Trust-Assisted Anomaly Detection and Localization in Wireless Sensor Networks

Shanshan Zheng and John S. Baras
Institute for Systems Research
Department of Electrical and Computer Engineering
University of Maryland, College Park, MD, 20742
Email:{sszheng,baras}@umd.edu

Abstract—**Fast anomaly detection and localization is critical to ensure effective functioning of wireless sensor networks. The low bandwidth and power constraints in wireless sensor networks are the main challenges for achieving this task, especially for large scale networks. In this paper, we propose a trust-assisted framework for detecting and localizing network anomalies in a hierarchical sensor network. The proposed method makes inference based on end-to-end measurements collected by a set of measurement nodes. Network heterogeneity is exploited for better bandwidth and energy efficiency. The trustworthiness of network links is utilized to design an efficient two-phase probing strategy that can achieve a flexible tradeoff between inference accuracy and probing overhead. We performed experiments with different network settings and demonstrated the effectiveness of our proposed algorithms.**

I. Introduction

Wireless sensor networks have been widely employed for many real-world applications such as critical infrastructure monitoring, scientific data gathering, smart building, and personal medical systems, etc. Due to the possibly unattended and hostile operating environment, a sensor network may suffer from system failures due to loss of links and nodes, or malicious intrusions. Therefore, it is critical to continuously monitor the overall state of the sensor network to ensure its correct and efficient functioning. Compared to the diagnosis protocols for the Internet, monitoring and diagnosing wireless sensor networks have unique challenges due to the low communication bandwidth and the limited resources such as energy, memory and computational power.

Existing work on anomaly detection and localization in wireless sensor networks can be roughly divided into two categories: centralized and distributed. In centralized approaches, a central controller takes responsibility of monitoring and tracing anomalous behaviors in the network [1]. However, resource constrained sensor networks can not always afford to periodically collect all the measurements in a centralized manner. Distributed approaches address this issue by encouraging local decision making: for example, nodes can rely on neighbor coordination such as Watchdog [2] to detect misbehaving nodes. However, in a hostile environment, a node may not report its status or its neighbors' status honestly, and an intermediate node can intentionally alter its forwarded messages. One possible solution would be to use only end-to-end measurements if the end nodes can be trusted. Moreover, for large scale sensor networks where it may be difficult to access individual nodes, end-to-end data provides valuable information for inferring the internal status of the networks.

Making inference using end-to-end measurements has been extensively studied for wired networks such as the Internet [3], [4]. However, these techniques cannot be directly applied to sensor networks due to the resource constraints. In this paper, we present a trust-assisted framework for anomaly detection and localization in a hierarchical sensor network, where network heterogeneity is exploited for better bandwidth and energy efficiency. End-to-end measurements are collected through a two-phase probing. The goal of the first phase probing is to select probes that can cover as many anomalous links as possible and narrow down suspicious areas to be examined in the second phase. The probe selection problem in this phase is formulated as a budgeted maximum coverage problem, and we propose an efficient approximation algorithm to solve it based on linear programming duality. Experimental results show that our algorithm is much faster than the exact solution at the cost of a slight performance degradation. The second phase probing is aimed at locating individual links that are responsible for the observed end-to-end anomalies with minimum communication cost. In this phase, the probe selection problem is formulated as a Markov Decision process, where the probes are sequentially selected according to the previous observations and the predicted diagnosis quality. The prediction of diagnosis quality is carried out using the Loopy Belief Propagation (LBP) algorithm. Since the probe selection in this phase is sequential, it would require repeated executions of the LBP algorithm, which can be extremely redundant. We propose a heuristic method to improve algorithm efficiency based on the observation that adding one probing result at a time will affect only a small region in the graphical model for the inference problem. In both phases, the historic information on link trustworthiness is exploited to achieve a good tradeoff between the communication overhead and the inference accuracy. In summary, our key contributions in this paper include

- A two-phase probing strategy that enables a flexible tradeoff between communication overhead and inference accuracy for anomaly detection and localization in resource constrained wireless sensor networks.
- An approximation algorithm that utilizes link trustworthiness to effectively select the first-phase probe set and achieves good early detection of anomalous end-to-end behaviors within communication overhead constraints.
- An efficient second-phase probe selection strategy to locate individual anomalous links based on end-to-end measurements with minimum communication cost.
- Thorough validations through simulations under different network settings and performance comparisons with existing algorithms.

The paper is organized as follows. In Section II, we review literature work. Problem formulation is presented in Section III and the proposed approaches are described in Section IV. Section V shows the experimental results and Section VI concludes the paper and presents future work.

978-1-4577-0093-4/11/$26.00 ©2011 IEEE

II. Related Work

A. Network Tomography

Anomaly inference from end-to-end measurements can be generally categorized as a problem of network tomography, i.e., inferring a network's internal properties using information derived from end point data. It requires solving a system of equations that relate the end-to-end measurements to the link properties such as packet loss rate, link delay, etc. Most of the network tomography techniques are developed for wired networks such as the Internet [3], [4], [5]. Duffield et al. [3] formulated the tomography problem as a set-cover problem and solves it on a tree topology to identify the most unreliable links of the network. In [4], Nguyen et al. proposed a Boolean algebra based approach to improve inference accuracy by an order of magnitude over previous algorithms. Gu et al. [5] presented an optimal probing scheme for unicast network delay tomography that can provide the most accurate estimation. However, these techniques can not be directly applied to wireless sensor networks as they incur high probing overhead and computational complexity, while the sensor networks usually have severe resource constraints.

B. Monitoring Wireless Sensor Networks

Monitoring wireless sensor networks has recently generated a surge of interest from the research community [1], [6], [7]. Ramanathan et al.[1] proposed Sympathy, which carefully selects a minimal set of metrics at a centralized sink and uses an empirically developed decision tree to determine the most likely causes of detected failures. Nguyen et al. [6] proposed inference schemes based on maximum likelihood and Bayesian principles, which can isolate links with high loss rates even with routing changes and noisy measurements. Wang et al. [7] formulated the anomaly detection and localization problem as an optimal sequential testing guided by end-to-end data. They proposed a greedy algorithm that can determine an optimal selection sequence of network measurements to minimize testing cost, while the measurements are not limited to end-to-end behaviors. All of the above mentioned work assume the existence of a centralized sink node which is responsible for collecting required measurements, and rarely consider the strict constraint on communication overhead. In our work, we do not use any centralized sink node but exploit a hierarchical network structure to improve bandwidth and energy efficiency. Furthermore, our work focus on achieving a good tradeoff between communication overhead and inference accuracy, which provides a more practical and flexible solution for real-world applications.

C. Trust in Wireless Sensor Networks

In our work, we use link trust information to improve inference efficiency. The notion of trust, as a network security concept rather than a philosophical concept, corresponds to a set of relations among network nodes that are participating in certain protocols. With the knowledge of trust relations, it could be easier for the nodes to take proper security measures and make correct decisions.

There are various ways to numerically represent trust. For example, in [8], trust opinion is represented by a triplet in $[0,1]^3$, where the elements in the triplet represent belief, disbelief, and uncertainty, respectively. Trust can also be interpreted as probability, e.g., subjective probability is used in [9], while objective probability is used in [10]. Trust has also been used to secure wireless sensor networks. For instance, Tanachaiwiwat et al. [11] utilized trust to address the non-cooperative and malicious behavior in location-aware sensor networks. In [12], Probst et al. presented a distributed approach that establishes reputation-based trust among sensor nodes to identify compromised and malicious nodes and minimize their impacts. In our algorithm, the trustworthiness of a link is computed according to the historic observations of anomaly occurrences on the link. The information of link trustworthiness enables us to achieve an effective tradeoff between detection performance and communication overhead.

