disseminate it over networks wirelessly. Once a sensor receives a hello packet, extracts ID from the packet header and record it as neighbor. After dissemination phase, all sensors step into encoding phase. The complete structure of the proposed framework is depicted in Figure 6.

A *simple random walk* on graphs forms a path that consists of a sequence of random and independent steps, that is, the next node that decides to stop by is chosen from current node randomly at uniform. The random walk based data dissemination mechanisms have several advantages such as simplicity, no need for global information and robustness for dynamic network topologies. In [4], the simulation results show the better performance of the simple random walk for data dissemination, Therefore, the proposed framework follows the concept to adopt the *simple random walk* for data dissemination.

We observe that the length of random walks affects the average decoded rate of sensing data as shown in Figure 1. The random walk moves from one node to its neighboring node in a random manner, it may revisit some nodes frequently and may be trapped in a local area around the starting node. Figure 2 shows an example. To achieve the same level of efficiency as the centralized LT codes do, the source data need to be uniformly combined (XOR) into encoded data. Hence, the length of random walk should be long enough so as to uniformly distribute of source data over the full service network, also to successful decode with less redundancy.

It is at great expense of communication cost however. Also, as mentioned before, the decoding process can be divided into three phases. Consider Figure 1 again. The ending phase begins around hop count of 120, here takes a long way (>200) to the end (100% decoded). This long tail effect will cause much more overhead so we suggest the framework with partial decoding and compensating for uncovered.

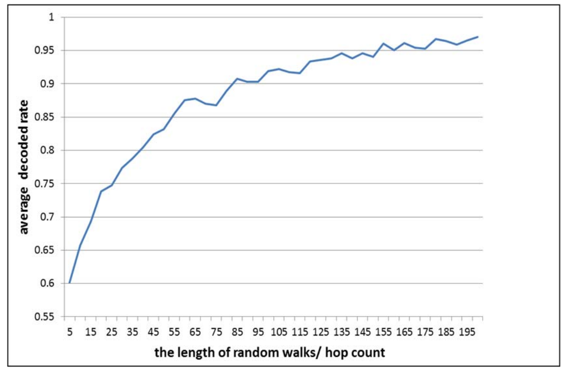

Figure 1 The average decoded rate of distributed LT coding (c = 1, δ = 0.05 and redundancy $\eta \leq 1.1$) that disseminates source packets by simple random walk over a sensor network with K = 83, N = 166 and r = 10.49meter, where sensors deployed in a size of 100×100 meter square field.

A. Design Concept

We separate time-series data into two categories. For a given number t, the data corresponding to the numbers divisible by t will be almost recovered with high probability; however, the data corresponding to the other time slots is set as partially recovered. Figure 3 describes the concept of proposed framework and shows that the missing data can be interpolated by the nearby neighboring data through temporal dependency.

The proposed framework employs LT codes and RF codes to increase the data persistence in case of massive sensor failures. As mentioned previously, the length of random walks (initial hop count) affects the data successful recovered rate. In Figure 4 and Figure 5, we observe that RF codes have better decoding performance than LT codes corresponding to short hop count; however the successful decoded rate of LT codes is as good as or even better than RF codes corresponding to lengthy hop count. Therefore, the proposed framework fully utilizes these observations, that is, the mixture of LT and RF, denoted as *long-short* dissemination plan: applying LT codes of long length dissemination at time slots divisible by t; applying RF codes of short length dissemination at remaining time slots. Note that, in Figures 4 and 5, the tremble of redundancy corresponding to the longer hop count is due to the characteristics of Fountain codes. Since the average decoded rate reaches above 0.98, the probability of recovering a new source symbol from an encoded data would be declined.

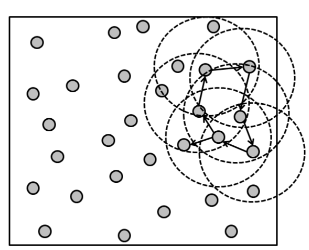

Figure 2 An illustrated example of being trapped in a local area around the starting node.

