ISSN 2278-3091
Volume 8, No.1.4, 2019
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse3381.42019.pdf
https://doi.org/10.30534/ijatcse/2019/3381.42019

Energy Efficient Path Reconstruction in Wireless Sensor Network using iPath

Sazlinah Hasan[1], Wamidh Jwdat Abd[2], Ahmad Alauddin Ariffin[3]
Faculty of Computer Science and Information Technology,
Universiti Putra Malaysia (UPM), Malaysia,
[1]sazlinah@upm.edu.my, [2]wameidh6000@googlemail.com, [3]alauddin@upm.edu.my

ABSTRACT

Wireless sensor networks operate through commonly self-organized sensor nodes to transfer data in a multi-hop approach to a central sink. In order to support fine-grained diagnostic analysis and optimize the performance level of the networks, the reconstruction of per-packet routing path is essential. However, in large-scale networks, the performance levels of the current path reconstruction method decline rapidly, with loss of links. An efficient approach to fully comprehend the complex internal behavior of network is through the reconstruction of the routing path of each received packet at the sink side. This paper discussed the added of energy efficiency parameter to enhance the inference Path (iPath). Thus, the iPath by added the energy efficiency enables the reconstruction of the per-packet routing paths of large-scale networks, by providing a stable and efficient route to exchange messages between source and destination in a timely manner. This work uses iterative boosting algorithm to find an alternative path with less distance and energy consumption. To achieve energy efficiency, it compresses the packet information by using GZIP tools in JAVA. Energy efficient iPath (E-iPath) is evaluated with several variations of nodes in WSN deployments as well as large-scale simulations. The findings demonstrate that E-iPath surpasses other current approaches such as EEPMM. E-iPath has accomplished low transmission overhead which it has reduced 13% of the energy consumption and has gained significant reconstruction ratio compared with iPath.

Key words: Wireless Sensor Networks, Path Reconstruction, iPath, Energy Efficiency.

1. INTRODUCTION

A wireless sensor networks (WSN) refers to a group of sensor nodes possessing limited supply of power as well as limited computational abilities. Furthermore, limited communication ranges and large densities of sensor nodes lead to packet forwarding in sensor networks, typically via multi-hop data transmission. Consequently, WSN routing has recently received much attention from the research community within the 21st Century [1]. Routing within WSN determines the path, starting from the source of data transmission and finishing with the sink. In the case of multi-hop networks, source nodes do not typically connect sink nodes immediately. Instead, intermediate sensor networks (SNs) transmit the data via packets. Routing paths can provide a solution to the problem, thus leading to reconstruction of routing paths as an important field of study, with regards to routing algorithms. Many sensors are required to transmit messages effectively from the source to the sink. To achieve effective transmission and maximize network lifetime, message delivery assurance for the routing protocol is crucial. This denotes that the protocol should search for routing paths even if the route in question is not present.

With regards to network performance, routing paths for packets serve as important guide for comprehension. The metrology of routing dynamics requires per-packet routing paths to check and determine the frequency of path alterations occurring. Furthermore, the metrology of packet losses requires path information to extrapolate the location of packet loss. To add to this, network diagnostic tools require information for the routing path to determine the main cause of network disturbances. Information on the path of packets enable enhancement of detection and flow control.

While reconstructing routing paths for the received packet constitutes an efficient method for comprehending the complex internal network behavior, the routing path of each packet requires diagnostic tools, which can execute effective management and protocol optimizations for WSNs in deployment. These WSNs are also characterized by a high number of unsupervised sensor nodes.

Two currently available methods for path reconstruction in WSN's include multi-hop network tomography (MNT) [2] and PathZip [3]. First method MNT is for each individual packet, a reliable packet set is computed to attempt reconstructing the path. Consequently, no packet loss or packet reordering occurs. Furthermore, the set entails that the routing path to be carefully reconstructed with regards to the computer utilized. The second method is PathZip that attempts to reconstruct the packet path by assigning compressed path information, as a hash value to the individual packet. An effective math is sought for the path by employing a comprehensive search across neighboring nodes by the sink. The two methods provide an effective means of reconstructing paths in events of low loss rates and non-severe routing dynamics.

A direct method, which solves the problem, entails sticking the whole routing path to each individual packet. An issue arises with this method, however, in that the message overhead may be too huge for packets with longer routing paths. Because WSNs tend to be deployed with limited resources, the method is impractical in many cases.

This paper discusses on E-iPath, which is an energy efficient inference method that enables the reconstruction of the per-packet routing paths of large-scale networks. This approach constantly searches for longer paths rather than shorter ones. An iterative smoothing algorithm is also employed to find the longest unknown path based on the known shortest path.

