Link Loss Inference in Wireless Sensor Networks with Randomized Network Coding

Vahid Shah-Mansouri and Vincent W.S. Wong
Department of Electrical and Computer Engineering
The University of British Columbia, Vancouver, Canada
e-mail: {vahids, vincentw}@ece.ubc.ca

Abstract—Due to fading and interference, data transmission via wireless links may sometimes be prone to error. For some applications in wireless sensor networks, it is of interest to monitor the link status and infer the packet loss rate. It has been shown that randomized network coding can improve the reliability of wireless sensor networks with lossy links. With network coding, the loss rate of a chosen path in a wireless sensor network is the maximum link loss rate among all the links in that path. This behavior changes the link identification problem and imposes challenges on the link loss inference. In this paper, we study the passive loss tomography problem in coded packet wireless sensor networks. We show that by inspecting the content of the coded packets at the sink (i.e., destination), one can estimate the path loss rates not only from the source nodes but also from various intermediate nodes to the sink. By utilizing such information at the sink, we determine the set of links whose loss rates can be identified. We propose a passive loss inference with random linear network coding (PLI-RLC) algorithm to estimate the link loss rates. Results show that in coded packet wireless sensor networks, our proposed algorithm can identify the status of a higher number of links compared to a Bayesian inference algorithm.

I. INTRODUCTION

In wireless sensor networks, sensor nodes are battery-powered and prone to node failures. Due to the stochastic nature of the channels, communication links between sensors are also subject to fading and interference [1], [2]. Recent studies have shown that *network coding* can not only increase the network throughput in wired and wireless networks [3]–[5], but can also improve the *robustness* in systems with erasure channels [6]–[8]. A network node performs network coding when it combines incoming packets (e.g., using XOR operation) and creates an encoded packet. At the destination, the sink can decode and extract the original packets if it has received sufficient encoded packets with independent information. Network coding outperforms automatic repeat request (ARQ) and forward error correction (FEC) techniques in terms of a lower delay and a higher network throughput [8]. Network coding also outperforms erasure codes since intermediate sensor nodes do not need to decode packets. It reduces the complexity of operation at intermediate sensor nodes and reduces the end-to-end delay.

For both wireless and wired networks, it is of interest to monitor the behavior of the network, identify the link status, estimate the delay and loss rate of different links. Such a technique is called *network tomography* or *network monitoring*

in the literature. Network tomography for wired and wireless networks has received much attention recently [9]–[13]. Network monitoring is especially important for wireless sensor networks as the information can allow sensor nodes to make decisions on routing and to bypass the links which are either congested or prone to failure.

Network monitoring can be performed either in an *active* or *passive* manner [10]. Active network monitoring refers to the case where probe packets are transmitted by some chosen source nodes and are gathered from the destinations. Although active monitoring schemes may reveal more information about the link loss rates, they utilize additional network resources such as bandwidth and energy. On the other hand, for passive network monitoring, information from the received data packets can allow the destination to infer the link loss information [9]. Passive monitoring schemes are more attractive for wireless sensor networks because of the limited power supply and bandwidth constraints in these networks [14], [15]. The problem of supporting passive network monitoring by data aggregation techniques is studied in [16], [17]. Identification of poorly performing links using end-to-end observations and maximum likelihood estimate in a wireless sensor network is studied in [18]. However, none of the above works use network coding for network monitoring. In [19], Gui *et al.* used a linear algebraic approach and proposed an active monitoring scheme for packet loss inference on mesh topologies.

In this paper, we focus on designing algorithms to infer the loss rate of different links in wireless sensor networks by passively monitoring the traffic at the sink. All sensor nodes (including source nodes and intermediate nodes) are capable of performing randomized network coding. The link identification problem in coded packet networks is different from the traditional routed packet networks. In a coded packet network, the loss rate of a chosen path is the maximum loss rate of all the links of that path [8]. Therefore, it can be infeasible to infer the loss rate of the intermediate links by simply observing the path loss rate from the source node towards the sink. This behavior of the coded packet networks makes the link identification problem more challenging. However, we can benefit from the coding operations performed at the intermediate nodes and extract information about the path loss rate from the intermediate nodes to the sink. The contributions of this paper are as follows:

- By inspecting the contents of the received coded packets

at the sink and adopting the *subspace* properties of network coding, we propose an estimation technique to estimate the path loss rates not only from the source nodes but also from several intermediate nodes to the sink.

