Route Optimization for Energy Efficiency in IoT and Software Defined RPL Heterogeneous Networks

Amir Gasouma (amirkku@gmail.com)
Universiti Teknikal Malaysia - Main Campus: Universiti Teknikal Malaysia Melaka
Kamaludin Yusof
Universiti Teknikal Malaysia - Main Campus: Universiti Teknikal Malaysia Melaka
Belgacem Bouallegue
King Khalid University
Abdelmoty Ahmed
King Khalid University
Mahmoud Khattab
King Khalid University

Research Article

Keywords: Internet of Things, RPL, Software-Defined Networks, Adaptive Objective Function, Energy Efficiency.

Posted Date: July 25th, 2023

DOI: https://doi.org/10.21203/rs.3.rs-2813981/v1

Route Optimization for Energy Efficiency in IoT and Software Defined RPL Heterogeneous Networks

Amir Gasouma[1,2], Kamaludin M. Yusof [1], Belgacem Bouallegue[2], Abdelmoty M. Ahmed[2], Mahmoud M. Khattab[2]

[1]Universiti Teknologi Malaysia, 81310 UTM Skudai, Johor Bahru, Malaysia.
[2]College of Computer Science, King Khalid University, Abha 61413, Saudi Arabia

Corresponding author: Amir Alrashid Abdalla Gasouma (amirkku@gmail.com).

Abstract The Internet of Things (IoT) technology is one of the most important emerging technologies in today's world, and it is one of the most important and hot topics in information technology research. The Internet of Things (IoT) refers to the concept of connecting smart things to monitor, control, or exchange data over the Internet. These smart things could be tiny devices, with limited battery capacity and power supplies. These devices' high energy consumption shortens their lifespan, affecting the entire IoT network. The Internet Engineering Task Force (IETF) developed the main routing protocols used in the IoT, such as the IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL), and standardized it in RFC6550, as one of the IoT's core routing protocols, and it is the only standard protocol that assists the routing process in Low Power and Lossy Networks (LLNs) of IoT applications. An approach that addresses the challenges of IoT networks and exploits new flexible network architectures, such as Software-Defined RPL networks, there is a considerable gap in adapting objective functions (OFs) for routing and controlling control messages for RPL operations, which enhance the energy efficiency of the IoT networks. This paper proposes a unique software-defined RPL system with optimized RPL operations for heterogeneous IoT environments to enhance energy efficiency. The proposed work performed adaptive OF selection and routing, for that purpose the proposed work formulated three categories of objective functions (OF1, OF2, OF3) namely TriOF. The optimal OF is selected based on the status of the network using the Killer Whale Optimization (KWO) algorithm. It improved the performance of adaptive OF selection and enhanced the network energy efficiency. We evaluate the outcomes through a series of simulated experiments using the Network Simulator (NS3). The proposed model approach results in a reduced number of control messages, control overhead, packet delivery ratio, and packet loss rate. Compared to the contrast works, energy consumption is reduced by 40% and 60%, respectively.

Keywords Internet of Things, RPL, Software-Defined Networks, Adaptive Objective Function, Energy Efficiency.

1 Introduction

The Internet of Things (IoT) is a developing paradigm with numerous real-world applications [1],[2]. IoT uses the internet to connect millions of smart devices such as tiny sensors, smart gadgets, and so on. IoT connectivity enables many smart environments such as smart cities, smart industries, e-healthcare, and so on [3]. As the Internet of Things expands, energy efficiency issues continue to arise [4]. The primary reason for these challenges is that IoT creates large-scale heterogeneous environments with various types of nodes, which have limitations such as limited energy and computational resources, limited bandwidth, and low-quality radio communication.

The Software-Defined Networking (SDN) paradigm emerged as a promising approach for implementing alternative routing control strategies, thereby expanding the set of IoT applications that can be delivered by enabling global protocol strategies and network programmability [5], [6]. SDN is an important solution for solving energy consumption issues that existed in IoT. Thus, IoT is integrated with SDN to solve the energy efficiency of the network [3], [7]. SDN uses logically centralized software, hosted in network nodes called SDN controllers, to control the behavior of a network by reducing the network configuration and management complexity. In simple, SDN differs from the traditional network in the way that it separates the control plane and data plane. The model of SDN-IoT is presented in Figure 1. In SDN, the data plane consists of OpenFlow switches which are the forwarding devices, and the control plane consists of controllers which maintain the global view of the network [8]. The SDN network can be constructed with single or multiple controllers in the control plane [9]. When it comes to IoT, the multi-controller SDN model is effective since the network is large in scale [10].

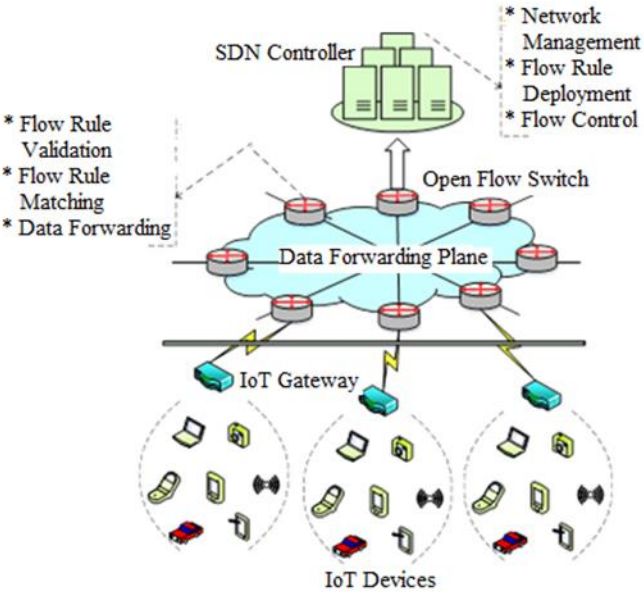

Fig. 1 SD-RPL Integrated Network.