III. Problem Formulation

In this section, we describe the network model and define the problem of anomaly detection and localization. We use a two-layer heterogeneous network as illustrated in Fig. 1. The lower-layer network is the main network responsible for environment sensing and task execution. It consists of regular mote-type sensor nodes with severe energy and communication constraints. The upper-layer network is responsible for monitoring the status of the lower-layer network by taking end-to-end measurements. It consists of a set of measurement nodes with much stronger computation and communication capabilities. These measurement nodes are assumed to be highly trusted, e.g., they may associate with tamper resistant hardwares, so they would not give false information. We found by experiments that a small number of the measurement nodes would be enough to monitor a large network.

The main concern of using such hierarchical structure is to concentrate resource intensive computation and communication tasks only in the upper-layer network, thus to prolong the lifetime of the lower-level network. Heterogenous networks have become popular recently, particular in real world deployments because of their potential to increase network lifetime and throughput without significantly increasing the cost [13]. In this paper, the focus is not on quantitatively analyzing the performance of heterogeneous networks , but to enable flexible tradeoff between cost and network lifetime.

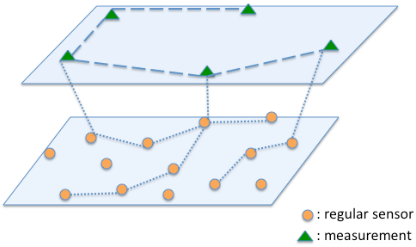


Fig. 1. A hierarchical network structure

To formulate the problem of anomaly detection and localization, we assume that the anomalies may occur on any links in the monitored network. The occurrence of an anomaly on one link is assumed to be independent of others. We will consider coordinated attacks in our future work. These network anomalies typically lead to deviations of the end-to-end measurements from the normal case, which can be explored for anomaly detection. Let $\mathcal{P}$ be the set of all end-to-end paths from the measurement nodes to the lower-layer network nodes. Denoting the set of links that appear in $\mathcal{P}$ by $\mathcal{E}$, we use a matrix A of dimension $|\mathcal{P}| \times |\mathcal{E}|$, called the routing matrix, to represent the information relating paths to links. Each row of A represents a path in $\mathcal{P}$, and each column represents a link in $\mathcal{E}$. The entry a_{ij} equal to 1 if path P_i contains link e_j. Let x_i be the indicator for the anomaly in path P_i, and y_j be the indicator for the anomaly in link e_j, i.e., $x_i = 1$ indicates that P_i is anomalous, and $y_j = 1$ indicates that link e_j is anomalous. Assuming that $|\mathcal{P}| = n_p$ and

$|\mathcal{E}| = n_e$, if the network has no noise, a path P_i is anomalous if any of its links is anomalous, i.e., $x_i = 1-\Pi_{j=1}^{n_e}(1-a_{ij}y_j)$, so we have at most n_p constraints on n_e variables. For a sensor network with severe communication constraints, the number of observations on path behaviors may not be sufficient to achieve a unique solution for the y_js. More generally, a practical network usually has some noise and follows the noisy-OR model [14], i.e., $P(x_i = 0|\mathbf{y}) = (1-\rho_j)\prod_j (a_{ij}\rho_{ij})^{y_j}$ where ρ_j is a leak probability representing the probability that path P_j performs as anomalous even if all its links are normal, and ρ_{ij} is an inhibition probability representing the probability that link e_j in path P_i is anomalous but performs as normal. Therefore, exact inference for the locations of the anomalous links may not be possible and we focus on the maximum a posterior estimation. The anomaly detection and localization problem can then be summarized as follows. We are given the following information:

- The set of wireless links $\mathcal{E} = \{e_1, \ldots, e_{n_e}\}$;
- The set of all paths $\mathcal{P} = \{P_1, \ldots, P_{n_p}\}$;
- The routing matrix $A = [a_{ij}]_{n_p \times n_e}$;
- Constraints on communication overhead.

The objective is to select a subset of paths in $\mathcal{P}$ to collect end-to-end measurements under communication constraints, and find the most probable candidates of anomalous links.

IV. Proposed Framework

In this section, we present a trust-assisted two-phase probing strategy for solving the anomaly detection and localization problem under communication constraints. In both phases, link trustworthiness is utilized to achieve the best possible performance under the given communication constraint. In the first phase, the probes are selected to cover as many anomalous links as possible and narrow down suspicious areas to be explored. In the second phase, probes are sequentially selected based on previous probing results and sent only to the suspicious areas to locate individual links responsible for the observed end-to-end anomalous behaviors. This incremental probing strategy is particularly important for large scale sensor networks, because collecting all the end-to-end measurements in $\mathcal{P}$ at one time may congest the network.

A. First phase probing

Ideally, to monitor network status, probes should cover all links in the network in order to detect all possible anomalies. However, probing all links at a rate that is sufficiently fast for the detection may cost high communication overhead and cause serious network congestion. Our first-phase probing scheme is motivated by the observation that different links might be exposed to different levels of risks, e.g., the attacker may be more likely to target some "important" links in the network. Therefore, the links should be probed with *different frequencies* and *priorities*. The priority of a link can be computed based on its trustworthiness and the obsoleteness of previously collected data of the link. The probes are then selected to cover links of highest priorities and satisfy a given communication constraint. The goal is to *optimize* the tradeoff between detection performance and probing overhead.

1) Problem Formulation: For each link $e_i \in \mathcal{E}$ at time k, we assign a trust value $t_i(k) \in [0,1)$ and an obsolete value $o_i(k) \in [0,1)$. The trust value represents the probability of e_i being normal, while the obsolete value indicates the obsoleteness of previously collected information about e_i. We use the Beta Reputation System [9] to capture link trustworthiness, which is based on using beta distribution to combine feedback and derive trust values. In Bayesian statistics, the beta distribution can be seen as the posterior probability of the parameter p of a binomial distribution, where p is the probability of success. In the Beta Reputation System, let $p_i(k)$ be the probability that link e_i is normal at time k, then $p_i(k)$ is assumed to have a beta distribution with parameters $\alpha_i(k)$ and $\beta_i(k)$, i.e.,

$$\begin{aligned} &f(p_i(k)|\alpha_i(k), \beta_i(k)) \\ &= \frac{\Gamma(\alpha_i(k), \beta_i(k)) p_i(k)^{\alpha_i(k)-1}(1-p_i(k))^{\beta_i(k)-1}}{\Gamma(\alpha_i(k))\Gamma(\beta_i(k))}, \end{aligned}$$

where $\Gamma(\cdot)$ is the gamma function. Given previously observed states of e_i till time k, $\alpha_i(k)$ and $\beta_i(k)$ can be represented by $\alpha_i(k) = r_i(k)+1$ and $\beta_i(k) = s_i(k)+1$, where $r_i(k)$ is the number of events that e_i is normal, and $s_i(k)$ is the number of events that e_i is anomalous. The trust value $t_i(k)$ is defined to be the mean value of $p_i(k)$, i.e., $t_i(k) = E[p_i(k)] = (r_i(k)+1)/(r_i(k)+s_i(k)+2)$. Since links may change behaviors over time, old observations are less relevant for the current trust value. A forgetting factor can be introduced so that $r_i(k+1) = \kappa_1 r_i(k) + I_i(k+1)$ and $s_i(k+1) = \kappa_2 s_i(k) + 1 - I_i(k+1)$, where $I_i(k+1)$ is the indicator function that equals to 1 if the $(k+1)^{th}$ observation of link e_i is normal. The forgetting factors κ_1 and κ_2 are positive and can be set differently, e.g., if we want to punish more on the occurrence of an anomalous event, we can set $\kappa_2 > \kappa_1$. More details on the Beta reputation system can be found in [9].