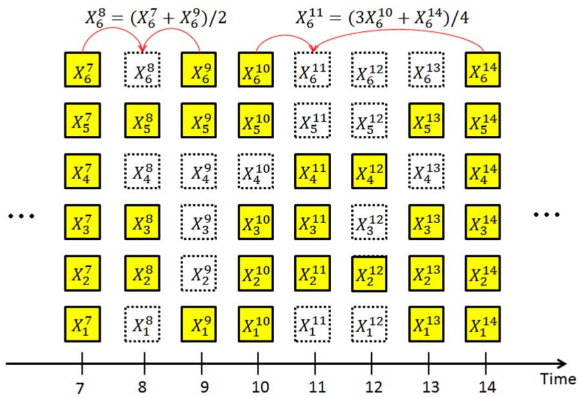

Figure 3 An example of proposed framework with t = 7. X^j_i is the sensing data of node i at time slot j.

B. Estimation of Source Decoded Rate

In this section, a mathematical model for analyzing the relationship between the average decoded rate and the mean square error (MSE) of compensated data is presented. Thus, the appropriate source decoded rate under a desired value of MSE can be answered through this theoretical model, and then find a sufficient hop count to meet the corresponding decoded rate so as to reduce the communication cost eventually. Figure 7 briefly describes the procedure. Note that the relationship between hop count and average decoded rate is complicated because of several big issues such as, network topology, dissemination behavior, and Fountain codes, being tangled together.

The proposed framework uses simple interpolation to recover the missing (failed to decode) data. However, if there is consecutive data which is sensed in a node cannot be decoded; the interpolation accuracy would decline with the event of continuously unrecovered data. Therefore, we can see the aggregated amount of compensated errors as the summation of the prediction errors caused by a series of events of n loss, n = 1, 2, …, in a time span T.

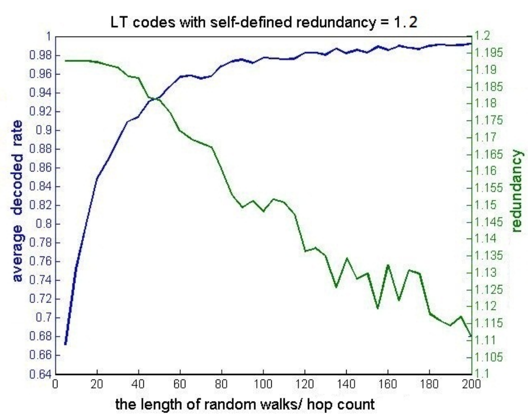

Figure 4 The decoding performance of distributed LT coding (c = 1, δ = 0.05 and redundancies 1.1 ≤ η ≤ 1.2) that disseminates source packets by simple random walk over a sensor network with K = 83, N = 166 and r = 11, which sensors deployed in a size of 100×100 meter square field.

Let X^n_i is i^{th} event of compensating exactly n consecutive loss. Then the mean error $E(X)$ resulted by data compensation can be represented as below:

$$E(X) = E(\sum_i X_i^1) + E(\sum_i X_i^2) + E(\sum_i X_i^3) \cdots \quad (1)$$

Therefore we need to calculate $E(\sum_i X_i^n)$ for n = 1, 2, … to get the mean error $E(X)$ given a source decoded rate p. We can calculate $E(\sum_i X_i^n)$ based on *Wald's Equation* [14], since $\{X_1^n, X_2^n, \ldots\}$ are independent of each other. Since for lager n (> 5), the probability of consecutive loss is quite small so that we ignore these events. Let N^n be the event of exactly n consecutive loss. By calculation of *Wald's Equation* we have:

$$E(X) \cong \sum_{i=1}^{5} \mathrm{E}(N^i)\, \mathrm{E}(X^i) \quad (2)$$

With single analysis model, the appropriate decoded rate corresponding to a desired value of MSE can be solved. In Subsection IV.D, we shall demonstrate the practical usage for several real datasets.