- We characterize the conditions whether the link loss rate of a particular link can be estimated.
- We propose a passive loss inference with random linear network coding (ML-RLC) algorithm using the proposed estimation technique to determine the loss rate of intermediate links in the network. Results show that our proposed algorithm has a lower percentage of false detection compared to a Bayesian inference algorithm [20].

This paper differs from the existing related work in the literature in several aspects. The problem of passive inference of the possible locations of link failures for multicast traffic is studied in [21]. Our work differs from [21] in that (a) we consider unicast traffic, and (b) sensor nodes are being able to buffer data packets and perform coding in order to improve the reliability of the system. In [20], the problem of passive loss inference in wireless sensor networks with network coding is studied. Our work differs from [20] in that we study the identifiability problem and compute the loss rate of links precisely instead of merely categorizing the links as either good or bad. To the best of our knowledge, this work is the first attempt to use the subspace properties in network coding for the link loss inference problem in wireless sensor networks.

The rest of the paper is organized as follows: In Section II, the system model is introduced. In Section III, we propose estimators to determine the path loss rate and an PLI-RLC algorithm to infer the link loss rate. In Section IV, we evaluate the performance of our proposed algorithm. Conclusions are given in Section V.

II. SYSTEM MODEL

A. Network Structure

Consider a wireless sensor network. Let $\mathcal{N} = \{1, 2, \dots, N\}$ denote the set of sensor nodes and $\mathcal{L} = \{1, 2, \dots, L\}$ denote the set of directed wireless links. Let $\mathcal{S}$ denote the set of source nodes where $\mathcal{S} \subset \mathcal{N}$. There is one sink node in the network. The data gathered at the sink over a directed tree routed at the sink. The leaves of the tree are the source nodes.

With a tree structure, there is a unique path from every source node to the sink. We call the packets transmitted from source s to the sink as the flow of source s . Let $\mathcal{P}$ denote the set of paths in the network. Let $P_s \in \mathcal{P}$ denote the path from source node s to the sink and $\mathcal{L}_s$ denote the set of links on path P_s . For any intermediate node $i \in \mathcal{N}$ in the tree, the upstream flows are flows of those sources in $\mathcal{S}$ using node i as a relay. Fig. 1 shows a sample wireless sensor network with nodes 1, 2, 3, 4, and 7 as the source nodes.

B. Randomized Network Coding

Coded packet networks can provide reliable data gathering for wireless sensor networks. Each source has a stream of data packets to transmit towards the sink. Each packet is a sequence

Fig. 1. A sample wireless sensor network where nodes 1, 2, 3, 4, and 7 are the sources.

of symbols over a Galois field $GF(q)$. For practical purposes, the source nodes group the packets into *generations* [7]. Each source transmits one generation at a time. When the sink can decode the packets of one generation, the source moves to the next generation. Let K denote the size of the generation. Every source transmits packets periodically. Without loss of generality, sources transmit with a rate of one packet per second.

Each sensor node can perform network coding by combining several incoming packets to create a coded packet. We assume the use of randomized network coding [22]. Each coded packet transmitted by node $i \in \mathcal{N}$ is a weighted linear combination of the packets in the buffer of node i using a set of random weights taken from $GF(q)$ called the *coding coefficients*. Nodes append the coding coefficients in the header of the coded packet to enable decoding at the sink. The vector of coefficients is called the *coding vector*. Each sensor node has several buffers to store the packets received from different upstream flows. Upon receiving a packet from flow s , the node checks whether the packet is linearly independent of the other packets currently stored in the buffer of flow s . An incoming packet is placed in the buffer if it is linearly independent of the currently buffered packets. When a node transmits a packet of flow s , it linearly combines the stored packets in the buffer of flow s using the random coding coefficients.

