Efficient Route Selection Strategy to Reduce Packet Delays for Using Data Compression in WSNs

You-Chiun Wang and Chia-Ting Wei
Department of Computer Science and Engineering,
National Sun Yat-sen University, Kaohsiung, 80424, Taiwan
Email: ycwang@cse.nsysu.edu.tw; alexx6319@gmail.com

***Abstract*—In a wireless sensor network, sensors may require to periodically report the monitoring results to the sink, causing network congestion and exhausting their energy. The data compression technique can alleviate this situation by proving in-network data reduction. However, existing data compression schemes for wireless sensor networks usually ask each sensor to accumulate up to a threshold number of monitoring results in order to produce the compressed packet. When sensors generate monitoring results in different rates, some packets will encounter high delays. To solve this problem, the paper develops an efficient route selection strategy to facilitate the data compression process. Our strategy helps sensors to determine whether to keep the monitoring results for compression locally or to send them to the best neighbors so as to reduce potential packet delays. Through simulations, we verify the effectiveness of our proposed route selection strategy in terms of diminishing packet delays.**

***Index Terms*—data compression, in-network data processing, packet delay, route selection, wireless sensor network.**

I. INTRODUCTION

In the recent decades, *wireless sensor networks (WSNs)* have successfully attracted considerable attention from academia, industry, and military [1], [2]. A WSN is composed of numerous small sensors, each containing processing, storage, sensing, and communication modules. Some sensors can even be equipped with mobile platforms so as to roam around in the region of interest to react to the environmental situations [3], [4]. These sensors then use a multihop routing protocol to transmit their monitoring results to a remote sink node. Various WSN applications have also been developed, from precision agriculture [5] to home surveillance [6], energy saving [7], oceanic exploration [8], and pollution detection [9], [10].

In many WSN applications, sensors have to continually transmit their monitoring results to the sink. It is expected that the application can be executed for a long period (for example, several months) but sensors are usually non-rechargeable. One possible solution is to exploit *sensor redundancy* by only activating a subset of sensors by turns in order to monitor the region of interest [11], [12]. However, this solution requires to deploy a lot of extra sensors (for example, providing k-coverage deployment [13]), which results in a pretty high network-construction cost. Another feasible solution is to provide *in-network data compression* by taking advantage of data correlation of monitoring results [14]. In other words, each sensor can combine several monitoring results together by ‘compressing’ them into one packet and then send out only this packet. It can significantly reduce the amount of data transmitted by sensors, thereby saving their energy.

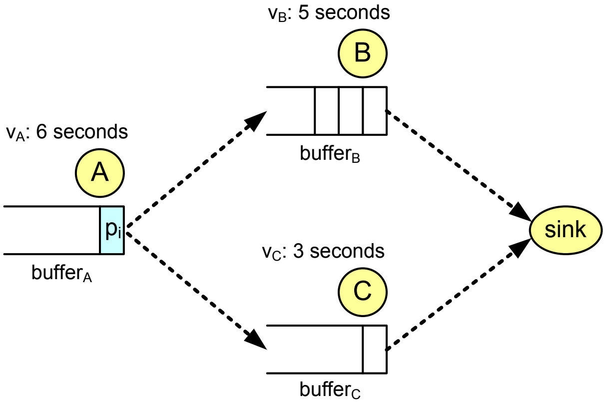

Fig. 1: An example of route selection with data compression.

This paper aims at using the data compression technique to extend a WSN’s lifetime. However, existing data compression schemes for WSNs [15] usually require each sensor to accumulate a threshold number of monitoring results so as to conduct the compression operation on them. When sensors generate monitoring results in different rates, some packets will be queued for a longer time, resulting in high packet delays. Fig. 1 presents an example, where there are three sensors A, B, and C, which generate one monitoring result every six, five, and three seconds (that is, $v_A = 6$, $v_B = 5$, and $v_C = 3$), respectively. Suppose that each sensor has a buffer to keep its monitoring results for compression, where there are already one, three, and one monitoring results in the buffers of sensors A, B, and C, respectively. In addition, we assume that a data compression scheme will compress every six monitoring results into one packet. Sensor A can choose between sensors B and C to be its next-hop neighbor to the sink. For ease of calculation, let us assume that the transmission time can be ignored in the example. Then, we discuss the *queuing time* of a monitoring result p_i currently in sensor A’s buffer. For a conventional data compression scheme, it will ask sensor A to accumulate up to six monitoring results (including p_i) so as to send out the compressed packet. In this case, p_i has to stay at sensor A’s buffer with the total time of

$$v_a \times (6 - 1) = 6 \times 5 = 30 \text{ seconds}.$$

Here, since sensor A already has one monitoring result (i.e., p_i) in its buffer, it has to wait for generating the remaining $(6-1)$ monitoring results to produce the compressed packet, where it generates one monitoring result every six seconds. Then, even if sensor B immediately relays this compressed packet to the sink, the queuing time of p_i (in the network) will be at least 30 seconds.

