International Journal of Engineering & Technology, 7 (4.10) (2018) 857-862

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Energy Potent Shortcut Tree Routing in Zigbee Networks

B. Rajeshkanna[1]*, M. Anitha[2]

[1]*Department of Electronics and Communication Engineering, Annamalai University, Tamilnadu, India*
[2]*Department of Electrical Engineering, Annamalai University, Tamilnadu, India*
**Corresponding author E-mail: rajeshkanna.ece.au@gmail.com*

Abstract

ZigBee is an international standard for low-power and low-data rate WPAN. ZigBee works on IEEE 802.15.4 MAC and physical layers. Its network layer is liable to facilitate routing in networks. To provide a near optimal routing path as well as to balance the load and energy over the nodes, Energy-efficient shortcut tree routing (ESTR) protocol suggested an optimization method with three criteria such as minimum hop-counts, minimum congestion and maximum link quality. This paper proposes an Energy potent shortcut tree routing (EPSTR) that adds one more criterion called minimum failure-transmissions with EPSTR's criteria. The values for these criteria can be derived from the neighbor table data of a node. Performance evaluation reveals that EPSTR appreciably expands the network lifetime and shows better routing performances compared to ESTR.

Keywords: *Energy Balancing; Network Lifetime; Shortcut; Tree routing; ZigBee*.

1. Introduction

ZigBee is an international standard for Wireless Personnel Area Network [1] aiming at low-power, low-data rate, cost-effective, reliable and scalable devices and its applications. ZigBee works on IEEE 802.15.4 MAC and physical layers. Its network layer [2] is designed to offer multi-hop communication and application layer has number of interoperable diverse application profiles such as Smart Energy [3], Home Automation [4] and Health Care [5].

In a data communication networks, routing is an important function that provides multi-hop transmissions among the nodes. In ZigBee, AODV and Zigbee tree routing (ZTR) [7] are prominent routing mechanisms. Since AODV is a reactive routing, it can provide an optimal routing path between a source and a destination. For that, AODV applies a route discovery procedure [6] prior to the actual data-packet transmission takes place. This route discovery overhead proportionately increases as the number of concurrent transmissions increase. Also, the route discovery packets are flooded to entire network that produces rigorous congestion in network nodes.

On contrary, ZTR routes the data packets through its child or parent using child-parent relations. But, ZTR cannot provide near optimal routing path. Since ZTR follows tree topology, it does not need to conduct route discovery. These inherent characters of ZTR make suitable for the networks containing low-memory and low-power tiny nodes. Despite these advantages, ZTR has two main problems: 1) It tends to increase the hop-counts as compared to AODV. 2) Due to hierarchical tree arrangement, the router nodes close to the root node experience rigorous congestion and consume extra power since they should manage the traffic of the lower level nodes. Consequently, the batteries of ZigBee devices deplete soon.

A routing is proposed in [8] called Energy-efficient Shortcut Routing (ESTR) that employs an optimization method to solve the problems of ZTR. Further, ESTR can balance the energy and the load over nodes to extend the network life time. For that, ESTR uses the

hop-counts, the link quality and the congestion of a neighbor node as the optimization criteria to designate an energy-efficient neighbor node as the next hop to the destination. The number of previous failure-transmissions of a neighbor node is also an important criterion that influences the routing efficacy.

This paper proposes a routing mechanism called Energy Potent Shortcut Tree Routing (EPSTR) to enhance the load and energy balancing features by adding one more criterion namely the number of previous failure-transmissions with EPSTR's criteria. The main idea of EPSTR is to designate a high potent neighbor node as the next hop. A node is said to be high potent when it has more data-packet transmission ratio without wasting the energy in unnecessary transmissions.

The factors to decide a neighbor node is a high potent node are follows: minimum hop-counts to a destination, high link quality, low congestion and minimum number previous failure-transmissions. In summary, EPSTR can solve the problems of ZTR by providing near optimal routing path through high potent nodes. Also, EPSTR retains all the benefits of ZTR and extend the network life time compared to ESTR.

The rest of paper is organized as follows: Section 2 presents the related works of tree routings protocols. Section 3 depicts the problems of ZTR and its solutions. Section 4 describes the algorithm of proposed EPSTR. Section 5 evaluates the performance of EPSTR. Section 6 concludes this paper.

2. Related Works

In recent times, numerous routing methods have been proposed to mitigate the difficulties of the ZigBee tree routing. STR (Shortcut Tree Routing) [7] offers short routing path by designating a next hop node that has minimum leftover tree hops to the destination. Similar to STR, ETR (Enhanced Tree Routing) [9] uses tree index to ease the route calculations. To reduce the routing cost, DFGTR (Destination family group tree routing) [10] is proposed. It selects the neighbor node as the next hop node if it belongs to the destination family group.