We slightly modify the randomized linear network coding for the monitoring purposes. In this modified version, an intermediate node with more than one incoming link randomly selects two flows from two different incoming links and combines the buffers of those flows. These flows do not have common information about one source since they are from different incoming links. We notice that the intermediate node still relays two separate flows. In this case, some of the transmitted packets have information of more than one source. Since we assume that the minimum loss rate is 0.5 in the network, the input rate for innovative packets of both flows which are combined is at least 0.5. Therefore, for each of the transmitted flows, the intermediate node always has innovative packets to transmit and acts as a source. We show that merging the buffers can help to derive information of the path from the intermediate node to the sink. The sink

receives packets from $|\mathcal{S}|$ flows and stores the packets of every flow s at buffer $\mathcal{B}_s$. Since some of the intermediate nodes have combined different flows, buffer of flow s may contain information of other sources as well. To decode the packets, the sink uses the buffers of all flows. The sink can decode the data if it has $K|\mathcal{S}|$ linearly independent packets in different buffers. We notice that the sink needs to keep different buffers for monitoring purposes but not for decoding purposes.

III. PROBLEM FORMULATION

The objective of our paper is to study the link identification and loss inference problem for wireless sensor networks using the information obtained at the sink. We first need to estimate the end-to-end path loss rates from the sources to the sink. Then the loss rate of links can be inferred using the information of path loss rates. In the next two subsections, we consider the path loss estimation and link identifiability problems.

A. Path Loss Estimation

In this subsection, the objective is to accurately estimate the loss rate of the paths in the wireless sensor networks by passively observing the data traffic received at the sink. This is based on tracking the propagation of packets known as *innovative packets* from different source nodes. Such packets are innovative as they carry new, yet unknown, information generated by the source nodes. Let $\Pi_s(t)$ represent the subspace spanned by those coefficients of the packets in $\mathcal{B}_s$ corresponding to the source s at time t . For example, if the first K coefficients of each coding vector belongs to the source $s \in \mathcal{S}$, then $\Pi_s(t)$ is the subspace spanned by the first K coefficients of the packets in $\mathcal{B}_s$ at time t .

The time at which the sink expects to receive the first packet from a source depends on the distance (i.e., number of hops) between the source node and the sink. Let t_s^1 denote the time at which the sink expects to receive the first packet of source s . Let $\bar{t}_s$ denote the time at which the subspace Π_s becomes full rank. The maximum rank of subspace Π_s is equal to the number of packets generated by source s which is K . That is $\dim(\Pi(\bar{t}_s)) = K$.

The rate at which buffer $\mathcal{B}_s$ receives innovative packets from source s is the minimum rate at which the nodes on the path P_s receives packets [8]. This is equal to the minimum rate at which packets are transmitted on the links in $\mathcal{L}_s$ multiplied by the corresponding link success rates. Since the rate at which intermediate nodes relay the packets of flow s is one, the rate at which buffer $\mathcal{B}_s$ receives innovative packets of source s is the minimum of the link success rate for the links in $\mathcal{L}_s$. This is also considered as the path loss rate of path P_s which can be written as¹:

$$1 - \alpha_s = \min_{l \in \mathcal{L}_s} \{1 - e_l\}, \quad \forall s \in \mathcal{S} \quad (1)$$

where α_s is the loss rate of the path P_s and e_l denotes the link loss rate of link l . We assume the link loss rate is less

¹In the case of routing without network coding, we have $(1 - \alpha_s) = \prod_{l \in \mathcal{L}_s} (1 - e_l)$.

than 0.5, since a link with loss rate greater than 0.5 should not be used in practice. At every second, buffer $\mathcal{B}_s$ receives an innovative packet of source s with probability α_s . Therefore, the dimension of subspace $\Pi(\bar{t}_s)$ has a negative binomial distribution and it can be used to estimate the path success rate as:

$$\tilde{\alpha}_s(t) = 1 - \frac{\dim(\Pi_s(t))}{t - t_s^1 + 1}, \quad \forall s \in \mathcal{S}. \quad (2)$$

where $\tilde{\alpha}_s(t)$ is the estimation of α_s at time t . We notice that for values of $t > \bar{t}_s$, since the dimension of the subspace is not incremented, the estimator uses the last time slot which contains useful information for estimation which is $\bar{t}_s$. Therefore, we have $\tilde{\alpha}_s(t) = \tilde{\alpha}_s(\bar{t}_s)$ for $t > \bar{t}_s$. The estimation error in $\tilde{\alpha}(t)$ asymptotically follows a normal distribution with zero mean and variance $\sigma_s^2(t)$ as [23]:

$$\sigma_s^2(t) \approx \frac{\alpha_s(1 - \alpha_s)}{t - t_s^1 + 1}, \quad \forall t \in [t_s^1, \bar{t}_s], s \in \mathcal{S}. \quad (3)$$