In fact, sensor A has other choices by directly sending p_i to either sensors B or C to speed up the compression operation. If sensor A chooses to send p_i to sensor B for compression, then p_i has to stay at sensor B's buffer for

$$v_b \times (6-(3+1)) = 5 \times 2 = 10 \text{ seconds.}$$

Here, since sensor B already has three monitoring results in its buffer, if it can get p_i, then sensor B only requires to wait for generating the remaining $(6-(3+1))$ monitoring results to produce the compressed packet. Because we ignore the transmission time in the example, the queuing time of p_i can be shrunk to 10 seconds. Similarly, if sensor A directly sends p_i to sensor C for compression, then the total queuing time of p_i can be reduced to

$$v_c \times (6-(1+1)) = 3 \times 4 = 12 \text{ seconds.}$$

Obviously, we can significantly reduce the packet delay of p_i in either case and the optimal solution in Fig. 1 is that sensor A directly sends p_i to sensor B for compression.

Motivated from the example in Fig. 1, we develop an efficient route selection strategy with the goal of decreasing packet delays when applying the data compression technique to WSNs. Our strategy allows each sensor to compare the benefit if it keeps the monitoring results in its buffer or directly sends them to a neighbor. In addition, the route selection strategy also helps a sensor estimate the 'best' neighbor to be its next hop to the sink so that the data compression process can be facilitated. We apply our route selection strategy to the popular AODV (ad hoc on-demand distance vector) protocol [16] in the simulation and experimental results demonstrate that our strategy can significantly reduce packet delays compared with other methods. This paper contributes in 1) pointing out the problem encountered in existing WSN data compression schemes where some packets will suffer from pretty longer delays and 2) developing a novel route selecting strategy to solve this problem.

The remainder of this paper is organized as follows: Section II surveys existing data compression schemes in WSNs and also presents the network assumptions. Our route selection strategy is proposed in Section III. Experimental results are discussed in Section IV. We finally give a conclusion of this paper in Section V.

II. PRELIMINARY

A. Data Compression Technique in WSNs

Many WSN data compression schemes exploit the correlation among monitoring results to condense them [15]. To do so, several research efforts allow sensors to 'code' their data. For example, S-LZW [17] modifies the popular dictionary-based *Lempel-Ziv-Welch (LZW) algorithm* [18] for data compression. In [19], Huffman coding is used to reduce the size of monitoring results. On the other hand, both DIMENSIONS [20] and MRCQ (multi-resolution compression and query) [21] organize the WSN into a hierarchial architecture and view the monitoring results transmitted from sensors as an image with multiple pixels. Then, they utilize wavelet transformations to obtain the features of monitoring results and discard minor ones so as to reduce the data size. In addition, the Slepian-Wolf theorem [22] shows that any two correlated data streams can be encoded independently and then be decoded jointly at a receiver with a rate equal to their joint entropy. This property is widely used in [23] and [24] to support data compression in WSNs. As mentioned above, these data compression schemes aim at how to reduce the amount of data generated by sensors, but they do not consider reducing packet delays caused by the compression operation. This motivates us to develop an efficient route selection strategy to address the delay issue when applying the data compression technique to WSNs.

B. Network Assumptions

We consider a WSN used to continuously monitor the region of interest, where sensors have to regularly report their monitoring results to the sink. However, sensors may have different generating rates of monitoring results because of, for example, hardware designs or application requirements. More specifically, we assume that each sensor s_i will generate one monitoring result every v_i seconds. In addition, sensor s_i has to maintain a small buffer to keep its monitoring results for the compression purpose, where we use the variable b_i to denote the length of sensor s_i's buffer (i.e., how many monitoring results residing in the buffer). Suppose that a data compression scheme is adopted to reduce the size of data transmitted by sensors, and this scheme requires to deal with at least $\alpha \in \mathbb{N}$ monitoring results together so that they can be compressed into one packet. Here, we assume that the threshold value α is known by all sensors during the configuration phase. Each sensor s_i can choose to 1) keep its monitoring results in the local buffer until reaching α monitoring results or 2) send the monitoring results to one neighbor, say, s_j. In case 1, sensor s_i will do the compression by itself and send one compressed packet to its neighbor. Notice that this packet cannot be further compressed so that it will be directly relayed to the sink (probably through multiple hops). In case 2, sensor s_j may do compression for sensor s_i by combining its monitoring results together with s_i's monitoring results.