Another proposal for ZigBee routing is EAMTR (Energy aware multi-tree routing) [11] that constructs multiple routing trees in a sink-based network. It is to select the least congested route to the root of the network on behalf of each node. It reveals that EAMTR extends the network lifetime by balancing the energy consumption of nodes. An alternative method is introduced in [12] that considers the load balancing in the ZigBee network called IRP (Improved routing protocol) that is designed for energy-balancing and uses an algorithm to ensure that the routing is locally optimal. It considers the residual energy of nodes to avoid the participation of nodes with low residual energy from routing. ESTR (Energy-efficient shortcut tree routing) protocol [8] is an improved version of STR to provide near optimal route and to extend the network life time by balancing the load and energy over nodes.

The protocols described above may be classified into two groups. Intention of the first group is to reduce the hop counts(delay). STR, ETR and DFGTR protocols belong to this group that use neighbor table to select any one neighbor node as the next hop node that is closest to the destination and to establish the shortcut paths through their neighbors. Though, shortcut paths can reduce energy consumption, but the first group routing strategies do not take care of load balancing problem on the nodes directly. The performance estimation of these approaches shows some declines in the number of hops between the source and destination, but they have no intention to evaluate network lifetime and load balancing.

On the other hand, the routing strategies belongs to second group that focus on balancing energy consumption on the nodes for extending the network lifetime. Even though, load balancing indirectly decreases delays, however, second group routing approaches do not take hop counts and delay as constraints directly to find optimal path. EAMTR, IRP and ESTR belong to this group. Objective of this paper is to propose a new routing protocol by improving the optimization method of ESTR for extending the network lifetime.

3. Tree Routing Protocols in Zigbee

ZTR is designed for resource-limited ZigBee devices for delivering the packets to the destinations through multi-hop routing path. ZigBee employs distributed block addressing method [7] to allocates the address to its nodes as explained in (1) and (2). In following equations, the definitions of *Cm, Rm* and *Lm* are the maximum number of children of a node, the maximum number of routers of a node and the maximum depth or level of a tree.

$$Cskip\left(d\right)=\begin{cases}1+C_m\cdot\left(L_m-d-1\right), & \text{if } R_m=1,\\ \dfrac{1+C_m-R_m-C_m\cdot R_m^{\,L_m-d-1}}{1-R_m}, & \text{otherwise}\end{cases} \tag{1}$$

$$A_k=A_{parent}+Cskip\left(d\right)\cdot\left(k-1\right)+1 \qquad \left(1\le k\le R_m\right) \tag{2}$$

$$A_n=A_{parent}+Cskip\left(d\right)\cdot R_{m+n} \qquad \left(1\le n\le C_m-R_m\right) \tag{3}$$

$$A_k<A_n<(A_n+Cskip(d-1)) \tag{4}$$

At each level of a tree, addressing scheme pre-allocates the network address space. As the tree level increases, the network address space is split recursively. At tree level *d,* the *Cskip(d)* in (1) calculates the size of address space allocated by each router node. It covers the *Rm* number of router-capable children and (*Cm-Rm)* number of end-devices. Thus, the size of *Cskip(d)* is same as "*Rm· Cskip(d+1) + (Cm-Rm) + 1".* At tree level *d,* the parent allocates the network address for each k^{th} router-capable child and n^{th} end device as explained in (2) and (3) with the help of *Cskip(d).* From source node or intermediate node, ZTR can decide the destination is one of its direct-child if the address of the destination satisfies (4), then ZTR sends the data-packet to direct-child if the destination address is equal to the address of direct-child; Otherwise it sends to its direct-parent. Since ZTR use tree topology to forward a packet towards the destination, it does not need any routing table. However, it does not provide optimal path.

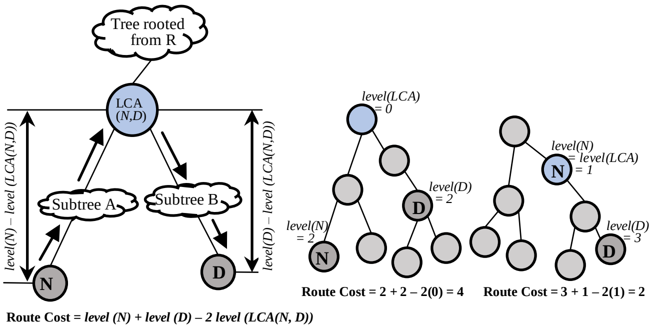


Fig. 1: Computation of tree routing cost [7].

In ZigBee, STR offers optimum short route between the source and the destination. For that, STR selects a neighbor node as the next hop node with least leftover tree hops to the destination; Then STR forward the data-packet to it; Otherwise, STR follows ZTR. In addition, STR solves the hot-spot problem of ZTR and retains all the benefits of ZTR. The key idea of STR is to compute the routing cost (RC) between the neighbor(*N)* and the destination(*D*) using (5).

$$RC=level(N)+level(D)-2\cdot level\Big(LCA\big(N,D\big)\Big) \tag{5}$$

In (5), *level(N), level(D),* and *level(LCA(N, D))*[13] are the level of *N,* the level of *D* and the level of lowest common ancestor between *N* and *D* in a tree. Fig. 1 describes an example of routing cost computation [7] between the given *N* and *D* in a tree.

