A Survey on Network Tomography with Network Coding

Peng Qin*, Bin Dai*, Benxiong Huang*, Guan Xu*, Kui Wu†
*Department of Electronics and Information Engineering
Huazhong University of Science and Technology, Wuhan, China
†Department of Computer Science
University of Victoria, Victoria, Canada

Email: {qinpeng, daibin, huangbx}@hust.edu.cn, guanxu86@gmail.com, wkui@cs.uvic.ca

Abstract—The overhead of internal network monitoring motivates techniques of network tomography. Network coding (NC) presents a new opportunity for network tomography as NC introduces topology-dependent correlation that can be further exploited in topology estimation. Compared with traditional methods, network tomography with NC has many advantages such as the improvement of tomography accuracy and the reduction of complexity in choosing monitoring paths. In this paper we first introduce the problem of tomography with NC and then propose the taxonomy criteria to classify various methods. We also present existing solutions and future trend. We expect that our comprehensive review on network tomography with NC can serve as a good reference for researchers and practitioners working in the area.

Index Terms—network tomography, network coding, topology recovery, link loss estimation, link delay inference, bottleneck discovery, failure localization

I. INTRODUCTION

Network tomography [1] studies internal characteristics of Internet using information derived from end nodes. One advantage of network tomography is that it requires no participation from network elements other than the usual forwarding of packets. This feature is particularly important, when anonymous internal routers [2] [3] do not respond to ICMP messages which are required by traditional traceroute based topology estimation methods.

Y. Vardi was one of the first to rigorously study the problem of inferring routing topology and coined the term network tomography [4] due to the similarity between network inference and medical tomography. According to the type of data acquisition and the performance parameters of interest [5], network tomography can be classified as a) link-level parameter estimation based on end-to-end, path-level traffic measurements [6], [7], and b) sender-receiver path-level traffic intensity estimation based on link-level traffic measurements [8], [9]. Based on whether or not explicit control messages are required, network tomography could be classified as active tomography [7], [10], [11] and passive tomography [8], [12], [13], [14]. The former needs to explicitly send out probing messages to estimate the end-to-end path characteristics, while the latter merely utilizes the regular data flow for further analysis. According to different application contexts, network tomography can also be categorized into link loss rates estimation [10], topology recovery [15], [16], and delay tomography [7], etc. In this field there are several issues in terms of tomography performance such as the accuracy of topology recovery, and probing complexity of active tomography. It is a non-trivial task to find an appropriate approach to deal with the performance problems.

Network Coding (NC) emerged at the beginning of the last decade with the primary aim of improving the throughput of networks [12], [13], [19], [20]. It breaks the tradition that intermediate nodes only forward data and the processing of information is performed only at end nodes. In multicast networks where simple linear operations of NC are performed on incoming packets, we can achieve the min-cut throughput of the network to each receiver. Since the receiver has to recover original packets by solving a system of linear equations over a finite field, NC packets introduce topology-dependent correlation which can be exploited for network tomography. Compared with traditional methods, network tomography based on NC has many advantages, such as the improvement of accuracy and the reduction of complexity in choosing monitoring paths [21]. These advantages motivate research in NC based tomography.

There are many surveys on NC and its applications in the literature. Tutorials on NC theory can be found in [22], [23], [24] while surveys of NC applications could be found in [25], [26]. Since we focus on applications of NC, research of NC theory is beyond the scope of this survey. Authors in [25] reviewed various NC applications such as content distribution and NC for wireless networks. However, its main purpose of applying NC is to enhance the performance for existing network systems.

In this paper we focus on the new emerging NC application areas especially in the domain of Network Tomography (NT) and sum up the state-of-the-art research on NC tomography. To the best of our knowledge, this is the first comprehensive survey on NC based NT problems.

In this paper we focus on the new emerging NC application areas especially in the domain of Network Tomography (NT) and sum up the state-of-the-art research on NC tomography. To the best of our knowledge, this is the first comprehensive survey on NC based NT problems.

The remainder of the paper is structured as follows. In Section I we introduce the problem of tomography with NC and propose the taxonomy criteria to classify various methods. Section III and Section IV present existent applications and new proposed applications of NT with NC, respectively. In Section V we provide some NC based methods that are in practical use and discuss lessons and existing problems which...
II. PROBLEM STATEMENT AND CLASSIFICATION

Prior work on NT considered networks that implement multicast and unicast forwarding. In this paper, we consider networks where internal nodes implement network coding and we re-visit some classic network tomography problems such as link loss inference and topology inference. We develop new techniques that make use of the network coding capabilities and show that they can improve several aspects of interest (including identifiability of links, accuracy of estimation, and complexity of probe path selection) over traditional techniques. We also seek to propose new tomography applications with network coding such as bottleneck discovery and failure localization. These extend the scope of traditional tomography.

We first summarize the basic idea of network coding as shown in Fig. 1. $S$ is the source node while $R_1$ and $R_2$ correspond to the receiver nodes. Intermediate node $C$ has two input links and one output link, and we call nodes like $C$ the coding nodes. On the other hand, nodes $A$, $B$ and $D$ only forward received packets without coding. $S$ transmits original packets $x_1$ and $x_2$ to its two output links ($S$, $A$) and ($S$, $B$), respectively, and by encoding $x_1$, $x_2$ into $x_1 \oplus x_2$ with XOR operation at coding node $C$, $R_1$ receives $x_1$ and $x_1 \oplus x_2$ while $R_2$ receives $x_2$ and $x_1 \oplus x_2$. In this way, we obtain a system of equations at receiver $R_1$ shown in Eq. (1) and the original packets of $x_1$, $x_2$ can be retrieved by solving it.

$$
\begin{align*}
\begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix} &=
\begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 1 \\ 0 & 1 \end{pmatrix}
\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}
\end{align*}
$$

According to the performance parameters of interest, implementation manners, and application scenarios, network tomography with network coding (NCT) can be classified into the following two categories (the existing applications and proposed new applications of NCT) with 10 subclasses, as shown in Fig. 2.

Active NT needs to explicitly send out probing messages to estimate the end-to-end path characteristics, while passive NT merely utilizes the regular data flow for further analysis. Compared with single-source methods, multiple source network tomography can infer more accurate topology and provide more information for improving network performance. Topology inference (TI) is the core part of NT technology and is the basic step to other performance inference such as the link loss tomography. For Link Loss Inference (LLI), the reason why we analyze wireless network scenario separately is that NC changes the fundamental mode between end-to-end observations and network characteristics from $\beta = \Pi_{i \in P}(1 - x_i)$ to $\beta = \min_{i \in P}(1 - x_i)$, where $\beta$ and $x_i$ denote the path successful transmission probability and the link loss probability, respectively. It means that the path successful transmission probability is not the product of all link successful transmission probabilities but the minimum of all link successful transmission probabilities on this path. For link delay inference (LDI), which is an important parameter for performance evaluation and load balance, we introduce a DCE measurement method [27] in our previous work and discuss the trend for adding NC in future.

To further exploit benefits of NC, we also present new application scenarios for NCT in addition to the above traditional tomography areas. For example, in a NC based P2P network [28], we are able to make use of the subspace characteristics for bottleneck discovery and re-route packets in a distributed manner with less overhead.

III. EXISTENT APPLICATIONS OF NT WITH NC

A. Topology Inference (TI)

Topology identification is the core component of network tomography technology and is the first step to other performance inference such as the link loss tomography.

1) Active TI with tree structure: For active detection in traditional network tomography area, probing packets are usually sent to multiple receivers by a multicast tree, and then they are used to recover the topology structure with information of received data packets at different nodes. These methods require that each receiver should obtain enough probing packets. Comparatively, authors in paper [29] propose a method of TI with NC for a tree structure. The topology is discovered by sending probes between multiple sources and receivers at the edge of the network, while intermediate nodes locally combine incoming probes before forwarding them. Since NC brings topology dependence into data packets which can be observed at the receiver nodes, this information is used to infer the network structure. The basic idea is described as follows.