The obsoleteness of a link is set to 0 initially and updated according to the following rule. Let $A_i(k)$ be the event that e_i is not probed in the k^{th} interval and $A_i^C(k)$ be the event that e_i is probed in the k^{th} interval, then

$$o_i(k+1) = \left\{ \begin{array}{ll} 1-(1-o_i(k))e^{-\tau}, & \text{if } A_i(k), \\ 0, & \text{if } A_i^C(k). \end{array} \right.$$

where $\tau > 0$ is a parameter controlling the fading speed of the information. In other words, the more recent a link is probed, the smaller its obsolete value is. Let $w_i(k) = \rho \cdot (1-t_i(k)) + (1-\rho)\cdot o_i(k)$, with ρ being a parameter that adjusts the relative importance of the trust value and the obsolete value, then $w_i(k)$ can be used as a weight that indicates the urgency of probing link e_i.

The first-phase probing selection can be formulated as an optimization problem. Let h_i be the length of path P_i, and h_0 be the communication constraint such that the number of links traversed by the probes can not be larger than h_0. Let u_i be the indicator function for selecting path P_i, i.e., $u_i = 1$ means that path P_i is selected, and v_j be the indicator function for selecting link e_j, the optimization problem can be defined as

$$\max \quad z = \sum_{j\in\mathcal{E}} w_j \cdot v_j, \tag{1}$$

$$s.t. \quad \forall\ i \in \mathcal{P},\ u_i \in \{0,1\},\ \sum_{i\in\mathcal{P}} h_i u_i \le h_0, \tag{2}$$

$$\forall\ j \in \mathcal{E},\ v_j \in \{0,1\}\ \sum_{i\in\mathcal{P}} a_{ij}u_i - v_j \ge 0. \tag{3}$$

Constraint (2) represents the communication constraint and constraint (3) is due to the fact that one link may belong to multiple paths. The above optimization problem belongs to the class of the budgeted maximum coverage problem [15], which is NP-hard. Khuller et al. [15] proposed a $(1-1/e)$ approximation algorithm that achieves a best possible approximation ratio. However, their method involves an enumeration of all subsets of $\mathcal{P}$ that have cardinality k, where $k \ge 3$, which is too computationally expensive in our settings.

We propose an efficient approximation algorithm that has very low computational complexity to solve the problem and prove a performance bound. Experimental results in Section V show that our approximation algorithm achieves not only low computation overhead but also high approximation factor.

2) The approximation algorithm: The approximation algorithm and its analysis are based on linear programming duality. To solve the optimization problem, we first relax the integer constraints in (2) and (3) to pairs of linear constraints $0 \le u_i \le 1$ and $0 \le v_j \le 1$. It can be further shown that $0 \le u_i \le 1$ and $0 \le v_j \le 1$ can be equivalently changed to $u_i \ge 0$ and $v_j \le 1$. Then the dual problem of the original optimization problem can be written as

$$\min\nolimits_{\lambda,\gamma} \quad \lambda h_0 + \sum_{j\in\mathcal{E}} \gamma_j \tag{4}$$

$$\text{s.t.}\ \ \forall\ i \in \mathcal{P},\ \lambda h_i + \sum_{j\in\mathcal{E}} a_{ij}\gamma_j \ge \sum_{j\in\mathcal{E}} a_{ij} w_j = \tilde{w}_i, \tag{5}$$

where λ, γ_j for $j = 1, \dots, n_e$, are Lagrange multipliers, and $\tilde{w}_i$ is the sum of link weights for path P_i.

We denote the set of selected paths as $\mathcal{X}$, the set of links covered by $\mathcal{X}$ as $\mathcal{E}(\mathcal{X})$, and the total hop counts of the paths in $\mathcal{X}$ as hops$(\mathcal{X})$. Note that hops$(\mathcal{X})$ is usually larger than $|\mathcal{E}(\mathcal{X})|$ as paths can have overlapping links. Let $q_0 = \arg\max_{i\in\mathcal{P}} \tilde{w}_i/h_i$, the algorithm starts with the feasible dual solution $\mathcal{X} = \{P_{q_0}\}$, hops$(\mathcal{X}) = h_{q_0}$, $\lambda = \tilde{w}_{q_0}/h_{q_0}$ and $\gamma_j = 0$ for $j \in \mathcal{E}(\mathcal{X})$. Then in each iteration, the basic idea is to reduce the dual objective value while improving the primal objective value. Initially, the objective value of the dual problem is λh_0 and only the q_0^{th} constraint in (5) is active. To reduce the dual objective value in each iteration, we choose one inactive constraint in (5), say, the i^{th} constraint and make it active. We reduce the value of λ by a constant β and raise the value of γ_j by the same amount β. Let overlap $= |\mathcal{E}(\{P_i\}) \cap \mathcal{E}(\mathcal{X})|$ be the number of overlapping links between P_i and $\mathcal{X}$, the change to the left side of the i^{th} constraint will be $-h_i \cdot \beta + \text{overlap} \cdot \beta < 0$ as $h_i = |\mathcal{E}(\{P_i\})| \ge$ overlap. Therefore, the i^{th} constraint can be made active with a properly selected positive value of β. In order to keep the dual solution feasible, all other constraints should not be violated, so the chosen constraint must be associated with the smallest β among all the inactive constraints. For the already active constraints, the change of their left side equations will be $-h_i \cdot \beta + \sum_j a_{ij} \cdot \beta = 0$, so they are still active and not violated. After each iteration, the change of the dual objective value is $\beta \cdot (|\mathcal{E}(\mathcal{X})| - h_0)$. Since $|\mathcal{E}(\mathcal{X})| <$ hops$(\mathcal{X})$, the dual objective value will be reduced as long as the primal solution is feasible, i.e., hops$(\mathcal{X}) <= h_0$. The algorithm will terminate as soon as the communication constraint is violated, i.e., hops$(\mathcal{X}) > h_0$. Table I is a summary of the algorithm for first-phase probe selection.

The probe selection algorithm is performed among the measurement nodes only. Each measurement node maintains a partial view of the network, i.e., the information of the links that constitute its monitored paths, which can be extracted from the routing protocol running in the lower-layer network, and the corresponding link weights, which are updated locally by the Beta reputation system. In the algorithm described in Table I, except the computation of $(\max \tilde{w}_i/h_i)$ in the 1st line and the computation of $(\min \beta_i)$ through line 4-10 , other computations are local, i.e., they can be executed by individual measurement node without exchange of information. The distributed computation of minimum can be achieved easily by maintaining a spanning tree among the measurement nodes [16]. Furthermore, the iterations in the 'While-end' statement through line 2-19 can be finished in less than h_0/h_{min} rounds, with h_{min} being the length of the shortest path in $\mathcal{P}$. Since the number of measurement nodes is small, the communication cost for this algorithm can be kept low.

TABLE I
A SEQUENTIAL ALGORITHM FOR FIRST-PHASE PROBE SELECTION

```
1  Initialization: $q_0 = \arg\max_{i\in\mathcal{P}} \tilde{w}_i/h_i$,
     $\mathcal{X} = \{P_{q_0}\}$, hops$(\mathcal{X}) = h_{q_0}$, $\lambda = \tilde{w}_{q_0}/h_{q_0}$, $\gamma_j = 0$.
2  While hops$(\mathcal{X}) < h_0$,
3    $\beta = inf, idx = 0$
4    for each $i \notin \mathcal{X}$
5      overlap $= |\mathcal{E}(\{P_i\}) \cap \mathcal{E}(\mathcal{X})|$
6      $\beta_i = \frac{\lambda \cdot h_i + \sum_{j\in\mathcal{E}} a_{ij}\gamma_j - \tilde{w}_i}{h_i - \text{overlap}}$
7      if $\beta > \beta_i$
8        $\beta = \beta_i$, idx $= i$;
9      end
10   end
11   if hops$(\mathcal{X}) + h_{idx} <= h_0$
12     $\lambda = \lambda - \beta$,
13     $\forall\ i \in \mathcal{E}(\mathcal{X}), \gamma_i = \gamma_i + \beta$,
14     $\forall\ i \notin \mathcal{E}(\mathcal{X}), \gamma_i = \gamma_i$,
15     $\mathcal{X} = \mathcal{X} \cup P_{idx}$,
16   else
17     terminate
18   end
19 end
```

Next, we derive the performance bound for the proposed approximation algorithm.