ZTR and STR have difficulties in wireless environment. Even single lossy wireless link on a path cause the failure of the end-to-end packet delivery since the wireless link is lossy and time-varying [14]. Even though STR provides a near optimal path, it does not consider whether the next hop node has much potential in forwarding a packet to the destination. Hence, it is extremely possible to drop a packet in vulnerable link or traffic congestion circumstances. If a routing protocol can designate a high potent neighbor node as the next hop node, then the difficulties of ZTR and STR can be solved. Thus, this paper shows an intention to solve the difficulties of tree routing in ZigBee by proposing EPSTR.

4. Proposed Algorithm

EPSTR combines STR and ESTR. ESTR use three optimization criteria such as minimum hop-counts, minimum congestion and maximum link quality. For further reduction in congestion and energy consumption in nodes, EPSTR includes one more optimization criterion called minimum failure-transmissions to select the high potent next hop node for forwarding the packets to the destination. EPSTR derives the values for the criteria from the neighbor table data of a node.

ZigBee Alliance [2] has defined a neighbor table for a node to keep the data of single-hop neighbors within the single-hop transmission range. This table updates data during the link-state procedure, network formation, joining, rejoining and communications among the nodes. The proposed EPSTR use some important fields of neighbor table. These fields are: Network address (NA), Event counter (EC), Transmit failure (TF) and Link quality indicator (LQI) [16]. NA is the16-bit network address of the neighboring device, EC counts the number of transmitted packets, TF counts the number of previous failure-transmissions. LQI indicates the quality of the received signal. MAC and Physical layers of IEEE802.15.4 [15] measure the LQI parameters and determine the value of LQI using energy detection and signal-to-noise estimation techniques.

4.1. Cost Functions of EPSTR

The criteria of the proposed algorithm EPSTR directly impact the energy consumption in nodes consequently the network life time. It is apparent that many network events such as transmissions and receptions exhaust the node's battery soon. It also obvious that the link quality, transmission failures and subsequent retransmission trails greatly impact the energy consumption in network nodes.

EPSTR invokes a subroutine namely *find_LeftOverHops* [17] that returns the hop-counts between the neighbor of a source to the destination. EPSTR considers hop-count as the Routing cost (RC). In neighbor table, EC indicates the degree of congestion in a neighbor node, TF indicates the degree of node's potential in energy saving and LQI indicates the quality of the link. It is clear that higher EC indicates more congested node, so that it is defined as Node cost (NC). The higher value of TF indicates more previous failure-transmissions. So, it is considered as Transmission failure cost (TFC) by the EPSTR. Apparently, the higher LQI designates more reliable link. It means that less packet loss if this link is chosen as the next hop. Hence, EPSTR use inverse of LQI as the link cost (LC).

To apply four criteria simultaneously, EPSTR defines a linear function called total cost (*TC*) as in (6), where coefficients α, β, γ and δ are the weights of the criteria. Depending upon the importance of each criterion, the values of the corresponding coefficients can be chosen by satisfying the condition given in Equation (7). $\widetilde{RC}, \widetilde{NC}, \widetilde{LC}$ and $\widetilde{TFC}$ are locally normalized values of *RC*, *NC*, *LC* and *TFC* respectively and their normalizing functions are described in equation (8), where RC_i, NC_i, LC_i *and* TFC_i are the costs of i^{th} neighbor and $\max_i RC_i$, $\max_i NC_i$, $\max_i LC_i$ and $\max_i TFC_i$ are the maximum cost of all neighbors.

$$TC = \alpha * \widetilde{RC} + \beta * \widetilde{NC} + \gamma * \widetilde{LC} + \delta * \widetilde{TFC} \tag{6}$$

$$\alpha,\ \beta,\ \gamma,\ \delta (0 \le \alpha,\beta,\gamma,\delta \le 1 \text{ and } \alpha+\beta+\gamma+\delta = 1) \tag{7}$$

$$\widetilde{RC} = \frac{RC}{\max_i RC_i},\ \widetilde{NC} = \frac{NC}{\max_i NC_i},\ \widetilde{LC} = \frac{LC}{\max_i LC_i} \text{ and}$$

$$\widetilde{TFC} = \frac{TFC}{\max_i TFC_i} \tag{8}$$

From a source node, the proposed algorithm EPSTR calculates the *TC* for each neighbor by using (6). Then, EPSTR designates a neighbor node with minimum *TC* as the high potent next hop node to the destination and forwards the packet to it. In addition, EPSTR uniformly allocates the workload to all intermediate nodes and relieves the congested links/nodes from the routing competency. Hence, EPSTR can efficiently balance the traffic and congestion in network. Moreover, it can reduce end-to-end delay and shows a remarkable improvement in packet delivery ratio, as it chooses link with better quality.