Consider the network shown in Fig. 3(a). Assume that nodes $A$ and $P$ act as sources while the rest nodes as receivers. Thus, nodes $A$ and $P$ send $x_1=[1 \ 0]$ and $x_2=[0 \ 1]$ respectively. Intermediate nodes duplicate and forward the arriving packet if only one packet is received. If two packets arrive at a node it will perform NC operations and forward the NC coded packet. In this case since $x_1$ and $x_2$ meet at node $H$, leaf nodes $B$, $C$, $D$ will receive packet $x_1$, leaf nodes $M$, $O$ will receive packet $x_2$ and leaf nodes $K$, $L$ will receive packet $x_3 = x_1 \oplus x_2=[1 \ 1]$. Thus, the tree will be divided into three areas, $\Lambda_1 = \{A, B, C, D\}$ containing $x_1$, $\Lambda_2 = \{M, O, P\}$ containing $x_2$, and $\Lambda_3 = \{K, L\}$ containing $x_3$, as shown in Fig. 3(b).

To infer the structure that connects leaf nodes $\{A, B, C, D\}$ to node $G$ and the structure that connects leaf nodes...
If one area $A_i$ contains only one or two leaf nodes, it can be replaced with either one or two edges. The overall topology of Fig. 3(a) can be deduced from Fig. 3(d) by removing vertices of degree two.

2) Active TI with general network structure: To extend TI research from the tree structure to general networks, authors in paper [30] propose a tomography scheme with NC for directed acyclic graphs with multiple sources and multiple receivers.

The problem of TI in general Internet-like topology is divided into two steps. The first step is built on the observation that any M-by-N network can be decomposed into a collection of 2-by-2 sub-network components [31], [32], each of which could be one of the four possible types shown in Fig. 4.

The second step is to develop algorithms to identify the correct one from the above four possible basic 2-by-2 components.
For traditional tomography methods, one may propose to coordinate transmission of multi-packet probes from the two sources and measure the packet arrival order at the two receivers to obtain information about the 2-by-2 topology. However, it is not able to distinguish between the last three unshared types since one cannot extract features between them just from the packet number received at sink nodes $R_1$ and $R_2$. Fortunately, by allowing intermediate nodes to perform linear network coding, we can accurately distinguish them.

The observations at receivers are $R_1 = c_{11}x_1 + c_{12}x_2, R_2 = c_{21}x_1 + c_{22}x_2$, respectively. In particular, in a lossless scenario $R_1$ and $R_2$ obtain the following information in each 2-by-2 type:

- type (I): $R_1 = x_1 + x_2, R_2 = x_1 + x_2$
- type (II): $R_1 = x_1 + x_2, R_2 = x_1 + 2x_2$
- type (III): $R_1 = x_1 + 2x_2, R_2 = x_1 + x_2$
- type (IV): $R_1 = x_1 + x_2, R_2 = x_1 + 2x_2$

In this case type (I) and type (IV) have the same result and are indistinguishable. To address the problem we note that type (IV) has two different joining points (namely $J_1$ and $J_2$) if we let $S_2$ send packet later than $S_1$ with an offset of $u$ to force the packets to meet only at one of the joining points but not at the other, the receivers will have different observations as Case 3 and Case 4 shown in Table I.

**Table I**

| Obs. number | Type (I) | Type (IV) |
|-------------|----------|-----------|
| $R_1$       | $R_2$    | $R_1$     | $R_2$     |
| Case 1      | $x_1 + x_2$ | $x_1 + x_2$ | $x_1 + x_2$ |
| Case 2      | $x_1$    | $x_1$    | $x_1$    |
| Case 3      | $x_1 + x_2$ | $x_1$    | $x_1$    |
| Case 4      | $x_1$    | $x_1$    | $x_1 + x_2$ |

Based on the above observations, by comparing coefficients of $c_{22}$ and $c_{12}$ one can easily identify four different types. For example, if $c_{22} > c_{12}$ we know that the topology component should be type (II), and Type (IV) is identified by the first different observation between $R_1, R_2$.

However, in a lossy network one can no longer guarantee the meeting of $x_1$ and $x_2$ at the joining points and predictable observations at the receivers. In this case all possible observations are enumerated in Table II and divided into three groups: (i) at least one of the receivers does not receive any packet due to loss, (ii) both receivers have the same observation $R_1 = R_2$, and (iii) the two receivers have different observations $R_1 \neq R_2$.

We focus on the coefficient of $x_2$ and look at the difference of $c_{12} - c_{22}$. Table I shows that $c_{12} - c_{22} < 0$ can only occur in type (II) or type (IV) topology; while $c_{12} - c_{22} > 0$ can only occur in type (III) or (IV) topology. The details of inference are described as follows.

By performing several independent experiments and collecting several observations of group (iii), candidate topologies are distinguished. If after $\tau$ experiments ($\tau$ is a threshold to indicate the probing times from source $S_1$ and $S_2$) there are only observations of group (ii) or (iii) with $c_{12} - c_{22} \leq 0$, the topology is declared as type (II). If there are only observations of group (ii) or (iii) with $c_{12} - c_{22} \geq 0$, it is declared as type (III). If there are observations of group (ii) or (iii) with both $c_{12} - c_{22} < 0$ and $c_{12} - c_{22} > 0$, it is declared as type (IV). If we have only observed group (ii) packets, then the topology is declared as type (I) since no case in group (iii) can occur.

Authors in [33] extend the above work to merge network topologies. Compared to active tomographic techniques [32] [34] without NC, TI with NC has the following advantages: (i) it can exactly identify the 2-by-2 type, as opposed to just distinguish between shared and non-shared types; and (ii) the merging algorithms can precisely locate the joining points with respect to the branching points, as opposed to only provide bounds.

An open question remains: are there only four 2-by-2 basic components? For example why Fig. (5) in [33] is not included?

Fig. 5. Example: a new 2-by-2 component

3) Passive TI: Compared with active tomography the passive TI does not need to explicitly send probing packets at source node but makes use of regular data packets observed at edge nodes to recover the network structure.

A passive approach for topology inference on top of random NC has been proposed in [35]. As regular data flows are transmitted, intermediate nodes choose coding coefficients $\beta$ uniformly at random out of a finite field $F_q$. The basic idea is that under assumptions of a large enough $F_q$ and strong connectivity, the transfer matrix $[T]$ from sender to receiver is distinct between any pair of network structure. Thus, based on the original messages $[X]$ and the received results $[Y]$, an appropriate topology that matches $[Y] = [T][X]$ can be found.

Take four the basic 2-by-2 network components (shown in Fig. 4) for example, transfer matrices are provided in Fig. 5. The approach in [36] tries to distinguish them based on $[T]$. Note that $[T]$ is obviously unique in terms of $\beta$ for each component. For example, if all $\beta$ are equal to $1$, then $[T]_{1} \neq [T]_{2}$ with

$[T]_{1} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$

$[T]_{2} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix}$

Papers [37] [38] extend the above work to erroneous networks. With the assumptions that each intermediate node