2. WIRELESS SENSOR NETWORK (WSN)

This section discusses the literature survey of WSN path reconstruction and energy consumption. It also discusses the survey of existing path reconstruction techniques. Another crucial part that needs to be identified before progressing to the proposed work is the performance's time factor and followed by suitable algorithms and techniques that fit the purpose of developing the proposed E-iPath.

2.1 WSN Path Reconstruction

A per-packets routing way fills in as the meta-data for interpreting details of WSNs practices in numerous networks upkeep and diagnostic circumstances, such as packet loss holes or detections on wormholes, routing dynamics, even per hop per-packet transmission delay or end-to-end packet transmission delay, and network diagnosis [4].

Reproducing per-packets routing way data, in any case, has been known as non-inconsequential. The trouble in acquiring per-bundle routing way is due to two reasons, which is firstly, sensor systems are normally self-sorted out and progressively evolving. There is typically no earlier learning about the hidden directing topology. Secondly, it is exorbitant to straightforwardly join way data in every packet since the overhead increments as the system scales [5]. There have been numerous endeavors made to address the per-packets recreation way issue in WSNs. This part clarifies some related work on way reproduction. [6] Proposed Domo, a Delay tomography method for accomplishing lightweight and precise per-hop per-bundle defer remaking in wireless ad-hoc networks. Domo comprises of two sections; in the node part, Domo appends a little overhead to every packet for building imperatives for the streamlining issues. Meanwhile, in the PC part, Domo utilizes numerous procedures to productively tackle the enhancement issues.

Ma et al. [7] have explored the issue of distinguishing singular connection measurements in correspondence networks through end-to-end way estimations, with the assumption that connection measurements are a supplementary element and stable. Liu et al. [8] proposed PAD, a probabilistic analysis method for deriving the underlying drivers of irregular marvels. PAD utilizes a packets technique for effectively building, and powerfully keeping up the inference scheme. It expects a generally hard framework, utilizing every bundle to move through one bounce of a way. Another study has been done by Keller et al [2], presenting multi-jump network tomography (MNT), a new, non-meddlesome method for remaking the travelling way, the per-hop landing request, and the per-jump entry times of individual bundles at runtime. This method presents a lower level of message overhead, and versatility with comprehensive scale systems. However, MNT is not able to operate efficiently with systems that are in high routing elements, and great misfortune rate.

Another study, PathZip [3], which works when every sensor node performing lightweight hash-depend on calculations to latently mark each packet sent. In the meantime, the sink extracted the label data in order to use the pre-information on the network to calculate the full parcel way. Both topology-mindful and geometry-associate procedures are used by PathZip, keeping in mind the end goals are to misuse distinctive system learning and decrease the calculation and capacity overhead significantly. However, the major concern of this method is that when the system scales up, the pursuit space generates rapidly. The computational overhead develops exponentially with expanding way length. At the end of the day, given a restricted calculation time for every parcel, the recreation proportion of per-bundle way will be low for substantial scale systems.

The end goal is to remake per-packet way in huge scale sensor systems with unassuming parcel conveyance proportion, generally high steering elements. Reference [5] present Pathfinder reproduction strategy that is proposed as utilizing fleeting connection among packet ways to proficiently pack the way data. Pathfinder derives packet ways through the compacted data, utilizing astute way hypothesis to remake the packet ways with high reproduction proportion.

In addition, another recent test-based system, FineComb utilizes deferral and misfortune geology method which focuses mainly to resolve the parcel reordering [9]. This study gathers the estimation for system execution through compressing the outline space of examining calculations. It is not a common practice for any test's utilizations in WSNs, due to the complexity of identifying the remote dynamic through several number of tests, and the subsequent examining will demonstrate a huge utilization of vitality. In addition, this study also aims to discover the issue of identifying the measurements of per-jump from the estimations of end-to-end way, with the presumption that connection measurements are complementing and stable elements. This study constructs a direct framework without any dynamic test utilization through end-to-end estimations from numerous inward screens. In order to build the straight framework, way data is utilized as the prior learning. This then becomes orthogonal to iPath [10], and new methods of estimation in WSNs could be identified through the process of unifying and strengthening them.