4.2. EPSTR Algorithm

Table 1 describes the algorithm of proposed EPSTR. It works from a view of an intermediate node or a source node. If the node address is equal to address of the destination, do nothing. The packet is delivered successfully; Otherwise, look up the neighbor table; If destination address is equal to any one of its neighbors' address, then EPSTR transmit packet to that neighbor; Otherwise, for each neighbor, look up the table to get the neighbor's information such as *NA*, *EC*, *TF* and *LQI*; Calculate the leftover hops to the destination; Calculate the TC using the (5); Designate a neighbor node with minimum TC as the next hop node and transmit the packet to it.

Fig. 2 shows an inspiring example of EPSTR where *S* is a source node that has 8 neighboring nodes (n_1, n_2, …., n_8) and *D* is the destination node. For each neighbor of *S*, EPSTR computes the four costs (*RC*, *NC*, *LC* and *TFC*) using the neighbor table data. Table 2 shows the neighbors' data of *S* and the costs of EPSTR. Note that the neighbor's *RC* can be calculated using *'level(n_k)+level(D – 2(LCA(n_k, D))*' and LC is the inverse of neighbor's LQI. Also, $\widetilde{RC}, \widetilde{NC}, \widetilde{LC}$ and $\widetilde{TFC}$ are computed using (8). Since n_6 has minimum total cost, EPSTR designates n_6 as the high potent next hop node to forward the packet to it. To balance the load and energy among the nodes, equal importance is given to each EPSTR's optimization criterion. So that the coefficients of these criteria α, β, γ and δ are initialized with 0.25. However, EPSTR allows to modify

Table 1: Energy Potent Shortcut Tree Routing Algorithm

```
EPSTR (nodeAddr, dstAddr)
Inputs: nodeAddr -network address of a node
        dstAddr – network address of a destination
Output: nextHopAddr – network address of the next hop node
1:  If (nodeAddr = dstAddr)
2:     do nothing go to End;
3:  end if
4: Initialize  $\alpha,\beta,\gamma,\delta$
5: If  $((\alpha+\beta+\gamma+\delta)=1 \text{ and } (0\le\alpha,\beta,\gamma,\delta\le 1))$
6:   Initialize maxRouteCost, maxNodeCost,
maxLinkCost, maxTransFailureCost with 0
7:   Initialize RouteCost, NodeCost, LinkCost,
TransFailureCost, TotalCost, minTotalCost with infinity
8:   for each neighbor of node $n_k$ in neighbor table
9:     if (Addr($n_k$) = dstAddr)
10:        nextHopAddr $\leftarrow$ Addr($n_k$)
11:        go to line 39:
12:      end if
13:      RouteCost($n_k$) $\leftarrow$ find_LeftOverHops(nbrAddr($n_k$), dstAddr)
14:      NodeCost($n_k$) $\leftarrow$ nbrEventCounter($n_k$)
15:      LinkCost($n_k$) $\leftarrow$ 1/nbrLinkQualityIndicator($n_k$)
16:      TransFailureCost $\leftarrow$ nbrTransmitFailure($n_k$)
17:       If (maxRouteCost < RouteCost(nk))
18:        maxRouteCost $\leftarrow$ RouteCost($n_k$)
19:        If (maxNodeCost < NodeCost($n_k$))
20:          maxNodeCost $\leftarrow$ NodeCost($n_k$)
21:          If (maxLinkCost < LinkCost($n_k$))
22:            maxLinkCost $\leftarrow$ LinkCost($n_k$)
23:            If (maxTransFailureCost < TransFailureCost($n_k$))
24:              maxTransFailureCost $\leftarrow$ TransFailureCost($n_k$)
25:      end if
26:   end for each
27:   for each neighbor node $n_k$ in neighbor table
28:      normRouteCost $\leftarrow$ RouteCost($n_k$)/maxRouteCost
29:      normNodeCost $\leftarrow$ NodeCost($n_k$)/maxNodeCost
30:      normLinkCost $\leftarrow$ LinkCost($n_k$)/maxLinkCost
31:      normTransFailureCost $\leftarrow$ TransFailureCost($n_k$)/max
TransFailureCost
32:     TotalCost($n_k$) $\leftarrow \alpha$ (normRouteCost) + $\beta$ (normNodeCost)
+ $\gamma$ (normLinkCost)+ $\delta$ normTransFailureCost
33:       If (TotalCost($n_k$) < minTotalCost)
34:         minTotalCost $\leftarrow$ TotalCost($n_k$)
35:         nextHopAddr $\leftarrow$ nodeAddr($n_k$)
36:      end if
37:    end for each
38: end if
39: Transmit_Packet(nextHopAddr)
40: End
```

the values of α, β, γ and δ dynamically by satisfying the condition (7).