III. THE PROPOSED ROUTE SELECTION STRATEGY

Our route selection strategy can be applied to most ad hoc routing protocols since it aims at selecting the most suitable neighbor to help compress monitoring results so as to reduce packet delays. In this paper, we use the popular AODV routing protocol as the example to conduct our route selection strategy, where AODV lets sensors exchange RREQ (route request) and RREP (route reply) packets to construct routing paths.

Since all sensors only know the α threshold value in advance, they have to exchange some information from their one-hop neighbors. Therefore, our route selection strategy can be executed in a distributed manner. Then, each sensor will determine whether to keep its monitoring result(s) or to relay these results to one of the neighbors according to the exchanged information. In particular, our route selection strategy involves the following four steps:

- **Step 1:** Each sensor s_i announces its generating rate v_i of monitoring results and the current buffer length b_i to its one-hop neighbors. There are two possible manners to deal with the above announcement. One is to let sensor s_i periodically broadcast *hello messages* containing its v_i and b_i values. The other solution is to embed both v_i and b_i values in the RREP packets.
- **Step 2:** Sensor s_i then calculates the potential queuing time of the compressed packet if it chooses to keep all monitoring results in the buffer:
$$Q_i^L = v_i \times (\alpha - b_i). \qquad (1)$$
In Eq. (1), it is clear that we should guarantee the relationship of $b_i < \alpha$. In case of $b_i \geq \alpha$, sensor s_i immediately compresses α monitoring results into one packet and sends the packet to the neighbor following the AODV routing protocol.
- **Step 3:** Let us denote by $c(i, j)$ the data rate of the communication link between sensors s_i and s_j. To estimate $c(i, j)$, sensor s_i can compute the average number of packets successfully transmitted through the link (s_i, s_j) in the near past. Alternatively, we could apply the analyses in [25], [26] to predicting the link data rate. Then, for each one-hop neighbor s_j closer to the sink (this can be known by the AODV protocol), sensor s_i calculates the potential queuing time of the compressed packet if it chooses to send the current monitoring results (in the buffer) to sensor s_j as follows:
$$\begin{aligned} Q_{i,j}^R = &v_j \times (\alpha - (b_j + \min\{(\alpha - b_j), b_i\}))+ \\ &\left(\frac{\min\{(\alpha - b_j), b_i\}}{c(i, j)}\right). \end{aligned} \qquad (2)$$
In Eq. (2), the first term indicates the time that sensor s_j has to wait for compression if it receives the monitoring results from sensor s_i, and the second term is the time spent to transmit the monitoring results from sensor s_i to sensor s_j. Here, since it is possible that $b_i + b_j > \alpha$, which means that sensor s_i has more monitoring results than sensor s_j wants to get. In this case, sensor s_i only needs to send $(\alpha - b_j)$ monitoring results to sensor s_j for compression. That is why we select the minimum value between $(\alpha - b_j)$ and b_i in Eq. (2).
- **Step 4:** By combining Eqs. (1) and (2), sensor s_i can choose between keeping the monitoring results or sending them to a neighbor according to the following equation:
$$\arg\min \left\{ Q_i^L, \min_{s_j \in N_i} \{Q_{i,j}^R\} \right\}, \qquad (3)$$
where N_i is the set of sensor s_i's one-hop neighbors which are closer to the sink. In particular, if Q_i^L is the minimum value, then sensor s_i should accumulate its monitoring results in the local buffer for future compression. Otherwise, s_i should send its monitoring results to the neighbor s_j such that $Q_{i,j}^R$ is the minimum value, where the number of monitoring results that should be transmitted is determined by Eq. (2).