Besides that, another important indicative instrument is cushion, which acquires system topology by incorporating a bundle of stamping plan. This instrument assumes a system that is static and utilizes each bundle for transmitting on bounce of a way. When the system completes the course in a unique manner, it is almost impossible to accurately remake the frequently changing steering way. Firstly, MNT acquires from the got parcels at sink an arrangement of stable bundles, before remaking the got parcel way by utilizing the set of solid bundles. However, MNT can achieve high remaking precision and reproduction proportion when the system is entirely unique, together with a high proportion of bundle conveyance. Besides that, MNT does not work well when facing remote flow and parcel misfortune.

In every parcel, the steering way is hashed by Path Zip into an 8-B hash an incentive. When this happens, the sink performs an extensive search for a match over the neighboring hubs. The main drawback of implementing PathZip is that, when the system scales up, the pursuit space will be generating rapidly. All hubs in Pathfinder are assumed to produce neighborhood parcels, and with a usual inter-packet interim, namely IPI. The Pathfinder effectively packs the way date into each bundle through the transient connection with various parcel ways. From the packed date, Pathfinder can obtain parcel ways at the PC side. In contrast with PathZip, iPath provides a higher versatility by utilizing the high way similitude that exist among various parcels to gain a speedy deduction. Meanwhile, when compared with MNT, iPath provides a lower rigidity on necessities for way deduction. While an arrangement of sequential parcels from the same parent is demanded by MNT, iPath, on the other hand, only calls for one neighboring bundle that uses the same way for each bounce. In addition, while normal IPI is sufficient for Pathfinder, iPath on the other hand abuses the way comparability among various lengths ways to achieve better proportion and precision remaking.

PathZip [3], on the other hand, attaches in each packet a 64-bit hash value, which includes the path information in a compressed form, and it estimates the node's neighbor table availability. The PC-side algorithm carries out extensive search for an equal hash value over nearby nodes of all forwarders along the path, to reconstruct the routing path of every packet. With the expanding path length, the computational overhead increases rapidly. Thus, if each packet's computation time is limited for large scale networks, the per-packet path reconstruction ration will be low. MNT [2] carries out the routing path reconstruction by exploiting the packet header's existing information, namely the first-hop forwarder, by assuming that local traffic is generated by all nodes. MNT infers the passing node packets' routing paths by utilizing the parent information in local packets, which is derived from helper packets. While MNT can be scaled to large scale networks and produces low message overhead, it on the other hand, does not handle networks with high routing dynamics and high loss rate well. Thus, this work proposed Pathfinder to be utilized in the reconstruction of the per-packet path that is in large scale sensor networks, with high routing dynamics, but moderate packet delivery ratio. Pathfinder is an innovative approach to reconstruct path which competently compress the path information by utilizing the packet paths' temporal correlation. In addition, Pathfinder infers routing path for a forwarded packet by utilizing the forwarders' local packets. While MNT has a rigid set of specifications for the forwarder's helper packets, Pathfinder however, only requires the parent information found in a single packet in every forwarder. This is based on the information on the path difference of a single packet. Thus, this helps Pathfinder to attain a greater ratio in path reconstruction with higher routing changes and packet losses. To add to that, this approach also has a protection byte as one of its features, and it aids ensuring a high probability of the reconstructed path's accuracy.

Besides that, a limited number of bytes are also a feature of Pathfinder which enables it to get any changes in a routing path's part recorded. From an observation of most packets on a real-world sensor network, despite the network being large scaled (≤2 for 12-hop paths), the amount of the routing path's changed part continues to be low. Thus, to ensure efficient path reconstruction, it is necessary to record additional information of several hops. When it comes to large scale networks, Pathfinder has been found to be more computationally scalable to them, as it solely records any changed part with a way that is in line with the requirements of PC-side algorithm to conduct an efficient path reconstruction. This is in a stark contrast with PathZip, where the entire routing path will be compressed into a 64-bit hash value.

2.2 WSN Energy Consumption

Several efforts have been made in this area, including iPath, which demonstrated good results, but neglecting the energy efficiency. WSN sensors which are powered by small batteries along with techniques for power harvesting, are commonly set up in a non-accessible setting. The battery restricts any changes in the design and management to any gain additional efficiency, while also limiting the life of the sensor itself. Due to the energy supply limitations, several researchers have attempted to address this issue:

Kaur [11] have compared different geographic routing protocols in their study, namely, Energy-Aware geographic routing, Greedy perimeter stateless routing, and Location aided routing based on a number of performance metrics through NS2, a simulation tool. The perform metrics utilized are namely energy utilization, end to end delay and packet delivery ratio. The study has identified that with the dynamic changes in the topology and high mobility, the geographic routing is able to provide high delivery ratio of packets, enhanced network lifetime and energy efficiency, compared to other protocols.