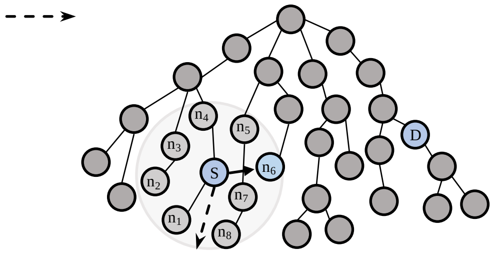


Fig. 2: Example of EPSTR

Table 2: Neighbors data in a source node *S* and their costs

n_k	RC	NC	LQI	LC	TFC	$\widetilde{RC}$	$\widetilde{NC}$	$\widetilde{LC}$	$\widetilde{TFC}$	TC
n_1	9	4	3	0.33	7	1	0.5	0.66	0.78	0.74
n_2	8	1	2	0.5	8	0.89	0.13	1	0.89	0.73
n_3	7	5	9	0.11	4	0.78	0.63	0.22	0.44	0.52
n_4	7	2	6	0.17	8	0.78	0.25	0.34	0.89	0.57
n_5	6	8	2	0.5	4	0.67	1	1	0.44	0.78
n_6	7	1	9	0.11	1	0.78	0.13	0.22	0.11	0.31
n_7	7	3	5	0.2	1	0.78	0.38	0.4	0.11	0.42
n_8	8	4	7	0.14	9	0.89	0.5	0.28	1	0.67

5. Simulation Results

Simulations are performed using NS 2.0 to evaluate the diverse routing performances of EPSTR by comparing with ZTR, STR and ESTR. The simulated network area is 100×100 m^2 with Cm, Rm and Lm equal to 4, 4 and 6 respectively. The number of nodes deployed uniformly for five different simulated networks are: 50, 100, 150, 200 and 250. The coordinator is positioned at the center of the network area and the node deployment is carefully organized to keep up to 8 neighbors for each node. For each routing protocol, there are 30 runs performed and the results are averaged. The general simulation parameters are listed in Table 3, in which an event is defined as the transmission of a packet between the randomly chosen communication pair (source and destination) and the events are sequentially conducted to avoid packet collision or channel contention during the packet transmissions. It allows us to emphasis on performance of routing protocols.

Table 3: Simulation Parameters

Parameters	Value
Network Area	100 m X 100 m
Number of Nodes	50, 100, 150, 200, 250
Deployment Type	Random
PAN Coordinator Position	Center
Initial Node Energy	2376 *Joules*
Value of each cost coefficients	0.25
Communication Pair Selection	Random
Number of Ent-to-end Events	25000
Events Happening Fashion	Sequential
PHY/MAC Layer	
Protocol	IEEE 802.15.4
Propagation Model	Log-Normal Shadowing Radio
Max. Rx. range	25 *m*
Max. Carrier Sensing Range	30 *m*
ZigBee Network Layer	
Routing Protocol	ZTR/STR/ESTR/EPSTR
Cm, Rm, Lm	4, 4, 6
Association Duration	0-50 seconds
Application Session	
Packet Type	CBR
Packet Interval	1 packet/sec
Traffic Type	Any-to-any

5.1. ZigBee Battery Life Time

In the literature, an analysis [18][19] discussed the different energy consumption scenarios of a ZigBee node. That are: transmission of a packet, reception of a packet, sleep mode and idle mode. The total power consumption (P_{con}) of a ZigBee battery is expressed in (8), where P_{tx} is the power consumption for transmitting a packet, P_{rx} is the power consumption for receiving a packet, P_{sleep} is the power consumption during the sleep mode and P_{idle} is the power consumption due to idle state of a node. Using the basic relationship of energy and power, the consumed energy (E_{con}) is expressed in (9), where P_{con} is the total power consumption in *Watts*, E_{con} is the consumed energy in Joules and *t* is time in hours. After t time, the residual energy (E_{res}) in a ZigBee battery is expressed in (10), where E_{init} is the initial energy of the battery. In industry, strength of the battery is expressed in volts and its capacity expressed in *mAH*. In addition, the battery lifetime in hours is expressed in (11) where I_c

Table 4: Freescale Data of ZigBee node battery CR2032[20]

Characteristics	Value
Peukert's Exponent (k)	1
Battery Capacity (I_c)	220 *mAh*
Initial Battery Energy	2376 *Joules*
Standard Load Current	
Transmission	0.2 *mA*
Reception	0.2 *mA*
Sleep Mode	0.048 *mA*
Idle Mode	0.048 *mA*
Total Current (I)	0.248 *mA*
Battery Life	
Theoretical (= I_c / I^k)	887 *hours*
Practical	848 *hours*

is the battery capacity in *mAh*, I^k is the load current in mA and *k* is the Peukert's exponent that varies from 1 to 1.3 according to the age and type of the battery.

