ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 3, March 2017

IPath: Path Inference in Wireless Sensor Networks

R Sai Vijay [1], Y Madhavi[2], Ch Venkateswarlu[3]

MCA Student, Dept. of MCA, Narayana Engineering College, Nellore, A.P, India[1]

MCA Student, Dept. of MCA, Narayana Engineering College, Nellore, A.P, India[2]

Associate Professor, Dept. of MCA, Narayana Engineering College, Nellore, A.P, India[3]

ABSTRACT: Recent wireless sensor networks (WSNs) are be-coming increasingly complex with the growing network scale and the dynamic nature of wireless communications. Many measurement and diagnostic approaches depend on per-packet routing paths for accurate and fine-grained analysis of the complex net-work behaviours. In this paper, we propose iPath, a novel path inference approach to reconstructing the per-packet routing paths in dynamic and large-scale networks. The basic idea of iPath is to exploit high path similarity to iteratively infer long paths from short ones. IPath starts with an initial known set of paths and performs path inference iteratively. iPath includes a novel design of a lightweight hash function for verification of the inferred paths. In order to further improve the inference capability as well as the execution efficiency, iPath includes a fast bootstrapping algorithm to reconstruct the initial set of paths. We also implement iPath and evaluate its performance using traces from large- scale WSN deployments as well as extensive simulations. Results show that iPath achieves much higher reconstruction ratios under different network settings compared to other state-of-the-art approaches.

KEYWORDS: Measurement, path reconstruction, wireless sensor networks.

I. INTRODUCTION

WIRELESS sensor networks (WSNs) can be applied in many application scenarios, e.g., structural protection [1], ecosystem management [2], and urban CO_2 monitoring [3]. In a typical WSN, a number of self-organized sensor nodes report the sensing data periodically to a central sink via multihop wireless. Recent years have witnessed a rapid growth of sensor net-work scale. Some sensor networks include hundreds even thou-sands of sensor nodes [2], [3]. These networks often employ dynamic routing protocols [4]–[6] to achieve fast adaptation to the dynamic wireless channel conditions. The growing network scale and the dynamic nature of wireless channel make WSNs become increasingly complex and hard to manage.

Reconstructing the routing path of each received packet at the sink side is an effective way to understand the network's complex internal behaviours [7], [8]. With the routing path of each packet, many measurement and diagnostic approaches [9]–[13] are able to conduct effective management and protocol optimizations for deployed WSNs consisting of a large number of unattended sensor nodes. For example, PAD [10] depends on the routing path information to build a Bayesian network for infer-ring the root causes of abnormal phenomena. Path information is also important for a network manager to effectively manage a sensor network. For example, given the per-packet path information, a network manager can easily find out the nodes with a lot of packets forwarded by them, i.e., network hop spots. Then, the manager can take actions to deal with that problem, such as deploying more nodes to that area and modifying the routing layer protocols. Furthermore, per-packet path information is essential to monitor the fine-grained per-link metrics. For example, most existing delay and loss measurement approaches [9], [14] assume that the routing topology is given as *a priori*. The time-varying routing topology can be effectively obtained by per-packet routing path, significantly improving the values of existing WSN delay and loss tomography approaches.

DOI: 10.15680/IJIRCCE.2017. 0503089

A straightforward approach is to attach the entire routing path in each packet. The problem of this approach is that its message overhead can be large for packets with long routing paths. Considering the limited communication resources of WSNs, this approach is usually not desirable in practice.

In this paper, we propose iPath, a novel path inference approach to reconstruct routing paths at the sink side. Based on a real-world complex urban sensing network with all node generating local packets, we find a key observation: It is highly probable that a packet from node i and *one of* the packets from i's Parent will follow the same path starting from i's parent toward the sink. We refer to this observation as *high path similarity*. Fig. 1 shows a simple example where S is the sink node. a_1 denotes a packet from A, and $b_1 b_2 b_3$ denotes packets from B (A's parent). *High path similarity* states that it is highly probable that a_1 will follow the same path (i.e., $\mathtt{Path}(a_1)\_A$, which means the sub path by removing node A from $\mathtt{Path}(a_1)$) as one of B's packet, say b_1, i.e., $\mathtt{Path}(a_1)=(A\,\mathtt{Path}(b_1))$.