We use Fig. 1 again as the example to illustrate how our route selection strategy works, but here we assume that sensor A already has three monitoring results in its buffer. Suppose that $\alpha = 5$ and the link data rates are as follows: $c(A, B) = c(A, C) = 1$ packet/second. Then, according to Eq. (1), we can derive that

$$Q_A^L = 6 \times (5 - 3) = 12 \text{ seconds.}$$

In addition, from Eq. (2), we can calculate that

$$Q_{A,B}^R = 5 \times (5 - (3 + 2)) + \frac{2}{1} = 2 \text{ seconds.}$$
$$Q_{A,C}^R = 3 \times (5 - (1 + 3)) + \frac{3}{1} = 6 \text{ seconds.}$$

Therefore, by Eq. (3), sensor A should send out two monitoring results to sensor B so that sensor B can directly compress five monitoring results into one packet. On the other hand, if we change the link data rates by $c(A, B) = 0.4$ packets/second and $c(A, C) = 3$ packets/second, then we can recalculate the potential queuing delays for sensors B and C as follows:

$$Q_{A,B}^R = 5 \times (5 - (3 + 2)) + \frac{2}{0.4} = 5 \text{ seconds.}$$
$$Q_{A,C}^R = 3 \times (5 - (1 + 3)) + \frac{3}{3} = 4 \text{ seconds.}$$

In this case, sensor A should transmit all of its three monitoring results to sensor C so as to reduce the packet delay. From the above two examples, we can observe that our route selection strategy can dynamically select the 'best' neighbor to relay monitoring results for compression, according to the buffer sizes and link data rates of sensors.

IV. SIMULATION STUDIES

We use the network simulator (NS-2) [27] to evaluate the performance of our route selection strategy. The region of interest is a rectangle whose length is 150 meters and width is 100 meters. There are 150 sensors deployed in the region of interest using the hexagon-based deployment [28], and their communication ranges are set to 10 meters. The data generating rates of sensors can be 0.01 Mbps, 0.02 Mbps, and 0.03 Mbps. We use CSMA/CA (carrier sensing multiple access with collision avoidance) as the MAC-layer protocol for sensors, which is also adopted in the IEEE 802.15.4 standard [29]. We compare three different schemes in the experiments:

- **AODV (without compression):** Sensors do not use any data compression method to reduce the amount of data transmitted from them.
- **AODV with compression:** Sensors adopt the AODV protocol to transmit their data. However, each sensor has

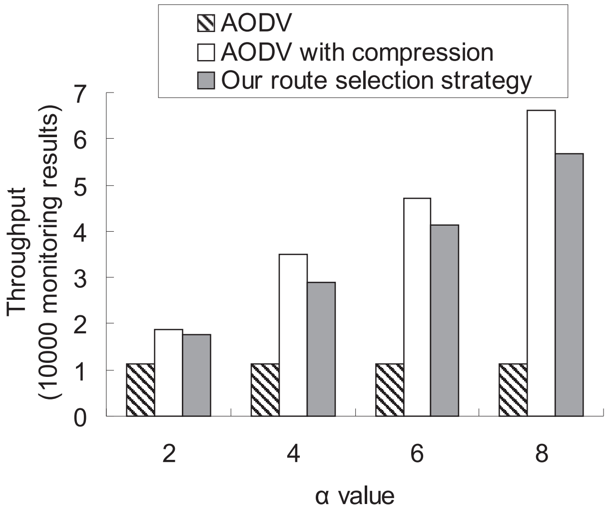

Fig. 2: Comparison on the network throughput (in the number of monitoring results) by different schemes.

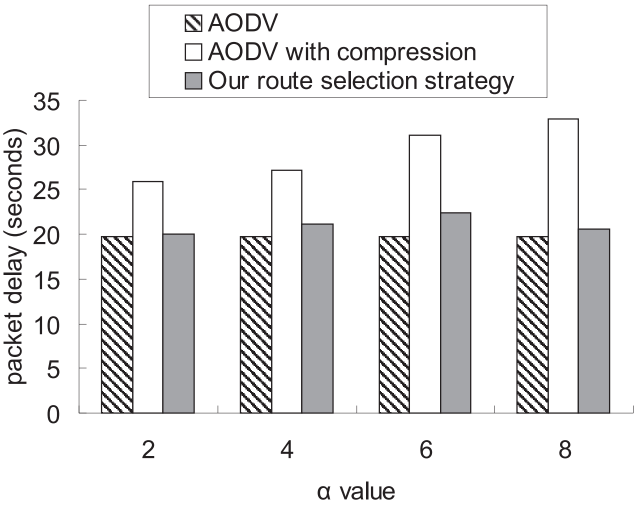

Fig. 3: Comparison on the average packet delay by different schemes.

to accumulate up to α monitoring results so that it can compress these monitoring results into one packet.

