

## Research Article

# Sink-to-Sink Coordination Framework Using RPL: Routing Protocol for Low Power and Lossy Networks

Meer M. Khan,<sup>1</sup> M. Ali Lodhi,<sup>2</sup> Abdul Rehman,<sup>3</sup> Abid Khan,<sup>4</sup> and Faisal Bashir Hussain<sup>3</sup>

<sup>1</sup>Department of Computer Science, The University of Lahore, Sargodha 40100, Pakistan

<sup>2</sup>Department of Computer Science, COMSATS Institute of Information Technology, Sahiwal 57000, Pakistan

<sup>3</sup>Department of Computer Science, Bahria University, Islamabad 44000, Pakistan

<sup>4</sup>Department of Computer Science, COMSATS Institute of Information Technology, Islamabad 44000, Pakistan

Correspondence should be addressed to Meer M. Khan; [meer.khan78@gmail.com](mailto:meer.khan78@gmail.com)

Received 8 December 2015; Revised 12 May 2016; Accepted 5 June 2016

Academic Editor: Belén Calvo

Copyright © 2016 Meer M. Khan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

RPL (Routing Protocol for low power and Lossy networks) is recommended by Internet Engineering Task Force (IETF) for IPv6-based LLNs (Low Power and Lossy Networks). RPL uses a proactive routing approach and each node always maintains an active path to the sink node. Sink-to-sink coordination defines syntax and semantics for the exchange of any network defined parameters among sink nodes like network size, traffic load, mobility of a sink, and so forth. The coordination allows sink to learn about the network condition of neighboring sinks. As a result, sinks can make coordinated decision to increase/decrease their network size for optimizing over all network performance in terms of load sharing, increasing network lifetime, and lowering end-to-end latency of communication. Currently, RPL does not provide any coordination framework that can define message exchange between different sink nodes for enhancing the network performance. In this paper, a sink-to-sink coordination framework is proposed which utilizes the periodic route maintenance messages issued by RPL to exchange network status observed at a sink with its neighboring sinks. The proposed framework distributes network load among sink nodes for achieving higher throughputs and longer network's life time.

## 1. Introduction

Wireless Sensor Networks (WSNs) are a primitive part of smart environments such as smart homes, buildings, and cities [1]. Smart environments depend on the sensed information from the real world. WSNs consist of specialized components that have sensing, computational, and communicational capabilities for monitoring distributed locations. Many applications have emerged over the past decade, requiring self-organizing distributed sensor networks for monitoring physical environments [2].

WSNs are a part of LLNs that are generally characterized as low power and resource constrained devices that are interconnected with wireless links [3]. Due to miniature sized and cheap quality of radios in LLN devices, the wireless links are lossy as compared to other wireless networks like IEEE 802.11 [4]. Devices in LLNs can act as data originators as well as data routers. Many routing protocols are proposed

for LLNs in general and for WSNs in particular [5–7]. These routing protocols only provide routing within the network and the devices are not directly accessible through Internet. IETF working group titled routing over low power and lossy networks proposed RPL [3, 5] (routing protocol for LLNs). This routing protocol supports IPv6-based connectivity and is aimed at being a standard protocol for providing routing over IPv6-based Low Power Wireless Personal Area Networks (IPv6LoWPANs) in the Internet of Things (IoT) architecture.

RPL supports the deployment of multiple sinks (root nodes) within the network. Each sink node using certain control messages indicates its presence to the network devices and the network is divided into several segments depending on the number of available sinks. Nodes within a segment communicate with their closest or best possible sink node. The control information transmitted by network devices for the management of routing topology is an overhead in RPL

communication. The frequency of control information is dependent on network dynamics. During stable and idle network operation, the frequency of control information is very low. In case when devices are mobile, then topology is constantly changing resulting in high frequency of control messages. This is especially true if the sink nodes are mobile, this forces RPL sinks to send control information to their neighbors at very high frequency resulting in an unstable routing paths and topology.

RPL does not provide any coordination framework that allows sink nodes to know about the presence of other sinks within the network. This type of coordination is important for network optimization. Considering both static and mobile sinks within the network, coordination among sink nodes can allow sinks to learn about the traffic load, level of mobility, network size, and the status of other application defined parameters observed by each sink. As a result, coordinated decision can be taken by the sink nodes to increase/decrease their network size for optimizing over all network performance in terms of load sharing to achieve higher throughput, increasing network lifetime, and lowering end-to-end latency of communication.

In this paper, a sink-to-sink coordination framework is proposed that utilizes the periodic route maintenance messages issued by RPL for coordination. Each sink node monitors its subnetwork size, sink's mobility factor, and packet delivery ratio of its subnetwork. The aforementioned metrics are exchanged among sink nodes using the proposed coordination framework. As a result, sink nodes take the decision of increasing or decreasing their network size for optimized load sharing among sinks. Hence, the overall objective of this coordination is to enhance RPL's performance so that, in the presence of multiple static or mobile sinks within the network, the network load is distributed among sink nodes for achieving higher throughputs and longer network's life time.

The remaining part of this paper is organized as follows: an overview of RPL protocol is presented in Section 2. Section 3 presents the related works on coordination frameworks. In Section 4, network model is illustrated and proposed sink-to-sink coordination framework using RPL is presented. In Section 6, simulation analysis and performance are analyzed. The last section concludes this paper along with future directions.

## 2. RPL Overview

RPL is a gradient-based proactive routing protocol with bidirectional links that builds directed acyclic graphs (DAGs) based on routing metrics and constraints [3]. RPL can create one or more destination oriented DAGs (DODAG) for every root (sink) node within the network. DODAG root is the main root node which constructs the complete DODAG. To build DODAG, root node first multicasts DODAG information object (DIO) with initial rank value 1. Rank defines the individual node positions within the respective DODAG [3]. Apart from rank value, the DIO contains information about objective function, IDs, routing cost, related metrics, and network information. Different objective functions are

proposed in RPL such as objective function zero (OF0) and the minimum rank with hysteresis objective function (MRHOF) [6] for the construction of DODAG.

RPL strives to minimize the cost for reaching the root (sink) from any node in the LLN using an object function. Neighbors of root node receive DIO message and use this information to update their rank, join DODAG, and choose preferred parent by sending feedback to the root. The best preferred parent is used by a child node for routing based on expected transmission count (ETX) and energy. The DIO messages are periodically multicast by nodes for topology maintenance. Periodic feedback is also unicasted by child nodes to their respective parent node using destination advertisement object (DAO) to maintain point-to-multipoint and point-to-point connectivity. Figure 1 shows the flow and direction of DIO and DAO message in RPL network.

## 3. Related Work

Impact of node mobility and mobility management are two core areas of research in RPL networks [8–10]. Also, issues of multipath routing are recently highlighted in RPL networks [11, 12]. In [13], coordination among multiple sinks for mobility management using hybrid routing approach is presented. In [14], our initial work on sink-to-sink coordination framework is presented for static multisink networks. This paper is an extended version of [14] and it addresses the issue of sink-to-sink coordination in a network with both static and mobile sinks. Sink mobility is incorporated in network optimization decision. Also, a more complete analysis and results section are presented, which besides random topology includes grid topology. Apart from [14], to the best of our knowledge, any in-network coordination framework that utilizes RPL control messages or separate communication messages only for network optimization has not been proposed for RPL network. In the remaining of this section, related work on mobility management and the use of multiple sinks within RPL networks is discussed.