A Density-based Energy-efficient Game-theoretic Routing Algorithm (DEGRA) has been proposed by Xu et al., which is an approach that operates through game theory and utility function, which is surrounded based on density of nodes, residual energy of its neighboring nodes, and average energy consumption [12]. Besides that, the cluster head selection is iteratively taken.

Reference [13] has shown that bandwidth and energy are basically various metrics. Furthermore, in routing protocols, the resource spending is partially deal with based on bandwidth centric study. Energy Entropy-Based Minimum Power Cost Multipath (EEPMM) have developed a protocol to determine the energy entropy-based multipath routing provisioning, which will assist in lowering packet transmission power consumption, balancing of loads, and extending network lifetime in MANETs [14].

Meanwhile, a Conditional Max Min Battery Cost Routing protocol (CMMBCR) has been proposed by Jean Louis et al [15]. CMMBCR attempts to mediate between the Max Min Battery Cost Routing (MMBCR) and the Minimum total transmission power routing

3. METHODOLOGY

This section introduces an E-iPath that used algorithms for iPath [10]. This algorithm deal with recreating the per-packet directions ways as a (routing paths) in powerful and expansive scale networks with energy proficient. The E-iPath iteratively find long paths from short ones. This research uses iterative smoothing algorithm to find unknown longest path based on known shortest path. To achieve energy efficiency, it compresses the packet information with high security.

The existing path reconstruction algorithm has many disadvantages such as message overhead, packet delay occurred in data traffic and consumes more energy. To tackle existing problems, this study proposed E-iPath which is an inference method to deal with recreating the per-packet directions ways as routing paths that is performed in expansive and dynamic scale network with high energy efficiency. The fundamental idea behind this study is to constantly infer long paths from short ones by exploiting the similarity of high path. The process is first initiated with the primary known set of paths, which is one hop path, before iteratively executing path inference. By doing so, the current study avoids data traffic based on path reconstruction while reducing energy consumption.

3.1 Evaluation Metrics

Evaluation metrics were utilized to assess and test the quality and operation of the wireless sensor networks. They make use of network aimed functionality and high-level criteria, as well as long-running period. These metrics must be met at a level in order to make sure that the benefits of the wireless sensor networks are achieved compared to other technologies. The evaluation metrics also have a key variable that needs to be monitored and recorded in order to ensure its validity.

3.2 Energy Consumption

In WSN, Energy Consumption is one of the important metrics. While each node is receiving and transmitting energy, consumption is computed [16]. A node which is in idle, does not consume any energy. 0.6 J and 0.3 J, respectively, are the fixed values of the powers for transmission and receiving [17]. When a node sends Packet (P) to next node, the energy capacity of the node will be reduced by following equation (1)

$$EC = Er + (Dist * PS * Et) \qquad (1)$$

In (1), *EC* is the Energy Consumption of a node in J; *Dist* is the distance between a node to next node in meter, and *PS* is the Packet Size in Kb. In this way, a node's energy consumption is calculated [17]. *Et*: the energy spent by a sensor node in transmit. *Er*: the energy spent by a sensor node in receiving.

3.3 Reconstruction Ratio

The probability of a successful reconstruction of the iPath algorithm is the product of the ratios of stable periods on all nodes as stated at (3). A stable period of a node is a period *t* in which the node does not change parent. The ratio of stable period without packet loss can be obtained as (2), where λ is the timer difference of a packet between two nodes Δt.

$$\textit{Ratio of Stable Period without packet loss } \Delta t = (\lambda - 1) / \lambda. \qquad (2)$$

$$\textit{Reconstruction} = \prod_{j \in \textit{path length nodes}} \textit{Ratio of stable period} \qquad (3)$$

Network Scale means number of nodes is available in wireless sensor network. Furthermore, network scale affects the path length, because, small-scale network achieves good performance, but large-scale network is often unable to do a path reconstruction with the higher path length.

3.4 Algorithm Strategy

In this work two algorithms were used, first one is the iterative boosting algorithm which utilizes short paths to iteratively reconstruct long path. Second, it used the fast bootstrapping algorithm which utilizes to speed up the iterative boosting algorithm.

iPath features an iterative boosting algorithm as in Algorithm 1, which utilizes short paths in order to iteratively reconstruct long paths, based on the similarity of path. PSP-Hashing supplies a path similarity while still preserving hash functions, which enables the iterative boosting algorithm in verifying in high precision on the similarity of two paths.