- **AODV with our route selection strategy:** The AODV protocol helps each sensor find the nearest neighbors to the sink, and our route selection strategy then either picks the best neighbor to relay and compress data or asks the sensor to keep the monitoring results in the local buffer for compression by itself.

We mainly measure the *network throughput* and the *average packet delay*. Here, the network throughput is defined by the total number of monitoring results successfully transmitted to the sink. In this case, if the sink correctly receives a compressed packet, it means that α monitoring results are successfully sent to the sink. On the other hand, the delay of a compressed packet is defined by the longest delay of its monitoring results. The delay of a monitor result is defined by the period from the time that it is generated by a sensor to the time that it is received by the sink. The total simulation time is 300 seconds in our simulations.

Fig. 2 shows the network throughput by different schemes when we change the α threshold value. Apparently, the network throughput of AODV does not change as the α value increases because it does not adopt the data compression scheme. In this case, since each sensor has to transmit a large amount of monitoring results, the network becomes very congested and therefore only a few amount of monitoring results can successfully arrive at the sink. That is why AODV always has the lowest network throughput. On the contrary, when we apply the data compression scheme to AODV, the network throughput is significantly improved because the aforementioned network congestion situation can be alleviated. Such improvement is more obvious when the α value increases, since each sensor can compress more monitoring results together. On the other hand, when applying our route selection strategy to AODV, the network throughput slight decreases (as compared to AODV with data compression). The reason is that our route selection strategy allows sensors to transmit (uncompressed) monitoring results to their neighbors for compression, which may sometimes let more packets contend with the wireless medium.

Fig. 3 presents the average packet delay by different schemes when the α value ranges from two to eight. Again, increasing the α value has no effect on AODV because it does not use the data compression scheme. Although AODV has the shortest packet delay, it in fact drops a lot of packets due to serious network congestion (this can be verified from Fig. 2). Since monitoring results do not need to wait for compression in AODV, if they can be successfully received by the sink, they of course suffer from lower delays. When applying data compression to AODV, it significantly increases the average packet delay. Obviously, since each sensor has to accumulate up to α monitoring results in order to compress them, the existing monitoring results have to stay in the buffer for a long time. The situation becomes worse when sensors generate monitoring results in a lower data rate. By allowing sensors to send monitoring results to neighbors for speeding up the compression process, our route selection strategy can reduce the average packet delay. From Fig. 3, it can be observed that our route selection strategy only slightly increases the average packet delay compared with AODV (without compression). This demonstrates the effectiveness of our route selection strategy.

V. CONCLUSION

The data compression technique can significantly reduce the amount of data transmitted by sensors, thereby alleviating network congestion while saving the energy of sensors. However, in this paper we point out the problem that applying data compression to WSNs may substantially increase packet delays. To solve this problem, we develop an efficient route selection strategy to help sensors determine whether to keep monitoring results for compression by themselves or to send monitoring results to neighbors in order to facilitate the data compression process. Through simulations, we show that our route selection strategy can indeed reduce the average packet delay with the expense of slightly decreasing the network throughput.

ACKNOWLEDGEMENT

You-Chiun Wang's research is co-sponsored by the Ministry of Science and Technology under grant no. 100-2218-E-110-006-MY3, Taiwan.

REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–114, 2002.

[2] Y. C. Wang and Y. C. Tseng, "Intentional mobility in wireless sensor networks," in *Wireless Networks: Research, Technology and Applications*. USA: Nova Science Publishers, 2009.

[3] Y. C. Wang, F. J. Wu, and Y. C. Tseng, "Mobility management algorithms and applications for mobile sensor networks," *Wireless Communications and Mobile Computing*, vol. 12, no. 1, pp. 7–21, 2012.

[4] Y. C. Wang, "Mobile sensor networks: system hardware and dispatch software," *ACM Computing Surveys*, vol. 47, no. 1, pp. 12:1–12:36, 2014.

[5] J. Burrell, T. Brooke, and R. Beckwith, "Vineyard computing: sensor networks in agricultural production," *IEEE Pervasive Computing*, vol. 3, no. 1, pp. 38–45, 2004.

[6] Y. C. Tseng, Y. C. Wang, K. Y. Cheng, and Y. Y. Hsieh, "iMouse: an integrated mobile surveillance and wireless sensor system," *IEEE Computer*, vol. 40, no. 6, pp. 60–66, 2007.

[7] L. W. Yeh, Y. C. Wang, and Y. C. Tseng, "iPower: an energy conservation system for intelligent buildings by wireless sensor networks," *International Journal of Sensor Networks*, vol. 5, no. 1, pp. 1–10, 2009.