---

1Coding coefficients $\beta$ can be different in practice. They are selected from a large finite field $F_q$ to guarantee that any type of topologies is likely to be distinguishable.
TABLE II

| #  | Group | Type (I) | Group | Type (II) | Group | Type (III) | Group | Type (IV) |
|----|-------|---------|-------|---------|-------|-----------|-------|---------|
| 1  | (i)   | -       | (i)   | -       | (i)   | -         | (i)   | -       |
| 2  | -     | x₁ + x₂ | -     | x₁ + 2x₂ | x₁ + 2x₂ | -         | -     | x₁ + x₂ |
| 3  | -     | x₁     | -     | x₁ + x₂ | x₁ + x₂ | -         | -     | x₁     |
| 4  | -     | x₂     | -     | x₁     | -       | x₁ + 2x₂ | -     | x₂     |
| 5  | x₁ + x₂ | -     | x₁ + x₂ | x₁ + x₂ | x₁ + 2x₂ | -       | -     | x₁ + x₂ |
| 6  | x₁     | -     | x₁ + x₂ | -     | x₁ + x₂ | -         | -     | x₁     |
| 7  | x₂     | -     | -     | x₁     | -       | x₁ + 2x₂ | -     | x₁     |
| 8  | x₁ + x₂ | x₁ + x₂ | x₂     | -     | x₁ + 2x₂ | (ii)     | x₁ + x₂ | x₁ + x₂ |
| 9  | x₁     | x₁ + x₂ | x₁ + x₂ | (iii) | x₁ + x₂ | (ii)     | x₁ + x₂ | x₁ + x₂ |
| 10 | x₂     | x₂     | x₁     | x₁     | x₁     | x₂        | x₂    | x₂     |
| 11 | x₂     | x₂     | x₂     | x₂     | x₂     | (iii)     | x₁ + x₂ | x₁ + x₂ |
| 12 | x₁ + x₂ | x₁ + 2x₂ | (iii) | x₁ + 2x₂ | x₁ + 2x₂ | x₁ + 2x₂ | x₁ + x₂ | x₁ + x₂ |
| 13 | x₁     | x₁     | x₁ + 2x₂ | x₁ + x₂ | x₁     | x₂        | x₁    | x₁     |
| 14 | x₁     | x₂     | x₂     | x₂     | x₁     | x₁        | x₂    | x₁     |
| 15 | x₁ + x₂ | x₂     | x₂     | x₁ + 2x₂ | x₁ + 2x₂ | x₂        | x₁ + 2x₂ | x₁ + x₂ |
| 16 | x₂     | x₂     | x₁     | x₁     | x₁     | x₂        | x₂    | x₁ + x₂ |

where $\beta$ is 

$[T]_1 = \begin{bmatrix}
\beta_{S_1J,JB} \cdot \beta_{JB,BR} & \beta_{S_1J,JB} \cdot \beta_{JB,BR} \\
\beta_{S_2J,JB} \cdot \beta_{JB,BR} & \beta_{S_2J,JB} \cdot \beta_{JB,BR}
\end{bmatrix}$

$[T]_2 = \begin{bmatrix}
\beta_{S_1J,J_B} \cdot \beta_{J_B,B_R} & \beta_{S_1J,J_B} \cdot \beta_{J_B,B_R} \\
\beta_{S_2J,J_B} \cdot \beta_{J_B,B_R} & \beta_{S_2J,J_B} \cdot \beta_{J_B,B_R}
\end{bmatrix}$

$[T]_3 = \begin{bmatrix}
\beta_{S_1J,J_B} & \beta_{J_B,B_R} & \beta_{S_1J,J_B} & \beta_{J_B,B_R} \\
\beta_{S_2J,J_B} & \beta_{J_B,B_R} & \beta_{S_2J,J_B} & \beta_{J_B,B_R}
\end{bmatrix}$

$[T]_4 = \begin{bmatrix}
\beta_{S_1J,J_B} & \beta_{J_B,B_R} & \beta_{S_1J,J_B} & \beta_{J_B,B_R} \\
\beta_{S_2J,J_B} & \beta_{J_B,B_R} & \beta_{S_2J,J_B} & \beta_{J_B,B_R}
\end{bmatrix}$

Fig. 6. Example: transfer matrices for four types of 2-by-2 topologies when NC is applied at intermediate nodes with coding coefficient $\beta$.

knows its one-hop neighbors and that all nodes in the network share common randomness, i.e., the receiver knows the random code-books of other nodes, they present algorithms for passive topology tomography in the presence of network failures. The basic idea is described as follows.

Let $C$ be the min-cut from $s$ to $r$. The length-$C$ impulse response vector (IRV) $I(e)$ for an edge $e \in \xi$ is defined: $s$ transmits the all-zero packet on all outgoing edges and edge $e$ injects a nonzero packet $z(e)$. In this case the received packet at $r$ is $[Y] = I(e)z(e)$, where $I(e)$ is the IRV. Thus all the nonzero columns of $[Y]$ are equivalent to $I(e)$ multiplied by a scalar. Each IRV can be computed using Eq. (4):

$$I(e) = \sum_{1 \leq i \leq d} \beta(e, v, e_i)I(e_i)$$

where $e$ is an incoming edge of node $v$ and $e_i (1 \leq i \leq d)$ are the outgoing edges.

Take the network of Fig. 7 for example. Local encoding vectors are labeled at each intermediate node. Based on Eq. (4) we can calculate that $I(e_1) = [0 1]^T$, $I(e_2) = [0 1]^T$, $I(e_3) = [3]$, $I(e_4) = [0 0]^T$, $I(e_5) = [4]$, $I(e_6) = [2]$, $I(e_7) = [1 8]^T$.

Before topology recovery, there is a need to find the IRV of each link using the following two steps for $\tau$ independent successful communication rounds:

- Step (I): $r$ computes $M(i)$ using ECC. Then $E(i) = Y(i) - Y(i)M(i)$ where the source message is $X(i) = [I_C M(i)]$, $I_C$ is the coefficient matrix, $M(i)$ is the data message, $Y(i)_b$ is the matrix consisting of the first $C$ columns of $Y(i)$ and $Y(i)_l$ consisting of the last $n - C$ columns of $Y(i)$.

- Step (II): If the calculated result $E(i) \cap E(j) = 1$ for any $i, j \in [1 \tau]$ pair, $E(i) \cap E(j)_r$ is added to the IRV set $I$.

Based on $I$ of IRV result and the common random code-books $R$, vertex set $V$ is initialized with source $s$, receiver $r$.

A successful round means the number of errors does not exceed the bound and receiver node $r$ can decode the source message correctly using network error correcting code (ECC).

Fig. 7. A network with IRV of each link.
and all its upstream neighbors. $\xi$ is initialized with the edges incoming to $r$. Let $\bar{I}$ be the set of IRVs that are calculated from the currently recovered network structure and the common random code-book $R$. The basic idea for recovering topology is shown in the following.

For each node $v \neq s$, let $e_1, ..., e_d$ be the outgoing links of it. If $\bar{I}(e_1), ..., \bar{I}(e_d)$ has rank greater than 1, then the IRV for each candidate incoming link $e$ of $v$ is calculated using Eq. (2). If the obtained IRV belongs to the IRV set $I$, then the link and associated node are added to $\xi$ and $V$, respectively. This process proceeds up to the source $s$.

It has been demonstrated that the above algorithm can recover the accurate topology by performing $O(|\xi|^4|V|/C)$ operations over $F_q$ with probability $1 - O(|\xi|^4|V|/C/q)$ if $I$ contains all links’ IRVs.

B. Link Loss Inference (LLI)

Inferring the link loss rate in network layer and application layer is particularly important for a variety of traffic control methods. Many network tomography schemes with NC capability have been proposed for LLI [10], [39] where coding vectors implicitly containing topology correlated information can be utilized to collect coded packets at receiver nodes for estimating link loss rates.

1) Wireline LLI of tree structure: Consider the network shown in Fig. 8. Nodes $A$, $B$ send probes $x_1 = [1, 0]$, $x_2 = [0, 1]$, respectively. If node $C$ receives only $x_1$ (or $x_2$) it will forward $x_1$ (or $x_2$) to its child, while if node $C$ receives both, it will encode them into packet $x_3 = x_1 \oplus x_2 = [1, 1]$ and forward it. Nodes $E, F$ receive packets from node $D$. The basic idea is described as follows [21].

Assuming that nodes $A, B$ are synchronized and each operation occurs in one time slot, probe packets $x_1, x_2$ sent from $A, B$ arrive at nodes $E, F$ depending on a random link loss following an i.i.d. Bernoulli distribution. Table III gives the relationship between received packets at nodes $E, F$ and link transmission status ($S, N$ denote the successful and unsuccessful link transmission). For example, the scenario that $x_3 = x_1 \oplus x_2$ can be received at node $E$ but nothing is received at node $F$ occurs only when packet on link $DF$ is lost with the probability of $(1 - \alpha_{AC})(1 - \alpha_{BC})(1 - \alpha_{CD})(1 - \alpha_{DE})\alpha_{DF}$, where $\alpha_{link}$ denotes the packet loss rate of a link. By repeating the experiment many times, the Maximum Likelihood (ML) estimation can be used to infer the underlying loss rates of each link.