Algorithm 1: The iterative boosting algorithm [10]

Input: An initial set of packets P_{init} whose paths have been reconstructed and a set of other packets P_x
Output: The routing paths of packets

```
Procedure ITERATIVE-BOOSTING
    $P_n \leftarrow P_{\text{init}}$
    while $P_n \neq \{\}$ do
        $P_{nn} \leftarrow \{\}$
        for each packet $k$ in $P_n$ do
            for each packet $i$ in $P_x$ do
                $res =$ RECOVER($k$; $i$)
                if $res \equiv True$ then
                    $P_{nn} \leftarrow P_{nn} \cup i$
                    $P_x \leftarrow P_x - i$
        $P_n \leftarrow P_{nn}$
procedure RECOVER$(k, i)$
    if $len(i) - len(k) \notin \{1, 2\}$ then
        return False
    if $len(i) - len(k) \equiv 2$
        if $hash(o(i), p(i), path(k)) \equiv h(i)$ then
            $path(i) \leftarrow (o(i), p(i), path(k))$ // Case 2
            return True
        return False
    if $len(i) - len(k) \equiv 1$ then
        if $hash(o(i), path(k)) \equiv h(i)$
            $path(i) \leftarrow (o(i), path(k))$ //Case 1
            return True
        if $hash(o(i), p(i), path(k) - o(k)) \equiv h(i)$ then
            $path(i) \leftarrow (o(i), p(i), path(k) - o(k))$// Case 3
            return True
        return False
```

In order to speed up the iterative boosting algorithm and aid on reconstructing paths, a fast bootstrapping algorithm as in Algorithm 2 is further utilized. This algorithm then supplies more initial reconstructed paths to be utilized by the iterative boosting algorithm. The initial reconstructed paths aid in lowering the iterations amount needed, while accelerating the iterative boosting algorithm.

Algorithm 2: Fast Bootstrapping Algorithm [10]

Input: All received packets and a packet whose path is being reconstructed
Output: *path(k)*: the routing path of packet *k*

```
procedure FAST-BOOTSTRAPPING
    $path(k) \leftarrow\ o(k)\ \ p(k)$
    $n \leftarrow p(k)$
    while $n \neq sink$ do
        $u \leftarrow \arg\max_x \hat{t_g}(x) + \Delta_x^u \qquad x : o(x) \equiv n$
            $\cap \hat{t_g}(x) + \Delta_x^u < \hat{t_g}(k) - \Delta_k^l$
        $v \leftarrow \arg\min_x \hat{t_g}(x) + \Delta_x^v \qquad x : o(x) \equiv n$
            $\cap \hat{t_g}(x) - \Delta_x^l < t_s(k)$
        if $u \equiv \{\} \quad v \equiv \{\} \quad pc(u) \neq pc(v)$
            break
        $path(k) \leftarrow path(k) \cup p(n)$
        $n \leftarrow p(n)$
return $path(k)$
```

4. SIMULATION IMPLEMENTATION

In this section, we explain the steps how can run and briefly describes the basis of simulation code. The summary of the main processes as the following:

Input: Available Sensor Node Details (SND), Sink Node (S)
Step 1: D = {}, i = 0
Step 2: For each Sensor Node SN from SND
Step 3: Dist = Compute SN distance from S
Step 4: Dir = Compute SN direction from S
Step 5: D[i] = Dist +”,”+ Dir; i++;
Step 6: initial energy= 100 J
Step 7: End For
Step 8: Compute One hop Path (OHP) from D
Step 9: Construct Path (P) using Algorithm 2 Fast Bootstrapping Algorithm
Step 10: Send P to each Sensor Node.
Step 11: After receive packet via P the Sink Node S compute the energy consumption and reconstruct a new Secure Path (SP) using Algorithm 1 Iterative Boosting Algorithm.
Output: Reconstruction path.

The simulation has five (5) modules to represent the process of energy efficient path reconstruction, which are:

- Network Formation: construct wireless sensor network scenario that has one sink node and N number of sensor nodes. This sensor node has limited energy and fixed in various locations and Sink node has each sensor node details. Then the sink node computes each sensor nodes distance and direction.
- Compute one hop path & Path Construction: E-iPath starts with an established set of paths while constantly carrying out path inference. Here, known set of path means one hop path. One hop path with direction details used to construct path because, the large path should be constructed from known sort path. Finally, sink nodes inform path details to each sensor nodes.