$$P_{con} = P_{tx} + P_{rx} + P_{sleep} + P_{idle} \tag{8}$$

$$E_{con} = t \times P_{con} \tag{9}$$

$$E_{res} = E_{init} - E_{con} \tag{10}$$

$$T = {}^{I_c}\!\big/_{I^k} \tag{11}$$

Freescale document [20] describes the use of coin cell batteries for low power applications and recommends that the standard CR2032

battery suitable for ZigBee nodes. Further, this document lists the testing data of CR2032 and its significant data is summarized in Table 4. Based on continuous current demand, this battery is nominally rated at 3*V* with a total capacity of 220*mAh*. We assumed that the ZigBee nodes in our simulations use coin cell type lithium battery CR2032 with initial energy of 2376 Joules (= *3V* × *220mAh* × 3*600 s*) using the Freescale practical data.

5.2. Performance Metrices

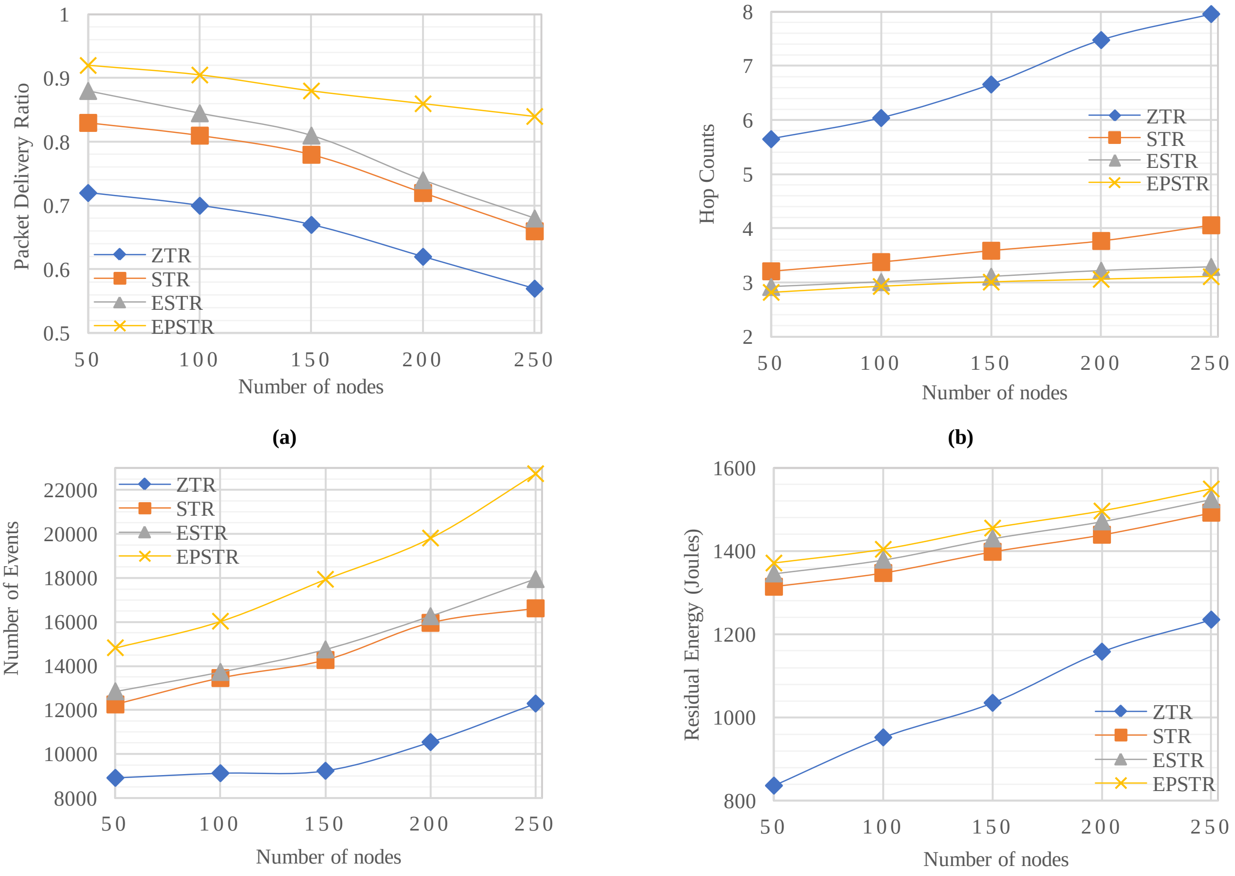
The definitions of routing metrics used in our performance evaluation are:

1. *Packet delivery ratio*: The ratio of packets successfully received by the destination nodes to the total packets sent by the source nodes.
2. *Hop count*: The average of the number of hops taken for each data packet from its source node to the destination.
3. *Residual energy*: The average of residual energy of each node at the end of network lifetime.
4. Network lifetime [21]: The time duration between the network initialization and the moment at the death of first node due to battery exhaustion.