Paper [40] also studies the link loss tomography using multiple sources of probes for tree structure network. It further concludes that by appropriately choosing the number of sources and receivers, as well as their locations, a significant improvement on the estimation accuracy can be obtained.

![Fig. 8. Structure for link loss rate inference](image)

| Received at | Link status |
|-------------|-------------|
| Node $E$   | Node $F$   | AC | BC | CD | DE | DF |
| $x_1$      | $x_1$      | $S$ | $N$ | $S$ | $S$ | $S$ |
| $x_1$      | $\phi$     | $S$ | $N$ | $S$ | $S$ | $N$ |
| $x_2$      | $x_2$      | $N$ | $S$ | $S$ | $S$ | $S$ |
| $x_2$      | $\phi$     | $N$ | $N$ | $S$ | $S$ | $N$ |
| $x_3$      | $x_3$      | $S$ | $S$ | $S$ | $S$ | $S$ |
| $x_3$      | $\phi$     | $S$ | $S$ | $S$ | $S$ | $N$ |
| $\phi$     | $x_1$      | $S$ | $N$ | $S$ | $N$ | $S$ |
| $\phi$     | $x_2$      | $N$ | $S$ | $S$ | $N$ | $S$ |
| $\phi$     | $x_3$      | $S$ | $S$ | $S$ | $N$ | $S$ |
| $\phi$     | $\phi$     | $S$ | $N$ | $N$ | $N$ | $N$ |

|   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|
|   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |

![Fig. 9. Example: Tree network for link loss rate inference with oriented link direction](image)

Consider the tree structure shown in Fig. 9. Demonstrated by simulations for three cases: (1) a multicast tree with source at node 1. (2) a multicast tree with source at node 2, (3) two sources at nodes 1 and 2 and a coding point at 4. Authors of [40] find the last one outperforms the former two cases. This is because coding points partition the tree into smaller multicast components and between two multicast trees with the same number of receivers, better performance is achieved by the tree that is more “balanced” with the smallest height.

Based on these observations, guidelines on how to choose the best “points of view” of a network for LLI tomography are listed as follows:

- Select a fraction of sources to receivers that partition the tree into roughly equal-size subcomponents, where each subcomponent should have at least 2-3 receivers.
- Distribute the sources in roughly equal distances along the periphery of the network.
Nevertheless, in paper [21] and [40], only sub-optimal algorithms are developed for multiple source loss tomography. To address this issue a low complexity maximum likelihood estimator in [41] is provided for LLI with network coding capabilities.

2) *Wireline LLI of general structure*: In paper [42] authors investigate an active probing method for LLI with NC in general network topology. With this approach each link is traversed by exactly one packet which results in a great bandwidth saving compared to traditional multicast or unicast tomographic techniques.

Two main issues of LLI in general graphs are discussed. The first is that a network with cycles may result in probes being trapped inside a cycle, i.e., a positive feedback loop that consumes network resources without aiding the estimation process. To solve this problem, they propose an Orientation Algorithm (OA) that creates an acyclic graph with the maximum number of identifiable links. OA achieves the goal by sequentially visiting the vertices of the graph, starting from the sources, and selecting an orientation for all edges of the visited vertex. To maximize the number of identifiable edges, OA selects an orientation such that each intermediate vertex (i.e., not a source or a receiver) has at least one incoming and at least one outgoing edge.

The second challenge is the identifiability of a link. In tree networks we know that it is sufficient for intermediate nodes to perform XOR operations (with finite field $F_2$) while it is insufficient in general networks.

Take Fig. 10 for example where intermediate nodes only do XOR operations. Since path (1-6-7-8-9) and (1-6-10-8-9) meet at nodes 6 and 8, XOR operations will cancel each other and result in the same observations as the case that both paths break down.

![Fig. 10. General network for link loss rate inference](image)

To solve this issue a larger finite field $F_q$ for coding coefficients is chosen. For instance, nodes 7 and 8 may use coefficients of [1 1] and [1 2], respectively.

Based on the above, the basic idea of LLI in general networks can be summarized as follows. Given a selected set of sources, the OA algorithm eliminates cycles and determines the transmitted paths of probes. Then sources send probing packets with NC operations at intermediate nodes over a large enough finite field $F_q$. Finally, receivers use both the number and content of received packets for LLI of all links.

3 A link is said to be identifiable under a given monitoring scheme namely, choice of sources, receivers, intermediate node operations, if its associated loss rate can be reliably inferred from the measurements observed at the receiver.

Compared with traditional tomography using multicast or unicast in general networks [43], NC based LLI uses exactly one probe per link to save bandwidth, and it also avoids suboptimal combination of measurements from different trees.

3) *Wireless network LLI*: Wireless network scenario is quite different from that of wireline environment for its highly stochastic nature. Therefore, it is desirable to monitor LLI in wireless networks.

Paper [44] makes the pioneer work to address link loss tomography in wireless networks with NC capability. Authors of [44] prove that network coding changes the fundamental connection between path and link loss rates from $\beta = \Pi_{e \in P}(1 - \alpha_e)$ to $\beta = \min_{e \in P}(1 - \alpha_e)$, where $\beta$ and $\alpha_e$ denote the path successful transmission probability and link loss probability, respectively. This is because a wireless link $\varepsilon$ can be modeled as a binary erasure channel with capacity $1 - \alpha_e$ [45] and with NC the transmission on each link is equipped with an erasure code that achieves the link capacity when code length $K$ is large enough. Moreover, the capacity of any graph is the min-cut [46]. Thus, the capacity of this path $P$ (it can also be modeled as a binary erasure channel) is the minimum capacity of any link: $\min_{e \in P}(1 - \alpha_e)$. Therefore, $\beta = \min_{e \in P}(1 - \alpha_e)$ [45] and new tomography methods need to be developed.

Unlike research work in wireline networks, wireless links have an important consequence on LLI: the most lossy link on a path essentially blocks the information of all other links on the same path. Hence, it is infeasible to determine the exact loss rates of all links. Therefore, only algorithms for determining highly lossy links in wireless sensor networks are proposed.

![Fig. 11. Example: wireless sensor network model for link loss rate inference, where nodes A, B are terminals and nodes C, D are relay node and sink node, respectively](image)

In this scenario the sensor network is assumed to be a directed graph, where each node represents either a terminal, a relay node, or the sink node, each directed edge represents the link between them (as shown is Fig. 11). The main idea of NC based data collecting protocol is that: Each terminal node partitions the data packets to segments, which are attached with a sequence number consisting of $K$ data packets. Assume that there are $n$ terminal nodes in the network. The $i^{th}$ group of segments consists of $Kn$ source packets. Note that when
multiple nodes transmit coded packets to a common child, all these packets are encoded together and the coding coefficients corresponding to the absent source packets are zeros. For instance, in Fig. 11 the coded packets sent from A to C have the first two coefficients of zero when the segment length \( K = 2 \).

For LLI in wireless scenario with NC, the basic idea is described as follows. The decoding matrix is divided into \( n \) submatrices so that each submatrix with \( K \) columns corresponds to the same segment packets from one terminal. For example, the grey and white columns of the coefficients at the sink in Fig. 11 represent the two submatrices from nodes \( A, B \). If the successful transmission probability \( \beta \) of path \( i \) is higher than that of path \( j \), the rank of submatrix \( i \) reaches \( K \) earlier than that of \( j \). The information of \( \beta \) is contained in the relative times that are useful to infer link loss rates.