The network coordination framework is only proposed for WSNs and is referred to as sensor-actor coordination model [15]. It is proposed for event-driven wireless sensor and actuator network using on-the-fly clustering. The cluster formation is triggered by an event so that clusters are created on the fly to optimally react to the event itself and provide the required reliability with minimum energy expenditure. Also, it proposes a model for actor-actor coordination that is aimed at selecting the best actuator from the event region to take application specific action.

Earliest works on sink mobility in RPL networks investigate benefits of moving sink for quick data collection using RPL in IPv6-based wireless networks [16]. It proposes a distributed and weighted strategy which improves the performance of RPL in terms of network lifetime by moving the sinks towards the leaf nodes. It assumes that the devices are position aware and the sink autonomously takes the decision of moving close to the event region. In the proposed scheme [16], the energy consumption is more balanced among the sensors, leading to a significant increase in network lifetime.

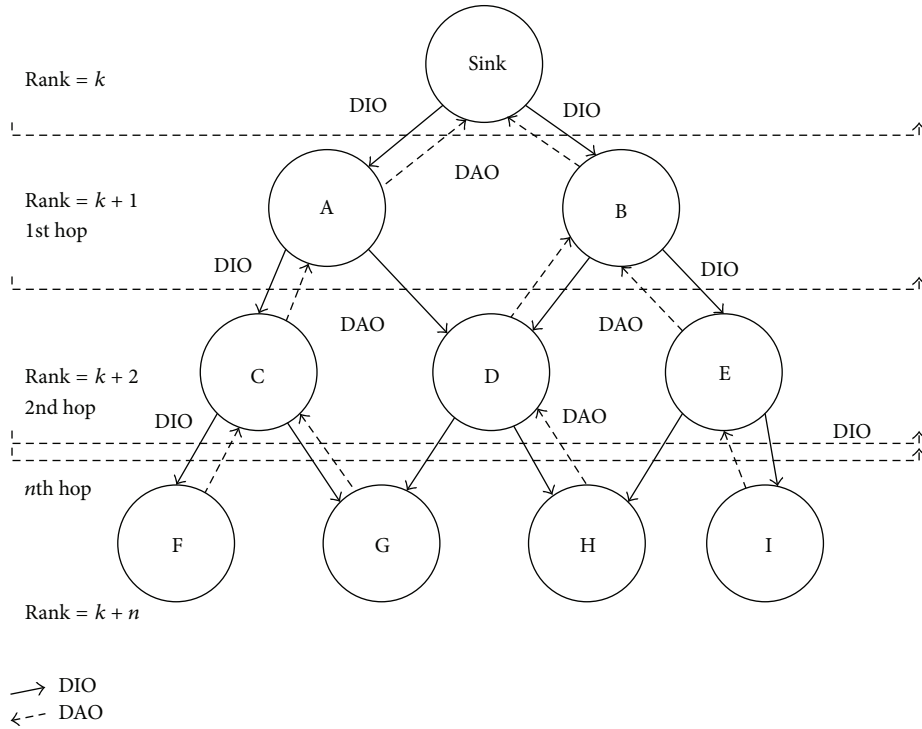


FIGURE 1: Flow of DIO and DAO message in RPL network.

Another approach [8] is proposed for sink mobility using RPL to improve the network lifetime by balancing network energy and avoiding hole creation problem. In [8], leaf nodes within the DAG compute their traffic load in terms of weight. Nodes communicate their weight to the sink node using the available control messages in the RPL. The sink after a certain decision interval calculates which region has the highest weight value and it moves towards that region. Although the proposed scheme decreases energy consumption by moving the sink closer to the event region but as the sink moves the topology is changed. This forces RPL to recreate the topology resulting in extra control overhead.

In [9], a technique is proposed for mobility management of sink nodes in WSNs. When a sink moves, it updates its location in its close vicinity but not in the entire network. In [9], the routing is divided into two stages: in the first stage, source node does not directly send data to the sink but forwards it to a small area containing sink using a geocasting protocol. If the destination is not reached, then in the second stage the forwarding occurs within the complete network.

In [17], an algorithm for finding mobile sinks within WSNs is proposed. Network nodes construct a predictive mobility graph of sink's movement. Mobility graphs have the information about the observed movement patterns of sink node(s) within the network. The mobility graph can be extracted from training data and is used to predict future relay nodes for the mobile node. The mobility graph allows source nodes or forwarding nodes to select relay nodes that can maintain uninterrupted data streams to the sink. The future relay nodes are selected predictively based on previously learned information. The approach requires high

computational capability and storage that are limited in WSNs devices.

In [18], a data driven approach is proposed for the mobility management of the sink nodes. The work takes advantage of the broadcast nature of wireless communication. Each data packet carries additional information about the distance of source/forwarding node and the mobile sink. Nonforwarding or neighboring nodes on the routing path overhear the communication between the source and the sink node. Each node receives and updates its knowledge about the sink's distance.

Mobile Sink based Routing Protocol (MSRP) [19] addresses the hotspot problem and purposes a mechanism for prolonging network lifetime in clustered WSNs. In MSRP, mobile sink moves in a clustered WSN to collect sensed data from the sensor nodes within its close vicinity. During data gathering, mobile sink also maintains information about the residual energy of the network nodes. Sink temporarily deploys/positions itself in only energy rich neighborhood. Consequently, the hotspot problem is minimized as the immediate neighbors of the sink have high energy compared to other nodes in the network.

#### 4. Network Model

In this section, basic network assumption, definition, and network characteristic are discussed. Sink-to-sink coordination framework is suitable for different network topologies: random and grid. The work assumes that the network comprises multiple static and mobile sinks. Also, the coordination is achieved using in-network messages that are

exchanged among sink nodes using intermediate devices. It is assumed that sinks cannot directly communicate with each other. Therefore, the coordination in this scenario is eminent for load balancing among different static or mobile sinks. Also, the coordination framework uses standard RPL control messages (DIO and DAO); therefore, it is considered that network devices periodically issue the aforementioned messages for network maintenance, as recommended by RPL protocol.

Since multiple sinks are used within the network, therefore, it is necessary to understand how RPL builds a network topology in the presence of multiple sink nodes. Also, it is interesting to investigate the effect of sink mobility on routing paths established by RPL. RPL is a proactive routing protocol; therefore, it not only builds the topology but maintains the topology to provide always-on routing to the network devices. During the network deployment phase when multiple sink (root) nodes are introduced in the network, each sink node multicasts DIO control messages to the neighboring nodes. On receiving DIO messages, neighboring nodes extract and use the received information to update their rank, to join DODAG, and to choose preferred parent based on best rank. Immediately, neighboring nodes send a DAO message to their preferred parent indicating that they have joined the network. Root node is the preferred parent of all first hop nodes in the network. The first hop nodes further transmit DIO messages in the downward direction and receiving nodes send DAO message upwards to their preferred parent. In the presence of multiple sinks, network devices will receive DIO messages sent by nodes belonging to different DODAGs. A node joins its closest DODAG based on the rank advertised in the DIO message.

