MARS: Multi-radio Architecture with ML-powered Radio Selection for Mesoscale IoT Applications

Jothi Prasanna Shanmuga Sundaram*, Arman Zharmagambetov*, Magzhan Gabidolla*,
Miguel Á. Carreira-Perpiñán*, Alberto E. Cerpa*
*Computer Science and Engineering, University of California, Merced, CA 95343, USA

***Abstract*—IoT is rapidly expanding from traditional small-scale (0–100m) applications like smart homes and large-scale (1–5km) applications like Microsoft's FarmBeats to emerging mesoscale (0.1–1.5km) applications such as smart-grid NANs and peer-to-peer energy trading in smart homes. These applications demand high throughput and low latency but currently lack dedicated radio technologies. Our qualitative analysis identified Zigbee and LoRa as promising candidates. Further quantitative analysis revealed that a multi-radio architecture combining these radios achieves the best throughput. However, within the 500–1200m range, termed the *gray region*, it is unpredictable which radio offers higher throughput at any given moment. To address this, we developed MARS, a Multi-radio Architecture with Radio Selection, powered by TAO-optimized decision trees that select the high-throughput radio at the time of transmission. These decision trees require instantaneous path quality estimates, but traditional multi-hop Zigbee networks cannot provide these promptly due to propagation and queuing delays. We overcome this challenge by introducing Decision Tree-based updates to instantaneously estimate end-to-end path quality. Large-scale, real-world experiments with MARS demonstrated average throughput gains of 48.2% and 49.79% at two different locations.**

I. Introduction

Traditional IoT networks span small-scale (e.g., smart homes, 10–100m) and large-scale (e.g., Microsoft FarmBeats, 1–5km) applications [1], whereas emerging use cases—such as smart-grid Neighborhood Area Networks [2], target tracking [3], industrial automation [4], [5], and peer-to-peer energy trading [6], [7], [8] operate at intermediate distances of 0.1–1.5km, which we term mesoscale IoT applications. These applications often rely on mains power or large battery reserves [9], [10], enabling latency and throughput to serve as key performance indicators that implicitly capture reliability [11]. Unlike the relatively stable conditions of smart homes or agricultural deployments, mesoscale environments are urban and dynamic, with wireless links impacted by buildings, human activity, and vehicles. Despite their growing significance, these applications lack purpose-built radio solutions. To close this gap, we explored the performance of the available COTS radios for mesoscale IoT applications.

First, we conducted a qualitative analysis on the available IoT radios. This analysis identifies Zigbee and LoRa as potential candidates for mesoscale IoT applications. Further quantitative analysis of the radio candidates on both single-hop and multi-hop topologies show that Zigbee and LoRa achieve competitive throughput at the distance of 500-1200m from the gateway, termed the *gray-region*. In this *gray-region*, LoRa and Zigbee radios achieve high end-to-end throughput at different time instants due to erratic channel conditions. The fundamental finding is that Zigbee and LoRa can work together as a multi-radio system and efficiently switch radios to maximize throughput. However, instantaneously selecting a high-throughput radio is not a trivial problem because of erratic link quality dynamics in mesoscale IoT environments.

Second, we developed MARS to instantaneously select high-throughput radio during transmission, on a per-packet basis, using end-to-end path quality metrics. MARS's node comprises both Zigbee 2.4 GHz and LoRa 915 MHz radios. While it is intuitive to employ a Machine Learning (ML) model to perform radio selection, it entails the challenge of obtaining instantaneous path quality estimations. Multi-hop Zigbee radios propagate the quality information of all the links along the path to compute path quality [12]. However, the propagated link quality info expires before it reaches the destined node because of temporal link quality variations, propagation, and queuing delays along the path. While it is ideal to compute the path quality of the entire path, we observed that a part of the entire path length is sufficient to identify the high throughput radio. We performed extensive analysis and evaluation on multiple topologies, to find the fewer hops required to estimate the end-to-end path quality as input to our ML model and still obtain good radio selection accuracy. This way we balance the trade-off between perfectly accurate global metrics that cannot be gathered on time and acceptably accurate local metrics that can be collected on time while still providing good input for our ML model, trained by CART[13], to achieve acceptable accuracy.