Formally speaking, let \( t_i \) denote the time when submatrix \( i \) is full rank, and let \( d_i \) represent the length of the \( i \)th path, thus \( t_i = t_i - d_i \) represents the wasted time due to link loss. The inference methods use the times \( D = \{t_1, \ldots, t_n\} \) and a link rate threshold \( T_i \) as the inputs. \( T_i \) is a threshold value so that a link is declared to be good with \( s_\varepsilon = 1 \) if its rate is higher than \( T_i \) and bad with \( s_\varepsilon = 0 \) otherwise. As a result, the output of LLI is the states of all links \( S = \{s_e\} \), where \( \varepsilon \in E \).

Three different inference algorithms to infer highly lossy links are proposed below. Two of them are based on the Bayesian principle while the last one is the Smallest Consistent Failure Set inference method.

The first one is Bayesian inference using factor graphs which is called BIFG for short.

\[
\begin{align*}
P(\beta_i | \alpha_e) &= \begin{cases} 1 & \text{if } \beta_i = \min(\alpha_e) \\ 0 & \text{otherwise} \end{cases} \quad P(t_i | \beta_i) = \left( \frac{t_i - 1}{K - 1} \right) \beta^K (1 - \beta)^{t_i - K}. 
\end{align*}
\]

![Fig. 12. Example: the factor graph of Fig. 11](image)

BIFG firstly constructs the factor graph 47 using Eq. (3) and Eq. (4).

\[
P(\alpha_e | \{\alpha_{\varepsilon}\}) = \int_{\alpha_e} P(D | \alpha_e) P(\{\alpha_{\varepsilon}\}) P(\alpha_e) d\alpha_e
\]

where \( \{\alpha_{\varepsilon}\} = \bigcup_{|f \neq e} \{\alpha_f\} \).

The third one is the smallest consistent failure set method, which is called SCFS for short. It modifies the SCFS approach proposed in [49]. The basic idea of the SCFS approach is that, if there are multiple choices to assign whether links are good or bad in order to satisfy the end-to-end observations, one should select the smallest set of bad links, under the assumption that a large fraction of the network behaves well.

Authors in [50] address the LLI in wireless networks using subspace properties of NC. They propose the PLI-RLC algorithm to infer link loss rate which performs 11% better than Bayesian inference algorithms of previous paper 44 in terms of false detection. The algorithm is composed of two parts. In the first part, it 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. In the second part, it estimates the packet loss rate based on the dimension of the subspaces using Eq. (7) and Eq. (8):

\[
\hat{\alpha}_s(t) = 1 - \frac{\dim(\bigcap_{\varepsilon} \{t_i(t)\})}{t - t_{s}^4 + 1}
\]

where \( s \) is a source node, \( \alpha_s \) is the loss rate of the path \( P_s \), \( \hat{\alpha}_s(t) \) is the estimation of \( \alpha_s \) at time \( t \). \( \bigcap_{\varepsilon} \{t_i(t)\} \) represents the subspace spanned by coefficients of the packets at node \( s \) and \( t_{s}^4 \) denotes the time at which the sink expects to receive the first packet of source \( s \).

\[
\hat{\alpha}_v(t) = 1 - \frac{\dim(\bigcap_{\varepsilon} \{t_i(t)\})}{t - t_{v}^4 + 1}
\]

where \( v \) is a virtual node, \( \alpha_v \) is the loss rate of the path \( P_v \), \( \hat{\alpha}_v(t) \) is the estimation of \( \alpha_v \) at time \( t \). \( \bigcap_{\varepsilon} \{t_i(t)\} \) represents the subspace spanned by coefficients of the packets at node \( v \) and \( t_{v}^4 \) denotes the time at which the sink expects to receive the first packet of source \( v \).

Then, for each link \( l \), if \( \alpha_{\text{head}(l)} > \alpha_{\text{tail}(l)} \), its link loss rate \( \varepsilon_l = \alpha_{\text{head}(l)} \), where \( \text{head}(l) \) and \( \text{tail}(l) \) denote the nodes attached to the head and tail of link \( l \), respectively.

In this paper each intermediate network coding node is called a virtual source.
Thus the system matrix is extended to Eq. (13):

\[ M = \begin{bmatrix}
  e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & e_7 \\
  P_1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \\
  P_2 & 1 & 1 & 0 & 1 & 0 & 1 & 1 \\
  P_3 & 1 & 0 & 1 & 0 & 0 & 1 & 1 \\
\end{bmatrix} \quad (9) \]

If \( e_1 \) and \( e_7 \) are referred as one virtual link \( e_{v1} \), the type 1 modified path-link matrix is shown in Eq. (10):

\[ \overline{M} = \begin{bmatrix}
  e_{v1} & e_2 & e_3 & e_4 & e_5 & e_6 \\
  P_1 & 1 & 1 & 0 & 0 & 1 & 0 \\
  P_2 & 1 & 1 & 0 & 1 & 0 & 1 \\
  P_3 & 1 & 0 & 1 & 0 & 0 & 1 \\
\end{bmatrix} \quad (10) \]

and the system can be represented by Eq. (11):

\[ \overline{M}\hat{a} = \hat{b} \quad (11) \]

By defining \( \varphi \) as the power set of path \( P: |\varphi| = 2^{2^{|P|}} \), each element of \( \varphi \) is a subset of \( P \) that can be used to represent combinations of paths. Accordingly, a modified path-link type 2 matrix \( M(2) = (m_{i,j})_{(|\varphi|-1)\times(|\varphi|-1)} \) is given as follows: if there is a path in the \( i \)th set including the \( j \)th link. Otherwise, it is equal to 0.

\[ M_2 = \begin{bmatrix}
  1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\
  1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\
  1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\
  1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 \\
  1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 \\
  1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\
\end{bmatrix} \quad (12) \]

Thus the system matrix is extended to Eq. (13):

\[ M_2\hat{a} = \hat{c} \quad (13) \]

At last it has been shown that: If \( M_2\hat{a} = \hat{c} \) is given, then \( \hat{a} \) can be determined by least-squares \[52\], where

\[ \hat{a} = (M_2^T M_2)^{-1} M_2^T \hat{c} \quad (14) \]

C. Link Delay Inference

The distribution of link delay is an important parameter for performance evaluation and load balance. Besides delay itself, the delay correlation between end hosts is particularly useful for some 3D tomography tasks, for example the topology recovery. Current literatures either use one way end-to-end measurement (OTT) \[53\] or round trip time (RTT) \[54\] to obtain it. The main difference between OTT and RTT is that for OTT, series of packets are sent from sources and collected at receivers while for RTT, packets are both sent and received at the source. However, OTT usually requires precise time synchronization while RTT needs the cooperation from receivers, which inject extra operation delay and causes inaccurate measurement. In this section we first introduce a DCE measurement method \[27\] without any time synchronization and then discuss the trend for adding NC in future.

1) DCE delay correlation inference: The basic idea of delay correlation estimation (DCE) is described as follows.

A simple model we use is shown in Fig. 14. The routing structure from the sender \( f \) to the receivers \( a \) and \( b \) must be a tree rooted at \( f \). Otherwise, there is a routing loop which must be corrected. Assume that router \( s \) is the ancestor node of both \( a \) and \( b \). Assume that the sender uses unicast to send messages to receivers, and assume that packets are sent in a back-to-back pair. For the \( k \)-th pair of back-to-back packets, denoted as \( a^k \) and \( b^k \), sent from \( f \) to \( a \) and \( b \), respectively. We use the following notation:

- \( t_a(k) \): the time when \( a \) receives \( a^k \) in the \( k \)-th pair.
- \( t_b(k) \): the time when \( b \) receives \( b^k \) in the \( k \)-th pair.
- \( d_a(k) \): the latency of \( a^k \) along the path from \( f \) to \( a \).
- \( d_b(k) \): the latency of \( b^k \) along the path from \( f \) to \( b \).
- \( t_f(k) \): the time when \( f \) sends the \( k \)-th pair of packets.