Hence, after network deployment, the network will be partitioned into multiple DODAGs, each maintained by a single sink node. The nodes which are at end of sub-DODAG are the border nodes. These nodes receive DIO messages from their own DODAG and from neighboring DODAG, but they do not further multicast the DIO messages of neighboring DODAGs within their own DODAG. This is because the connectivity with neighboring sink has higher rank or distance value as compared to their own sink. DIO and DAO message are periodically issued by network devices to maintain topology and routing paths. However, the frequency of these messages gradually decreases as the network becomes more stable. But, in case of sink mobility, the frequency is immediately increased by the sink node in order to keep neighboring devices informed about its presence and to update the network topology. Mobility of a single sink can suddenly trigger all network devices within the network to recompute their forwarding paths and topology. Thus, it introduces considerable overhead in terms of extra network transmissions and energy expenditure.

## 5. Sink-to-Sink Coordination Framework (SSCF)

In this section, the operation of the proposed SSCF is explained in length. Operation of the proposed coordination framework comprises several messages between different

sink nodes that can be categorized into framework for sink-to-sink coordination and network optimization. The former is aimed at providing the syntax and semantics for information exchange between multiple sinks. The intermediate nodes using the defined rules relay information between sink nodes. The latter uses the coordination framework to enhance or optimize the network performance using application defined NOMs. In the remaining of this section, both the coordination framework and NOMs are explained in length.

*5.1. Coordination Framework.* Flow chart of coordination framework proposed in RPL shown in Figure 2. Initially, a sink node multicasts DIO messages for the first time to establish a RPL network. A node joins the network by receiving DIO message from the sink node. The DIO messages are multicast by each node for the creation of a multihop DODAG. The proposed work considers multiple sinks with in network; therefore, multiple DODAGs will coexist within single DAG. Each sub-DODAG has a different sink ID that is included in DIO and DAO messages. A node receiving DIO message from different sub-DODAGs is referred to as a border node and has an important role in proposed framework. In normal RPL operation, the DIO and DAO messages transmitted by border node contain only their own sub-DODAG information. Border nodes are edge nodes and their transmissions are also received by the nodes in neighboring sub-DODAGs. In RPL, the neighboring nodes drop the DIO messages received by border nodes of other sub-DODAGs. But in the proposed coordination framework, border node receives and processes the NOMs values in the DIO message. Also, they relay the NOMs value of neighboring sub-DODAGs to their respective sink nodes. The proposed coordination framework introduces network coordination interval (NCI) in normal RPL operation. This is an application defined interval after which every border node updates its respective sink about the status (NOMs) of its neighboring sub-DODAG. Although, NCI is application dependent, but its value should be less than DIO interval. This is because the proposed framework piggybacks the NOMs values in the DIO message; otherwise, extra transmissions will be required. During NCI, any border node can receive multiple DIO messages from other sub-DODAGs but only updates its sink after the expiry of NCI. The border nodes save the NOMs information in its parent table.

Proposed framework modifies the RPL for this purpose and the parent table used in our framework is shown in Table 1. Two new fields are added in the parent table which are sub-DODAG ID and the NOMs values. The border node sends the sub-DODAG ID and NOMs values in its DAO message to its respective sink node. RPL DAO message contains version number updated by the source of DAO message. In the proposed framework, the sink node uses this version number to detect the latest DAO that is being sent by border router. Hence, version number is used for freshness of DAO in the proposed framework.

If a border router does not receive any DIO message from its neighboring sub-DODAG in three consecutive DIO intervals, then it removes the sub-DODAG from the parent table. Also, it informs its sink node about the unavailability

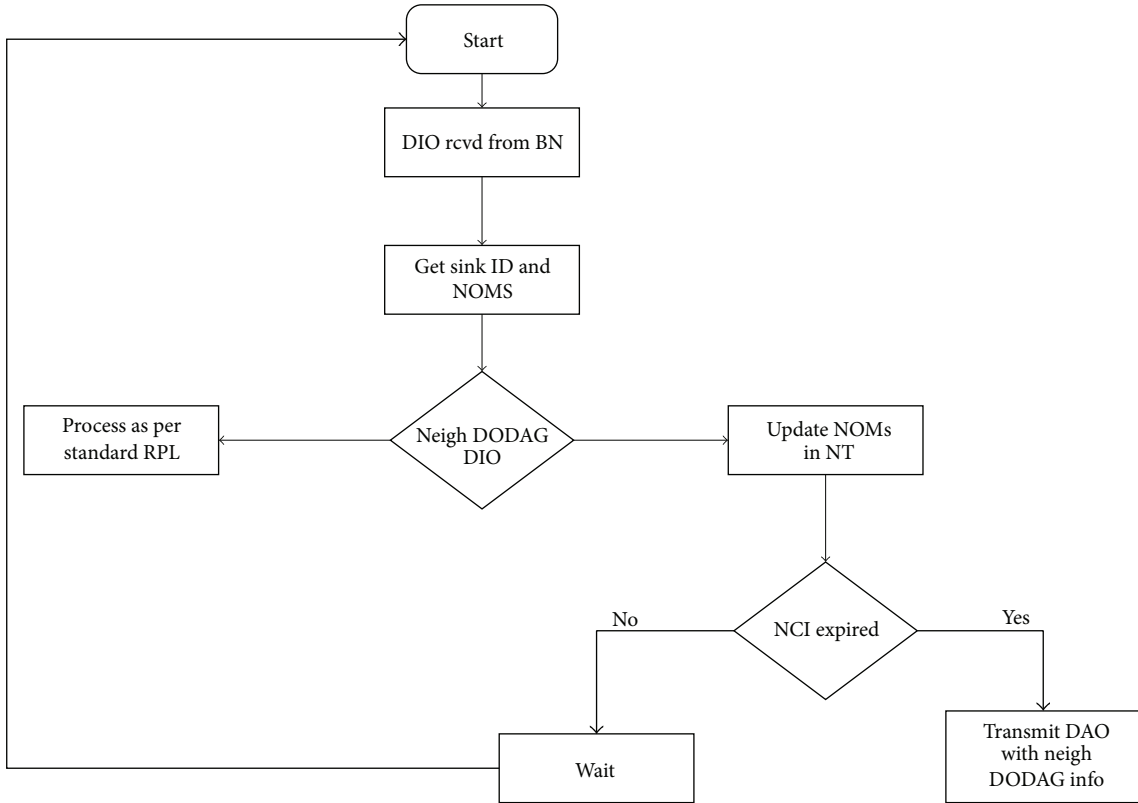


FIGURE 2: Flow chart of sink-to-sink coordination.

TABLE 1: Description of all fields in parent table.