- Packet Transmission: After receiving path details, each sensor nodes generate packet and compress packet to reduce data traffic and energy. After compressing, it transmits a compressed packet to sink node via its path, which is informed by sink node.
- Path Reconstruction: sink node receives compressed packets from each sensor nodes and decompresses. Then, it deletes packet received path, and reconstructs new path based on known shortest path (one hop path).
- Energy Consumption: The compute of energy consumption is done by the sink node when received path details. The path includes nodes state of energy. Each node consumes in the transmission process 0.6 J and in the receiving process 0.3 J. When a node sends Packet to the next node, the energy capacity of the node will be reduced.

5. RESULTS & DISCUSSIONS

Results from the simulation using input parameters such as packet sizes and network sizes with different metrics. Networks are randomly generated for simulation studies to provide independent results that are not influenced by the characteristics of the network topology. E-iPath has been implemented using Java language in order to assess its algorithm. The routing algorithm has been compared with iPath [10]. Meanwhile, [14] energy consumption is evaluated by comparing with EEPMM because the main idea of the EEPMM algorithm is to determine the multipath of stability as well as the construct of paths by the nodes. The EEPMM aims of this are to reduce energy consumption.

5.1 Reconstruction Ratio

We re-implement the iPath (R-iPath) with simulation for 400 mobile nodes randomly distributed in a 900 m × 600 m unit area. For each node, 100 meters is the range of radio propagation. The simulation is run six (6) network scales, and comparisons are made from the data collected. Data sessions with Source node 19 and Destination node 14 are simulated. The major factor to evaluate the reconstruction performance is the network scale (number of nides). Table 1 shows sink node reconstructs path for node 19 and informs this reconstructed path (3-59-40-46-Sink) to node 3. Next, the Source node 3 sends packet to sink. Then, it forwards a packet via 3-59-40-46-Sink. After Packet Transmission, the simulation collects evaluation metric values, as shown in Table 1.

Table 1: Reconstruction Ratio (RR)

Network Scale	**Reconstruction path**	**RR**	
		R-iPath	**iPath**
5	3-59-40-46-**Sink**	0.96	1
10	12-4-51-2-84-1-91-40-54-**Sink**	0.89	0.98
15	81-9-23-89-62-3-7-51-20-11-72-1-97-44-16-**Sink**	0.56	0.7
20	69-21-35-46-2-66-73-13-27-60-43-20-51-93-76-6-1-59- 6-46- **Sink**	0.42	0.46
25	217-103-78-236-56-69-21-35-46-2-66-73-13-27-60-43-20-51-93-76-6-1-59-6-6-46- **Sink**	0.22	0.26
30	376-216-321-65-248-203-142-109-71-39-11-79-41-30-63-81-9-23-89-62-3-7-51-20-11-72-1-9-44-16- **Sink**	0.14	0.21

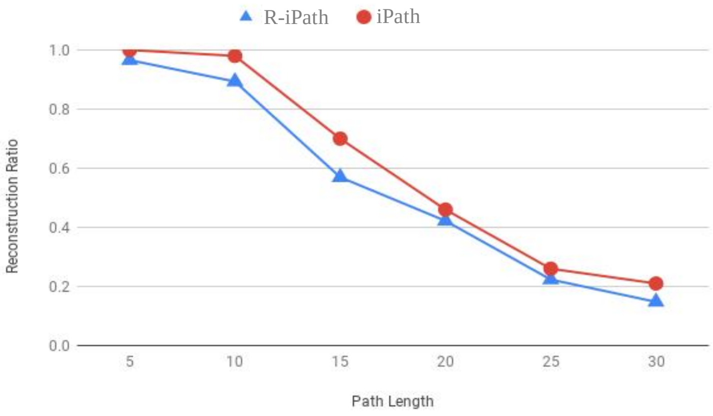

Figure 1: Results of the reconstruction ratios between iPath and R-iPath.

The network scale is a de facto parameter that affects the path length. Therefore, the reconstruction ratio is affected too, since the length of the path has a strong influence on it. The longer of the path length cause of the lower of the reconstruction ratio.

The result at Figure 1 shows that the reconstruction ratio with a number below 10 nodes sustains a possible highest ratio, ranging from 1- 0.89. However, WSN tends to be formed of large scales as a real demand in the applications world. When the number of nodes exceeded 10 until 15, the ratio is still not poor which is 0.56. The reconstruction ratio degraded dramatically when the number of nodes 20 or above which is 4.2. From our observation, even though the reconstruction ratio becomes 4.2, iPath strategy outperforms the previous strategies. When the path length becomes long, the reconstruction probability of each approach degrades.