In software defined RPL (SD-RPL) network, the data is generated by the IoT devices, aggregated by the IoT gateways, and forwarded through the SDN switches to the SDN controller. Here, routing between the IoT devices and the gateways plays a significant role in data collection. For that, the RPL routing protocol specifies objective-based parent selection that is specifically designed for IoT communication [11], [12]. For efficient routing of IoT data, RPL protocol is used by constructing the Destination-Oriented Directed Acyclic Graph (DODAG). The DODAG is defined as, the rooting of the directed acyclic graph in a specific destination [13],[14]. Here the data from IoT devices is gathered by the root node (Gateway) in which the parent selection is performed by defining an objective function (OF). This OF can be defined based on different metrics [15]. For analyzing the objective function performance several metrics are calculated. Some important metrics are selected to analyze the performance such as Expected Transmission Count (ETX), Hop Count (HC), and energy consumption of nodes. By these metrics the performance analysis of the objective function is calculated and depends upon the results i.e., optimal OF is selected with high performance to select the optimal parent. [16]. The fuzzy logic is also used in parent selection which can take into account more than one parameter for parent selection [17]. IoT devices are limited to batteries and hence energy-based metrics also play a vital role in parent selection. The data transmission toward the root is succeeded only when the parent selection is efficient. The following factors are important in SDN-IoT networks while using RPL protocol,

•The IoT devices are resource-constrained which is employed limited batteries and processors hence to make those devices live for a longer time, the energy utilization should be better.

•The data transmission is required to be optimal towards the end device without excessive re-transmission counts.

Statistically, the number of devices connected to IoT networks rapidly increases yearly. IoT networks with many Heterogeneous devices capture information and deliver it to backend servers. IoT devices are resource-constrained, and they lose energy for each process. The critical problems stated for energy consumption in SDN-IoT networks are the main motivations behind this work. RPL handles parent selection by using the objective function for forwarding the captured data to the root, which is always based on the objective function. While the IoT networks are heterogeneous, the properties of each node will not be similar, so the objective functions must be selected based on the current behavior of the IoT devices.

Based on this motivation, this research work aims at maximizing the energy efficiency of the RPL-based SDN-IoT network by achieving efficient transmission of data and high packet delivery ratio by defining multiple Objective Functions, selecting optimal OF and routing based on the network status, and considering various routing metrics.

1.1 Major Contributions

To achieve the objectives of this work, we have presented the following major contributions to RPL-based SDN-IoT networks.

- An SDN-IoT network is designed and assisted by RPL-based routing operations. The network is constructed to minimize energy consumption for resource-constrained IoT environments.
- The optimal objective function is adaptive based on the current network status. The KWO algorithm is executed with multiple criteria to select the optimal objective function. Here, we have three categories of objective functions (Tri-OF) that can be dynamically selected for parent selection as follows,

OF(1): Composed of residual energy, transmitter and receiver energy, and energy consumption for data transmission.

OF(2): Composed of mobility, load, and delivery ratio.

OF(3): Composed of ETX, link stability, and queue factor.

By applying these major contributions, the proposed work minimizes energy consumption in the network exponentially.

1.2 Paper Organization

The rest of this paper is organized as follows: Section 2 reviews significant literature works carried out on software-defined RPL networks and summarizes the research gap. Also, the research problems formulated in this work are highlighted in this section. Section 3 explains the proposed system in detail. Sections 4 and 5 lists the simulation parameters and evaluate the performance of the proposed system, and results are compared in terms of performance metrics with the existing works. In section 6, we have concluded the contributions and highlighted the future research directions.

2 Related Works

This section reviews the significant existing works presented on optimal routing in RPL-IoT networks.

RPL-based dynamic data gathering uses learning automata, and it is named LA-RPL [18]. In this work, two objective functions (OFs) were defined, the first was used for graph construction, and the second for learning automata. The first OF is based on the node degree, and the second OF defines the number of data packets. The first OF enabled the maintenance of the topology, and the second was for transmission. Here, both OFs use a single parameter which is insufficient for the heterogeneous IoT environment. A learning automaton for OF (LA-OF) was proposed using the expected transmission count [19]. In this learning automata, both online and offline phases are executed by computing the ETX as the states. According to the actions, rewards, and penalties for them are given. As a result of this, only the packet transmission-based parameter ETX is used. learning automaton was used to learn the network and update the parameters automatically upon ETX. This work mainly uses the Objective function Zero (OF0) which is inefficient to achieve better transmission performance.

The fuzzy logic technique was introduced to define the objective function using context-oriented parameters, i.e., COOF [20]. This OF considers the expected transmission count, queue fluctuation index, and residual energy index. Based on these three constraints, nine rules were established, and using this fuzzy parent was selected. From the fuzzy rules, the quality of the parent was determined as excellent, very good, good, fair, and bad. Then, data transmission was carried out through excellent parent nodes if available. The defined OF is not suitable for all instances, and using fuzzy could be made only when the parameters range between the specified values. If it is new, then the decision will not be efficient. Mobility-aware RPL (MARPL) was presented in an IoT environment to aid with the mobility of nodes and reduce packet loss [21]. MARPL performs three processes mobility detection, parent prediction, and trickle adjustment. The neighbors in RPL were determined based on the received signal strength value. Then the parent was selected based on the rank parameter that was updated concerning mobility. However, mobility is the significant parameter, it is only taken into consideration, even if a node with low mobility will also have lesser energy. In such situations, the preferred parent will fail in transmission.