Field	Description
Parent ID	Each node in RPL has the unique 128-bit ipv6 address that becomes each node ID.
DODAG ID	128-bit ipv6 address uniquely identifies a DODAG.
Rank	16-bit rank which determines the parent's individual position with respect to DODAG root.
DTSN	Destination advertisements trigger sequence number: 8-bit unsigned integer which is set by each node that sends DIO. The purpose of DTSN is to control downward traffic.
Sub-DODAG sink ID	8-bit sub-DODAG sink ID stored at each sensor node.
NOM	Network optimization metric values used for coordination among sink nodes.

of neighboring sub-DODAG through DAO message. The proposed framework requires certain modifications within the operation of sink nodes. All sink nodes are required to save the NOM values of neighboring DODAGs. This information is saved in the network optimization metric table (NOMT) maintained at the sink nodes. Four fields are used in NOMT border node ID, sub-DODAG sink ID, NOM, and stale timer. The stale timer defines the maximum time after which this entry is considered out of date and is not used until

fresh information is received. The stale timer corresponds to three DIO intervals.

In RPL, the value of DIO interval ranges from few milliseconds to a maximum of several minutes (17 min). If NCI is mirrored with DIO interval, then the response of border node towards changes in neighboring sub-DODAG will be very slow. Also, in RPL, the value of the DIO interval can be fixed and defined during network deployment. Lower value of DIO interval can be used for better synchronization and coordination among sink nodes. In this case, both NCI and DIO interval can be mirrored; however, proposed framework does not restrict the use of fixed DIO interval. Also, in the proposed framework, it is not mandatory that a node sends DAO message after the expiry of every NCI interval because if the NOMs value of neighboring sub-DODAG is not changing then the border node can skip NOM transmission in DAO. However, in order to avoid entries in the NOMT table of the sink from getting stale, a border router must send the status in alternate DAOs.

The energy cost of SSCF is very low as it uses inherent DIO and DAO messages issued by RPL. However, it is worth mentioning that RPL network can be established without DAO messages, but transmission of DIO is mandatory. DAO messages ensure that the sink has always a route to any node within the network. If DAOs are not used, then the sink does not have any route to the network devices. In this scenario, SSCF will require explicit exchange of DAO messages among sink nodes only for coordination.



**5.2. Network Optimization.** Network optimization is an application-dependent requirement. Optimization can be performed to enhance network lifetime, decrease latency, and increase throughput and reliability. The objective of the proposed work is to define and establish a sink-to-sink coordination framework that can be used to carry any information from one sink to the other sink. Therefore, we use a very simple optimization strategy to demonstrate the working of the coordination framework. The optimization considered in this work is for increasing overall network throughput and extending network life time.

Network optimization metrics (NOMs) in proposed solution define the set of values needed to be exchanged between the sink nodes for optimization. Each sink node in RPL has a complete topology of the network that is used to obtain the total number of nodes in sub-DODAG of the sink. Also, list of data/event generating nodes is available at each sink in RPL. The hop distance between the source nodes and sink can be calculated using the rank value of the source node. The coordination framework uses these metrics and some additional metrics. These additional metrics include packet delivery ratio (PDR) and mobility factor (MF) of a sink node. The PDR is the number of packets received to the number of packets sent by any source node. The sink node calculates the average PDR of all the sources sending data to the sink node. MF defines the mobility level of a sink node. For simplicity, we assume that all sink nodes move with a constant speed.

In SSCF, the optimization metrics used are network size (total number of nodes in its sub-DODAG), PDR, and MF. They are stored and updated by each sink node. They are communicated between sinks using DIO messages. The optional field of DIO message is used to carry network optimization parameters. For PDR and total number of nodes, 8-bit unsigned integer is used, whereas for MF a single flag bit is used. The DIO message generated by sink node travels downwards towards the border nodes. Border nodes add additional information regarding the distance between this border node and the source node (in its own sub-DODAG). The neighboring border nodes of the other sub-DODAGs piggyback this (network optimization metrics) information in the DAO message and send it to their sink nodes. Hence, each sink node has the information about network size, PDR, MF, and average event distance of all sub-DODAGs.

Optimization in the proposed framework is achieved by decreasing the sub-DODAG size of the sink based on the observed PDR and mobility. Sink mobility has adverse effects on the performance of RPL. As the sink moves, its neighboring nodes are changed, thus forcing RPL to increase the frequency of DIO messages to maximum. Also, the version number in the DIO message is increased, indicating all nodes within the network to recompute their rank value. In short, the topology formation is started from scratch. If the sink is constantly moving, then this procedure will continue until sink becomes stationary. Therefore, it is critical that, during mobility, the DIO advertisements of the sink node should not be multicast for all the network nodes. In the proposed framework, a mobile sink decreases the radius of its DIO advertisements to only neighboring nodes until it

```

Algorithm: Network optimization
algorithm executed at static sink
nodes
Variables:
Cpkt_counter (Total number of packets
received by the sink node)
Cpkt_sent (Total number of packets sent by
source nodes)
PDR (packet delivery ratio of the
sink)
Min_PDR_N (Minimum PDR among
neighboring sinks)
Max_Rank_Rest (Maximum rank
restriction)
(1) FOR EACH packet received DO
(2) Cpkt_counter++; Increment pkt counter
(3) END
(4) IF NCI Expired THEN
(5)   PDR = Cpkt_counter/Cpkt_sent
(6)   Cpkt_counter = 0; (resetting)
(7)   IF PDR < Min_PDR_N THEN
(8)     Decrease sub-DODAG size
(9)     Issue DIO with Max_Rank_Rest
(10)  ELSE
(11)    Issue DIO without Max_Rank_Rest
(12)  END IF
(13) END IF

```

ALGORITHM 1: Algorithm for network optimization.

becomes stationary. Radius of the DIO message is decreased by placing a special rank value in the DIO message issued by the sink node. This rank value indicates the Max\_Rank\_Rest (Maximum Rank Restriction) of the DODAG that the sink wants to create. In RPL, there is no restriction on the rank of a node; it increases downwards from the sink to the lower or leaf nodes of the DODAG. But, in the case of sink mobility, using SSCF, the maximum rank is one more than the rank advertised by the sink; therefore, only one-hop neighbors of the sink will receive it and will not further propagate the DIO to their child nodes. In this scenario, the child nodes will not receive DIO by parent nodes and will attach to other available sub-DODAGs through their neighboring nodes. When the mobile node becomes stationary, it removes the maximum rank limit from the DIO messages and the sub-DODAG will now attach more than one-op nodes in its network.

PDR is only considered if there is no mobility of sink within the network, as explained in network optimization algorithm shown in Algorithm 1. Sink nodes take a decision of decreasing their network size based on PDR. Since all the sinks have the PDR of neighboring sub-DODAGs sink nodes, therefore, the sink with the lowest PDR value below application defined threshold decreases its network size. The intuition behind this decision is that PDR generally decreases, if the nodes are unable to send their data successfully to the destination. Congestion is the major reason of packet drops. Congestion occurs at a node, if the arrival rate of packets at the node exceeds the forwarding rate of packets.