5.2 Energy Consumption

This simulation assumes that 50 nodes are randomly distributed in a 900 m × 600 m unit area. The range of each node's radio propagation is 100 meters. The simulation is run using five (5) number of nodes for every scenario, and comparisons are made from the data collected. The powers for transmission and receiving are fixed values, of 0.6 J and 0.3 J and the packet size is 522 bytes as shown at Table 2

Table 2: Initial value

No. Nodes	50
Initial Energy	100 J
Receiving Power	0.3 J
Transmission Power	0.6 J
Packet Size	522 bytes

Table 3: Results Energy Consumption

No. of Nodes	**Energy Consumption**	
	EEPMM	**E-iPath**
10	23%	21%
20	32%	24%
30	37%	28%
40	43%	30%
50	45%	32%

Figure 2 shows the energy consumption in E-iPath is less than EEPMM, firstly due to the paths found by EEPMM have routing protocol process to find the next hop by the nodes and its needs more energy consumption, but E-iPath used the path inference by sink node only. Secondly, the compressed the packets led to reduce time the transmitter and receiver between the nodes.

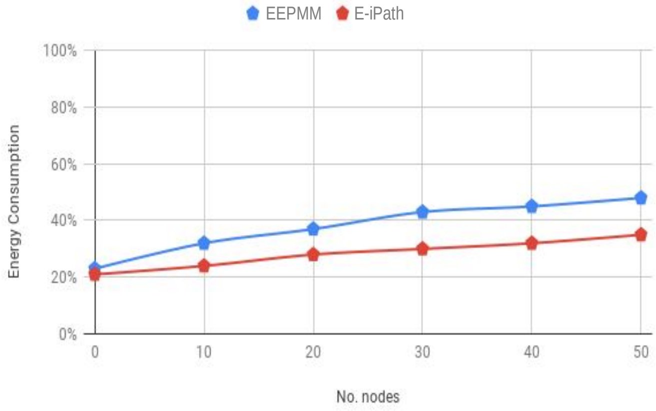

Figure 2: Energy Consumption between EEPMM and E-iPath

We have observed that E-iPath consumes energy lesser than EEPMM by 13% with 50 nodes. As mentioned in two stated reasons above, the size of the packet has a significant influence on energy consumption. Practically, we have found out when a node sends a message with compression, it consumes lesser than without compression by 7%. However, that does not conform to all number of nodes that have been chosen in this study since we can see this clearly in Table 3 the small sizes. This is because both approaches E-iPath and EEPMM work efficiently with small sizes such as 10 nodes. Furthermore, we have concluded that 7% of the total difference between the two approaches which is 13% is gained by compression packet size. Thus, the rest of the energy consumption which is 5% is caused by the second reason; the node consumes energy when it tries to find a new path.

The results exhibit that E-iPath is able to perform with stable energy efficiency and the increase size of network does not have any significant impact on it, while EEPMM's performance declines with the increase of network size. The first reason is that the paths found by EEPMM have routing protocol wants to find the next hop, this operation depend on nodes so needs more energy consumption, but E-iPath used the path inference by sink node. Secondly, the compressed the packets led to reduce time the transmitter and receiver between the nodes.

6. CONCLUSION

The current study proposed E-iPath, an inference method to deal with recreating the per-packet directions ways as routing paths in powerful and expansive scale networks with energy proficient. The E-iPath iteratively finds long paths from short ones. This study has utilized iterative smoothing algorithm to find unknown longest path based on known shortest path. To achieve energy efficiency, it compressed the packet information with high security. E-iPath is evaluated with several variations utilizing traces from broad simulation tool and substantial WSN which is network scale for organizations. The simulation results have demonstrated that the proposed E-iPath avoids data traffic based on path reconstruction, and reduced energy consumption efficiently.

ACKNOWLEDGEMENT

This research was financially supported by the Universiti Putra Malaysia through PUTRA Grant with Project Number GP/2018/9621400.

REFERENCES

1. M. Radi, B. Dezfouli, K. A. Bakar, & M. Lee. **Multipath Routing in Wireless Sensor Networks: Survey and Research Challenges**. *Sensors, 2012*, Vol. 12(1), pp. 650–685. https://doi.org/10.3390/s120100650
2. M. Keller, J. Beutel, & L. Thiele. **How Was Your Journey?: Uncovering Routing Dynamics in**

Deployed Sensor Networks with Multi-hop Network Tomography. *In Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems*, New York, 2012, pp. 15–28. https://doi.org/10.1145/2426656.2426659