RPL improvement was achieved for congestion control and energy efficiency [22]. For performing RPL operations, a new metric was formulated as the linear combination of ETX, delay, and node's residual energy. For parent selection, a two-level process in which the first level selects parents based on ETX and the second level uses residual energy as the tiebreaker metric. To avoid congestion, the control mechanism was presented based on time-bound (i.e.), a threshold value was set to control the broadcast. This method first uses a single metric (hop count) for selecting the optimal parent then uses energy as a tiebreaker which results in higher data loss due to the lack of current network status. In the heterogeneous IoT environment, RPL was presented to handle the heterogeneous traffic [23]. For that, queue and workload-based RPL (QWL-RPL) were presented to collect the heterogeneous traffic. The parent selection was performed based on buffer rate and workload which are in a linear combination. The absence of link and network-oriented metrics increases data loss.

Table 1 Summary of related works.

Methods Presented	Research gap
LA-RPL [18] **LA-OF [19]** **COOF [20]** **MARPL [21]** **Two-Level RPL [22]** **QWL-RPL [23]**	• Lack of significant parameters degrades data transmission performance • Using the same OF for all network conditions is not suitable • Mainly increases time and energy consumption

In Table 1, the related works are summarized, and the major research issues are listed. From the analysis, there is still a research gap that exists in achieving energy efficiency in RPL-based SDN-IoT.

2.1 Problem Definition

A cross-layer control of data flows (CORAL) was an SDN-inspired RPL routing protocol that works upon the ETX parameter [24]. DODAG Information Objective (DIO) broadcast was handled by doubling the message time at regular intervals. This work fails in attaining better performance in parent selection. a single OF (ETX) is considered which is not optimal, because the node could be poor at the energy level and other conditions. A versatile out-of-band (VERO-SD) controls the network topology and uses the Dijkstra algorithm for shortest path selection [25]. The congestion management was enabled by a threshold-based approach. In this work, only a single controller is used which decides path selection. In RPL-IoT most of the devices require selecting a path and hence using only one controller takes a longer time. Due to this, the transmission delay occurs and even the packets may be dropped. The routing in the controller is performed by the Dijkstra algorithm which is a blind search algorithm that is not able to find the shortest path in all the instances. In the case of sensitive data, then it cannot reach its destination at a prompt time. The dynamic threshold is determined based on the communication range which is not the only constraint to defining the broadcast threshold. Since not all the broadcasts are within the time they will fail.

3 System Model

In this section, we present the overall process of the proposed system.

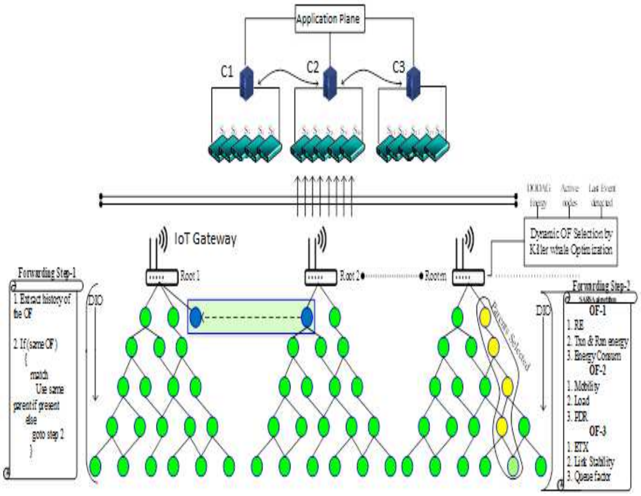

Fig. 2 Proposed TriOF Model.

The proposed model integrates IoT and the SDN environment for energy-efficient processing. This architecture is composed of n number of IoT sensors $N_1, N_2, .., N_n$, m number of IoT gateways $[G_1, G_2, .., G_m]$, l number of OpenFlow switches $[S_1, S_2, .., S_l]$ and multiple k numbers of controllers $[C_1, C_2, .., C_k]$. The IoT sensors are responsible for capturing the data and forwarding it to the root node. In this work, RPL is used, which constructs DODAG in which an objective function is defined to select the parent as a forwarder. The sensed data is forwarded toward the root, i.e., the IoT gateway, in this proposed work. The overall architecture of the model is illustrated in Figure 2. After the construction of DODAG, the routing is performed by using the objective function. In this work, Tri-OF is introduced (i.e.) three categories of OFs are formulated. At each time, the source node N_{Src} selects an OF based on the current network status. In RPL-based networks, OF plays an important role, and each node needs to compute OF each time. Considering more parameters in a single OF increases computational overhead, and the OF requirements vary over the network states. Thus, an adaptive OF selection procedure is presented with the aid of an optimization algorithm. Data forwarding is performed in two steps. In the first step, the N_{Src} determines the optimal OF for current parent selection. We have proposed KWO, which is the bio-inspired algorithm for optimal OF selection in each route selection. KWO is inspired by the behavior of killer whales [26] which works better than other benchmark optimization algorithms. Mainly, KWO resolves local optimal solution problems by searching population in clusters called Matriline. At first, the solutions (OF_1, OF_2, OF_3) are initialized as population. In each matriline, a leader whale is selected to search for the optimal solution. All other whales presented in that matriline are considered members. The leader whale is responsible for finding prey (optimal OF) direction, and the members need to chase the prey by updating their locations. In this work, each matriline is considered a DODAG, and optimal OF is selected for all DODAGs by searching within the search space. The proposed KWO algorithm involves the following steps,

1) In the first step, all members in the matriline scan the prey and the leader selects potential prey for chasing. The potentiality of the prey is evaluated in terms of fitness function $(\mathcal{F}n)$. For i^{th} prey (P_i), the Fn_i is computed as follows,

$$Fn_i = \{E(\mathbb{D}), \mathbb{AN}, \mathcal{L}\} \tag{1}$$

Fitness is evaluated as the function of DODAG energy level $(E(\mathbb{D}))$, the number of active nodes $(\mathbb{AN})$, and the last even detection time difference $(\mathcal{L})$. Each OF has a certain effect on the fitness function. If $E(\mathbb{D})$ and $\mathbb{AN}$ are low, then the difference between the latest event detection time and the current time is computed. The $\mathcal{L}$ is also low. Then the DODAG demands energy-efficient OF to minimize energy consumption. In this way, the potential prey is selected by the head.