In event based sensor networks upon the occurrence of an event, nodes from the event region suddenly start to transmit data. This causes in-network congestion and packet drops. The PDR can be increased by decreasing the traffic flows and load on the sink node. In the proposed optimization scheme, a sink having the lowest PDR and below application defined threshold decreases its network size in an attempt to force some of its event reporting nodes to send data to other sub-DODAG sink nodes. The decrease in network size is achieved by limiting the multicast of DIO messages to the required distance using Max\_Rank\_Rest. If the observed PDR of a sink node is above application defined threshold, then the sink nodes remove the maximum allowed rank restriction from the DIO message. Another important design challenge in the proposed optimization mechanism is how much the sub-DODAG size should be decreased. An event can occur at any place within the sensor field. It can be very close or far from the sink node. If event nodes are very close to the sink node and sink decides to decrease the network size, then a drastic decrease in network size is required to move some event nodes to other sub-DODAGs. As a consequence, the hop distance of event nodes from the new sink will also increase, resulting in lower probability of success information delivery over lossy wireless links. Therefore, in the proposed optimization mechanism, a sink node with the lowest PDR decreases its sub-DODAG size, if the average distance of the event region nodes is greater than half of maximum sub-DODAG size. This ensures that event nodes are at considerable distance from the current sink node and moving them to other sub-DODAG will give overall better network performance despite the increase in hop distance with the new sink. In this scenario, the sink node decreases its sub-DODAG size using

Decrease in Sub\_DODAG Size

$$= \frac{\text{Sub}_{\text{DODAG}} \text{ Distance} + \text{Avg Event Distance}}{2} \quad (1)$$

## 6. Performance Evaluation

In this section, the performances of SSCF and RPL are analyzed in both random and grid topology. We have simulated SSCF and RPL in Cooja [20] simulator in Contiki operating system [20]. The sensor nodes are randomly deployed in 200×200 m. Figures 3 and 4 show the simulation topology; four sink nodes are placed on the corner of the network and five events are generated at five different regions which are shown with circles. Background traffic is generated from different nodes that are uniformly within the sensor field. The details of network parameters used in the simulation analysis are illustrated in Table 2. In [21], very low data rate of two packets per 2.5 to 5 minutes is used but is not suitable for high data rate applications. Therefore, we have high data rates ranging from 2 packets per second to 1 packet per two seconds. The performances of RPL and SSCF are evaluated against throughput, end-to-end latency, and energy consumption. All simulations are run for 10 times and averaged results are presented. Results of random topology are represented with

TABLE 2: Network parameter used in simulation analysis.

Parameter	Value
Network layer	RPL/SSCF
MAC layer	802.15.4
Simulation time	600 s
DIO minimum interval	4 sec
DIO maximum interval	17.5 min
Data intervals (single packet per interval)	0.5, 1, 2, 4, 6, 8 sec
Number of source nodes	2–8
Packet size (excluding header)	50 bytes
Minimum application defined PDR for SSCF	0.6
Tx power	0.0174 W
Rx power	0.0188 W

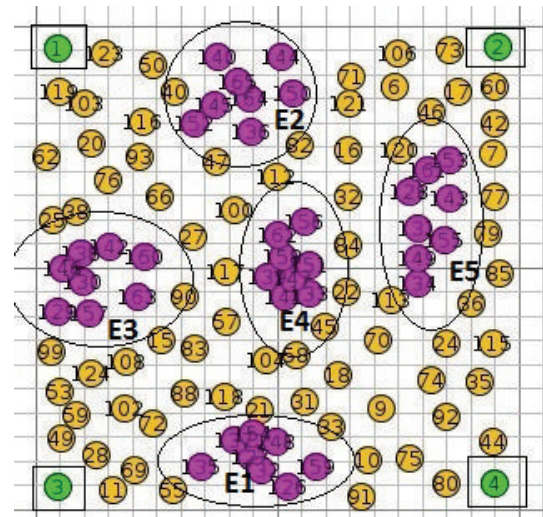


FIGURE 3: Random topology used for simulation analysis.

RT-SSCF and RT-RPL, whereas the results of grid topology are represented with GT-SSCF and GT-RPL. In grid topology, overall better results are achieved for both RPL and SSCF as compared to random topology because of better node placement and availability of multiple alternate forwarding nodes for all network devices.

The collective throughput observed at two sinks using SSCF and RPL is shown in Figure 5. In this scenario, eight source nodes are generating event data at 1 pkt/sec from the event region for 500 seconds. At the time of event occurrence, all the event nodes are present within a single sub-DODAG.

Event regions are randomly selected in each of the 10 simulations conducted and average of the observed throughput for both random and grid topology is presented in Figure 5. Event reporting nodes in RPL send data to their nearest sink, but paths get congested as traffic converges at certain nodes near the sink that result in packet drops. Therefore, the collective throughput observed at both sinks using RPL is less. On the other hand, SSCF distributes the event traffic to both sinks resulting into the formation of different paths

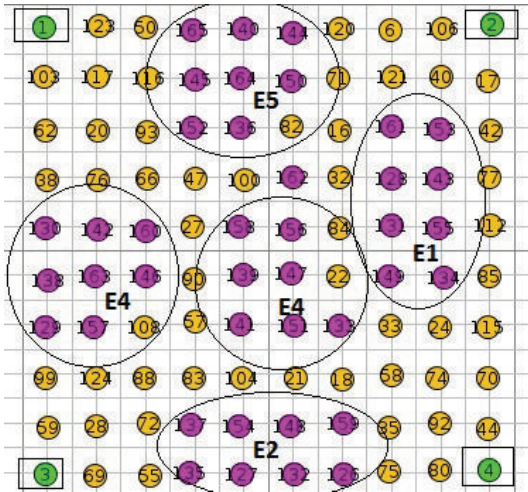


FIGURE 4: Grid topology used for simulation analysis.

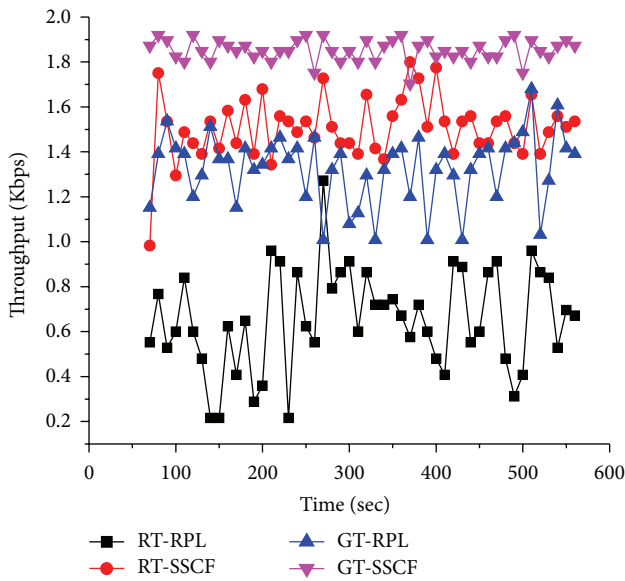


FIGURE 5: Average throughput of SSCF and RPL.

with lower traffic load and lesser congestion. As a result, using SSCF, the combined throughput observed at both sinks is considerably higher than RPL. Average overall throughput observed at two sink nodes using different packet generation intervals of 0.5 sec, 1 sec, 2 sec, 4 sec, 6 sec, and 8 sec is shown in Figure 6.