Using network model of Fig. 14 we obtain Eq. (15) for \( a^0 \) (we start the index with 0 for convenience):

\[ t_a(0) = t_f(0) + d_a(0) \quad (15) \]

Similarly, for \( a^k \) we have

\[ t_a(k) = t_f(k) + d_a(k) \quad (16) \]

Let \( t_a(k) \) from Eq. (16) and \( \delta_a(k) = t_a(k) - t_a(0) \), we can obtain Eq. (17):

\[ \delta_a(k) = (t_f(k) - t_f(0)) + (d_a(k) - d_a(0)) \quad (17) \]
Denote the time interval between two consecutive pairs of packets as $\delta$. We assume that $\delta$ is a constant for simplicity at this moment, and relax this assumption later. In this case we use $k\delta \equiv k\cdot \delta$ to replace $t_f(k) - t_f(0)$ in Eq.(17), then we have

$$d_a(k) = \delta_a(k) - k\delta + d_a(0)$$

(18)

Let $\delta_a'(k) \equiv \delta_a(k) - k\delta$, Eq.(18) is transformed into Eq. (19):

$$d_a(k) = \delta_a'(k) + d_a(0)$$

(19)

We can achieve similar results at receiver $b$ as in Eq.(20):

$$d_b(k) = \delta_b'(k) + d_b(0)$$

(20)

where $\delta_b'(k) \equiv \delta_b(k) - k\delta$.

Based on above equations, we prove that the correlation between delay variables $d_a(k)$ and $d_b(k)$ is equivalent to the correlation between variables $\delta_a'(k)$ and $\delta_b'(k)$, which means

$$\sigma^2_{d_a(k),d_b(k)} = \sigma^2_{\delta_a'(k),\delta_b'(k)}$$

(21)

The unbiased estimator of the correlation on shared path is also provided as Eq. (22) shows:

$$\delta_{\delta_a',\delta_b'} = \frac{1}{n-1} \sum_{k=1}^{n}[\delta_a'(k) - \overline{\delta_a'}][\delta_b'(k) - \overline{\delta_b'}]$$

(22)

where $\overline{\delta_i'}$ is the sample mean of $\delta_i'(k)_{k=1}^n$ for $i = a, b$.

To reduce explicit probing, we develop a mechanism for passive realization in real networks.

### IV. NEW PROPOSED APPLICATIONS OF NT WITH NC

#### A. NT of bottleneck discovery

Peer-to-peer (P2P) network is widely used in practice due to its excellent capability of content distribution. It, however, faces the issue of node collaboration. The introduction of NC can eliminate the data difference by combining multiple packets, thus is able to solve the above problem. Typical application of NC in P2P is “Avalanche” developed by Microsoft [56], [57], [58]. It uses random network coding for content distribution. However, P2P content distribution relies heavily on connectivity of the overlay network, where detecting bottleneck links is crucial for improving the overall performance (see the example shown in Fig. 16). All the above reasons motivate our research in NC capabled P2P scenario for bottleneck discovery, which aims at detecting bottleneck links and rewiring them to guarantee network capability.

Traditional method without NC in paper [59] determines the multicast tree structure and detects bandwidth bottlenecks by distinguishing the loss model correlation of sinks and measuring how network disrupts the fine packet timing structure. However, a drawback is that its implementation is centralized, and it needs to send probes actively. Therefore, the scalability to P2P network and other non-tree structure application areas is limited.

To address the above shortcomings, article [28] studies P2P network using NC. Based on characteristics of subspace [60] and the observation that subspace spanned by coded packets received at each node reveals topological information of the network, authors in [28] propose distributed algorithms for
discovering bottlenecks and breaking clusters. The basic idea is in the following.

Assume that at a given time $t$, each node $i$ knows its own subspace and the subspaces it has received from its parent nodes. Let $\Pi_i = \Pi_1 \cup \ldots \cup \Pi_c$ denote the subspace spanned by the coding vectors $i$ has collected, where $\Pi_1, \ldots, \Pi_c$ denote the subspaces that it has received from its $c$ neighbors $u_1, \ldots, u_c$, respectively. The overlap of subspaces from the neighbors reveals information about a bottleneck. Consider the network depicted in Fig. 17(a), where edges correspond to logical links. The source $S$ has $n$ packets to distribute to four peers. Nodes $A$, $B$, and $C$ are directly connected to $S$, while node $D$ is connected to nodes $A$, $B$, and $C$. However, logical links $SA$, $SB$, $SC$ share the bandwidth of the same underlying physical link $SV$ as shown in Fig. 17(b), which forms a bottleneck between the source and the remaining nodes of the network.

In this case when nodes $A$, $B$, $C$ receive the same packet from $S$, the coding vectors collected by them will span the same subspace. Hence the coded packets they offer to node $D$ will overlap subspaces significantly. From this information, node $D$ knows that there is a bottleneck between nodes $A$, $B$, $C$, and $S$.

Paper [28] also defines some criteria for bottleneck discovery. For example, a node can decide there is a bottleneck on link to a particular neighbor $i$, if it receives $k > 0$, non-innovative coding vectors from $i$, where $k$ is a tunable parameter.

### B. NT of failure localization

NT of failure localization is different from link loss inference (LLI) that we need to localize the accurate position of which link is unavailable or some random errors occur on a link, while LLI concerns the total link loss rate of each link.

Paper [61] is the first to show that coding coefficients used in randomized network coding play double duties by allowing link failure monitoring in addition to allowing the sink nodes to correctly decode the incoming data under different failure patterns, since link failure affects the coding vectors received at sink node. This paper derives the probability that an event is identified among a given set of failure events. It also sets bounds on the required finite field size $f_q$, and analyzes the complexity for designing robust network code. Details are described as follows.

Given a distinguishable failure event set $C$ and a specific failure event $c \in C$, the upper bound on the probability of tomography ambiguity in a random linear network code is

$$P \geq 1 - \left(1 - \frac{|C| - 1}{q}\right)^L$$

where $L$ denotes the maximum number of logical links on a source-sink path, $q$ is the finite field size of $f_q$. From Eq.(23), the probability of tomography ambiguity decreases inversely with field size $q$.

Similarly, the required field size and complexity for designing a robust network code that distinguishes among a given set of failure events are also bounded by Eq.(24) and Eq.(25), respectively, for a valid network code field size in Eq.(26).

$$q \geq |C| \left(\frac{|C| - 1}{2} + d\right)$$

$$O\left(\left(\frac{\gamma}{\gamma - 1}\right)^\eta |C| \left(\eta I + dr^{2.376} + |C|rt\right)\right)$$

$$q = \gamma |C| \left(\frac{|C| - 1}{2} + d\right)$$

where $d$ is the number of sinks, $\eta$ is the total number of links in the network, $I$ is the maximum in-degree of a node, $r$ and $t$ are the number of sources and terminal links, respectively.

Paper [37] provides the first polynomial time algorithm for locating the edges that are subject to random errors and random erasures. The basic idea is that sink node $r$ firstly calculates random error vector set $E_r$ according to the network structure, and then $r$ tests each impulse response vector $I(c)$ to see if it belongs to $E_r$. If so, it can be determined that the corresponding link belongs to the error link set.

\footnote{For a network with sources and sinks, we consider two failure patterns $p_1$ and $p_2$ to be indistinguishable if the set of source-sink paths containing at least one link in $p_1$ is identical to the set of source-sink paths containing at least one link in $p_2$, and distinguishable otherwise.}
Consider network of Fig. 18(a). Source $s$ sends symbols 1 and 2 to node $u$ via edges $e_1$ and $e_2$ respectively. Due to the error introduced in $e_1$, $r$ receives a vector $Y = [3 \ 2]^T$. Then $r$ computes the error vector to be $E_r = Y - [1 \ 2]^T = [2 \ 0]^T$. According to $E_r$, $r$ knows that error happens on $e_1$ or $e_3$ but cannot locate the error.