2) Next, the member whales move the position following the selected potential prey. First, the position of the prey is modeled based on the depth of the prey $(\mathfrak{D}_P)$, depth of the leader $(\mathfrak{D}_L)$ and the angle of the horizontal range (θ). This angle is determined from the following expression,

$$\theta = \sin^{-1}\left(\frac{\mathfrak{D}_P - \mathfrak{D}_L}{R_{P,L}}\right) \tag{2}$$

Here, $R_{P,L}$ denotes the range between the prey and the leader. Once the position of the prey is determined, then all members move toward the prey's position. This movement of members is formulated as follows,

$$\begin{cases} \overrightarrow{v_L} \leftarrow \overrightarrow{v_L} + \vec{U}(0,\sigma_1) \otimes \left(\overrightarrow{\mathrm{P_{best}}} - \overrightarrow{x_L}\right) + \vec{U}(0,\sigma_2) \otimes \left(\overrightarrow{\mathrm{P_g}} - \overrightarrow{x_L}\right) \\ \overrightarrow{x_L} \leftarrow \overrightarrow{x_L} + \overrightarrow{v_L}.t \end{cases}, \tag{3}$$

The movement is modeled from the current position of a killer whale (x_L), previous best position $(\mathrm{P_{best}})$ and velocity (v_L) at a given time t. By following this model, all members move toward the prey.

3) In this step, the best solution search is continued. The matriline creates a search pattern based on previous prey. In this step, the leader computes Fn for searched prey. If this prey has more potential than the previous prey, then the leader changes the entire movement toward new potential prey. Else, the members continue to chase the old prey.

4) If the stopping criteria have been met, then the optimal solution determined in steps 2 & 3 is provided as the best solution. Else, both steps are executed repeatedly to obtain the optimal solution.

At the end of KWO, the optimal OF that is suitable for the current network state. The three objective functions are formulated as follows,

$$OF_1 = E_{re} - \sum E_{Tx}, E_{rx}, E_c \tag{4}$$

The OF_1 is computed based on residual energy (E_{Re}), transmission energy (E_{Tx}), reception energy (E_{Rx}) and energy consumption (E_C). The transmitter and receiver energy are modeled as follows,

$$E_{Tx} = \mathfrak{B}\left(E_{elec} + \epsilon_{amp} * d^2\right) \tag{5}$$

$$E_{rx} = \mathfrak{B}(E_{da} + E_{elec}) \quad (6)$$

Here, $\mathfrak{B}$ is the number of bits per packet, d is the distance between the source and candidate parent node, E_{da} defines energy consumed for data aggregation and ϵ_{amp} is the multi-path fading signal amplification coefficient. In this work, OF_1 is mainly formulated to assure energy efficiency. If OF_1 is selected as the optimal objective function, then the node which has a higher E_{Re} and lower E_{Tx}, E_{Rx}, E_C is selected as the optimal parent node. Similarly, OF_2 is formulated as follows,

$$OF_2 = \frac{DR}{M+Ld} \quad (7)$$

The second OF is formulated when the DODAG has sufficient energy for further operations. This objective mainly focuses on maximizing the data delivery rate. Thus, a node with a high delivery ratio (DR) and low mobility (M), and load (Ld) is selected as the optimal parent node. The third OF is formulated as follows,

$$OF_3 = \frac{\psi}{ETX+\alpha} \quad (8)$$

This OF is formulated based on link stability (ψ), ETX, and queue factor (α). The N_{Src} first finds the optimal OF for current data transmission.

Pseudocode for KWO-based parent selection

Initialize $\{\mathrm{OF}_1, \mathrm{OF}_2, \mathrm{OF}_3\}$
Initialize Population
Form matriline
For each prey (P_i)
  Compute Fn_i
  Find potential prey // By Leader
  Chase potential prey // By Members
  Determine Prey Position (θ)
  Update the position of Whales x_L
  Move to new prey (P_{new})
  Compute Fn_{new}
  Compare $Fn_i \&\& Fn_{new}$
  If ($Fn_{new} > Fn_i$)
    Set P_{new} →Potential.
  Update position
  Else
    Continue with P_i
  End **If**
  While Stopping Criterion Met
    Return ($\mathrm{OF}_{\mathrm{current}}$)
  End **While**
End **For**
If ($\mathrm{OF}_{\mathrm{current}} = \mathrm{OF}_{\mathrm{Previous}}$)
  Extract history
  Set Previous Parents as Optimal
Else
  Select Optimal Parent as per $\mathrm{OF}_1/\mathrm{OF}_2/\mathrm{OF}_3$
  Transmit data
End **If**
End

In the first step, the optimal OF is selected by N_{Src} is compared with the previous best OF. For that, N_{Src} extracts the history of prior OFs. Then, it compares the current OF with the previous OF to speed up the parent selection process. If the last OF and the current OF are the same, then the N_{Src} considers previous optimal parents as current optimal parents and checks whether the previous parents are available. If the previous parents are available, data transmission is performed through these optimal parents. In the next step, the following processes are performed,

- If the current and previous OFs are the same and parents are unavailable, the source selects optimal parents for current data transmission.
- If the current and previous OFs are not the same, the source node selects optimal parents according to the current OF for data transmission.