The contributions of this work are the following.

- We observe high path similarity in a real-world sensor net-work. Based on this observation, we propose an iterative boosting algorithm for efficient path inference. We propose a lightweight hash function for efficient verification within iPath. We further propose a fast bootstrapping algorithm to improve the inference capability as well as its execution efficiency.

- We propose an analytical model to calculate the successful reconstruction probability in various network conditions such as network scale, routing dynamics, packet losses, and node density.

- We implement iPath and evaluate its performance using traces from large-scale WSN deployments as well as extensive simulations. iPath achieves higher reconstruction ratio under different network settings compared to states of the art.

The rest of this paper is organized as follows. Section II discusses the related works. Section III gives the results of a measurement study on two deployed networks. Section IV de-scribes the network model and assumptions made in this paper. Section V describes the design of iPath. Section VI formally analyses the reconstruction performance of iPath and two related works. Section VII evaluates iPath's performance compared to existing works in a trace-driven study. Section VIII reveals more system insights by extensive simulations, and Section IX concludes this paper.

II. EXISTING SYSTEM

- With the routing path of each packet, many measurement and diagnostic approaches are able to conduct effective management and protocol optimizations for deployed WSNs consisting of a large number of unattended sensor nodes. For example, PAD depends on the routing path information to build a Bayesian network for inferring the root causes of abnormal phenomena.
- Path information is also important for a network manager to effectively manage a sensor network. For example, given the per-packet path information, a network manager can easily find out the nodes with a lot of packets forwarded by them, i.e., network hop spots. Then, the manager can take actions to deal with that problem, such as deploying more nodes to that area and modifying the routing layer protocols.
- Furthermore, per-packet path information is essential to monitor the fine-grained per-link metrics. For example, most existing delay and loss measurement approaches assume that the routing topology is given as *a priori*.

The time-varying routing topology can be effectively obtained by per-packet routing path, significantly improving the values of existing WSN delay and loss tomography approaches.

III. PROPOSED SYSTEM

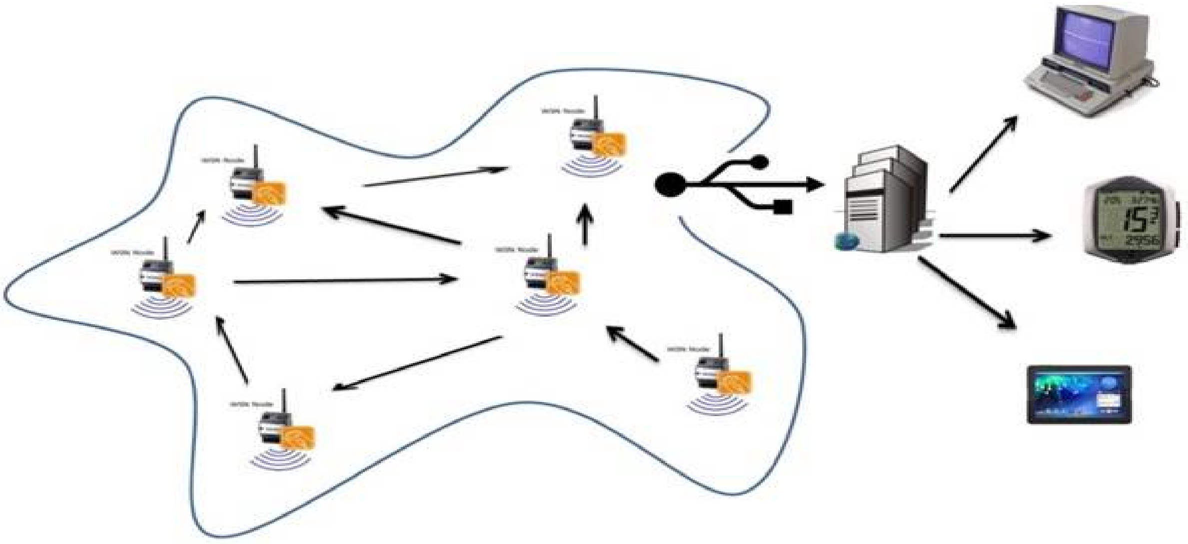
- In this paper, we propose iPath, a novel path inference approach to reconstruct routing paths at the sink side. Based on a real-world complex urban sensing network with all node generating local packets, we find a key observation: It is highly probable that a packet from node and *one of* the packets from‘s parent will follow the same path starting from ‘s parent toward the sink. We refer to this observation as *high path similarity*.
- The basic idea of iPath is to exploit high path similarity to iteratively infer long paths from short ones. iPath starts with a known set of paths (e.g., the one-hop paths are already known) and performs path inference iteratively. During each iteration, it tries to infer paths one hop longer until no paths can be inferred.
- In order to ensure correct inference, iPath needs to verify whether a short path can be used for inferring a long path. For this purpose, iPath includes a novel design of a lightweight hash function. Each data packet attaches a hash value that is updated hop by hop. This *recorded hash value* is compared against the *calculated hash value* of an inferred path. If these two values match, the path is correctly inferred with a very high probability.
- In order to further improve the inference capability as well as its execution efficiency, iPath includes a fast bootstrapping algorithm to reconstruct a known set of paths.