IV. SINULATIONS AND DISCUSSIONS

To evaluate the performance of proposed framework, two schemes are presented and compared. One is the proposed long-short scheme; another is the solution using distributed LT codes only. Note that, since the solution of complete decoding (100% fully decoded during a time span) needs relatively high hop count, we ignore the comparison in the following discussions. Furthermore, we evaluate the accuracy of theoretical average decoded rate using the proposed estimation model.

A. Parameter Settings

We use the Castalia simulator to simulate the proposed framework implemented over a wireless sensor network $G(K, N, r)$ of real data, where K = 83, N = 166. All sensors are randomly deployed in a 100m×100m area. The window size (time span T) is 20, where the parameter of long-short plan is seven.

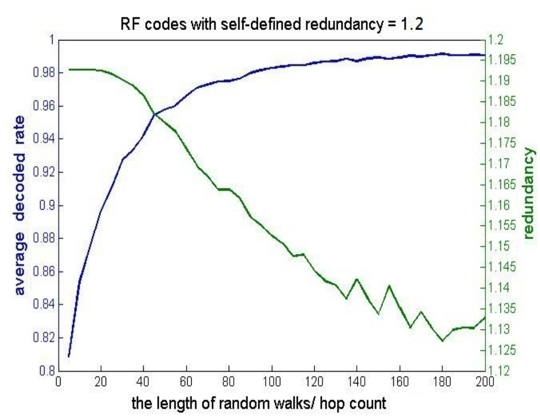

Figure 5 The decoding performance of distributed RF coding (c' = 2 and redundancies 1.12 ≤ η ≤ 1.2) that disseminates source packets by simple random walk over a sensor network with K = 83, N = 166 and r = 11, which sensors deployed in a size of 100×100 meter square field.

The parameters of LT codes with Robust Soliton distribution are C = 0.1 and δ = 0.05, and the parameter of RF codes is C'

= 2. The decoding and data compensation process are simulated by C++ code. We adopt the decoding algorithm proposed in [20].

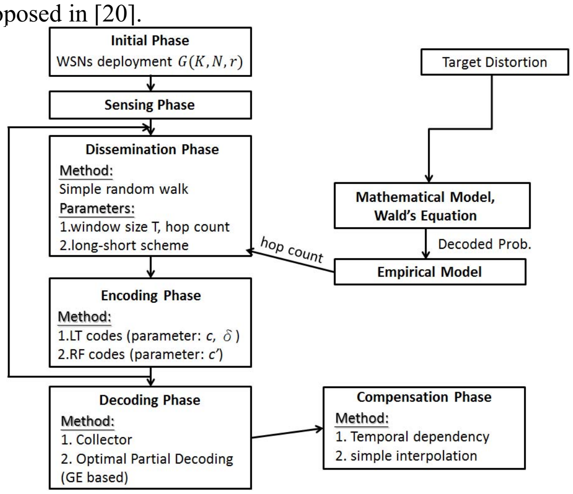

Figure 6 The flowchart of the proposed framework.

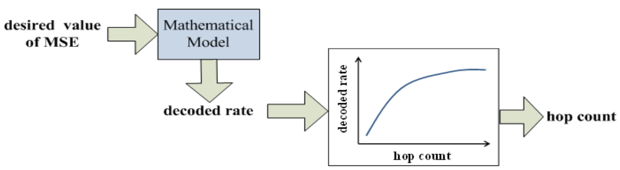

Figure 7 The basic concept for estimation.

The real dataset is from the Lausanne Urban Canopy Experiment (LUCE, http://lcav.epfl.ch/cms/lang/en/pid/86035). We choose three types of sensing data: ambient temperature, surface temperature and relative humidity, which were measured from 4 p.m. in 2007-05-04 to 11 a.m. in 2007-05-05 at interval of one hour.

In the large-scale WSN case, where K=500, N=1000, is also evaluated. Unfortunately, we do not have such amount of real dataset of K=500, however, we generated the synthetic data by a random-walk-like model. The value for each data point can be lower than or higher than that of the previous data point according to the probabilities p and (1-p), respectively. The magnitude of increase/decrease in the value is given by a uniform distribution $U(0, x)$, where x is a configurable parameter [21]. We let p=0.5; x=1 for temperature data set and x=5 for humidity data set, then generate the synthetic data from the data set of LUCE as mentioned. We adopt normalized root mean square error (NRMSE) as the error measurement of interpolation errors. All the results such as the average decoded rate, MSE, NRMSE are obtained based on an average of 1000 runs.