Theorem 1: Assuming that the algorithm in Table I terminates after l_1 iterations and the primal solution is $\mathcal{X} = \{P_{s_1}, \dots, P_{s_{l_1}}\}$ with objective value z^p. Denote the optimal objective value for the primal problem by z^*, then z^p can be lower bounded by $z^p > \frac{z^*}{\delta(1+r/l_1)}$, where δ is the maximum number of paths in $\mathcal{X}$ that intersect at a same link, and $r = h_{max}/h_{min}$, where h_{max} and h_{min} are the maximum and minimum path lengths.

Proof: The primal objective value z^p can be written as $z^p = \sum_{j\in\mathcal{E}(\mathcal{X})} w_j$ according to (1), which is lower bounded by $\frac{1}{\delta}\sum_{i\in\mathcal{X}} \tilde{w}_i$. Since the paths in $\mathcal{X}$ have active constraints in the dual problem, we have $[\sum_{i\in\mathcal{X}} \tilde{w}_i = \sum_{i\in\mathcal{X}}(\lambda h_i + \sum_j a_{ij}\gamma_j) < \delta \cdot z^p$. On the other hand, the dual objective value z^d can be written as $z^d = \lambda h_0 + \sum_{j\in\mathcal{E}} \gamma_j$. Since $\gamma_j = 0$ for $j \notin \mathcal{E}(\mathcal{X})$, we have

$$\begin{aligned} z^d &= \lambda(h_0 - \sum_{i\in\mathcal{X}} h_i) + \sum_{i\in\mathcal{X}}(\lambda h_i + \sum_j a_{ij}\gamma_j) \\ &< \lambda h_{max} + \delta \cdot z^p. \end{aligned}$$

Since $\lambda \sum_{i\in\mathcal{X}} h_i < \delta \cdot z^p$ and $|\mathcal{X}| = l_1$, we have

$$\lambda h_{max} < \frac{\delta \cdot z^p}{l_1 \cdot h_{min}} h_{max}.$$

Then z^d can be upper bounded by $z^d < \delta \cdot z^p \cdot (1 + r/l_1)$ and we have the optimal objective value z^* satisfying

$$z^* < z^d < \delta(1 + r/l_1) z^p.$$

■

Experimental results in Section V show that this approximation algorithm achieves very good approximation ratio as compared to the optimal solution.

After the probes are selected and sent, the measurement nodes will do hypothesis testings to detect anomalous paths based on the collected end-to-end measurements. The detection is based on identifying significant measurement deviations from the normal state. We have proposed to use either sequential probability ratio test (SPRT) or non-parametric change detection method for the hypothesis testing depending on whether a training data set is available. More details on the detection algorithms and their performance analysis can be found in our previous work [17].

B. Second phase probing

The goal of the second-phase probing is to find the individual links responsible for the observed path anomalies. In this phase, additional probes are sequentially selected according to previous observations and the predicted diagnosis quality. This online selection is typically more efficient than its offline counterpart [18], where the offline method attempts to select the set of probes before any observation is made.

To measure the diagnosis quality, we use the conditional entropy from the information-theoretic perspective. Assuming that the observed path states from the first phase is represented by $\mathbf{x}_1$. Denoting the second-phase probe selection strategy by π and the link states in $\mathcal{E}$ by $\mathcal{Y}$, then the diagnosis quality of π can be represented by $f(\pi) = H(\mathcal{Y}|\mathbf{x}_1) - H(\mathcal{Y}|\mathbf{x}_1, \phi(\pi))$, where $\phi(\pi)$ is the collected end-to-end measurements by implementing π. Let $h(\pi)$ be the communication overhead by implementing π, measured by the number of links traversed by the selected probes, and $\tilde{h}_0$ be the communication constraint for the second phase, then the probe selection problem in this phase can be formulated as to find a policy π^* such that

$$\pi^* = \arg\max_{\pi} E_\pi[H(\mathcal{Y}|\mathbf{x}_1) - H(\mathcal{Y}|\mathbf{x}_1, \phi(\pi))], \text{ s.t. } h(\pi) \le \tilde{h}_0.$$

However, solving this problem is equal to solving a finite-horizon Markov Decision Process that has exponential state space [19], which is NP-hard. A widely used method for solving this type of problem is to use a heuristic greedy approach that iteratively selects the probe that provides the largest reduction in uncertainty at each step [14], [20]. More specifically, let $\mathcal{X}_C$ represent the previously sent probes, including the probes sent in the first phase, and assuming that the observations are $\mathcal{X}_C = \mathbf{x}_C$ and the communication overhead is $h(\mathcal{X}_C)$, then the next probe is selected to be

$$i = \arg\max_{j: h_j \le \tilde{h}_0 - h(\mathcal{X}_C)} H(\mathcal{Y}|\mathbf{x}_C) - H(\mathcal{Y}|\mathbf{x}_C, X_j), \tag{6}$$

where X_j is a random variable representing the unknown state of path P_j and h_j is the hop count for path P_j. We now provide performance bound for this greedy algorithm.

Theorem 2: Assuming that the obtained diagnosis quality, i.e., the reduced uncertainty on link states, by the greedy algorithm is $\tilde{\Delta}$, and the optimal diagnosis quality is Δ^*, then $\tilde{\Delta}$ can be lower bounded by $\tilde{\Delta} \ge (1 - e^{-l_2 \cdot h_{min}/\tilde{h}_0})\Delta^*$, where l_2 is the number of probes selected by the greedy algorithm, h_{min} is the minimum path length, and $\tilde{h}_0$ is the communication constraint in the second probing phase.

Proof: First, the diagnosis quality $f(\pi)$, measured in the reduction of estimation uncertainty, satisfies the adaptive monotonicity property defined in [19], i.e. getting more probing observations can never decrease the diagnosis quality. Second, it also satisfies the adaptive submodularity property, i.e., when there are fewer probes that have been sent, probing one additional path can provide the same or more information than the case of probing the same additional path but there are more probes that have been sent. In other words, for $\mathcal{X}_A \subseteq \mathcal{X}_B \subset \mathcal{X}$ and $X_i \in \mathcal{X} \backslash \mathcal{X}_B$, we have

$$\begin{aligned} f(\mathcal{X}_A, X_i) - f(\mathcal{X}_A) &= H(\mathcal{Y}|\mathcal{X}_A) - H(\mathcal{Y}|\mathcal{X}_A, X_i) \\ \ge f(\mathcal{X}_B, X_i) - f(\mathcal{X}_B) &= H(\mathcal{Y}|\mathcal{X}_B) - H(\mathcal{Y}|\mathcal{X}_B, X_i), \end{aligned}$$

which is due to the fact that

$$\begin{aligned} &H(\mathcal{Y}|\mathcal{X}_A) - H(\mathcal{Y}|\mathcal{X}_A, X_i) - (H(\mathcal{Y}|\mathcal{X}_B) - H(\mathcal{Y}|\mathcal{X}_B, X_i)) \\ &= \sum_{\mathcal{X}_B} \sum_{X_i} P(\mathcal{X}_B, X_i) \log \frac{P(\mathcal{X}_B, X_i)}{P(\mathcal{X}_B) P(\mathcal{X}_i|\mathcal{X}_A)} \\ &= D_{KL}(P(\mathcal{X}_B, X_i) || P(\mathcal{X}_B) P(\mathcal{X}_i|\mathcal{X}_A)) \ge 0, \end{aligned}$$

where $D_{KL}(\cdot)$ represents the Kullback-Leibler divergence. Following Theorem 3 in [19], we have $\tilde{\Delta} \ge (1 - e^{-l_2/k})\Delta^*$, where l_2 is the number of steps for the greedy algorithm and k is that of the optimal algorithm. In our case, the number of steps in the greedy algorithm is equal to the number of probes selected by the greedy algorithm, and the number of steps for an optimal algorithm can be upper bounded by $k \le \tilde{h}_0 / h_{min}$, so we have $\tilde{\Delta} \ge (1 - e^{-l_2 \cdot h_{min}/\tilde{h}_0})\Delta^*$. ■