The pseudocode for proposed adaptive OF selection and parent selection explains the step-by-step procedure of proposed RPL operations. In this manner, an optimal route is selected between N_{Src} and corresponding G. After aggregating data from all sensor nodes in the DODAG, G forwards the data through OpenFlow switches deployed in the data plane.

4 Simulation Results and Analysis

In this section, we evaluate the performance of the proposed model through extensive simulations. Also, the performance is compared with the existing works.

4.1 Simulation Environment

The proposed model is experimentally analyzed for evaluating the performance. For that, the proposed network is modeled in Network Simulator NS3 which runs on the Ubuntu operating system. NS3 is suitable for simulating various types of networks and network protocols. Thus, NS3 is used for simulations. All algorithms have been written in C++ and the modules are built in the Python programming language. The simulation parameters used in the network model are listed in Table 2. By using these parameters, the network is constructed and simulated.

Table 2 Simulation Parameters.

Parameter	Value
Simulation Area	1000*1000 m
Number of IoT Nodes	100
Number of Gateways	3
Number of OpenFlow Switches	15
Number of Controllers	3
Initial Energy of IoT Nodes	15 Joules (Maximum)
Flow Table Size	1000 KB
Packet Size	512 KB (Maximum)
Number of Packets Generated	100
Data Rate	1.1 Mbps
Simulation Time	100 Minutes
Modules	IoT_Module Flow_Monitor_Module WiFi_Module OpenFlow_Module

KWO Configuration	Number of Matriline	10-50
	Initial Population	100
	Number of Leaders	10
	Maximum Iteration	100

In the simulation, TCP and UDP traffic types are generated by the IoT nodes. As the network is heterogeneous, each node moves with different mobility and generates data in different sizes.

4.2 Comparative Analysis

After simulation, the results are observed for comparative analysis. For that, we observe significant performance measures such as the number of control messages exchanged, control message overhead, average energy consumption, packet delivery ratio, and packet loss rate. Since the work focuses on the RPL aspect and SDN aspect, comparisons are made with RPL-based works including VERO-SD [25] and CORAL [24].

Table 3 Comparison of previous works.

Existing Work	Network Model	OF	Demerits
CORAL [24]	RPL-based SDN-IoT	ETX	• Increases data loss and energy consumption. • Not able to handle large-scale network
VERO-SD [25]	RPL-based SDN-IoT	Dijkstra RPL routing	• High E2E delay • Non-optimal route increases data loss

In Table 3, the existing works are compared with the proposed model. From the analysis, existing works have some demerits which degrade the performance. These demerits reflect as the results in performance metrics.

4.2.1 Comparison of Control Messages

The prime objective of the proposed system is to minimize energy consumption by minimizing unwanted control packet overhead. RPL-based operations require a greater number of control packets in DODAG formation and stabilization. Thus, control message analysis plays a vital role in the proposed work. The control message analysis is carried out based on two vital metrics the number of control packets exchanged and the control message overhead.

In Figure 3, the number of control messages in DODAG construction is compared in terms of the number of IoT nodes. An increase in the number of IoT nodes requires many control messages to be exchanged. The reason is that when the network is large then the broadcasting message will be high. Even with 100 nodes, our proposed model uses only 61 control messages while the VERO-SD method required 89 messages, and the CORAL method requires 101 messages. In the VERO-SD approach, the broadcast limit is defined based on a threshold value. However, the threshold value is insufficient to define the broadcast limit. Similarly, CORAL doubles the broadcast limit at regular time intervals. Due to these reasons, a large

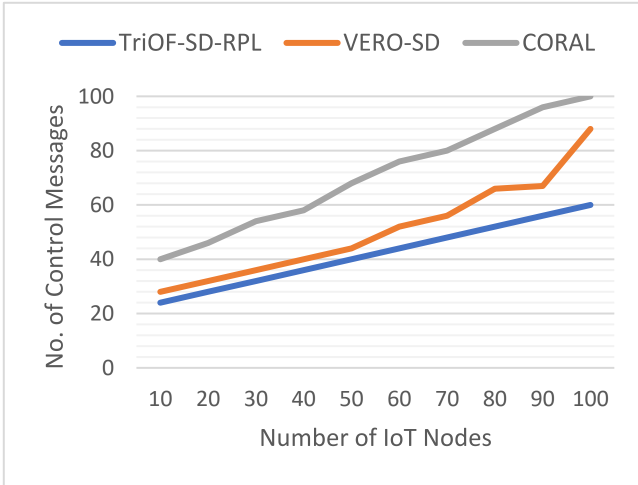

Fig. 3 Number of Control Messages Exchanged.

many control packets are exchanged in the existing works. At the same time, the proposed work uses adaptive decision which controls the broadcast.

4.2.2 Comparison of Control Overhead

In Figure 4, the control message overhead is compared concerning simulation time. The control overhead is defined as the ratio between the total number of transmitted packets and the total number of transmitted control packets. In the proposed model, the control overhead is minimized by 0.52 when n=100. For the same n value, existing VERO-SD achieve 0.96 and CORAL achieves overhead as 1 (i.e.) a greater number of control packets are exchanged throughout the network. However, exchanging many control packets is not suitable for the IoT network since it reduces the reliability of the entire network.

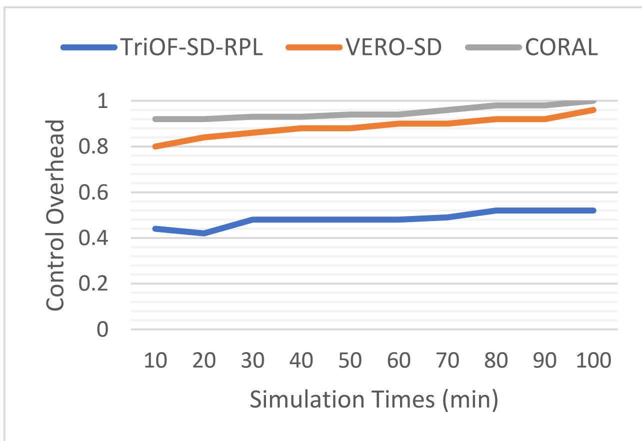

Fig. 4 Comparison of control overhead.