We notice that t in fact cannot go to infinity. The maximum value of t which can be used to estimate α_s is $\bar{t}_s$. At time $\bar{t}_s$, the network experiences the minimum achievable variance of estimation error. The minimum value for $\bar{t}_s$ which can be achieved when there is no loss is equal to the size of the generation K , which is a design parameter. Thus, higher values of K can increase the accuracy of the estimation. However, it can impose additional delay to the decoding. Also, the intermediate nodes need additional memory capacity and the sink needs more computational complexity for the decoding process. Hence, there is an inherent tradeoff between the accuracy and the complexity.

Given the path loss rates observed at the sink, it is challenging to infer the link loss rates as the number of variables (i.e., link loss rates) is more than the number of equations (i.e., path loss rates). From (1), the path loss rate α_s of a path P_s is an upper bound of the link loss rate e_l for all link $l \in \mathcal{L}_s$.

We now show that more information can be obtained for the link loss rates in the network by taking into account the properties of the coded packet networks. We call an intermediate node with more than one incoming link as a *virtual source*. We note that an intermediate node can be a source node as well. In this case, we do not consider it as a virtual source since this assignment cannot provide more information. Let $\mathcal{V}$ denote the set of virtual sources in the network. For virtual source v , let P_v denote the path from the virtual source to the sink and $\mathcal{L}_v$ denote the set of links constructing path P_v . From the randomized network coding, every virtual source $v \in \mathcal{V}$ combines the packets of two upstream flows. Therefore, among $|\mathcal{S}|$ flows received at the sink, there exists at least one flow which carries information of more than one upstream flows of the virtual source v . Let $\mathcal{B}_v$ denote the buffer for that flow at the sink and let $\mathcal{S}_v$ denote the set of flows whose buffers have been combined by the node v . We notice that this flow may contain information of other sources as well. For a virtual source $v \in \mathcal{V}$, let $\Pi_v(t)$ denote the subspace spanned by those coefficients of the packets in

$\mathcal{B}_v$ corresponding to the sources in S_v at time t . Let t_v^1 denote the time at which the sink expects to receive the first packet in $\mathcal{B}_v$. Let $\bar{t}_v$ denote the time at which the subspace Π_v becomes full rank.

The rate at which the node v receives the innovative packets of the sources in S_v is greater than or equal to one since the packets received are from two different flows each with rate at least equal to 0.5 (this is from the assumption that we made on the minimum loss rate of the links). Therefore, the packets transmitted from node v are always innovative packets with respect to the generated packets of sources in S_v . It means, the node v can be considered as a source generating packets which contribute to the dimension of Π_v . It means the behavior of the actual and virtual sources are similar while we monitor the dimension of Π_v instead of Π_s . We can use the following estimator to estimate the path loss rate for a path from virtual source v to the sink

$$\tilde{\alpha}_v(t) = 1 - \frac{\dim(\Pi_v(t))}{t - t_v^1 + 1} \quad \forall v \in \mathcal{V}, \quad (4)$$

where $\tilde{\alpha}_v(t)$ is the estimate of α_v at time t . We notice that similar to $\alpha_s(t)$, for values of $t > \bar{t}_s$, we have $\alpha_v(t) = \alpha_v(\bar{t}_v)$. Similar to the approximation proposed for the variance of the error obtained for $\tilde{\alpha}_s(t)$, the variance of error in the estimation of α_v can be obtained as follows:

$$\sigma_v^2(t) \approx \frac{\alpha_v(1 - \alpha_v)}{t - t_v^1 + 1}, \quad \forall t \in [t_v^1, \bar{t}_v], \quad v \in \mathcal{V}. \quad (5)$$

B. Identifiability Problem

The main objective of this paper is infer the loss rate of the network links. However, it is not always possible to infer loss rate for all the links using end-to-end measurements. Given the topology of the network, we can classify the link in three sets: links which are *not identifiable*, links which are *possibly identifiable*, and links which are *identifiable*. A link l is not identifiable if for any assignment for the loss rates of the links in the network, we are not able to infer the link loss rate of link l using end-to-end measurement. The possible identification is a new notion of identifiability introduced in this paper. A link is possibly identifiable if the identifiability of the link depends on the loss rate of other links. It means that the link loss rate can be inferred for some assignments, but not for other assignments. For an identifiable link, the loss rate can always be inferred independent of the loss rate of other links.