5. *Number of events*: The average number of events happened during the network life time.

5.3. Performance Evaluation

The pocket delivery ratio (PDR) of ZTR drastically falls from 0.72 to 0.57 as the number of nodes rises, as shown in Fig. 3(a). The major causes are the overlapped routing paths with large number of hop-counts to the destination. The rigorous concentration of packets occurs in the root node and the routers around the root. Thus, packet interferences and collisions increase enormously as the density of nodes increases. In other hand, STR, ESTR and EPSTR show PDR about 0.66, 0.68 and 0.84 respectively when the number of nodes is 250. Note that EPSTR show higher PDR than the other routings. It is because the routing paths are so short and distributed among the neighbor nodes not to interfere each other. The PDR evaluation shows that EPSTR outperforms ZTR, STR and ESTR.



(a) **(b)**

(c) **(d)**

Fig. 3: Routing performance for the different network density: (a) packet delivery ratio (b) hop count (c) average number of events happened during the network life time (d) Average residual energy in each node at the end of network lifetime.

Fig. 3(b) depicts the average hop count from a source to a destination. The hop count of ZTR rises from 5.65 to 7.96 hops, since the average tree level of the nodes grows as the number of nodes increases. On the other hand, STR, ESTR and EPSTR are insensitive to the network density. It proves that EPSTR provide the short routing path irrespective of the network topology. The average hop count of STR, ESTR and EPSTR are about 3.21-4.06 hops, 2.92-3.29 hops and 2.82-3.11 hops respectively. Compared to STR and ESTR, the hop count of EPSTR shows slight decrement. But it shows appreciable decrement in hop count when compare to ZTR since EPSTR finds the very short routing path by increasing the number of candidate nodes.

The average number of events during the network lifetime is described in Fig. 3(c). EPSTR shows many events than ZTR, STR and ESTR. The main reasons are keeping less concentration of packets in each node, choosing neighbor node with high link quality for
Forwarding the packets and selecting neighbor nodes with low number of transmission failures. Moreover, EPSTR provided more reliable short routing paths and the paths are highly distributed to entire network. Due to these reasons, EPSTR extends the network lifetime because the nodes consume very low energy.

Fig. 3(d) reports the average residual energy of each node at the end of network lifetime. It shows that EPSTR conducts many events during the network lifetime compared to ZTR, STR and ESTR and it makes the nodes to retain the energy. The main reason is the selection of potential candidate nodes to forward the packets through high reliable short routing paths. Furthermore, EPSTR can efficiently balance the load and energy in entire network. Due to these reasons, EPSTR shows extended network lifetime compared to ZTR, STR and ESTR.

6. Conclusion

In this paper, the proposed EPSTR solves the difficulties of ZTR and STR. The EPSTR included previous failure-transmissions as the additional criterion with ESTR's criteria to focus on energy and load balancing among the nodes as well as optimizing the routing cost. EPSTR efficiently designate a high potent next hop node to enhance the reliability of packet delivery. Performance evaluation shows that EPSTR outperforms ZTR, STR and ESTR in extending the network lifetime.

References

[1] Zhu Y, Chi H, Tian X & Leung V.C.M, (2016), "Network Coding-Based Reliable IPv6 Packet Delivery Over IEEE 802.15.4 Wireless

Personal Area Networks," *IEEE Transactions on Vehicular Technology*, Vol. 65, No. 4, pp. 2219-2230. https://doi:10.1109/TVT.2015.2419082.

[2] ZigBee Alliance, (2015), ZigBee Specification, ZigBee Document 05-3474-21.

[3] Jokar P & Leung V.C.M, (2018), "Intrusion Detection and Prevention for ZigBee-Based Home Area Networks in Smart Grids," in *IEEE Transactions on Smart Grid*, Vol. 9, No. 3, pp. 1800-1811. https://doi: 10.1109/TSG.2016.2600585.

[4] Alexandre Gouvea da Silva C, Leonardo dos Santos E, Christian Krainski Ferrari A & Tertuliano dos Santos Filho H, (2017), "A Study of the Mesh Topology in a ZigBee Network for Home Automation Applications," in *IEEE Latin America Transactions*, Vol. 15, No. 5, pp. 935-942. https://doi: 10.1109/TLA.2017.7910209.

[5] Chen *et al*., (2012), "A Reliable Transmission Protocol for ZigBee-Based Wireless Patient Monitoring," in *IEEE Transactions on Information Technology in Biomedicine*, Vol. 16, No. 1, pp. 6-16. https://doi:10.1109/TITB.2011.2171704.