ADVANTAGES OF PROPOSED SYSTEM:

- We observe high path similarity in a real-world sensor network.
- It’s an iterative boosting algorithm for efficient path inference.
- It’s a lightweight hash function for efficient verification within iPath.
- The proposed systems further propose a fast bootstrapping algorithm to improve the inference capability as well as its execution efficiency.
  iPath achieves higher reconstruction ratio under different network settings compared to states of the art.

IV. SYSTEM MODELS

In order to quantify the path similarity in real-world deployment, we conduct a measurement study on two deployed networks—CitySee [3] and GreenOrbs [2]. The CitySee project is deployed in an urban area for measuring carbon emission. All nodes are organized in four subnets. Each subnet has one sink node, and sink nodes communicate to the base station through 802.11 wireless links. We collect traces from one sink of a subnet with 297 nodes. The GreenOrbs project includes 383 nodes in an forest area for measuring the carbon absorbance.



The wakeup interval of the low power setting is 512 ms. each node reports data packets to a sink with a period of 10 min. Each data packet carries the routing path information directly for offline analysis. We first look at the routing dynamics of the networks. We measure a quantity λ that is defined to be the average number of periods (i.e., local packets) between two parent changes by a node. It is simply the inverse of the number of parent changes per period at a node. A smaller λ means more frequent parent changes. Fig. 2(a) and (b) shows the cumulative distribution function (CDF) of λ for all nodes in the two networks. We can see that these two networks have different degrees of routing dynamics. On average, there is a parent change every 46.9 periods in CitySee and 89.1 periods in GreenOrbs. As a comparison, the MNT paper [8] reports a parent change every $88.2 \sim 793.3$ periods of the networks tested, which have less frequent parent changes. We see that CitySee and GreenOrbs have high routing dynamics, making per-packet path inference necessary for reasoning about complex routing behaviours.

V. IMPLEMENTATION

5.1. MODULE DESCRIPTION:

5.1.1 Network Model:

- In this module we design the network model. We assume a multi hop WSN with a number of sensor nodes.

5.1.2 Iterative Boosting

- IPath reconstructs unknown long paths from known short paths iteratively. By comparing the recorded hash value and the calculated hash value, the sink can verify whether a long path and a short path share the same path after the short path's original node.

- When the sink finds a match, the long path can be reconstructed by combining its original node and the short path.

5.1.3. PSP-Hashing

- As mentioned in the iterative boosting algorithm, the PSPHashing (i.e., path similarity preserving) plays a key role to make the sink be able to verify whether a short path is similar with another long path.

- There are three requirements of the hash function. The hash function should be lightweight and efficient enough since it needs to be run on resource-constrained sensor nodes.

- The hash function should be order-sensitive. That is, hash (A, B) and hash (B, A) should not be the same. The collision probability should be sufficiently low to increase the reconstruction accuracy.

- Traditional hash functions like SHA-1 are order-sensitive. However, they are not desirable due to their high computational and memory overhead.

- For example, an implementation of SHA-1 on a typical sensor node TelosB takes more than 4 kB program flash and longer than 5 ms to hash 20 B of data. Note that this memory overhead is about 10% of the total program flash of a TelosB node, and 5 ms computational overhead nearly doubles the forwarding delay in a typical routing protocol.