On the other hand, exchanging excess control packets consumes more energy in each IoT node.

In the proposed model, the DIO broadcast is broadcasted when there is a greater number of nodes needed to receive the control message. In VERO-SD, the broadcast control is carried based on a threshold value, but the threshold value is determined non-optimally. Due to this factor, the overhead is 0.8 even when n=10. Similarly, CORAL fixes a time interval and doubles the DIO broadcast in a regular interval which increases the overhead exponentially to 0.92 for n=10. From the analysis, the proposed approach minimizes the overhead which will further minimize energy consumption.

4.2.3 Comparison of Energy Consumption

Energy consumption is defined as the amount of energy consumed by the network on average. Energy consumption includes energy consumed for data transmission, reception, and environmental sensing.

In Figure 5, the energy consumption is compared concerning the number of IoT nodes. The average energy consumption of the network is increased with an increase in simulation time. At the simulation time of 10 minutes, the energy consumption of our model is 6J and it is increased to 18J at the end of the simulation (i.e.) 100 minutes. That is 18J energy is dissipated throughout the network.

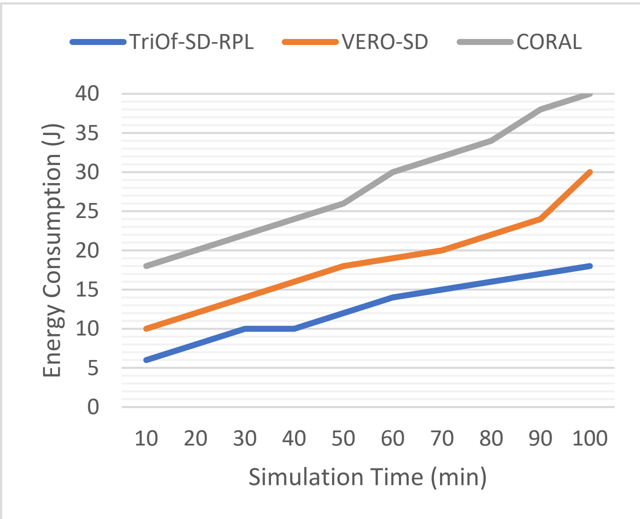

Fig. 5 Comparison of energy consumption.

At the same time, 30J energy is dissipated in VERO-SD, and 40J energy is dissipated in CORAL, which is near twice the time higher than the proposed work. In IoT devices, energy dissipation is mainly caused by sensing and transmitting the data. Besides, RPL-based networks dissipate energy for DODAG construction and control packet exchanges. Thus, optimizing control packet overhead and frequent retransmission minimizes energy dissipation. However, the existing approaches fail to assure minimized energy dissipation due to high control packet overhead.

CORAL uses ETX as the metric, which leads to selecting an energy-minimized node as a parent many times (i.e.), energy consumption at a particular node is increased exponentially. Also, control packets are transmitted without optimal limiting function. This leads to 40J energy consumption in the network. On the other hand, VERO-SD uses single OF and threshold-based broadcast control, which fails to achieve optimal routing and broadcast minimization. Therefore, the energy consumption of the CORAL and VERO-SD approaches is higher than the proposed work since the proposed work minimizes the control message exchange and selects the optimal route by adaptive OF selection procedure.

4.2.4 Analysis of Data Transmission

Data transmission efficiency is analyzed in terms of packet delivery ratio and packet loss rate. The packet delivery ratio is defined as the ratio between the total number of packets transmitted from the source and the total number of packets received by the destination.

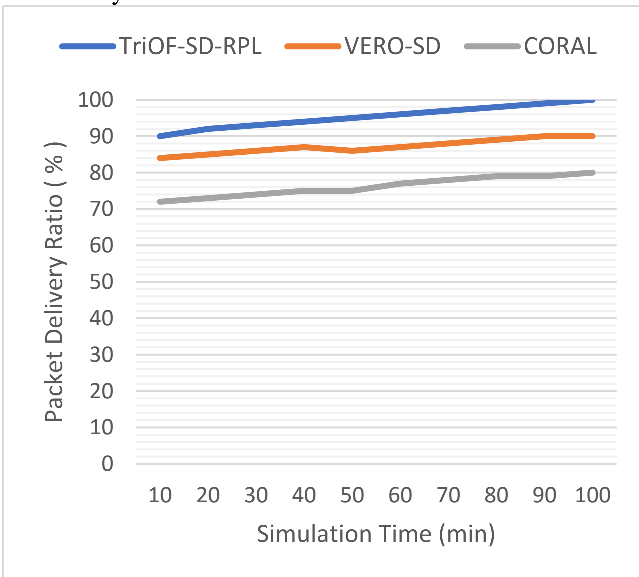

Fig. 6 Comparison of packet delivery ratio.

Similarly, the packet loss rate measures the total number of packets lost during data transmission. In Figure 6 and Figure 7, the packet delivery ratio and packet loss rate are compared, respectively.