[6] Ferronato J.J & Trentin M.A.S, (2017), "Analysis of Routing Protocols OLSR, AODV and ZRP in Real Urban Vehicular Scenario with Density Variation," *IEEE Latin America Transactions*, Vol. 15, No. 9, pp. 1727-1734. https://doi:10.1109/TLA.2017.8015079.

[7] Kim T, Kim S.H, Yang J, Yoo S & Kim D, (2014), "Neighbor Table Based Shortcut Tree Routing in ZigBee Wireless Networks," *IEEE Transactions on Parallel and Distributed Systems*, Vol. 25, No. 3, pp. 706-716. https://doi:10.1109/TPDS.2014.9.

[8] Khatiri A, Mirjalily G & Khademzadeh A, (2012), "Energy-Efficient Shortcut Tree Routing in ZigBee Networks," *2012 Fourth International Conference on Computational Intelligence, Communication Systems and Networks*, Phuket, pp. 117-122. https://doi:10.1109 /CIC SyN.2012.31.

[9] Sengaliappan M & Marimuthu A, (2014), "Enhanced tree routing algorithms in wireless sensor network," *2014 International Conference on Green Computing Communication and Electrical Engineering (ICGCCEE)*, Coimbatore, pp. 1-11. https://doi:10.1109/ICGCCEE. 2014.6922415.

[10] Ahn S, Ko D, Kim B, Lee S & An S, (2010), "Energy-Efficient Tree Routing Algorithm-Based Destination Family Group in ZigBee Networks," *2010 Fourth International Conference on Sensor Technologies and Applications*, Venice, pp. 1-6. https://doi:10.1109/ SENSORCOMM.2010.8.

[11] Fariborzi H & Moghavvemi M, (2009), "EAMTR: energy aware multi-tree routing for wireless sensor networks," in *IET Communications*, Vol. 3, No. 5, pp. 733-739. https://doi:10.1049/iet-com. 2008. 0238.

[12] Tao L, Qiao-lin C, Deng-di W & Yan-li B, (2010), "ZigBee Tree Routing Optimize Based on Energy-balanced," *2010 Ninth International Symposium on Distributed Computing and Applications to Business, Engineering and Science*, Hong Kong, pp. 301-307. https://doi: 10.1109/DCABES.2010.68.

[13] Aho A.V, Hopcroft J.E, & Ullman J.D, (1976), “On Finding Lowest Common Ancestors in Trees,” SIAM Journal of Computing, Vol. 5, No. 1, pp. 115-132. https://doi.org/10.1137/0205011.

[14] Reina D.G, Toral S.L, Jonhson P & Barrero F, (2013), "Hybrid Flooding Scheme for Mobile Ad Hoc Networks," in *IEEE Communications Letters*, vol. 17, no. 3, pp. 592-595. https://doi:10.1109/LCOMM .2013.012213.121505.

[15] Tavakoli H, Misic J, Misic B & Naderi M, (2015), "Energy-Efficient Cluster-Head Rotation in Beacon-Enabled IEEE 802.15.4 Networks," *IEEE Transactions on Parallel and Distributed Systems*, Vol. 26, No. 12, pp. 3371-3380. https://doi: 10.1109/TPDS.2014.2385692.

[16] Bhavathankar P, Chatterjee S & Misra S, (2018), "Link-Quality Aware Path Selection in the Presence of Proactive Jamming in Fallible Wireless Sensor Networks," in *IEEE Transactions on Communications*, vol. 66, no. 4, pp. 1689-1704. https://doi:10.1109/TCOMM .2017. 2736550.

[17] Kim T & Kim D, (2016), "Opportunistic Shortcut Tree Routing in ZigBee Networks," *IEEE Sensors Journal*, Vol. 16, No. 12, pp. 5107-5115. https://doi:10.1109/JSEN.2016.2557344.

[18] Hussein H & Ghassan S, (2015), “Mathematical Modeling and Analysis of ZigBee Node Battery Characteristics and Operation” MAGNT Research Report, Vol. 3, pp. 99-106.

[19] Elshabrawy T, Shereen E, Ashour M & Robert J, (2017), "Report Success Probability/Battery Liftime Analysis of Dense IEEE 802.15.4-Based Metering Networks With Hidden Nodes," in *IEEE Sensors Journal*, Vol. 17, No. 7, pp. 2259-2266. https://doi:10.1109/ JSEN.2017.2661879.

[20] Povalac K & Ligertwood P, (2012), “Low Power Considerations for ZigBee Applications Operated by Coin Cell Batteries”, Application Note of Freescale Semiconductor, Document No. AN4573.

[21] Sofiane Ouni & Zayneb Trabelsi Ayoub, (2013), “Predicting Communication Delay and Energy Consumption for IEEE 802.15.4/Zigbee Wireless Sensor Networks”, International Journal of Computer Networks & Communications (IJCNC) Vol.5, No.1. https://doi: 10.5121/ ijcnc.2013.5110.