B. Simulation Results, the Average Decoded Rate

The comparison between the long-short scheme and the distributed LT scheme with the same length of random walks in terms of the average decoded rate is given and discussed in this subsection. In decoding, the mobile collector queries a number of random storage nodes limited by $K\times\eta$. Apparently, the average decoded rate is proportional to the number of encoded data that we can collect (i.e. decoding redundancy). We limit the decoding redundancy to η=1.2 to save the query time. In our experiments, total hop count for window size 20 is ranging from 420 (55/15) to 1180 (195/35), where 55/15 is the combination of long-short, where 55 for long and 15 for short, totally with 55*3+15*17 = 420 hops. Also, three ranges of radius are considered: r = 10.49m, r = 11.01m and r = 12.12m corresponding to the average node degree of network graph 5.47, 6.02, and 7.13, respectively. Because of the page limitation, only the case of r = 11.01m is depicted, see Figure 8. The simulation results demonstrate that the proposed long-short scheme has better performance than distributed LT scheme except for some cases with larger redundancy η=1.2. Note that, our proposed long-short approach still can achieve the decoded rate around 0.85 even with decoding redundancy limited to 1.1.

C. Simulation Results, NRMSE

The proposed approach interpolates the missing data by the neighboring data through the temporal dependency. Here, we show the performance by NRMSE. Again, the proposed long-short scheme has lower level of errors than distributed LT scheme, see, Table 1.

D. Hop count vs. Target tolerance in NRMSE

In this subsection, the performance of the proposed theoretical model of estimating the mean square error with three real datasets is discussed. The experiments use the average decoded rate in Figure 1 as inputs. Since the proposed model is based on *Wald's Equation*, the results are the average of possible values. The results of theoretical model for three datasets, ambient temperature, surface temperature and relative humidity demonstrate the match up with the actual circumstances, see Table 1, 2 and Figure 9. Thus, this model is valuable for estimating the corresponding decoded rate to the desired error.

V. CONCLUSION

We proposed a framework to largely reduce the communication cost in this paper. With the help of exploiting the temporal dependency of time-series sensing data achieves the goal of affordable communication cost. The framework we suggested with partial decoding and compensation actually avoids the long tail effect thus largely reduce the communication overhead in the realization of distributed Fountain codes. We examine the efficiency of the proposed framework with two schemes in terms of the average decoded rate and NRMSE. Simulation results show that the proposed long-short scheme can reduce transmission cost in data dissemination phase with less amount of decoding redundancy while maintaining tolerable level of errors as well.

REFERENCES

[1] A. Kamra, J. Feldman, V. Misra and D. Rubenstein, "Growth Codes: Maximizing Sensor Network Data Persistence," *Proc. ACM SIGCOMM 2006*, Pisa, Italy, 2006.

[2] Y. Lin, B. Liang and B. Li, "Data Persistence in Large-Scale Sensor Networks with Decentralized Fountain Codes," *Proc. IEEE INFOCOM 2007*, Anchorage, AL, 2007.

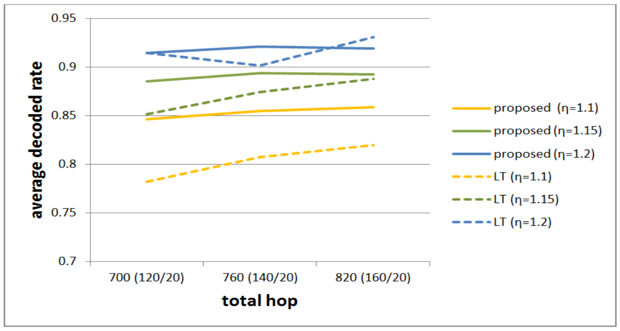

Figure 8 Average decoded rate for $r = 11.01m$, where the average node degree of network graph is 6.02.