Next, we discuss how to implement the greedy algorithm. The graphical model for computing $H(\mathcal{Y}|\mathbf{x}_C, X_j)$ is illustrated in Fig. 2, where X_j represents the state of path P_j and Y_i represents the state of link e_i. By simple algebraic manipulation, it is found that $H(\mathcal{Y}|\mathbf{x}_C, X_j)$ satisfies

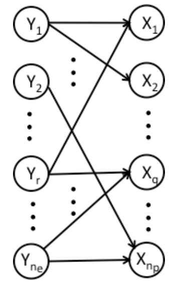


Fig. 2. Graphical Model

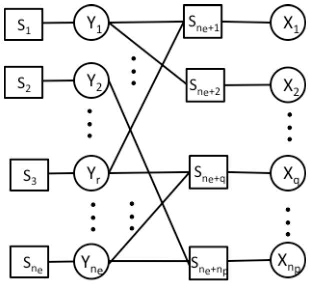


Fig. 3. Factor Graph for the Graphical Model

$$H(\mathcal{Y}|\mathbf{x}_C, X_j) = H(\mathcal{Y}|\mathbf{x}_C) - H(X_j|\mathbf{x}_C) + H(X_j|\mathcal{Y}_{pa_j}, \mathbf{x}_C),$$

where $\mathcal{Y}_{pa_j}$ is the set of the parent nodes of X_j in Fig. 2. Since $H(\mathcal{Y}|\mathbf{x}_C)$ is not related to X_j, only $H(X_j|\mathbf{x}_C)$ and $H(X_j|\mathcal{Y}_{pa_j}, \mathbf{x}_C)$ need to be computed. The computations can be carried out using the Loopy Belief Propagation algorithm (LBP). We will first briefly review the LBP algorithm.

LBP [21] is a message passing algorithm for performing inference on graphical models. It calculates the marginal distribution for each unobserved node, conditioned on any observed nodes. It usually operates on a factor graph, which is a bipartite graph containing nodes corresponding to variables and factors. In the factor graph, an undirected edge connects a variable node c and a factor node s if and only if the variable participates in the potential function f_s of s. Let $\mathbf{x}$ denote a set of n discrete variables and s a factor node, then the joint mass function of $\mathbf{x}$ can be written as

$$P(\mathbf{x}) = \frac{1}{Z} \prod_s f_s(\mathbf{x}_s),$$

where Z is a normalization constant, $\mathbf{x}_s$ is the set of neighboring variable nodes of s, and the index s ranges over all factor nodes in the graphical model. The factor graph for our graphical model is shown in Fig. 3. More details on factor graph can be found in [21]. In LBP, the probabilistic messages are iterated among the variable nodes and the factor nodes. Let $\mathcal{N}(x_i)$ denote the neighboring factor nodes of a variable node

x_i, then the LBP message from x_i to one of its neighboring factor node s is defined as

$$\mu_{x_i \to s}(x_i) = \prod_{\tilde{s} \in \mathcal{N}(x_i) \setminus s} \mu_{\tilde{s} \to x_i}(x_i),$$

and the message from factor node s to x_i is defined as

$$\mu_{s \to x_i}(x_i) = \sum_{\mathbf{x}_s \setminus x_i} f_s(\mathbf{x}_s) \prod_{x_j \in \mathbf{x}_s \setminus x_i} \mu_{x_j \to s}(x_j),$$

where $\mathbf{x}_s$ is the set of neighboring variable nodes of s. Based on these messages, the beliefs for each variable node and the probability potential for each factor node can be computed as,

$$\begin{aligned} P(x_i) &\propto \prod_{s \in \mathcal{N}(i)} \mu_{s \to x_i}(x_i), \\ P(\mathbf{x}_s) &\propto \prod_{x_i \in \mathcal{N}(s)} \mu_{x_i \to s}(x_i) \end{aligned}$$

In our case, since $\mathbf{x}_C$ are observed nodes, the original LBP algorithm already provides the computation of the marginal distribution $P(X_j|\mathbf{x}_C)$. $H(X_j|\mathbf{x}_C)$ can then be easily computed as $\sum_{x_j} P(x_j|\mathbf{x}_C) \log P(x_j|\mathbf{x}_C)$. To compute $H(X_j|\mathcal{Y}_{pa_j}, \mathbf{x}_C)$, we first write it as

$$\begin{aligned} & H(X_j|\mathcal{Y}_{pa_j}, \mathbf{x}_C) = -\sum_{\mathcal{Y}_{pa_j}} P(\mathbf{y}_{pa_j}, x_j|\mathbf{x}_C) \log P(x_j|\mathbf{y}_{pa_j}) \\ & = -\sum_{\mathcal{Y}_{pa_j}, x_j} P(\mathbf{y}_{pa_j}, x_j|\mathbf{x}_C) \log P(\mathbf{y}_{pa_j}, x_j|\mathbf{x}_C) \\ & + \sum_{\mathcal{Y}_{pa_j}} P(\mathbf{y}_{pa_j}|\mathbf{x}_C) \log P(\mathbf{y}_{pa_j}|\mathbf{x}_C), \end{aligned} \tag{7}$$

then consider the two terms in the right side of equation (7). For the first term, $\mathbf{y}_{pa_j}$ and x_j are variable nodes that are neighbors of the factor node s_{n_e+j}, which has the joint mass function

$$P(\mathbf{y}_{pa_j}, x_j|\mathbf{x}_C) = f(x_j|\mathbf{y}_{pa_j}) \cdot \mu_{x_j \to s_{n_e+j}} \cdot \prod_{y_k \in \mathbf{y}_{pa_j}} \mu_{y_k \to s_{n_e+j}}.$$

The original LBP message from s_{n_e+j} to x_j is

$$\mu_{s_{n_e+j} \to x_j} = \sum_{\mathbf{y}_{pa_j}} P(\mathbf{y}_{pa_j}, x_j|\mathbf{x}_C).$$

Let $\tilde{\mu}_{s_{n_e+j} \to x_j} = \sum_{\mathbf{y}_{pa_j}} P(\mathbf{y}_{pa_j}, x_j|\mathbf{x}_C) \log P(\mathbf{y}_{pa_j}, x_j|\mathbf{x}_C)$ and piggyback it in the original LBP message, then the first term can be obtained at node x_j as $\sum_{x_j} \tilde{\mu}_{s \to j}(x_j)$. Similarly, the second term in the right side of equation (7) can be obtained by first computing $P(\mathbf{y}_{pa_j}|\mathbf{x}_C) = \sum_{x_j} P(\mathbf{y}_{pa_j}, x_j|\mathbf{x}_C)$ at node s_{n_e+j} and then passing $\hat{\mu}_{s_{n_e+j} \to x_j} = \sum_{\mathbf{y}_{pa_j}} P(\mathbf{y}_{pa_j}|\mathbf{x}_C) \log P(\mathbf{y}_{pa_j}|\mathbf{x}_C)$ to node x_j. Therefore, $H(X_j|\mathcal{Y}_{pa_j}, \mathbf{x}_C)$ can be obtained at x_j.