For each directed link $l \in \mathcal{L}$, let $\text{head}(l)$ and $\text{tail}(l)$ denote the nodes attached to the head and tail of link l , respectively. That is, $\text{head}(l) \in \mathcal{N}$ denotes the transmitter node of link l and $\text{tail}(l) \in \mathcal{N}$ denotes the receiver node of link l . The following lemma helps to distinguish between these three groups of links.

Lemma 1: For the directed link $l \in \mathcal{L}$:

- (a) It is *identifiable* if $\text{head}(l) \in \mathcal{S} \cup \mathcal{V}$ and $\text{tail}(l)$ is the sink.
- (b) It is *possibly identifiable* if $\text{head}(l)$ and $\text{tail}(l) \in \mathcal{S} \cup \mathcal{V}$.
- (c) Otherwise, it *not identifiable*.

Algorithm 1 - Passive Loss Inference with Random Linear Network Coding (PLI-RLC) Algorithm

```

1: input: network topology, sets  $\mathcal{V}$  and  $\mathcal{V}$ ; output: link loss rates
2: for every generation
3:   While at time slot  $t$ 
4:     for each source node  $s \in \mathcal{S}$ 
5:       if  $t \geq t_s^1$ ,
6:         Calculate  $\dim(\Pi_s(t))$ .
7:         Estimate  $\alpha_s(t)$  using (2).
8:         Calculate  $\sigma_s^2(t)$  using (3) and  $\alpha_s(t)$ .
9:     end for
10:    for each virtual source  $v \in \mathcal{V}$ 
11:      if  $t \geq t_v^1$ ,
12:        Calculate  $\dim(\Pi_v(t))$ .
13:        Estimate  $\alpha_v(t)$  using (4).
14:        Calculate  $\sigma_v^2(t)$  using (5) and  $\alpha_v(t)$ .
15:    end for
16:    if  $t \geq \max_{i \in \mathcal{S} \cup \mathcal{V}} t_i^1$  and  $\sigma_i^2(t) < E_{th}$ ,  $\forall i \in \mathcal{S} \cup \mathcal{V}$ 
17:      Break while.
18:    end while
19:    for each link  $l \in \mathcal{L}$ ,
20:      if  $(\text{head}(l), \text{tail}(l)) \in \mathcal{S} \cup \mathcal{V}$ 
21:        if  $\alpha_{\text{head}(l)} > \alpha_{\text{tail}(l)}$ ,
22:          Set  $e_l := \alpha_{\text{head}(l)}$ .
23:        end if
24:      end for

```

The proof of Lemma 1 is given in Appendix A. Using the estimators presented in Section III-A and Lemma 1, we propose the passive loss inference with random linear network coding (ML-RLC). Algorithm 1 shows the PLI-RLC algorithm. The algorithm monitors the incoming packets to the sink and estimates the loss rate of the links. The algorithm is composed of two parts. In the first part (lines 3–18), the algorithm calculates the dimension of all subspaces at every time slot according to the buffered packets and the packets which are received in that time slot. Then, it estimates the packet loss rate based on the dimension of the subspaces. We notice that the PLI-RLC algorithm is a passive monitoring algorithm and time t is the system time instead of the algorithm time. If at a time instance t , the variance of estimation error of all the path loss rates obtained using equation (3) is less than a certain threshold E_{th} , then the algorithm stops monitoring the received packets. In the second part of the algorithm (lines 19–23), the algorithm determines the loss rates.