Table 1 NRMSE with $r = 11.01$m.

Total hop	η=1.1		η=1.15		η=1.2	
Ambient Temperature						
	Proposed	LT	Proposed	LT	Proposed	LT
700 (120/20)	3.52%	6.40%	2.31%	4.65%	1.62%	3.22%
760 (140/20)	2.59%	4.84%	1.75%	3.74%	1.26%	2.99%
820 (160/20)	2.78%	4.98%	1.87%	3.23%	1.38%	2.22%
Surface Temperature						
Total hop	η=1.1		η=1.15		η=1.2	
	Proposed	LT	Proposed	LT	Proposed	LT
700 (120/20)	2.71%	4.57%	1.89%	3.38%	1.42%	2.45%
760 (140/20)	2.03%	3.38%	1.46%	2.59%	1.11%	1.98%
820 (160/20)	2.21%	3.54%	1.59%	2.34%	1.22%	1.63%
Relative Humidity						
Total hop	η=1.1		η=1.15		η=1.2	
	Proposed	LT	Proposed	LT	Proposed	LT
700 (120/20)	4.59%	8.34%	3.01%	6.15%	2.15%	4.30%
760 (140/20)	3.34%	6.05%	2.24%	4.67%	1.61%	3.75%
820 (160/20)	3.65%	6.55%	2.44%	4.31%	1.80%	2.95%

Table 2 NRMSE of distributed coding over large-scale WSNs.

Total hop	η=1.1		η=1.15		η=1.2	
Ambient Temperature						
	Proposed	LT	Proposed	LT	Proposed	LT
880(180/20)	3.01%	5.30%	2.62%	4.26%	2.30%	3.45%
1220 (180/40)	2.78%	5.05%	2.31%	3.93%	2.01%	3.04%
1560 (180/60)	2.58%	4.14%	2.12%	3.14%	1.78%	2.50%
Surface Temperature						
Total hop	η=1.1		η=1.15		η=1.2	
	Proposed	LT	Proposed	LT	Proposed	LT
880(180/20)	2.31%	4.13%	2.01%	3.29%	1.77%	2.65%
1220 (180/40)	2.14%	3.82%	1.78%	2.96%	1.55%	2.29%
1560 (180/60)	2.03%	3.24%	1.67%	2.48%	1.39%	1.99%
Relative Humidity						
Total hop	η=1.1		η=1.15		η=1.2	
	Proposed	LT	Proposed	LT	Proposed	LT
880(180/20)	3.52%	7.01%	3.00%	5.56%	2.62%	4.45%
1220 (180/40)	3.25%	6.50%	2.78%	5.05%	2.35%	3.93%
1560 (180/60)	3.10%	5.38%	2.56%	4.10%	2.14%	3.30%

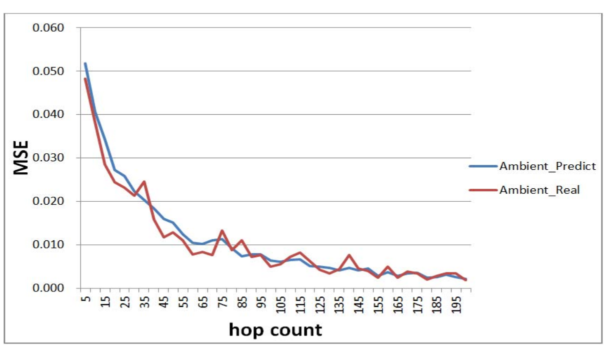

Figure 9 The result of theoretical model with ambient temperature data using MSE.

[3] S. Aly, Z. Kong, and E. Soljanin, "Fountain codes based distributed storage algorithms for large-scale wireless sensor networks," *Proc. IEEE/ACM IPSN*, S. Louis, 2008.

[4] D. Vukobratovic, C. Stefanovi, V. Crnojevi, F. Chiti, and R. Fantacci,"Rateless Packet Approach for Data Gathering in Wireless Sensor Networks," *IEEE Journal on Selected Areas in Communications*, Vol. 28, No. 7, pp. 1169-1179, Sept. 2010.