- In order to design an efficient and lightweight hash function, efficient operations, such as bitwise XOR operation, are preferred. Since XOR operation is not order-sensitive, the order information should be explicitly hashed into the hash value. We propose PSP-Hashing, a lightweight path similarity preserving hash function to hash the routing path of each packet. PSP-Hashing takes a sequence of node ids as input and outputs a hash value. Each node along the routing path calculates a hash value by three pieces of data. One is the hash value in the packet that is the hash result of the subpath before the current node. The other two are the current node id and the previous node id. The previous node id in the routing path can be easily obtained from the packet header. Shows this chained hash function along the routing path.

5.1.4 Fast Bootstrapping

- The iterative boosting algorithm needs an initial set of reconstructed paths.

- In addition to the one/two-hop paths, the fast bootstrapping algorithm further provides more initial reconstructed paths for the iterative boosting algorithm.

- These initial reconstructed paths reduce the number of iterations needed and speed up the iterative boosting algorithm.

- The fast bootstrapping algorithm needs two additional data fields in each packet, parent change counter and global packet generation time.

The parent change counter records the accumulated number of parent changes, and the global packet generation time can be estimated by attaching an accumulated delay in each packet.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

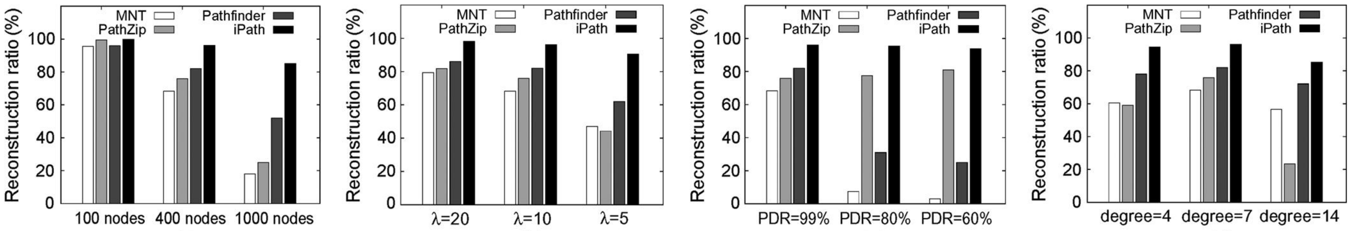
International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 3, March 2017

VI. RESULTS AND EXPERIMENTAL STUDIES



(a) (b) (c) (d)

In order to quantify the reconstruction performance of iPath and two related approaches, we analyse these approaches by a novel analytical model. Here, the performance means the probability of a successful reconstruction, which is the most important metric. We use the following definitions for analysis.

- Local packet generation period t. iPath does not require all nodes have the same local packet generation period. In order to simplify the presentation, we assume all nodes have the same packet generation period in this analysis section.

- Routing dynamics δ, which is the number of parent changes in a single period t. On average, there is one parent change every $\lambda = 1/\delta$ local packet. We call these λ consecutive periods as one *cycle* for analysis.

- Packet delivery ratio PDR of packet k. It can be calculated as the product of the packet reception ratios (PRR) along
  the routing path packet _ '—∈—(–) ———.
- The average node degree d.

- As mentioned in the fast bootstrapping algorithm, a *stable period* of a node is a period t in which the node does not change parent.

In this section, we conduct extensive simulations in TOSSIM [28], a standard simulator for TinyOS programs, to reveal more system insights. Specifically, we evaluate the reconstruction performance of iPath and three related works in networks with different configuration settings such as path length, routing dynamic, packet delivery ratio, and degree. A number of networks with up to 1000 nodes are used in the simulations. We also evaluate the impact of length of the hash value, which is the key parameter in the design of iPath. At the end of this section, we will show a visualization of the reconstruction process in a network with 400 nodes.

VII. CONCLUSION AND FUTURE WORK

In this paper, we propose iPath, a novel path inference approach to reconstructing the routing path for each received packet. iPath exploits the path similarity and uses the iterative boosting algorithm to reconstruct the routing path effectively. Furthermore, the fast bootstrapping algorithm provides an initial set of paths for the iterative algorithm. We formally analyse the reconstruction performance of iPath as well as two related approaches. The analysis results show that iPath achieves higher reconstruction ratio when the network setting varies. We also implement iPath and evaluate its performance by a trace-driven study and extensive simulations. Compared to states of the art, iPath achieves much higher reconstruction ratio under different network settings.