To implement the above mentioned algorithm for second-phase probe selection, there are two main concerns. First is the mapping from the graphical model to the sensor network as we should limit messaging across sensors whenever possible in order to save resources. Second is that whenever an additional probe is selected, the beliefs need to be updated, which requires repeated executions of LBP and may lead to very high computational complexity. We propose a heuristic algorithm to reduce the computation overhead and improve algorithm speed by utilizing the redundancy in each execution of LBP.

a) Mapping the graphical model to the sensor network: The second-phase probe selection algorithm is also performed among the measurement nodes. For the graphical model shown in Fig. 2, we call the X_is evidence nodes as they represent path states and can be observed, and call the Y_js latent nodes as they represent the hidden link states. First, we assign each evidence node to the measurement sensor that monitors the corresponding path. Observing that in the graphical model the evidence nodes are only coupled through the latent nodes, we decouple them by making duplicated copies of the latent nodes [22]. More specifically, assuming that the evidence nodes X_p and X_q are coupled through a latent node Y_r, while the corresponding measurement sensors for the two evidence nodes, denoted by S_p and S_q, are connected through a path constituted of measurement sensors $(S_{r_1}, \ldots, S_{r_n})$. Then we add new variables $(Z_{r_1}, \ldots, Z_{r_n})$ to the original graphical model, remove the link from Y_r to X_q, and add new links for $(Y_r, Z_{r_1}), (Z_{r_1}, Z_{r_2}), \ldots, (Z_{r_{n-1}}, Z_{r_n}), (Z_{r_n}, X_q)$ with

$$\begin{aligned} & f(Z_{r_1}|Y_r) = \boldsymbol{I}(Z_{r_1} = Y_r), \\ & f(Z_{r_{k+1}}|Z_{r_k}) = \boldsymbol{I}(Z_{r_{k+1}} = Z_{r_k}), \ k = 1, \ldots, n-1, \\ & f(X_q|Z_{r_n}) = P(X_q|Y_r), \end{aligned}$$

where $\boldsymbol{I}(\cdot)$ is the indicator function. The factor graph of the modified graphical model can be derived accordingly. The newly added variables $(Z_{r_1}, \ldots, Z_{r_n})$ are assigned to the measurements sensors $(S_{r_1}, \ldots, S_{r_n})$ and the latent variable Y_r is assigned to the measurement sensor S_p.

b) Belief Updating: Belief updating is the most computational intensive part in the second-phase probe selection algorithm as the beliefs need to be updated each time when one additional probe is collected. In [14], Zheng et al. also utilized belief propagation for online probe selection, however, the computation complexity is pretty high in their implementation due to the repeated executions of belief propagation. In [20], Cheng et al. proposed to improve algorithm efficiency based on an observation of approximated conditional independence of probes. However, their method does not hold true in the online scenario where each probe is sequentially selected according to previous observations.

We propose a heuristic algorithm that exploit the redundancy in the repeated executions of LBP to reduce computational complexity. It is based on the observation that adding one evidence at a time may only affect a small region in the graphical model. Therefore, messages should be updated only in that region. The similar principle is used for expanding frontier belief propagation (EFBP) in [23]. However, EFBP focus on the case where the choice of evidence variables is fixed but the evidence changes, while in our case the choice of evidence variable is not fixed, i.e., one more evidence is added each time. Our algorithm starts with one run of the full LBP algorithm to select the first probe given the observations from the first-phase probing. Then for each time when an additional probe is sent and the corresponding path state x_i is observed, the message from x_i to its neighboring factor node s, i.e., $\mu_{x_i \to s}$, will be updated and sent if and only if it differs by ϵ from the previous one when x_i last participated in belief propagation. Similarly, if s receives a new message, it will update and send messages to neighbors if and only if the new message differs by ϵ from the last one s received. In most cases, the effect of adding one evidence dies out very quickly, so the number of message passing is greatly reduced compared to the full LBP algorithm. It is found by experiments that this heuristic approximation algorithm can achieve similar performance as the repeated executions of full LBP, while the speed is more than one order of magnitude faster. Table II summaries the algorithm, where $\tilde{h}_0$ is the communication constraint for the second phase and $h(x_i)$ represents the hop count of the path corresponding to x_i.

TABLE II
AN IMPROVED LBP BASED ALGORITHM FOR SECOND-PHASE PROBE SELECTION

```
Initialization: perform LBP to select the first probe $x_1$,
  obtain initial belief messages $\mu^{[0]}_{\cdot\to\cdot}$,
  $k = 1,\ h_k = h(x_1), converged = false,\ \mathcal{S}^{[0]} = \{x_1\}$,
while $h_k <= \tilde{h}_0$
  while $converged == false$
    $converged = true,\ \mathcal{S}^{[1]} = \emptyset$
    for $s \in \mathcal{S}^{[0]}$
      for $t \in \mathcal{N}(s)$
        compute $\mu^{[1]}_{s\to t}$ based on new observed probing
         result or new received message from neighbors
        if $|\mu^{[1]}_{s\to t} - \mu^{[0]}_{s\to t}| > \epsilon$
          $converged = false$, send $\mu^{[1]}_{s\to t}$ to $t$
          $\mu^{[0]}_{s\to t} = \mu^{[1]}_{s\to t},\ \mathcal{S}^{[1]} = \mathcal{S}^{[1]} \cup t$
        end if
      end for
    end for
    $\mathcal{S}^{[0]} = \mathcal{S}^{[1]}$
  end while
  $k = k + 1$
  select the $k^{th}$ probe $x_k$, $h_k = h_{k-1} + h(x_k)$
  $\mathcal{S}^{[0]} = \{x_k\}$
end while
```

V. EVALUATION

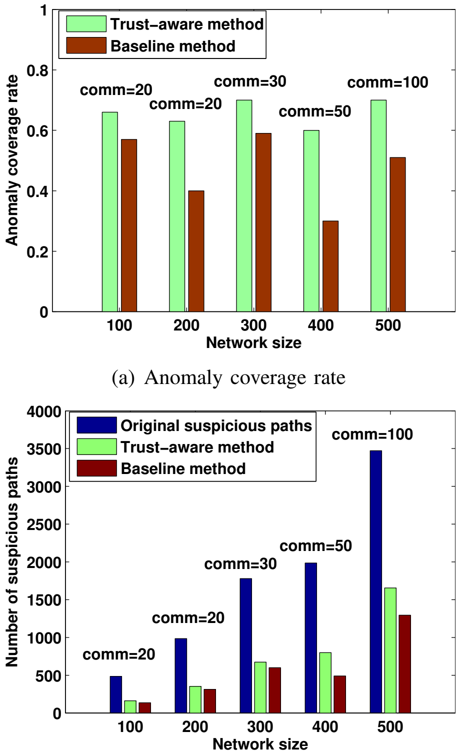
To evaluate the performance of the proposed algorithm, we generate a set of different networks. The sensor nodes are assumed to be uniformly deployed. A small number of nodes are randomly selected as the measurement nodes (M nodes) to collect end-to-end measurements. The monitored paths from measurement nodes to other regular sensor nodes follow the shortest paths. Table III summarizes the parameters for the evaluated networks with different sizes.

TABLE III
NETWORK PARAMETERS

Net size	Avg. node degree	Num. of M nodes	Num. of links	Num. of paths	Avg. path lengths
100	5.36	5	268	485	3.30
200	5.06	5	506	985	5.21
300	4.68	5	702	1779	5.30
400	5.09	5	1049	1985	6.39
500	6.27	7	1568	3472	7.83

In the experiments, the occurrence of an anomaly on each link in the network is modeled as a Bernoulli process. The probability associated with each Bernoulli process may be adjusted for different anomaly densities. For simplicity, we assume that once an anomaly occurs, it remains anomalous until being detected. We report results for the two probing phases under different experimental settings, including different network sizes, communication constraints and anomaly densities. Since existing work typically do not consider a strict communication constraint, we design a simple baseline algorithm that can work under communication constraints and compare our trust-assisted algorithm with the baseline algorithm. The baseline algorithm is assumed to be not aware of the trust information. It follows the same procedures as the trust-assisted algorithm but assumes that each link is equally likely to be anomalous with probability 0.5.