However, if NC is used at intermediate node $u$, we can locate the error link accurately (shown in Fig. 18(b)). In this scenario, $x_3 = x_1 + 2x_2$ and $x_4 = x_1 + x_2$, where $x_1$, $x_2$ are symbols received from $e_1$, $e_2$ and $x_3$, $x_4$ are symbols to be sent via $e_3$, $e_4$. With error at link $e_1$, $r$ receives $Y = [7 \ 5]^T$. Similarly, $r$ can compute the error vector $E_r = Y - [1 + 2 \ 2]^T = [2 \ 2]^T$. Then $r$ obtains each impulse response vector $I(e_1) = [1 \ 1]^T$, $I(e_2) = [2 \ 1]^T$, $I(e_3) = [1 \ 0]^T$, $I(e_4) = [0 \ 1]^T$. Using $E_r = [2 \ 2]^T$, $r$ knows that error is injected to $e_1$ with 2.

C. Comparison with traditional tomography methods

The introduction of NC to network tomography not only brings advantages such as high tomography efficiency and low cost, but also broadens the scope of tomography. For example, it extends the bottleneck discovery to P2P networks. To compare its performance with non-NC tomography methods, we present a summary of traditional solutions.

A comprehensive survey on network tomography can be found in [5] which focuses exclusively on inferential network monitoring techniques that require minimal cooperation from network elements. Tools for active/passive measurement of networks can be found in [63], for example, mper is a probing engine that clients can use to conduct topology and performance measurements using ICMP, UDP, and TCP probes.

For traditional topology inference [64], probing packets are sent to multiple receivers by a multicast tree, and then they are used to recover the topology structure with information of received data packets at different nodes. In [31] [32], authors extend this to propose that any M-by-N network can be decomposed into a collection of four 2-by-2 subnetwork components as shown in Fig. 4. However, the above traditional tomography methods are not able to distinguish between the last three unshared types. Independently, NC based tomography in [30] is able to exactly identify the 2-by-2 type, as opposed to just distinguish between shared and non-shared types. In addition, the 2-by-2 component merging algorithms in [33] can precisely locate the joining points with respect to the branching points, as opposed to only provide bounds by traditional methods.

Moreover, [36] proves that the transfer matrices $[T]$ for detectably different networks are distinct and can be utilized to distinguish between any network topology. [37] [38] extends the above work to erroneous networks to passively infer network topology without explicit probing. Also [60] shows that subspace spanned by coded packets received at each node reveals topological information of the network which can be used for passive topology recovery.

For traditional link loss inference [65], a maximum-likelihood estimator for loss rates on internal links observed by multicast receivers is developed. However, LLI over general graphs with an arbitrary structure is beyond its scope. For general graphs, authors of [43] [66] use multiple multicast trees and/or multiple unicast paths to cover the network graph, and then combine the link loss rates estimated from the different paths/trees. However, the above traditional methods are suboptimal with respect to the following optimality criteria: identifiability, estimation accuracy and bandwidth efficiency. Independently, NC based tomography in [21] is the first to use network coding for inferring link loss rates in overlay networks with advantages of less bandwidth consumption, less complexity for realization and high inference accuracy. Paper [40] extends [21] and further presents that appropriately choosing the number of sources and receivers, as well as their location, can have a significant effect on the accuracy of the estimation. In [42], with NC each link is traversed by exactly one packet, resulting in a great bandwidth saving compared to traditional multicast or unicast tomographic techniques.

Moreover, [44] is the first to address loss inference in wireless networks using network coding and it changes the fundamental connection between path and link loss rates from $\beta = \Pi_{e \in P}(1 - \alpha_e)$ to $\beta = \min_{e \in P}(1 - \alpha_e)$. To make use of subspace properties of network coding for LLI in wireless sensor networks, [50] proposes a PLI-RLC algorithm to infer link loss rate.

For traditional link delay inference, authors in [53] use the one way end-to-end measurement (OTT) method while paper [54] develops a round trip time (RTT) solution. The main difference between OTT and RTT is that for OTT, series of packets are sent from sources and collected at receivers while for RTT, packets are both sent and received at the source. However, OTT requires precise time synchronization and RTT needs the cooperation from receivers. To address the above issues, a novel DCE measurement method [27] is proposed without any time synchronization and complex cooperation. The trend of adding network coding to DCE was discussed in [27].
For new proposed applications of NT, authors in [23] [60] [62] extend the traditional tomography scope to bottleneck discovery in P2P environment. The paper [61] is the first to show how the coding coefficient embedded in randomized network coding can be used to infer failure patterns, which consumes no extra overhead for probing.

A more detailed summary of NC-based tomography methods and the comparison with traditional techniques is listed in Table IV where not only existent applications but also new proposed solutions of NT with NC are listed.

V. DISCUSSION AND FUTURE TRENDS

In this section, we first present some NC-based methods that are in practical use, then discuss some lessons and existing problems which demand further research.

For NC-based tomography, authors in [44] have developed a customized pack-level network simulator in C++, with the implementation of randomized network coding to evaluate LLI in wireless networks. Similarly, paper [50] implemented a discrete-event packet-level simulator for the wireless sensor networks. In paper [27], the authors have implemented the DCE tomography method using OMNeT++, which is an open-architecture discrete-event simulator consisting of extensible, modular C++ libraries. To evaluate its performance in practical networks, the authors also implemented it on PlanetLab platform. It concluded that if some traditional tomography methods, for example, tools listed in [63], are equipped with NC capability, their performance will be much enhanced in practice.

For implementation of NC-based tomography, there are some aspects of lessons we should take into account. For example, the finite field used for network coding is an important parameter which needs a tradeoff between decoding efficiency and transmitting expense. In wireless environment this issue is even worse since both the bandwidth and memory space are limited. In practical networks, it has been found that the platform itself may impact performance of network tomography. For example, there is a sharp performance drop over the PlanetLab platform for the DCE measurement. The main reason is that PlanetLab nodes are always heavily loaded with multiple applications and it uses virtualization techniques which may destroy time series for resource scheduling.

While NC-based tomography in general outperforms non-NC-based tomography, there exist some problems which need further research in future.

- For Topology inference with NC capabilities there already exist solutions for recovery of tree structure and basic network components. However, we still need to find methods to merge them together to obtain a general multiple sources multiple sinks network. Moreover, whether or not only four components exist is still an open question, since in [35] two more 2-by-2 networks are proposed.

- For link loss inference especially in wireless scenario, although [44] demonstrates that NC changes the fundamental connection between path and link loss rates from $\beta = \prod_{c \in P}(1 - \alpha_c)$ to $\beta = \min_{c \in P}(1 - \alpha_c)$, we would like to point out that this is almost impossible in practice since the length of codes cannot be infinite for the limited link bandwidth of wireless sensor network.

- For delay tomography we introduce DCE method for delay correlation inference and show the bonus of using NC coded packets. Further work, however, is required to combine mechanism of packet generation to time-related information.

- For research in new application areas besides bottleneck discovery and failure localization, utilizing the subspace property of NC should be a good direction in future.

VI. CONCLUSIONS

In this paper we introduce the application of network coding in network tomography and show that with NC many benefits can be gained. Firstly, we present taxonomy result for network tomography with NC capabilities; Secondly, we review the corresponding methods in each category. Last but not least, we present the primary research in delay inference and present the lessons and trend for future research with NC. We expect that this comprehensive survey on tomography with NC abilities will attract more attention to this area.