Packet generation interval defines the length of time after which a source node generates a packet for the destination. Lower packet generation interval means higher data rate, whereas higher packet generation interval corresponds to lower data rates. In this simulation, eight source nodes are used that are randomly placed in a single sub-DODAG. Collective throughput of SSCF is significantly better than standard RPL especially when the packet generation interval is low. At lower intervals, more packets are transmitted resulting in higher level of congestion and packet drops. Since

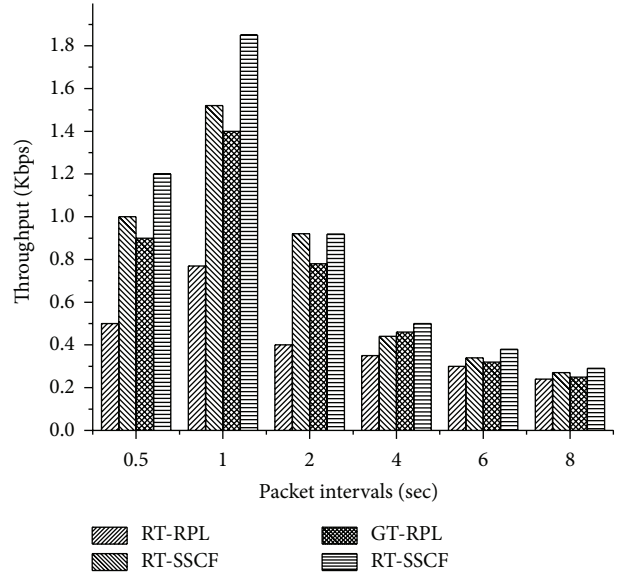


FIGURE 6: Average throughput of SSCF and RPL under different packet generation intervals.

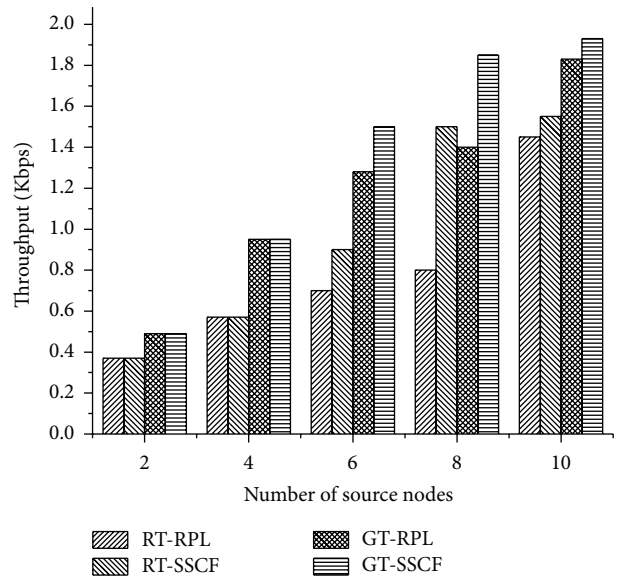


FIGURE 7: Average throughput observed at sink using different number of source nodes.

sinks in SSCF dynamically adjust their network size based on the observed PDR, therefore, network load is shared by both sinks. Thus, SSCF increases the throughput of the network by lowering the congestion as compared to RPL that does not apply any kind of coordination among the sink the nodes.

The effect of increasing source nodes on the throughput of the network in multisink network is shown in Figure 7. Throughput using RPL and SSCF increases with the increase in number of source nodes. Using few source nodes in the event region does not create congestion within the network due to low event traffic; therefore, both protocols provide similar and better result. When source nodes are increased,



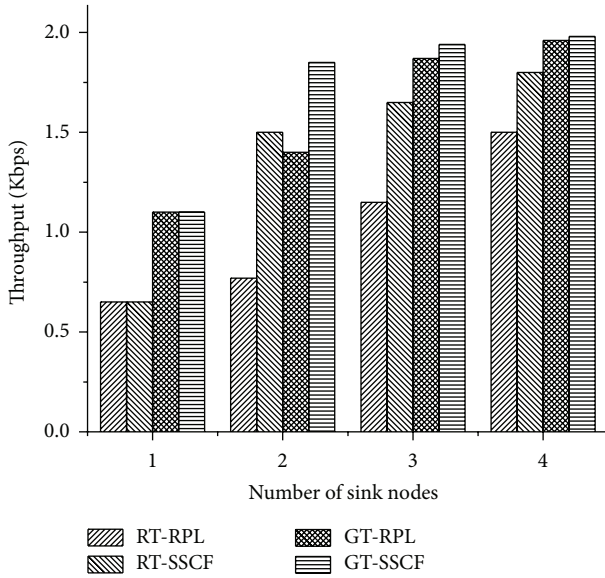


FIGURE 8: Throughput using different number of sink nodes.

the network starts to get congested and the performance of RPL in terms of throughput is lowered. On the other hand, using SSCF network, congestion is decreased by splitting the network traffic among multiple sinks; hence, better throughput is observed.

Total network throughput observed in networks comprising different number of sink nodes is shown in Figure 8. As the number of sink nodes is increased, the hop distance between the event reporting nodes and the nearest sink decreases which decreases congestion and enhances the probability of successful information delivery. It is evident from Figure 8 that increasing the number of sink nodes increases the overall network throughput provided that the sinks are uniformly placed within the network. SSCF generates superior performance than simple RPL because it distributes the network load among sink nodes in a probabilistic fashion.

The impact of sink mobility on the throughput of the network using RPL and SSCF is shown in Figure 9. Four sink nodes are used in the simulations with different number of mobile sinks: case 1, all sinks are static; case 2, one out of four sinks is mobile; case 3, two out of four sinks are mobile; and case 4, three out of four sinks are mobile. Mobility pattern for mobile sinks is random and they move with a constant speed of 1 m/s. Also, the pause time during mobility is between 1 to 5 seconds and is randomly selected by the sink node. Event nodes are sending data at 1 pkt/sec to their nearest sink node in the simulation setup. RPL is a proactive routing protocol; when a sink node moves, it forces the topology of the network to change using DIO control messages. The performance of standard RPL is more affected as compared to SSCF because the proposed coordination framework limits the network diameter of mobile sinks.

Figures 3 and 4 show the location of different events in the sensor field that are used to study the impact of event location on the throughput of RPL and SSCF in both random and grid topologies. Five events are generated from different positions

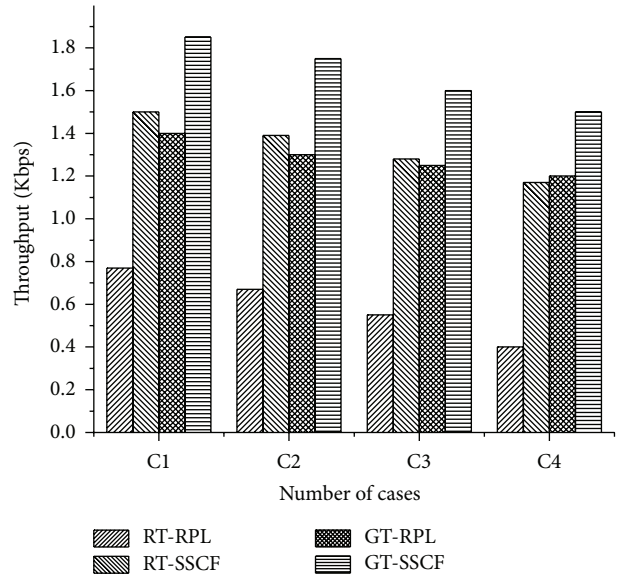


FIGURE 9: Average throughput with mobile sink nodes in different topologies (C1: sinks are static, C2: 1/4 sinks is mobile, C3: 2/4 sinks are mobile, and C4: 3/4 sinks are mobile).