Finally, we optimized the CART with Tree-Alternating Optimization (TAO) [18]. The TAO algorithm optimizes the traditional trees to achieve higher accuracy. The TAO-Optimized CART (TAO-CART) can be converted into IF...ELSE statements for efficient deployment on IoT end devices. A large-scale, real world evaluation of MARS on a complex mesh topology at two different locations showed that MARS can instantaneously select the high-throughput radio during transmission, achieving an average throughput gain of 48.2% and 49.79% than the competing single and multi-radio systems.

In summary, the contributions of our work are:
(i) Identifying the absence of a dedicated radio for emerging mesoscale IoT apps, we developed an intelligent multi-radio architecture with COTS IoT radios, Zigbee and LoRa, with an order of magnitude difference in theoretical throughput.
(ii) Through experimental analysis, we identified the existence of a *gray-region* between 0.5-1.2 Km from the gateway where

The authors acknowledge Busher Bridi for his support during experiments. This work is partially supported by NSF award IIS–2007147.

Table I: A comparison of Related Work with MARS

Multi-radio systems	Radios Used	Optimized Metric				Implemented Range >500m?	Radio Selector	Mobility?
		Energy	Reliability	Throughput	Latency			
Backpacking [14]	802.11 + 802.15.4	✔	✗	✗	✗	✗	✗	✗
Kusy et al. [15]	802.15.4 (915MHz) + 802.15.4 (2.4GHz)	✗	✔	✗	✗	✗	✗	✗
Gummeson et al. [16]	802.15.4 (915MHz) + 802.15.4 (2.4GHz)	✔	✗	✗	✗	✗	Reinforcement Learning	✔
Lymberopoulos et al. [17]	802.11b + 802.15.4	✔	✗	✗	✗	✗	Threshold-based	✗
MARS	802.15.4 (2.4GHz) + LoRa LPWAN	✗	✗	✔	✔	✔	Decision Trees	✗

Table II: Qualitative comparison of radios in the context of mesoscale IoT showing the suitability of Zigbee and LoRa.

	SigFox	WiFi HaLow	BLE	Zigbee	LoRa
Open-source?	No	Yes	Yes	Yes	Yes
Link Budget (dB)	58 [19]	24.5 [20]	108 [21]	103 [22]	150 [23]
Topology Type	LPWAN	LPWAN	PAN	PAN/LAN	LPWAN
Communication Range (m)	5000	1000	100	125 [24]	5000
Max bitrate (bps)	600	upto 4M	up to 1M	250K	upto 27K

it is uncertain which radio provides the best throughput at any given time due to the temporal link quality variations.

(iii) We developed and implemented CART tree model optimized with Tree Alternating Optimization (TAO) for radio selection. To the best of our knowledge, this is the first real-world use case of the TAO algorithm.

(iv) We showed that partial path quality can provide sufficient information for our TAO-optimized tree to accurately select the high-throughput radio during transmission.

II. Related work

Multi-radio wireless networks have been extensively explored using different radio combinations. Backpacking [14] was developed for high data rate sensor networks. Kusy et al. [15] developed a multi-radio architecture for WSN. They show that employing two multi-hop radios in the same node improves reliability with 3-33% energy overhead. Gummeson et al. [16] optimize energy consumption by employing a Reinforcement Learning (RL) based adaptive link layer to switch radios (CC2420+XE1205) based on channel dynamics. Lymberopoulos et al. [17] (Zigbee+WiFi) switch radios with a threshold-based algorithm to optimize energy efficiency. A comparison of related work with MARS is tabulated in Table I. MARS is the only work that optimizes throughput and latency for a deployment range greater than 500m.

III. IoT radios for mesoscale applications

A qualitative analysis of the available COTS IoT radios is tabulated in Table II to identify suitable candidates for mesoscale IoT applications. The closed-source SigFox radio is unsuitable for private deployments. WiFi HaLow's low link budget signals cannot penetrate environmental obstacles. PAN topology characteristic of BLE cannot cover the entire mesoscale range. Zigbee and LoRa satisfy all the requirements of mesoscale IoT apps. So, Zigbee and LoRa, following US region standards, are chosen for further quantitative analysis.