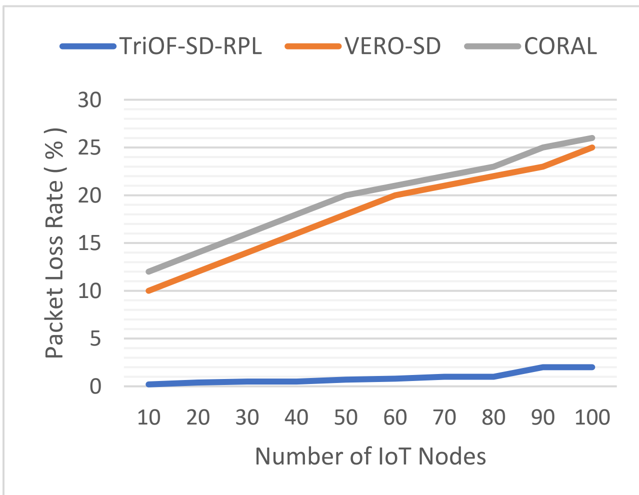

Fig. 7 Comparison of packet loss rate.

Here, the PDR achieved by the proposed work is between 90% to 100% (i.e.) a reasonable number of packets reached the destination without any loss. Also, the packet loss rate is minimized to 2%. The proposed work optimizes the overall data transmission using the optimal parent selection based on

the optimal OF for the current network status, which enhances the data transmission. Each parameter involved in the TriOF assists in improving data transmission. Thus, 100% of packets reach the destination without any loss.
On the other hand, VERO-SD uses the Dijkstra algorithm for routing, which relies on a single OF. CORAL uses ETX metric for optimal routing. However, both methods lack parameter consideration which leads to a huge packet loss. Also, both works use SDN without proper management. Thus, overloading at the data or control plane reflects the data loss in VERO-SD and CORAL methods.

Table 4 Summary of Comparison analysis with 100 nodes.

Work	Control Messages	Control Overhead	Energy Consumption
CORAL [24]	101	1	40J
VERO-SD [25]	89	0.96	30J
Proposed	61	0.52	18J

Table 4 summarized the comparison analysis of the previous works compared with the proposed model with 100 nodes and the energy consumption is measured at the end of the simulation (i.e.) 100 minutes.

5 Results & Discussion

The results show that the proposed model outperforms all energy efficiency aspects in terms of diverse performance metrics. Each of the proposed contributions helps in improving the Energy efficiency performance of the software-defined IoT networks. The major research highlights are in SD-RPL, the parent selection, first, the OF is decided using killer whale optimization with energy, active nodes, and last event time, it predicts the optimal OF for each region. Then the selection OF will be incorporated with the computation of the OF parameters defined into energy-based, load-based, and link-based metrics.

6 Conclusion

In this paper, a software-defined RPL network model is designed and simulated to improve the energy efficiency of IoT networks. Optimal routing is enabled by the formulation of TriOF in which each OF focuses on the different network conditions. Each time, the OF is adaptively selected by using the KWO algorithm based on multiple metrics. The proposed model achieves better performance in energy efficiency. It is also evaluated in the NS3 simulator based on performance metrics such as the number of control messages, control overhead, and energy consumption. In all aspects, the proposed work shows better performance.
In the future, the proposed model will be extended with data and control plane load balancing approach to minimize retransmission in the IoT sensor plane and give a better improvement in energy efficiency. Thus, achieving energy efficiency through data and control plane load balancing provisioning is the better future research direction of this proposal.

Funding Statement: This research is done with financial support from the Deanship of Scientific Research at King Khalid University under research grant number **(R.G.P.2/246/44).**
Competing Interests: The authors declare that they have no competing interests.
Ethics Statement: No unethical work has been performed in this research work.

References

[1] J. H. Nord, A. Koohang, and J. Paliszkiewicz, "The Internet of Things: Review and theoretical framework," *Expert Systems with Applications,* vol. 133, pp. 97-108, 2019/11/01/ 2019, doi: https://doi.org/10.1016/j.eswa.2019.05.014.

[2] Z. Huang, X. Xu, J. Ni, H. Zhu, and C. Wang, "Multimodal Representation Learning for Recommendation in Internet of Things," *IEEE Internet of Things Journal,* vol. 6, no. 6, pp. 10675-10685, 2019, doi: 10.1109/JIOT.2019.2940709.

[3] P. Asghari, A. M. Rahmani, and H. H. S. Javadi, "Internet of Things applications: A systematic review," *Computer Networks,* vol. 148, pp. 241-261, 2019/01/15/ 2019, doi: https://doi.org/10.1016/j.comnet.2018.12.008.

[4] P. K. Khatua, V. K. Ramachandaramurthy, P. Kasinathan, J. Y. Yong, J. Pasupuleti, and A. Rajagopalan, "Application and assessment of internet of things toward the sustainability of energy systems: Challenges and issues," *Sustainable Cities and Society,* vol. 53, p. 101957, 2020/02/01/ 2020, doi: https://doi.org/10.1016/j.scs.2019.101957.

[5] E. Molina and E. Jacob, "Software-defined networking in cyber-physical systems: A survey," *Computers & Electrical Engineering,* vol. 66, pp. 407-419, 2018/02/01/ 2018, doi: https://doi.org/10.1016/j.compeleceng.2017.05.013.

[6] J. Xie *et al.*, "A Survey of Machine Learning Techniques Applied to Software Defined Networking (SDN): Research Issues and Challenges," *IEEE Communications Surveys & Tutorials,* vol. 21, no. 1, pp. 393-430, 2019, doi: 10.1109/COMST.2018.2866942.

[7] A. H. Mohammed, K. R. M, M. k. Hussein, and I. A. Abdulateef, "A Review Software Defined Networking for Internet of Things," in *2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, 26-28 June 2020 2020, pp. 1-8, doi: 10.1109/HORA49412.2020.9152862. [Online]. Available: https://doi.org/10.1109/HORA49412.2020.9152862

[8] P. W. Tsai, C. W. Tsai, C. W. Hsu, and C. S. Yang, "Network Monitoring in Software-Defined Networking: A Review," *IEEE Systems Journal,* vol. 12, no. 4, pp. 3958-3969, 2018, doi: 10.1109/JSYST.2018.2798060.