IV. PERFORMANCE EVALUATION

In this section, we evaluate the performance of the proposed estimators and our proposed PLI-RLC algorithm. Moreover, we compare our proposed PLI-RLC algorithm with the Bayesian inference algorithm [20]. We developed a discrete-event packet level simulator for the wireless sensor networks using MATLAB. For randomized network coding, the size of the generation K is set to 100. The Galois field size q is equal to 8. We set the threshold value E_{th} to be equal to 5×10^{-4} .

A. Number of Identifiable Links

We investigate the performance of PLI-RLC algorithm on inferring the link loss rates. We consider the network in Fig. 1.

Fig. 2. Average number of links whose rates determined using PLI-RLC algorithm.

For the link loss rate model, we follow the model proposed in [17]. For each link $l \in \mathcal{L}$, the probability distribution function that the loss rate of the link becomes e_l is

$$f(e_l) = \lambda(1 - e_l)^{(\lambda-1)}, \quad e_l \in (0, 1], \quad (6)$$

where λ is a tunable parameter. The mean of the link loss error rate is $1/(\lambda + 1)$. The link loss rate on each link is randomly chosen from the distribution in (6). If the chosen link loss rate is less than 0.5, we set it to 0.5. We vary the value of λ from 5 to 10. The PLI-RLC algorithm is run 100 times for each value of λ . Fig. 2 shows the average number of links whose loss rates can be inferred using our proposed PLI-RLC algorithm. Higher values of λ decreases the average of link loss rate in the network. When λ is increased from 5 to 10, the average number of link with determined loss rate is slightly decreased from 4.12 to 4.04. We notice that there are 2 identifiable links and 5 possibly identifiable links in this network. It means, from 5 possibly identifiable links, on average the link loss rate of 2 can be determined by PLI-RLC algorithm.

B. Performance Comparison

In this section, we compare our proposed PLI-RLC algorithm with the Bayesian inference algorithm proposed in [20]. In [20], the posterior probability of the link loss rate is derived given the observed path loss rates from sources to the sink. Then, a link is categorized as a good link if with probability greater than 0.5 the link loss rate is greater than a threshold γ . To compare our scheme with the one proposed in [20], we classify each link to be either a good or a bad link using the observations obtained from the PLI-RLC algorithm. For link $l \in \mathcal{L}$ whose loss rate is determined by the algorithm, we classify the link as a good link if the loss rate of the link is higher than γ . For a link $l \in \mathcal{L}$ whose link loss rate cannot be determined using PLI-RLC algorithm, we derive the posterior probability of loss rate e_l using the observed path loss

Fig. 3. Comparison of false detection for PLI-RLC algorithm and Bayesian inference algorithm [20].

rates. Let $f(e_l | \{\alpha_i, \forall i \in \mathcal{S} \cup \mathcal{V}\})$ denote the probability distribution of link loss rate e_l for link l whose rate cannot be determined using the PLI-RLC algorithm. Assuming no a priori information on e_l , the posteriori probability distribution function of e_l can be written as

$$f(e_l | \{\alpha_i, \forall i \in \mathcal{S} \cup \mathcal{V}\}) = \begin{cases} \frac{1}{\min_{\{i \in \mathcal{S} \cup \mathcal{V} | l \in \mathcal{L}_i\}} \{\alpha_i\}}, & \text{if } e_l \leq \min_{\{i \in \mathcal{S} \cup \mathcal{V} | l \in \mathcal{L}_i\}} \{\alpha_i\}, \\ 0, & \text{otherwise.} \end{cases}$$

Link l is classified as a bad link if:

$$\int_0^\gamma f(e_l | \{\alpha_i, \forall i \in \mathcal{S} \cup \mathcal{V}\}) de_l \geq 1/2. \quad (7)$$

We use the PLI-RLC algorithm along with equation (7) to classify all the links as either good or bad links. We compare the performance of LPI-RLC and Bayesian inference algorithms using the *ratio of false detection*. A detection is false if we declare a good link as a bad link, or vice versa. We consider a wireless sensor network with 50 sensor nodes randomly deployed in a 100 m $\times$ 100 m field. 25 sensor nodes are randomly selected as the sources. Each source node selects the path with minimum power consumption among all the paths it has towards the sink. The link loss rates are selected randomly based on the distribution function in equation (6). The value of λ is varied from 5 to 10 and we measure the fraction of links whose status are detected incorrectly. The threshold value γ is set to 0.7. The results are averaged over 100 simulation runs. We notice that the number of links which participate in data communication in each simulation run depends on the topology of the network and the routing tree.