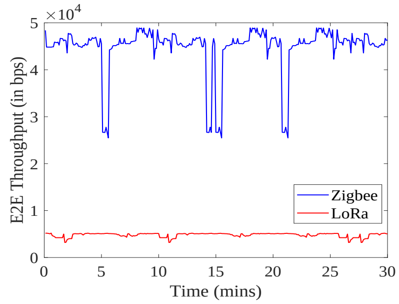
An experimental quantitative analysis of the chosen Zigbee [22] and LoRa [25] radios is conducted in two folds: single-hop and multi-hop experiments. For the ***Single-hop experiments,*** A sender and receiver of the radio candidates are placed in both free space and urban-like environments at a distance of ≈20m from each other. The latter spans multiple wooden walls, a glass door, and three humans between the nodes. One thousand 29-byte packets are transmitted from the sender to the receiver to average the achieved throughput and Packet Loss Rate (PLR). Zigbee motes are programmed using TinyOS [26] where CSMA and Link-layer ack's are disabled. There is no reliability mechanism for retransmitting the packets. The results are tabulated in Table III. First, experiments were conducted using a Zigbee radio operating at 2.4 GHz. In a free-space environment, Zigbee achieved a 0 PLR with an average throughput of 77.63 Kbps. However, in an urban-like environment, the throughput dropped to 56.53 Kbps with 33.40% PLR. This performance degradation is attributed to increased packet loss and reduced signal quality due to environmental obstructions. The higher PLR in urban settings results from signals being lost or corrupted, while the reduced throughput is due to signal attenuation caused by penetration through obstacles. An intuitive idea would be to utilize Zigbee radio at a lower frequency to improve the signal penetration capacity. So, we experimented with Zigbee radios operating at 915 MHz. It achieves 0 PLR in free space with an average throughput of 9.53 Kbps. It achieves an average throughput of 6.7 Kbps with 24.60% PLR in an urban-like environment. LoStik [27] LoRa USB nodes using SF7 in 125KHz bandwidth were used for testing LoRa. LoRa was able to achieve 4.58 Kbps average throughput with a 100% reception ratio in both free-space and urban-like environments. LoRa's penetration capacity is higher because of its robust modulation scheme [25], generating high link budget signals.

Table III: Throughput and PLR of Zigbee and LoRa radios

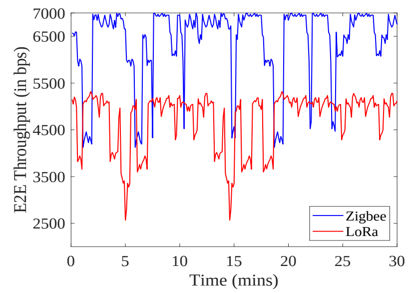
Radio candidate	Throughput (bps)	PLR (%)
Zigbee 2.4 GHz-free space	77,634	0
Zigbee 2.4 GHz-urban-like environment	56,530	33.40
Zigbee 915 MHz-free space	9,530	0
Zigbee 915 MHz-urban-like environment	6,777	24.60
LoRa 915 MHz-free space	4,579	0
LoRa 915 MHz-urban-like environment	4,579	0

Zigbee 2.4 GHz achieves higher throughput and PLR than Zigbee 915 MHz. This high throughput characteristic is highly desirable for mesoscale applications despite the higher PLR, since throughput will further degrade in multi-hop communications due to queueing and channel detection delays [28]. If we choose Zigbee 915 MHz because of a lower loss rate, its lower throughput will further degrade when employed in a multi-hop fashion [28]. So, we choose Zigbee 2.4GHz over Zigbee 915MHz. Comparing LoRa 915MHz and Zigbee 915MHz, LoRa 915 MHz achieves lower PLR than Zigbee 915 MHz as LoRa signals have a stronger link budget. Hence, Zigbee 2.4GHz is better for apps demanding multi-hop communications while LoRa is better for apps demanding long-range, single-hop communications. All the further experiments employ Zigbee 2.4GHz, LoRa 915MHz radios.

