PATH : PATH INFERENCE IN WIRELESS SENSOR NETWORKS

[1]N. Umamaheswari, M.Phil Scholar, Bharathiyar Arts And Science College For Women, Deviyakurichi, Thalaivasal, Salem,

[1]R.Radha, Asst. Professor, Bharathiyar Arts And Science College For Women, Deviyakurichi, Thalaivasal, Salem.

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.

1. INTRODUCTION

Wireless sensor networks (WSNs) can be applied in many application scenarios, e.g., structural protection [1], ecosystem management [2], and urban CO 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 infor- mation, 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. 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.

2. RELATED WORK

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. 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.

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 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.

3. PROPOSED SYSTEM

RTR algorithm with multiple measurement metrics helps reduce the potential size of the solution set significantly. While the theoretical probability analysis is still an open question, we empirically observed that the probability of having more-than-one potential solutions should be extremely small using the RTR with both SUMm and XOR measure- ment metrics. Therefore, we also develop a Fast Routing Topology Recovery (FRTR) algorithm which only provides the first solution candidate found and then stop the further searching. With proper edge labeling value function and path measurement metrics, FRTR would be able to obtain the unique correct recovery with very high probability. The merit of FRTR algorithm is that it is significantly faster than RTR algorithm since RTR algorithm may waste re- sources trying to search either non-existent or duplicated solution candidates in its effort to obtain the complete set of potential solution candidates.

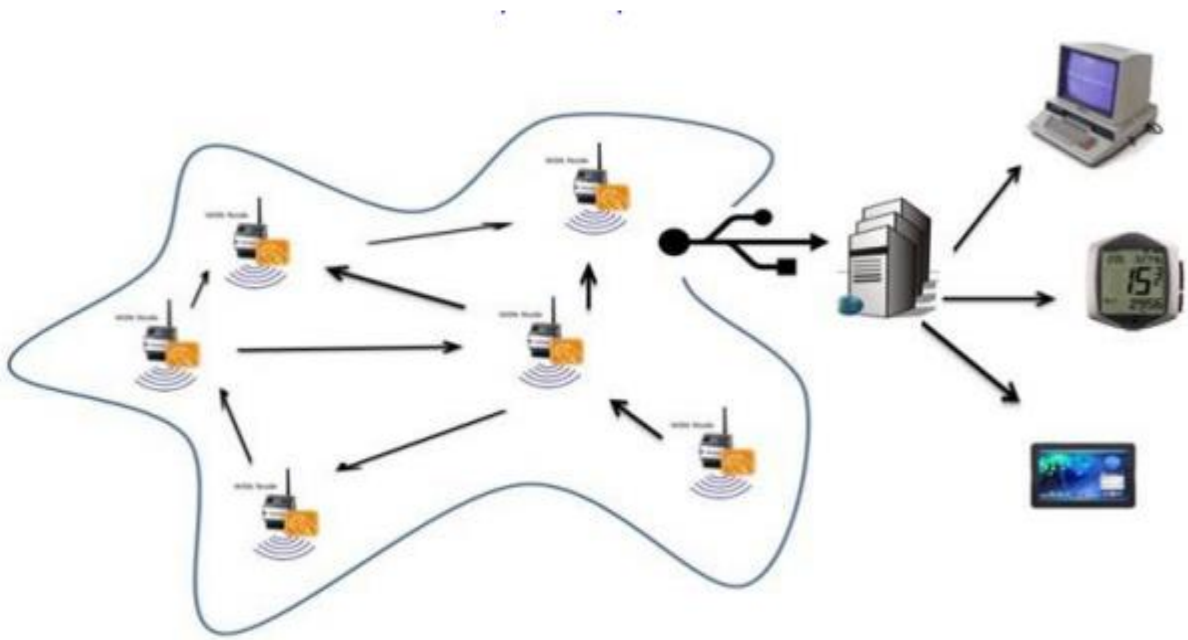

Fig.1.Routing system

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. 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. 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.

4. ANALYSIS

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. 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 . 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. As mentioned in the fast bootstrapping algorithm, a stable period of a node is a period in which the node does

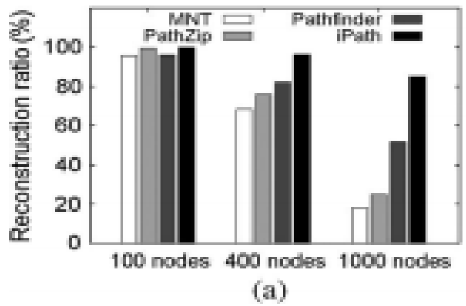

Fig.2.Analysis 1

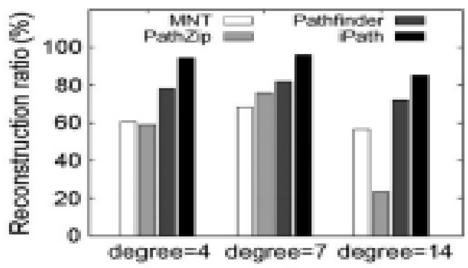

Fig.2.Analysis 2

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.

CONCLUSION

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.