[8] B. M. Howe, Y. Chao, P. Arabshahi, S. Roy, T. McGinnis, and A. Gray, "A smart sensor web for ocean observation: fixed and mobile platforms, integrated acoustics, satellites and predictive modeling," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 3, no. 4, pp. 507–521, 2010.

[9] S. C. Hu, Y. C. Wang, C. Y. Huang, and Y. C. Tseng, "A vehicular wireless sensor network for CO_2 monitoring," *Proc. IEEE Conference on Sensors*, pp. 1498–1501, 2009.

[10] S. C. Hu, Y. C. Wang, C. Y. Huang, and Y. C. Tseng, "Measuring air quality in city areas by vehicular wireless sensor networks," *Journal of Systems and Software*, vol. 84, no. 11, pp. 2005–2012, 2011.

[11] Y. Zou and K. Chakrabarty, "A distributed coverage- and connectivity-centric technique for selecting active nodes in wireless sensor networks," *IEEE Transactions on Computers*, vol. 54, no. 8, pp. 978–991, 2005.

[12] S. He, J. Chen, and Y. Sun, "Coverage and connectivity in duty-cycled wireless sensor networks for event monitoring," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 3, pp. 475–482, 2012.

[13] Y. C. Wang and Y. C. Tseng, "Distributed deployment schemes for mobile wireless sensor networks to ensure multilevel coverage," *IEEE Transactions on Parallel and Distributed Systems*, vol. 19, no. 9, pp. 1280–1294, 2008.

[14] Y. C. Wang, Y. Y. Hsieh, and Y. C. Tseng, "Compression and storage schemes in a sensor network with spatial and temporal coding techniques," *Proc. IEEE Vehicular Technology Conference*, pp. 148–152, 2008.

[15] Y. C. Wang, "Data compression techniques in wireless sensor networks," in *Pervasive Computing*. USA: Nova Science Publishers, 2012.

[16] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc on-demand distance vector (AODV) routing," *IETF RFC 3561*, 2003.

[17] C. M. Sadler and M. Martonosi, "Data compression algorithms for energy-constrained devices in delay tolerant networks," *Proc. ACM International Conference on Embedded Networked Sensor Systems*, pp. 265–278, 2006.

[18] T. A. Welch, "A technique for high-performance data compression," *IEEE Computer*, vol. 17, no. 6, pp. 8–19, 1984.

[19] F. Marcelloni and M. Vecchio, "A simple algorithm for data compression in wireless sensor networks," *IEEE Communications Letters*, vol. 12, no. 6, pp. 411–413, 2008.

[20] D. Ganesan, B. Greenstein, D. Estrin, J. Heidemann, and R. Govindan, "Multiresolution storage and search in sensor networks," *ACM Transactions on Storage*, vol. 1, no. 3, pp. 277–315, 2005.

[21] Y. C. Wang, Y. Y. Hsieh, and Y. C. Tseng, "Multiresolution spatial and temporal coding in a wireless sensor network for long-term monitoring applications," *IEEE Transactions on Computers*, vol. 58, no. 6, pp. 827–838, 2009.

[22] D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," *IEEE Transactions on Information Theory*, vol. 19, no. 4, pp. 471–480, 1973.

[23] Z. Xiong, A. D. Liveris, and S. Cheng, "Distributed source coding for sensor networks," *IEEE Signal Processing Magazine*, vol. 21, no. 5, pp. 80–94, 2004.

[24] J. Zheng, P. Wang, and C. Li, "Distributed data aggregation using Slepian-Wolf coding in cluster-based wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 5, pp. 2564–2574, 2010.

[25] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, pp. 535–547, 2000.

[26] F. Wang, D. Li, and Y. Zhao, "Analysis of CSMA/CA in IEEE 802.15.4," *IET Communications*, vol. 5, no. 15, pp. 2187–2195, 2011.

[27] The Network Simulator (NS2). [Online]. Available: http://www.isi.edu/nsnam/ns/

[28] Y. C. Wang, C. C. Hu, and Y. C. Tseng, "Efficient deployment algorithms for ensuring coverage and connectivity of wireless sensor networks," *Proc. IEEE Wireless Internet Conference*, pp. 114–121, 2005.

[29] LAN/MAN Standards Committee of the IEEE Computer Society, "IEEE Std 802.15.4-2003, Wireless medium access control (MAC) and physical layer (PHY) specifications for low-rate wireless personal area networks (LR-WPANs)."