3. X. Lu, D. Dong, X. Liao, & S. Li. **PathZip: Packet path tracing in wireless sensor networks**, *IEEE 9th International Conference on Mobile Ad-Hoc and Sensor Systems (MASS 2012)*, 2012, pp. 380–388.
4. Z. Liu, Z. Li, M. Li, W. Xing, D. Lu, Z. Liu, & D. Lu. **Path Reconstruction in Dynamic Wireless Sensor Networks Using Compressive Sensing**, *IEEE/ACM Trans. Netw.* , 2016, Vol. 24(4), pp. 1948–1960. https://doi.org/10.1109/TNET.2015.2435805
5. Y. Gao, W. Dong, C. Chen, J. Bu, G. Guan, X. Zhang, & X. Liu. **Pathfinder: Robust path reconstruction in large scale sensor networks with lossy links**. *21st IEEE International Conference on Network Protocols (ICNP)*, 2013, pp. 1–10. https://doi.org/10.1109/ICNP.2013.6733600
6. Y. Gao, W. Dong, C. Chen, J. Bu, T. Chen, M. Xia, & X. Xu. **Domo: Passive Per-Packet Delay Tomography in Wireless Ad-hoc Networks**, *IEEE 34th International Conference on Distributed Computing Systems*, 2014, pp. 419–428. https://doi.org/10.1109/ICDCS.2014.50
7. L. Ma, T. He, K. K. Leung, A. Swami, & D. Towsley. **Identifiability of Link Metrics Based on End-to-end Path Measurements**, *In Proceedings of the 2013 Conference on Internet Measurement Conference*, New York, 2013, pp. 391–404.
8. Y. Liu, K. Liu, & M. Li. **Passive Diagnosis for Wireless Sensor Networks**, *IEEE/ACM Trans. Netw.*, 2010, Vol. 18(4), pp. 1132–1144. https://doi.org/10.1109/TNET.2009.2037497
9. M. Lee, S. Goldberg, R. R. Kompella, & G. Varghese. **Fine-grained Latency and Loss Measurements in the Presence of Reordering**. *SIGMETRICS Perform. Eval. Rev.*, 2011, Vol. 39(1), pp. 289–300. https://doi.org/10.1145/2007116.2007150
10. Y. Gao, W. Dong, C. Chen, J. Bu, W. Wu, & X. Liu. **iPath: Path Inference in Wireless Sensor Networks**, *IEEE/ACM Trans. Netw.*, 2016, Vol. 24(1), pp. 517–528. https://doi.org/10.1109/TNET.2014.2371459
11. H. Kaur, H. Singh, & A. Sharma. **Geographic Routing Protocol: A Review**, *International Journal of Grid and Distributed Computing*, 2016, Vol. 9, pp. 245–254. https://doi.org/10.14257/ijgdc.2016.9.2.21
12. Z. Xu, Y. Yin, & J. Wang. **A Density-based Energy-efficient Routing Algorithm in Wireless Sensor Networks Using Game Theory**, *International Journal of Future Generation Communication and Networking*, 2012, Vol. 5(4), pp. 99-112.
13. L. M. Feeney, & M. Nilsson. **Investigating the energy consumption of a wireless network interface in an ad hoc networking environment**, *In Proceedings IEEE INFOCOM Conference on Computer Communications. Twentieth Annual Joint Conference of the IEEE Computer and Communications Society (Cat. No.01CH37213)*, 2001, Vol. 3, pp. 1548–1557.
14. B. Sun, M. Lu, K. Xiao, Y. Song, & C. Gui. **An Energy Entropy-Based Minimum Power Cost Multipath Routing in MANET**, *International Journal of Grid and Distributed Computing*, 2016, Vol. 9(2), pp. 169-180. https://doi.org/10.14257/ijgdc.2016.9.2.15
15. S. D. Jean Louis E. **Evaluation of Energy Consumption of Proactive, Reactive, and Hybrid Routing Protocols in Wireless Mesh Networks Using 802.11 Standards**, Retrieved January 5, 2019, https://www.scirp.org/journal/PaperInformation.aspx?paperID=83838
16. M. Navarro, Y. Li, & Y. Liang. **Energy profile for environmental monitoring wireless sensor networks**, *In Communications and Computing (COLCOM) IEEE Colombian Conference*, 2014, pp. 1–6. https://doi.org/10.1109/ColComCon.2014.6860416
17. M. Er-Rouidi, H. Moudni, H. Mouncif, & A. Merbouha. **An Energy Consumption Evaluation of Reactive and Proactive Routing Protocols in Mobile Ad-Hoc Network**, *In 2016 13th International Conference on Computer Graphics, Imaging and Visualization (CGiV)*, 2016, pp. 437–441. https://doi.org/10.1109/CGiV.2016.90