Fig. 3 compares PLI-RLC and Bayesian inference algorithms for varying λ . Higher values of λ reduce the number of bad links in the network. As Fig. 3 shows, the false detection ratio is decreased when λ is increased. When λ is equal to 5,

32% of the links are bad links and 68% of them are good links. In this case, our proposed PLI-RLC algorithm has 14% less false detection compared to the Bayesian inference algorithm. When λ is equal to 10, 11% of the links are bad links and 89% of them are good links and our proposed algorithm performs 11% better than the Bayesian inference algorithm in terms of false detection.

V. CONCLUSION

In this paper, we studied the loss inference problem in wireless sensor networks while nodes performs randomized linear network coding. By using the subspace property of network coding, we proposed estimators to determine the path loss rate between the source and the sink. We characterized the conditions for a link to be either identifiable, possibly identifiable, or not identifiable. We proposed a passive loss inference with random linear network coding (PLI-RLC) algorithm to determine the link loss rates. Simulation results showed that our proposed algorithm is accurate. It also has a lower percentage of false detection compared to the Bayesian inference algorithm.

APPENDIX

A. Proof of Lemma 1

Consider a link $l \in \mathcal{L}$ with two actual/virtual sources at the head and tail. Let $\alpha_{\text{head}(l)}$ and $\alpha_{\text{tail}(l)}$ denote the path loss rate from the nodes located at the head and tail of l to the sink, respectively. The relationship between the path and link loss rates can be expressed as

$$\alpha_{\text{head}(l)} = \max\{e_l, \alpha_{\text{tail}(l)}\}, \quad (8)$$

where $\alpha_{\text{head}(l)}$ and $\alpha_{\text{tail}(l)}$ are observed values and e_l is an unknown variable. The loss rate of link l can be determined as $\alpha_{\text{head}(l)}$ if $\alpha_{\text{head}(l)} > \alpha_{\text{tail}(l)}$. However, if $\alpha_{\text{head}(l)} \leq \alpha_{\text{tail}(l)}$, then the loss rate can have any value less than or equal to $\alpha_{\text{head}(l)}$. It means that for link l , we have

$$e_l \begin{cases} = \alpha_{\text{head}(l)} & \text{if } \alpha_{\text{head}(l)} > \alpha_{\text{tail}(l)}, \\ \leq \alpha_{\text{head}(l)} & \text{if } \alpha_{\text{head}(l)} \leq \alpha_{\text{tail}(l)}. \end{cases} \quad (9)$$

Therefore, such a link is possibly identifiable. A link l is identifiable if for any assignment of link loss rates, we have $\alpha_{\text{head}(l)} > \alpha_{\text{tail}(l)}$. This only happens when $\alpha_{\text{tail}(l)} = 0$ which means $\text{tail}(l)$ is the sink. Now, consider link l while either $\text{head}(l)$ or $\text{tail}(l)$ is not actual/virtual source. Consider the case that $\text{tail}(l)$ is not an actual/virtual source. $\text{tail}(l)$ has one incoming link which is l . Let l' denote the outgoing link of node $\text{tail}(l)$. Any path including l should contain l' as well. Therefore, from the point of view of sink, l and l' are not distinguishable. Therefore, link l is not identifiable for any pattern of link loss rates. The case that $\text{head}(l)$ is not an actual/virtual source is the same as this case. ■

ACKNOWLEDGEMENT

This work is supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) and the British Columbia Innovation Council (BCIC).