[9] J. Lu, Z. Zhang, T. Hu, P. Yi, and J. Lan, "A Survey of Controller Placement Problem in Software-Defined Networking," *IEEE Access,* vol. 7, pp. 24290-24307, 2019, doi: 10.1109/ACCESS.2019.2893283.

[10] M. Karakus and A. Durresi, "A survey: Control plane scalability issues and approaches in Software-Defined Networking (SDN)," *Computer Networks,* vol. 112, pp. 279-293, 2017/01/15/ 2017, doi: https://doi.org/10.1016/j.comnet.2016.11.017.

[11] H. Kharrufa, H. A. A. Al-Kashoash, and A. H. Kemp, "RPL-Based Routing Protocols in IoT Applications: A Review," *IEEE Sensors Journal,* vol. 19, no. 15, pp. 5952-5967, 2019, doi: 10.1109/JSEN.2019.2910881.

[12] B. Safaei, A. A. Mohammad Salehi, A. M. Hosseini Monazzah, and A. Ejlali, "Effects of RPL objective functions on the primitive characteristics of mobile and static IoT infrastructures," *Microprocessors and Microsystems,* vol. 69, pp. 79-91, 2019/09/01/ 2019, doi: https://doi.org/10.1016/j.micpro.2019.05.010.

[13] K. S. Bhandari, I. Ra, and G. Cho, "Multi-Topology Based QoS-Differentiation in RPL for Internet of Things Applications," *IEEE*

Access, vol. 8, pp. 96686-96705, 2020, doi: 10.1109/ACCESS.2020.2995794.

[14] Z. Shah, A. Levula, K. Khurshid, J. Ahmed, I. Ullah, and S. Singh, "Routing Protocols for Mobile Internet of Things (IoT): A Survey on Challenges and Solutions," *Electronics,* vol. 10, no. 19, p. 2320, 2021. [Online]. Available: https://www.mdpi.com/2079-9292/10/19/2320.

[15] H. Lamaazi and N. Benamar, "A comprehensive survey on enhancements and limitations of the RPL protocol: A focus on the objective function," *Ad Hoc Networks,* vol. 96, p. 102001, 2020/01/01/ 2020, doi: https://doi.org/10.1016/j.adhoc.2019.102001.

[16] S. S. Solapure and H. H. Kenchannavar, "Design and analysis of RPL objective functions using variant routing metrics for IoT applications," *Wireless Networks,* vol. 26, no. 6, pp. 4637-4656, 2020/08/01 2020, doi: https://doi.org/10.1007/s11276-020-02348-6.

[17] H. Lamaazi and N. Benamar, "OF-EC: A novel energy consumption aware objective function for RPL based on fuzzy logic," *Journal of Network and Computer Applications,* vol. 117, pp. 42-58, 2018/09/01/ 2018, doi: https://doi.org/10.1016/j.jnca.2018.05.015.

[18] M. H. Homaei, E. Salwana, and S. Shamshirband, "An Enhanced Distributed Data Aggregation Method in the Internet of Things," *Sensors*, vol. 19, no. 14, doi: https://doi.org/10.3390/s19143173.

[19] A. Saleem, M. K. Afzal, M. Ateeq, S. W. Kim, and Y. B. Zikria, "Intelligent learning automata-based objective function in RPL for IoT," *Sustainable Cities and Society,* vol. 59, p. 102234, 2020/08/01/ 2020, doi: https://doi.org/10.1016/j.scs.2020.102234.

[20] K. S. Bhandari and G. H. Cho, "A Resource Oriented Route Selection Framework Using Contextual Information Based on Fuzzy Logic," *Electronics*, vol. 8, no. 9, doi: https://doi.org/10.3390/electronics8091023.

[21] J. Kniess and V. Marques, "MARPL: A crosslayer approach for Internet of things based on neighbor variability for mobility support in RPL," *Transactions on Emerging Telecommunications Technologies,* vol. 31, p. e3931, 12/01 2020, doi: https://doi.org/10.1002/ett.3931.

[22] B. Behrouz Vaziri and A. Toroghi Haghighat, "Brad-OF: An Enhanced Energy-Aware Method for Parent Selection and Congestion Avoidance in RPL Protocol," *Wireless Personal Communications,* vol. 114, no. 1, pp. 783-812, 2020/09/01 2020, doi: https://doi.org/10.1007/s11277-020-07393-0.

[23] A. Musaddiq, Y. Zikria, Z. Nain, and S. W. Kim, "Routing protocol for Low-Power and Lossy Networks for heterogeneous traffic network," *EURASIP Journal on Wireless Communications and Networking,* vol. 2020, 01/20 2020, doi: https://doi.org/10.1186/s13638-020-1645-4.

[24] G. Violettas, S. Petridou, and L. Mamatas, "Evolutionary Software Defined Networking-Inspired Routing Control Strategies for the Internet of Things," *IEEE Access,* vol. 7, pp. 132173-132192, 2019, doi: https://doi.org/10.1109/ACCESS.2019.2940465.

[25] T. Theodorou and L. Mamatas, "A Versatile Out-of-Band Software-Defined Networking Solution for the Internet of Things," *IEEE Access,* vol. 8, pp. 103710-103733, 2020, doi: https://doi.org/10.1109/ACCESS.2020.2999087.

[26] T. R. Biyanto *et al.*, "Killer Whale Algorithm: An Algorithm Inspired by the Life of Killer Whale," *Procedia Computer Science,* vol. 124, pp. 151-157, 2017/01/01/ 2017, doi: https://doi.org/10.1016/j.procs.2017.12.141.