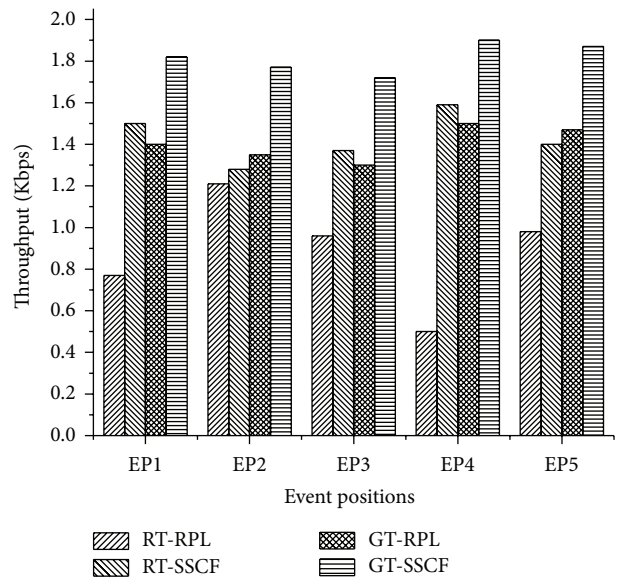


FIGURE 10: Average throughput on event positions in RPL and SSCF in different topology.

in the sensor field. Two sink nodes (1 and 2) are used and each source generates 1pkt/sec from the event region. Figure 10 shows that, in this scenario, SSCF gives better throughput as compared to RPL. Slightly lower throughput using SSCF is observed if event region is very close to the sink location (EP2 and EP3) as compared to when the event location is closer to the edge of sub-DODAG boundary (EP1, EP4, and EP5). This is because SSCF does not divert traffic from nodes very close to sink node.

Average packet drop ratio using RPL and SSCF is shown in Figure 11 using two sink nodes and eight source nodes that

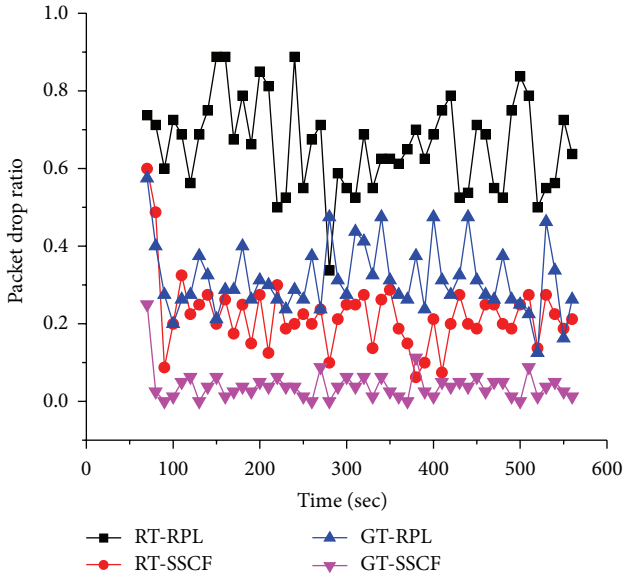


FIGURE 11: Packet drop ratio of RPL and SSCF.

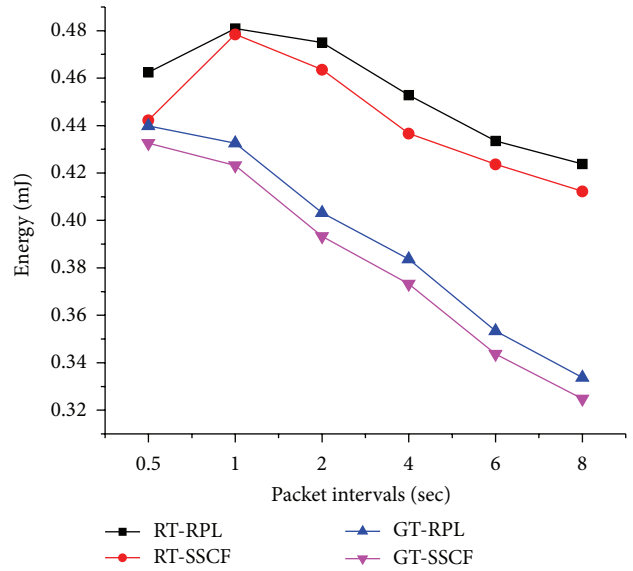


FIGURE 13: Average per bit energy consumption of RPL and SSCF at different packet generation intervals.

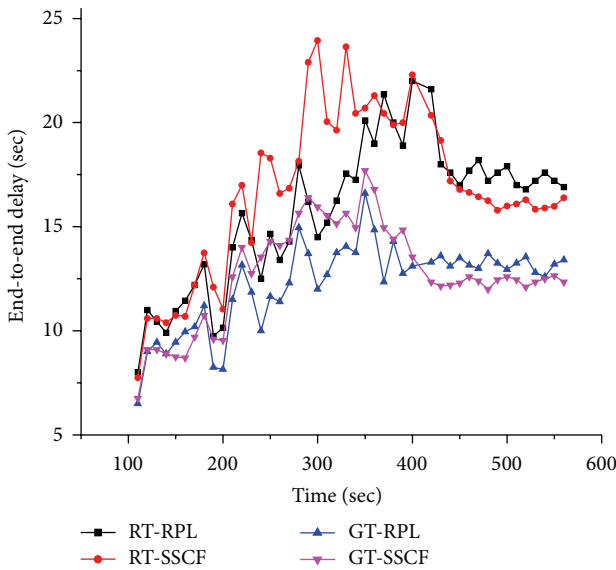


FIGURE 12: Average end-to-end delay of SSCF and RPL.

are generating data at the rate of 1 pkt/sec. It can be seen that packet drop ratio of RPL schemes is more than SSCF, as, in standard RPL, congestion is more severe and higher number of packets are dropped. The end-to-end delay of RPL and SSCF is shown in Figure 12. The latency of both protocols is high because sky notes [22] used in this simulation use very low duty cycle; therefore, nodes spent most of their time in sleep mode compared to active mode. It is noticeable that the delay of SSCF is similar to RPL initially. The data is continuously reported to a sink node; therefore, congestion slowly builds up and becomes persistent. Therefore, delay of RPL increases and then becomes stable. On the other hand, in the case of SSCF, initially, delay is similar to RPL because it works as a normal RPL works before network

optimization takes place. After the detection of low PDR at any sink node, network coordination starts to achieve network optimization. When network optimization takes place in the network, the delay of coordination framework increases. But, with continuous data reporting, the end-to-end delay of SSCF framework stabilizes and becomes slightly better than RPL.