REFERENCES

- [1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: A survey," *Computer Networks*, vol. 38, pp. 393–422, Mar. 2002.
- [2] A. Goldsmith and S. Wicker, "Design challenges for energy-constrained ad hoc wireless networks," *IEEE Trans. on Wireless Commun.*, vol. 9, pp. 8–27, Aug. 2002.
- [3] R. Ahlswede, N. Cai, S. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. on Information Theory*, vol. 46, pp. 1204–1216, July 2000.
- [4] R. Koetter and M. Medard, "An algebraic approach to network coding," *IEEE/ACM Trans. on Networking*, vol. 11, pp. 782–795, Oct. 2003.
- [5] J. Zhang, P. Fan, and K. Letaief, "Network coding for efficient multicast routing in wireless ad-hoc networks," *IEEE Trans. on Commun.*, vol. 56, no. 4, pp. 598–607, Apr. 2008.
- [6] D. S. Lun, M. Medard, R. Koetter, and M. Effros, "On coding for reliable communication over packet networks," *Physical Communication*, vol. 1, no. 1, pp. 3–20, Mar. 2008.
- [7] C. Fragouli, J.-Y. Le Boudec, and J. Widmer, "Network coding: An instant primer," *ACM SIGCOMM Computer Communication Review*, vol. 36, no. 1, pp. 63–68, Jan. 2006.
- [8] T. Ho and D. S. Lun, *Network Coding: An Introduction*. Cambridge University Press, 2008.
- [9] Y. Tsang, M. Coates, and R. Nowak, "Passive unicast network tomography using EM algorithm," in *Proc. of IEEE Int'l Conf. on Acoustics, Speech, and Signal Processing*, Salt Lake City, Utah, May 2001.
- [10] M. Coates, A. Hero III, R. Nowak, and B. Yu, "Internet tomography," *IEEE Signal Processing Mag.*, vol. 19, no. 3, pp. 47–65, May 2002.
- [11] Y. Chen, D. Bindel, H. Song, and R. Katz, "An algebraic approach to practical and scalable overlay network monitoring," in *Proc. of ACM SIGCOMM*, Portland, OR, Sept. 2004.
- [12] N. G. Duffield, J. Horowitz, F. Lo Presti, and D. Towsley, "Multicast topology inference from measured end-to-end loss," *IEEE Trans. on Information Theory*, vol. 48, no. 1, pp. 26–45, Jan. 2002.
- [13] N. Duffield, "Network tomography for binary network performance characteristics," *IEEE Trans. on Information Theory*, vol. 52, no. 12, pp. 5373–5388, Dec. 2006.
- [14] C. Hsin and M. Liu, "A distributed monitoring mechanism for wireless sensor networks," in *Proc. of ACM Workshop on Wireless Security*, Atlanta, GA, Apr. 2002.
- [15] J. Zhao, R. Govindan, and D. Estrin, "Computing aggregates for monitoring wireless sensor networks," in *Proc. of IEEE Int'l Workshop on Sensor Network Protocols and Applications*, Anchorage, AK, May 2003.
- [16] G. Hartl and B. Li, "Loss inference in wireless sensor networks based on data aggregation," in *Proc. of Int'l Conf. on Information Processing in Sensor Networks (IPSN)*, Berkeley, CA, Apr. 2004.
- [17] Y. Mao, F. R. Kschischang, B. Li, and S. Pasupathy, "A factor graph approach to link loss monitoring in wireless sensor networks," *IEEE J. on Selected Areas in Commun.*, vol. 23, no. 4, pp. 820–829, Apr. 2005.
- [18] H. X. Nguyen and P. Thiran, "Using end-to-end data to infer lossy links in sensor networks," in *Proc. of IEEE INFOCOM*, Barcelona, Spain, Apr. 2006.
- [19] J. Gui, V. Shah-Mansouri, and V.W.S. Wong, "An algebraic approach to loss tomography on mesh topologies," in *Proc. of IEEE ICC*, Cape Town, South Africa, May 2010.
- [20] Y. Lin, B. Liang, and B. Li, "Passive loss inference in wireless sensor networks based on network coding," in *Proc. of IEEE INFOCOM*, Rio de Janeiro, Brazil, Apr. 2009.
- [21] T. Ho, B. Leong, Y.-H. Chang, Y. Wen, and R. Koetter, "Network monitoring in multicast networks using network coding," in *Proc. of IEEE ISIT*, Chicago, IL, June 2004.
- [22] T. Ho, R. Koetter, M. Medard, D. R. Karger, and M. Effros, "The benefits of coding over routing in a randomized setting," in *Proc. of IEEE Int'l Symposium on Information Theory*, Yokohama, Japan, July 2003.
- [23] I. Blake, *An Introduction to Applied Probability*. John Wiley and Sons, 1987.