REFERENCES

[1] A. Chen, J. Cao, and T. Bu, “Network tomography: Identifiability and fourier domain estimation,” IEEE Transactions on Signal Processing, vol. 58, no. 12, pp. 6029–6039, 2010.
[2] B. Yao, R. Viswanathan, F. Chang, and D. Waddington, “Topology inference in the presence of anonymous routers,” in Proceedings of the IEEE INFOCOM 2003, San Francisco, CA, USA, April 2003, pp. 353–363.
[3] X. Zhang and C. Phillips, “A survey on selective routing topology inference through active probing,” IEEE Communications Surveys and Tutorials, vol. 14, no. 4, pp. 1129–1141, 2012.
[4] Y. Vardi, “Network tomography: estimating source-destination traffic intensities from link data,” Journal of the American statistical association, vol. 91, no. 433, pp. 365–377, March 1996.
[5] R. Castro, M. Coates, G. Liang, R. Nowak, and B. Yu, “Network tomography:recent developments,” Journal of Statistical Science, vol. 19, no. 3, pp. 499–517, 2004.
[6] N. G. Duffield, J. Horowitz, F. L. Presti, and D. F. Towsley, “Network delay tomography from end-to-end unicast measurements,” in Proceedings of the International Workshop on Digital Communications: Evolutionary Trends of the Internet, 2001, pp. 576–595.
[7] F. L. Presti, N. G. Duffield, J. Horowitz, and D. Towsley, “Multicast-based inference of network-internal delay distributions,” IEEE/ACM Transactions On Networking, vol. 10, no. 6, pp. 761–775, December 2002.
[8] J. Cao, D. Davis, S. V. Wiel, B. Yu, S. Vander, and W. B. Yu, “Time-varying network tomography: router link data,” Journal of the American statistical association, vol. 95, pp. 1063–1075, 2000.
[9] G. Liang and B. Yu, “Maximum pseudo likelihood estimation in network tomography,” IEEE Journal on Selected Areas in Communications, vol. 14, no. 4, pp. 1129–1141, 2000.
[10] R. Cáceres, N. G. Duffield, J. Horowitz, and D. F. Towsley, “Multicast-based inference of network internal loss characteristics,” IEEE Transactions on Information Theory, vol. 45(7), pp. 2462–2480, November 1999.
[11] M. Rabbat, R. Nowak, and M. Coates, “Multiple source, multiple destination network tomography,” in Proceedings of the IEEE INFOCOM, Piscataway, NJ, USA, March 2004, pp. 1628–1639.
[12] V. N. Padmanabhan, L. Qiu, and H. J. Wang, “Passive network tomography using bayesian inference,” Microsoft Research, 2002.
[13] F. Ricciato, F. Vaccarino, W. Fleischer, J. Motz, and M. Rupp, “Passive tomography of a 3G network: Challenges and opportunities,” in Proceedings of the IEEE INFOCOM 2006, Barcelona, Spain, April 2006.
[14] H. Yao, S. Iagig, and M. Chen, “Passive network tomography for erroneous networks: A network coding approach,” IEEE Transactions on Information Theory, vol. 58(9), pp. 5922–5940, 2012.
“Software tools overview for the Cooperative Association for Internet Data Analysis (CAIDA),” [http://www.caida.org/tools/](http://www.caida.org/tools/), the homepage of CAIDA.

[64] N. G. Duffield, J. Horowitz, F. L. Presti, and D. F. Towsley, “Multicast topology inference from measured end-to-end loss,” *IEEE Transactions on Information Theory*, vol. 48, no. 1, pp. 26–45, January 2002.

[65] R. Cáceres, N. G. Duffield, J. Horowitz, and D. F. Towsley, “Multicast-based inference of network-internal loss characteristics,” *IEEE Transactions on Information Theory*, vol. 45, no. 7, pp. 2462–2480, November 1999.

[66] M. Coates and R. Nowak, “Network loss inference using unicast end-to-end measurement,” in *Proceedings of ITC Seminar on IP Traffic, Measurements and Modeling*, Monterey, California, USA, September 2004.

**Peng Qin** received the B. S. degree in Electronics and Information Engineering from Huazhong University of Science and Technology, Wuhan, P. R. China, in 2009. He is currently a PhD Candidate in the Department of Electronics and Information Engineering at the Huazhong University of Science and Technology. He is now a visiting student at the University of Victoria in British Columbia, Canada. His research interests are in the areas of network tomography, network measurement, P2P network and applications of network coding.

**Bin Dai** received the B. Eng, the M. Eng degrees and the PhD degree from Huazhong University of Science and Technology, China, in 2000, 2002 and 2006, respectively. From 2007 to 2008, he was a Research Fellow at the City University of Hong Kong. He is currently an associate professor at Department of Electronics and Information Engineering, Huazhong University of Science and Technology, P. R. China. His research interests include P2P network, wireless network, network coding, and multicast routing.

**Benxiong Huang** received the B. S. degree in 1987 and PhD degree in 2003 from Huazhong University of Science and Technology, Wuhan, P. R. China. He is currently a professor in the Department of Electronics and Information Engineering, Huazhong University of Science and Technology, P. R. China. His research interests include next generation communication system and communication signal processing.

**Guan Xu** received the B.S. degree in Electronics and Information Engineering from Huazhong University of science and technology, Wuhan, P. R. China, in 2008. He is currently a PhD Candidate in the Department of Electronics and Information Engineering at the Huazhong University of science and technology. His research interests are in the areas of practical network coding in P2P network, IP switch networks and SDN networks with emphasis on routing algorithms and rate control algorithms.

**Kui Wu** received the PhD degree in Computer Science from the University of Alberta, Canada, in 2002. He joined the Department of Computer Science at the University of Victoria, Canada, in the same year and is currently a professor there. His research interests include mobile and wireless networks, network performance evaluation, and network security. He is a senior member of the IEEE.
| NC-NT          | Objectives                  | Network structure | Probing type | Scenario                          | Advantages VS. non-NC tomography                                                                 |
|----------------|-----------------------------|-------------------|--------------|-----------------------------------|-------------------------------------------------------------------------------------------------|
| [29]           | topology inference          | tree structure    | active       | wireline                          | The first to connect NC with TI tomography which needs less probing packets and has faster convergence rate. |
| [30] [33]      | topology inference          | general networks  | active       | wireline                          | It can distinguish four 2-by-2 components (shown in Fig. 4) accurately which, however, is impossible without NC [31], [32]. |
| [36]           | topology inference          | general networks  | passive      | wireline/wireless                 | It proves that the transfer matrices $[T]$ for detectably different networks are distinct and can be utilized to distinguish between any network topology. |
| [37] [38]      | topology inference          | general networks  | passive      | wireline                          | It extends the above work to erroneous networks to passively infer network topology without explicit probing. |
| [60]           | topology inference          | tree structure    | passive      | wireline                          | It shows that subspace spanned by coded packets received at each node reveals topological information of the network which can be used for passive topology recovery. |
| [21]           | loss inference              | tree structure    | active       | wireline                          | The first to use network coding for inferring link loss rates in overlay networks with advantages of less bandwidth consumption, less complexity for realization and high inference accuracy. |
| [40]           | loss inference              | tree structure    | active       | wireline/wireless                 | It extends [21] and further presents that appropriately choosing the number of sources and receivers, as well as their location, can have a significant effect on the accuracy of the estimation. It also give guidelines on how to choose the best “points of view” of a network for LLI tomography. |
| [42]           | loss inference              | general networks  | active       | wireline                          | It addresses two main issues of LLI in general graphs and proposes an Orientation Algorithm to delete cycles. It also chooses a larger finite field to identify more links. With this approach each link is traversed by exactly one packet which is a great bandwidth saving compared to traditional multicast or unicast tomographic techniques. |
| [51]           | loss inference              | general networks  | active       | wireline/wireless                 | They propose an LA (Linear Algebraic) approach to developing consistent estimators of link loss rates by estimating not only the success rate of a single path, but also the success rate of any combination of paths, which is unique to network coding network and cannot be achieved by only routing probes. . |
| [44]           | loss inference              | general networks  | passive      | wireless                          | The first to address loss inference in wireless networks using network coding and it changes the fundamental connection between path and link loss rates from $\beta = \Pi_{e \in P} (1 - \alpha_e)$ to $\beta = \min_{e \in P} (1 - \alpha_e)$. |
| [50]           | bottleneck discovery        | tree structure    | passive      | wireless                          | The first to use subspace properties in network coding for the link loss inference problem in wireless sensor networks. They propose the PLI-RLC algorithm to infer link loss rate. Great performance improvement is gained compared to Bayesian inference algorithms above in [43]. |
| [51] [60] [62] | bottleneck discovery        | P2P networks      | passive      | wireline                          | It extends the traditional tomography scope to bottleneck discovery in P2P environment. |
| [61]           | failure localization        | general networks  | passive      | wireline                          | The first paper to show how the coding coefficient embedded in randomized network coding can be used to infer failure patterns, which consumes no extra overhead for probing. |