Figure 13 shows the per bit energy consumption of SSCF and RPL against different data generation intervals. The energy consumption of SSCF is slightly better than RPL. If the packet interval is high, then the data rate is low and RPL does not drop much packets and as a result the per bit energy consumption is low as compared to lower packet generation intervals. Also, grid topology has higher PDR and less number of packets are dropped resulting in high throughput. Therefore, its energy wastage is less as compared to random topology.

## 7. Conclusion

RPL is the recommended routing protocol of IETF for IPv6-based LLNs. In this work, sink-to-sink coordination framework is proposed, aiming at increasing network throughput and enhancing network life time. Operation of the proposed coordination framework is based on defining syntax and semantics for exchange of network optimization parameters among sink nodes within the network using in-network forwarding. As a result, of the proposed coordination, the sink nodes adjust their network size for better load balancing to achieve higher overall network throughput. Simulation analysis has shown that the proposed framework provides notably higher throughput with lower energy consumption as compared to simple RPL in a multisink environment. Proposed framework provides 22% and 26% higher packet delivery ratio in random and grid topologies, respectively.

## Competing Interests

The authors declare that there are no competing interests regarding the publication of the paper.

## References

- [1] V. C. Gungor, D. Sahin, T. Kocak et al., "Smart grid and smart homes: key players and pilot projects," *IEEE Industrial Electronics Magazine*, vol. 6, no. 4, pp. 18–34, 2012.
- [2] D. Georgoulas and K. Blow, "In-motes EYE: a real time application for automobiles in wireless sensor networks," *Journal of Wireless Sensor Network*, vol. 3, no. 5, pp. 158–166, 2011.
- [3] T. Winter, P. Thuber, B. Brandt et al., "IPv6 routing protocol for low-power and lossy network," Internet Engineering Task Force (IETF) Request for Comments, 2008.
- [4] IEEE Standards, "IEEE Std. 802.11-2007, part 11 : wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications," IEEE Std. 802.11, 2007.
- [5] J. Pope and R. Simon, "CREST: an epoch-oriented routing control plane for Low-Power and Lossy Networks," in *Proceedings of the IEEE 38th Conference on Local Computer Networks Workshops (LCN '13)*, pp. 128–136, Sydney, Australia, October 2013.
- [6] N. Accettura and G. Piro, "Optimal and secure protocols in the IETF 6TiSCH communication stack," in *Proceedings of the IEEE 23rd International Symposium on Industrial Electronics (ISIE '14)*, pp. 1469–1474, Istanbul, Turkey, June 2014.
- [7] T. Clausen, J. Yi, and A. C. de Verdiere, "LOADng: towards AODV version 2," in *Proceedings of the 76th IEEE Vehicular Technology Conference (VTC Fall '12)*, pp. 1–5, Quebec City, Canada, September 2012.
- [8] L. B. Saad and B. Tourancheau, "Sinks mobility strategy in IPv6-based WSNs for network lifetime improvement," in *Proceedings of the 4th IFIP International Conference on New Technologies, Mobility and Security (NTMS '11)*, pp. 1–5, IEEE, Paris, France, February 2011.
- [9] G. Wang, T. Wang, W. Jia, M. Guo, H.-H. Chen, and M. Guizani, "Local update-based routing protocol in wireless sensor networks with mobile sinks," in *Proceedings of the IEEE International Conference on Communications (ICC '07)*, pp. 3094–3099, Glasgow, Scotland, June 2007.
- [10] K. C. Lee, R. Sudhaakar, J. Ning et al., "A Comprehensive evaluation of RPL under mobility," *International Journal of Vehicular Technology*, vol. 2012, Article ID 904308, 10 pages, 2012.
- [11] W. Tang, X. Ma, J. Huang, and J. Wei, "Toward improved RPL: a congestion avoidance multipath routing protocol with time factor for wireless sensor networks," *Journal of Sensors*, vol. 2016, Article ID 8128651, 11 pages, 2016.
- [12] M. A. Lodhi, A. Rehman, M. M. Khan, and F. B. Hussain, "Multiple path RPL for low power lossy networks," in *Proceedings of the IEEE Asia Pacific Conference on Wireless and Mobile (APWiMob '15)*, pp. 279–284, Bandung, Indonesia, August 2015.
- [13] V. Safdar, F. Bashir, Z. Hamid, H. Afzal, and J. Y. Pyun, "A hybrid routing protocol for wireless sensor networks with mobile sinks," in *Proceedings of the 7th International Symposium on Wireless and Pervasive Computing (ISWPC '12)*, Dalian, China, July 2012.
- [14] M. M. Khan, M. A. Lodhi, A. Rehman, and F. B. Hussain, "A multi-sink coordination framework for low power and lossy networks," in *Proceedings of the International Conference on Industrial Informatics and Computer Systems (CIICS '16)*, pp. 1–5, IEEE, Sharjah, United Arab Emirates, March 2016.
- [15] T. Melodia, D. Pompili, V. C. Gungor, and I. F. Akyildiz, "A distributed coordination framework for wireless sensor and actor networks," in *Proceedings of the 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '05)*, pp. 99–110, Urbana-Champaign, Ill, USA, May 2005.
- [16] T. Truong, K. Brown, and C. Sreenan, "Using mobile sinks in wireless sensor networks to improve building emergency response," in *Proceedings of the IEEE 1st International Conference on New Technologies, Mobility and Security*, Cork, Ireland, 2010.
- [17] B. Kusy, H. Lee, M. Wicke, N. Milosavljevic, and L. Guibas, "Predictive QoS routing to mobile sinks in wireless sensor networks," in *Proceedings of the International Conference on Information Processing in Sensor Networks (IPSN '09)*, pp. 109–120, San Francisco, Calif, USA, April 2009.
- [18] L. Shi, B. Zhang, K. Huang, and J. Ma, "An efficient data-driven routing protocol for wireless sensor networks with mobile sinks," in *Proceedings of the IEEE International Conference on Communications (ICC '11)*, pp. 1–5, IEEE, Kyoto, Japan, June 2011.
- [19] B. Nazir and H. Hasbullah, "Mobile Sink based Routing Protocol (MSRP) for prolonging network lifetime in clustered wireless sensor network," in *Proceedings of the International Conference on Computer Applications and Industrial Electronics (ICCAIE '10)*, pp. 624–629, Kuala Lumpur, Malaysia, December 2010.
- [20] Y. S. Yun and Y. Xia, "Maximizing the lifetime of wireless sensor networks with mobile sink in delay-tolerant applications," *IEEE Transactions on Mobile Computing*, vol. 9, no. 9, pp. 1308–1318, 2010.
- [21] V. Safdar, F. Bashir, Z. Hamid, H. Afzal, and J. Y. Pyun, "A hybrid routing protocol for wireless sensor networks with mobile sinks," in *Proceedings of the 7th International Symposium on Wireless and Pervasive Computing (ISWPC '12)*, pp. 1–5, Dalian, China, July 2012.
- [22] F. Sun, C.-L. Fok, and G.-C. Roman, "sChat: a group communication service over wireless sensor networks," in *Proceedings of the 6th International Symposium on Information Processing in Sensor Networks (IPSN '07)*, pp. 543–544, ACM, Cambridge, Mass, USA, April 2007.





**Hindawi**

Submit your manuscripts at  
<http://www.hindawi.com>