1) First-phase probing: The goal of the first-phase probing is to cover as many anomalous links as possible given the communication constraint and at the same time, narrow down the suspicious areas in the network. We propose to evaluate the algorithm performance using two metrics. The first metric is anomaly coverage rate, which is computed as the ratio of the anomalous links covered by the candidate probes to all anomalous links. The second metric is the suspicious area measured by number of suspicious paths after the first-phase probing. Since only end-to-end anomalous behaviors (path anomalies) are identified in the first phase, the suspicious paths represent those paths that intersect with the discovered anomalous paths. They are also candidate paths for the second-phase probing for locating individual anomalous links. Fig. 4(a) shows the anomaly coverage rate for our trust-aware probe selection method and the baseline method in different networks under given communication constraints. Fig. 4(b) shows the number of suspicious paths after the first-phase probing for these networks. The x-axis of the two figures show network sizes while other network parameters are shown in table III. The number above the bars are the corresponding communication constraints used for the first phase. For larger network sizes, we used larger communication constraints. Anomaly densities in these network are around 20%, i.e., about 20% of the links in the networks are anomalous.



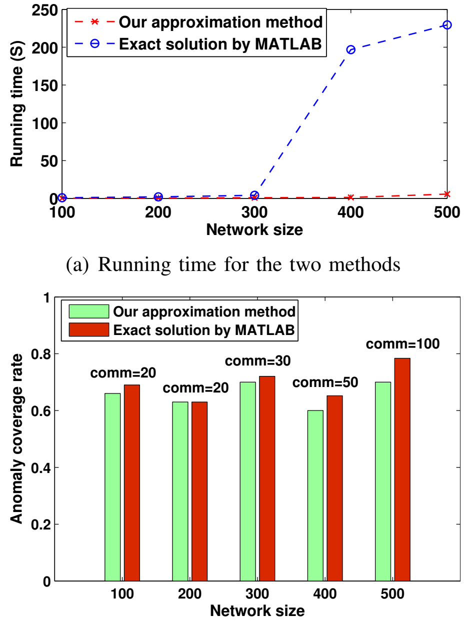
(a) Anomaly coverage rate

(b) Number of suspicious paths

Fig. 4. Comparison of the trust-aware method and baseline method

Fig. 4(a) shows that the trust-aware method can cover more anomalous links than the baseline method given the same communication constraints. Fig. 4(b) demonstrates that both the trust-aware algorithm and the baseline algorithm have scaled down the suspicious areas: the number of candidate probes have been greatly reduced. We also note that the baseline method may achieve smaller suspicious areas, that is because it identifies much fewer anomalous paths, which will lead to fewer intersecting suspicious paths.

Another concern for the first-phase probe selection, as mentioned before, is the computational complexity of the algorithm. We compare our approximation method implemented in MATLAB to the exact method provided by MATLAB integer programming solver, in terms of both speed and performance. The experimental settings are the same as the previous one. Fig. 5(a) shows the running time of our method and the exact solution for different networks. Fig. 5(b) shows the anomaly coverage rate for the two methods.



(a) Running time for the two methods

(b) Anomaly coverage rate for the two methods

Fig. 5. Comparison of the approximation method and the exact solution

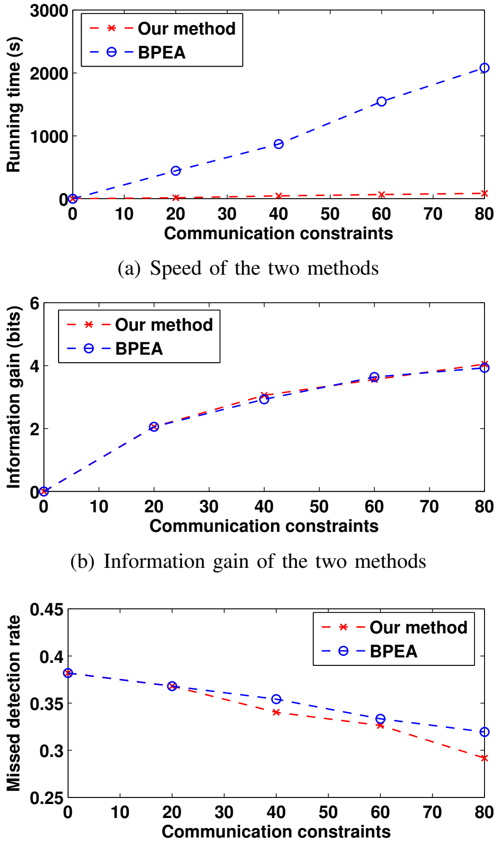
For small networks, since the communication constraint is also small, the budgeted maximum coverage problem has small scale, therefore, the exact solution is only about 3 times slower than the approximation method. However, the approximation method starts to provide much superior performance over the exact method when the network size becomes larger, e.g., for network size larger than 300, the approximation method is more than one order of magnitude faster than the exact method, while reduction on the anomaly coverage rate is less than 0.1. Considering the much larger running time of the exact solution, e.g, in the scale of minutes, the performance reduction of the approximation method is acceptable.

2) Second-phase probing: For the second-phase probing, the goal is to find the individual links that are responsible for the observed path anomalies. The performance is evaluated in terms of the missed detection rate (MDR) and false alarm rate (FAR). The missed detection rate is represented by the percentage of anomalous links that are not found, and the false alarm rate is represented by the percentage of normal links that are recognized as anomalous. Denote $\mathcal{E}$ as the set of monitored links, $\mathcal{E}_A$ as the set of anomalous links, and $\mathcal{E}_D$ as the set of anomalous links that are correctly localized, then

$$MDR = 1 - \frac{|\mathcal{E}_A \cap \mathcal{E}_D|}{|\mathcal{E}_A|},\ FAR = \frac{|\mathcal{E}_D \cap (\mathcal{E} - \mathcal{E}_A)|}{|\mathcal{E} - \mathcal{E}_A|}$$

We have implemented our probe selection algorithm in this phase on top of the belief propagation algorithm from Kevin Murphy's Bayes Net toolbox [24]. To demonstrate the effectiveness of our heuristic method on improving algorithm efficiency, we also implemented the BPEA algorithm proposed by Zheng et al. [14]. BPEA applied a similar repeated LBP algorithm for online probe selection, which, to the best of our knowledge, provides the best inference accuracy in the literature. However, BPEA did not fully exploited the redundancy in the repeated executions of LBP and it has very high computational complexity.

Fig. 6 shows the performance comparison of BPEA and our algorithm in a network with 400 nodes. Other network parameters are shown in the corresponding row in Table III. In this experiment, anomaly density is around 30%. The communication constraint is 50 for the first-phase probe selection. Fig. 6(a) compares the running time of the two methods. Fig. 6(b) compares the information gain in bits for the two methods, and Fig. 6(c) shows the missed detection rate. The false alarm rate is very small in both cases, i.e., around 0.04, and sending more probes did not change the false alarm rate much, so we don't show it here. The x-axis in the figures corresponds to the communication constraint for the second phase.



(a) Speed of the two methods

(b) Information gain of the two methods

(c) Missed detection rate of the two methods

Fig. 6. Comparison of BPEA and our approach

We can see that our heuristic method indeed speeds up the probe selection process a lot, i.e., more than one order of magnitude than BPEA. In the case that the communication constraint is 80 hop counts, BPEA need about 35 minutes, which is not realistic for real time probe selection, while our approach only need one and half minutes. The information gain, in terms of the reduced uncertainty, obtained by the two methods are almost the same for given communication constraints, while the missed detection rate is even a slightly lower using our heuristic method. Over various experiments that we have run with different networks, our heuristic method and the BPEA method can always achieve similar performances.

Next we change the anomaly densities in the network. It is observed that when the anomaly density reduces, both MDR and FAR become lower. Due to space limit, Fig. 7 only shows the change of MDR.