[5] S. A. Aly, H. Darwish, M. Youssef, and M. Zidan, "Distributed flooding-based storage algorithms for large-scale wireless sensor networks," *IEEE International Conference on Communications*, 2009.

[6] J. Liang, J. Wang, X. Zhang, and J. Chen, "An Adaptive Probability Broadcast-based Data Preservation Protocol in Wireless Sensor Networks," *IEEE International Conference on Communications*, 2011.

[7] A. G. Dimakis, V. Prabhakaran, and K. Ramchandran. "Decentralized erasure codes for distributed networked storage," *IEEE/ACM Transactions on Networking*, pp. 2809-2816, June 2006.

[8] A. G. Dimakis, V. Prabhakaran, and K. Ramchandran. "Distributed fountain codes for networked storage," *IEEE Conference on Acoustics, Speech and Signal Processing*, May 2006.

[9] J.W. Byers, M. Luby, M. Mitzenmacher, and A. Rege, "A Digital Fountain Approach to Reliable Distribution of Bulk Data," *Proceedings of ACM SIGCOMM '98*, pp. 56-67, 1998.

[10] J.W. Byers, M. Luby, and M. Mitzenmacher. "A Digital Fountain Approach to Asynchronous Reliable Multicast," *IEEE J. on Selected Areas in Communications, Special Issue on Network Support for Multicast Communications*, Vol. 20, No. 8, pp. 1528-1540, 2002.

[11] M. Luby, "LT Codes," *The 43rd Annual IEEE Symposium on Foundations of Computer Science,* pp. 271-280, November 2002.

[12] M. Asteris, and A.G. Dimakis, "Repairable Fountain codes," *IEEE International Symposium on Information Theory* (ISIT), pp.1752-1756, 2012.

[13] D. Aldous, J. Fill, "Reversible Markov chains and random walks on graphs," unpublished.

[14] Saeed Ghahramani, *Fundamentals of probability with stochastic process*, 3rd Edition, Pearson/Prentice Hall, 2005.

[15] S. Boyd, P. Diaconis, and L. Xiao, "Fastest Mixing Markov Chain on a Graph," *SIAM Review, problems and techniques section*, vol. 46(4), pp.667–689, 2004.

[16] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller, "Equations of state calculations by fast computing machines," *J. Chem. Phys*., Vol. 21, pp. 1087–1092, 1953.

[17] H. L. Harsh Sundani, Vijay K. Devabhaktuni, Mansoor Alam, and Prabir Bhattacharya, "Wireless Sensor Network Simulators A Survey and Comparisons," *International Journal of Computer Networks (IJCN)*, Vol. 2, 2010.

[18] Castalia, Available in http://castalia.research.nicta.com.au.

[19] M. Penrose. *Random geometric graphs*, Vol. 5, Oxford: Oxford University Press, 2003.

[20] V. Bioglio, M. Grangetto, R. Gaeta, and M. Sereno. "An optimal partial decoding algorithm for rateless codes," *IEEE International Symposium on Information Theory Proceedings* (ISIT), pp. 2731-2735, 2011.

[21] H. Elmeleegy, A. K. Elmagarmid, E. Cecchet, W. G. Aref, and W. Zwaenepoel, "Online piece-wise linear approximation of numerical streams with precision guarantees," in *Proceedings of the VLDB*, 2009.

[22] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Mixing Times for Random Walk on Geometric Random Graphs," *Proc. SIAM ANALCO Workshop*, 2005.

[23] P. Gupta and P. R. Kumar, "The Capacity of Wireless Networks," *IEEE Trans. Info. Theory*, Vol. 46, No. 2, pp. 388–404, 2000.

[24] C. Avin and G. Ercal, "On the cover time and mixing time of random geometric graphs," *Theor. Comp. Science*, Vol. 380, pp. 2–22, 2007.

[25] J. C. S. Cardoso, C. Baquero, and P. S. Almeida, "Probabilistic estimation of network size and diameter," in *Latin-American Symposium on Dependable Computing*, pp. 33-40, João Pessoa, Brasil, September 2009.