REFERENCES

[1] M. Ceriotti *et al.*, “Monitoring heritage buildings with wireless sensor networks: The Torre Aquila deployment,” in *Proc. IPSN*, 2009, pp. 277–288.
[2] L. Mo *et al.*, “Canopy closure estimates with GreenOrbs: Sustainable sensing in the forest,” in *Proc. SenSys*, 2009, pp. 99–112.
[3] X. Mao *et al.*, “CitySee: Urban CO2 monitoring with sensors,” in *Proc. IEEE INFOCOM*, 2012, pp. 1611–1619.
[4] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, and P. Levis, “Collec-tion tree protocol,” in *Proc. SenSys*, 2009, pp. 1–14.
[5] D. S. J. D. Couto, D. Aguayo, J. Bicket, and R. Morris, “A high-throughput path metric for multi-hop wireless routing,” in *Proc. Mo-biCom*, 2003, pp. 134–146.
[6] Z. Li, M. Li, J. Wang, and Z. Cao, “Ubiquitous data collection for mobile users in wireless sensor networks,” in *Proc. IEEE INFOCOM*, 2011, pp. 2246–2254.
[7] X. Lu, D. Dong, Y. Liu, X. Liao, and L. Shanshan, “PathZip: Packet path tracing in wireless sensor networks,” in *Proc. IEEE MASS*, 2012, pp. 380–388.
[8] M. Keller, J. Beutel, and L. Thiele, “How was your journey? Uncov-ering routing dynamics in deployed sensor networks with multi-hop network tomography,” in *Proc. SenSys*, 2012, pp. 15–28.
[9] Y. Yang, Y. Xu, X. Li, and C. Chen, “A loss inference algorithm for wireless sensor networks to improve data reliability of digital ecosys-tems.,” *IEEE Trans. Ind. Electron.*, vol. 58, no. 6, pp. 2126–2137, Jun. 2011.
[10] Y. Liu, K. Liu, and M. Li, “Passive diagnosis for wireless sensor net-works,” *IEEE/ACM Trans. Netw.*, vol. 18, no. 4, pp. 1132–1144, Aug. 2010.
[11] W. Dong, Y. Liu, Y. He, T. Zhu, and C. Chen, “Measurement and anal-ysis on the packet delivery performance in a large-scale sensor net-work,” *IEEE/ACM Trans. Netw.*, 2013, to be published.
[12] J. Wang, W. Dong, Z. Cao, and Y. Liu, “On the delay performance analysis in a large-scale wireless sensor network,” in *Proc. IEEE RTSS*, 2012, pp. 305–314.
[13] Y. Liang and R. Liu, “Routing topology inference for wireless sensor networks,” *Comput. Commun. Rev.*, vol. 43, no. 2, pp. 21–28, 2013.
[14] Y. Gao *et al.*, “Domo: Passive per-packet delay tomography in wireless ad-hoc networks,” in *Proc. IEEE ICDCS*, 2014, pp. 419–428.
[15] M. Lee, S. Goldberg, R. R. Kompella, and G. Varghese, “Fine-grained latency and loss measurements in the presence of reordering,” in *Proc. ACM SIGMETRICS*, 2011, pp. 329–340.
[16] Y. Shavitt and U. Weinsberg, “Quantifying the importance of vantage points distribution in internet topology measurements,” in *Proc. IEEE INFOCOM*, 2009, pp. 792–800.

BIOGRAPHY

Mr. CH. Venkateswarlu is an Assistant Professor of MCA at Narayana Engineering College, Nellore, A.P, India. He guided many projects for PG students. His research interests include Data mining, Computer Networks and Software Engineering etc.

Mr. R. Sai Vijay is a student pursuing MCA at Narayana Engineering College, Nellore, A.P, India. During my final semester project work we prepared this paper for publishing it in an international journal.

Ms. Y. Madhavi is a student pursuing MCA at Narayana Engineering College, Nellore, A.P, India. During my final semester project work we prepared this paper for publishing it in an international journal.