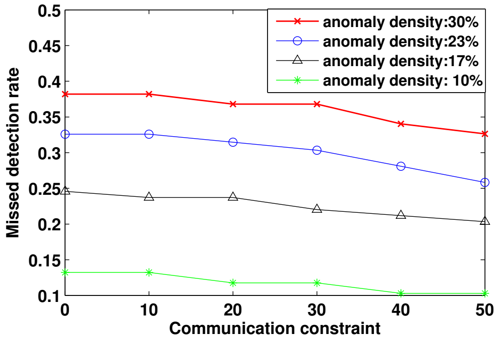
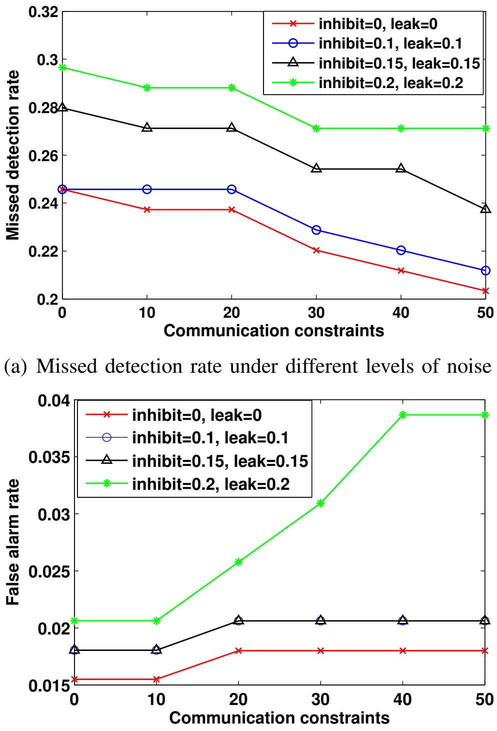


Fig. 7. Missed detection rate for different anomaly densities

Since the inference accuracy is also affected by the noise level in the network, i.e., the leakage probability and the inhibition probability in the Noisy-OR model. We now inject different levels of noise into the same network. Fig. 8 compares the MDR and FAR when the inhibition and leakage probability varies. The anomaly density is around 17% for this evaluation. Generally, the performance degrades when more noise are injected, which is as expected. It is also found that for low noise level, e.g., $inhibit < 0.1, leak < 0.1$, the performance does not change much.



(a) Missed detection rate under different levels of noise

(b) False alarm rate under different levels of noise

Fig. 8. Performance under different levels of noise

VI. Conclusions

In this paper, we present a trust-assisted framework for anomaly detection and localization in resource constrained wireless sensor networks using end-to-end measurements. In contrast to most of the prior work that focus on detection and localization accuracy, our focus is on the tradeoff between inference accuracy and the resource consumption in sensor networks. Especially, we are interested in the case that there is a strict constraint on the communication bandwidth for doing anomaly detection and localization. We proposed a hierarchical network structure and exploited network heterogeneity to improve energy and bandwidth efficiency. We designed an efficient two-phase probing scheme that utilize link trust information to achieve a good tradeoff between inference accuracy and probing overhead. Simulation results demonstrate the efficiency and effectiveness of our algorithms. In future work, we plan to implement the proposed algorithms in a real sensor network testbed to further evaluate performances.

References

[1] N. Ramanahan, K. Chang, R. Kapur, L. Girod, E. Kohler, and D. Estrin, “Sympathy for the sensor network debugger,” in *Proceedings of the 3rd Embedded Networked Sensor Systems (SenSys)*, 2005.

[2] S. Marti, T. J. Giuli, K. Lai, and M. Baker, “Mitigating routing misbehavior in mobile ad hoc networks,” in *Proceedings of the 6th International Conference on Mobile Computing and Networking (Mobicom)*, 2000.

[3] N. Duffield, “Network tomography of binary network performance characteristics,” *IEEE Transactions on Information Theory*, 2006.

[4] H. Nguyen and P. Thiran, “The boolean solution to the congested ip link location problem: Theory and practise,” in *Proceedings of IEEE INFOCOM*, 2007.

[5] Y. Gu, G. Jiang, V. Singh, and Y. Zhang, “Optimal probing for unicast network delay tomography,” in *Proceedings of IEEE INFOCOM*, 2010.

[6] H. X. Nguyen and P. Thiran, “Using end-to-end data to infer lossy links in sensor networks,” in *Proceedings of IEEE INFOCOM*, 2006.

[7] B. Wang, W. Wei, W. Zeng, and K. R. Pattipati, “Fault localization using passive end-to-end measurement and sequential testing for wireless sensor networks,” in *Proceedings of IEEE SECON*, 2009.

[8] A. Josang, “A logic for uncertain probabilites,” *International Journal of Uncertainty, Fuziness and Knowledge-Based Systems*, 2001.

[9] A. Josang and R. Ismail, “The beta reputation system,” in *Proceedings of the 15th Bled Electronic Commerce Conference*, 2002.

[10] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina, “The eigentrust algorithm for reputation management in p2p networks,” in *Proceedings of the 12th International World Wide Web Conference*, 2003.

[11] S. Tanachaiwiwat, P. Dave, R. Bhindwale, and A. Helmy, “Secure locations: routing on trust and isolating compromised sensors in location-aware sensor networks,” in *Proceedings of SenSys’03*, 2003.

[12] M. J. Probst and S. K. Kasera, “Statistical trust establishment in wireless sensor networks,” in *the 13th International Conference on Parallel and Distributed Systems*, 2007.

[13] M. Yarvis, N. Kushalnagar, H. Singh, A. Rangarajan, Y. Liu, and S. Singh, “Exploiting heterogeneity in sensor networks,” in *Proceedings of IEEE INFOCOM*, 2005.

[14] A. Zheng, I. Rish, and A. Beygelzimer, “Efficient test selection in active diagnosis via entropy approximation,” in *Proceedings of UAI-05*, 2005.

[15] S. Khuller, A. Moss, and J. Naor, “The budgeted maximum coverage problem,” *Inf. Process. Lett.*, 1999.

[16] M. Bawa, H. Garcia-molina, A.Gionis, and R. Motwani, “Estimating aggregates on a peer-to-peer network,” Computer Science Dept., Stanford University, Tech. Rep., 2003.

[17] S. Zheng, T. Jiang, and J. S. Baras, “Performance comparison of two sequential change detection algorithms on detection of in-band wormholes,” in *Proceedings of Information Science and Systems*, 2009.

[18] I. Rish, M. Brodie, S. Ma, N. Odintsova, A. Beygelzimer, G. Grabarnik, and K. Hernandez, “Adaptive diagnosis in distributed systems,” *IEEE Transactions on Neural Networks*, 2005.

[19] D. Golovin and A. Krause, “Adaptive submodularity: A new approach to active learning and stochastic optimization,” in *Proceedings of International Conference on Learning Thoery (COLT)*, 2010.

[20] L. Cheng, X. Qiu, L. Meng, Y. Qiao, and R. Boutaba, “Efficient active probing for fault diagnosis in large scale and noisy networks,” in *Proceedings of INFOCOM*, 2010.

[21] C. Bishop, *Pattern Recognition and machine learning*. Springer, 2006.

[22] J. Schiff, D. Antonelli, A. G. Dimakis, D. Chu, and M. J. Wainwright, “Robust message-passing for statistical inference in sensor networks,” in *Proceedings of the International Symposium on Information Processing for Sensor Networks (IPSN)*, 2007.

[23] A. Nath and P. Domingos, “Efficient belief propagation for utility maximization and repeated inference,” in *Proceedings of the 24th AAAI Conference on Artificial Intelligence*, 2010.

[24] K. P. Murphy, “The Bayes Net Toolbox for Matlab,” *Computing Science and Statistics*, 2001.