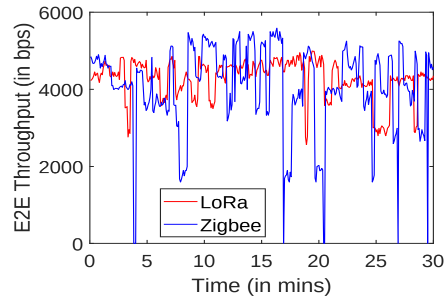
Multi-hop experiments: The multi-radio nodes shown in Fig. 2 are place in a simple line topology for these exper-



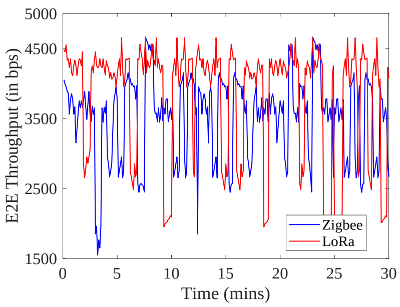
(a) TF at 100m



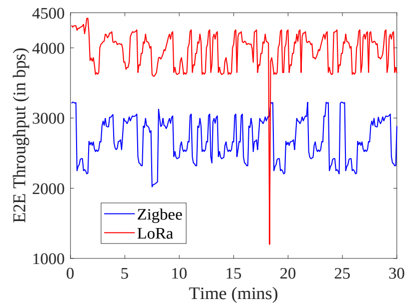
(b) TF at 500m



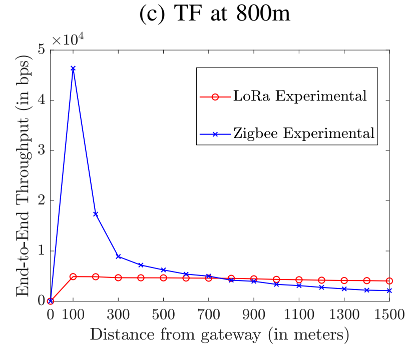
(c) TF at 800m



(d) TF at 1200m



(e) TF at 1400m



(f) Average throughput at different distances

Figure 1: Throughput Fluctuations (TF) at different distances from the gateway

Figure 2: Multi-radio node that contains both USB-based LoRa and Zigbee radios hosted by Raspberry Pi 3B.

iments. One gateway and fifteen nodes are placed in a line topology in free space such that each hop spans approximately 100m. Each node has both LoRa and Zigbee radios as shown in Figure 2. 29-byte sized packets are concurrently transmitted by both radios at the rate of 1 packet every 3 seconds. LoRa can reach the gateway in a single hop, whereas Zigbee takes multiple hops. LoRa nodes transmit in a 125KHz channel with spreading factor SF7. The LoRa gateway is capable of receiving eight packets concurrently [25]. This reduces collisions. Link-level acknowledgments are disabled for Zigbee radios and they use CSMA MAC while LoRa uses ALOHA MAC. The nodes are placed in such a way that they have *connected links* [29] to the neighboring nodes.

Multi-hop result analysis: Fig. 1f depicts the End-to-End (E2E) throughput as a function of distance. This end-to-end throughput at a specific distance is the average of 1000 packets. The drastic difference between LoRa and Zigbee at 100m is due to the fact that only one node is transmitting without any contention. A considerable drop in Zigbee's throughput is seen from 100m-300m as there are three contenders. CSMA blocks two other links from transmitting to avoid collisions, allowing only 1 of 3 links to transmit at any given time until 300m. After 300m, Zigbee's throughput steadily decreases with distance because of CSMA blocking delay and queuing delay. LoRa's throughput does not show any drastic decrease, but it slightly decreases with distance. This is because the LoRa signals are robust enough to pass through an urban-like environment, as LoRa's CSS modulation makes the signal highly resistant to attenuation. Also, it should be noted that the LoRa gateway can receive and decode eight packets concurrently. Fig. 1f shows that Zigbee wins until 500m and LoRa wins after 1200m. LoRa and Zigbee achieve competitive throughput between 500m- 1200m from the gateway, termed the *gray region*.

End-to-end throughput fluctuations in the gray region: Fig. 1f shows that LoRa and Zigbee achieves competitive throughput between 500m-1200m from the gateway. This showed the average E2E throughput of a thousand packets at different distances. This does not depict the end-to-end throughput fluctuations of different packets over time. So, we conducted an in-depth analysis of end throughput fluctuations over time at different distances from the gateway.

Figs.1a - 1e show the throughput fluctuations for 30 minutes between 100m to 1400m from the gateway. Zigbee wins at 100m (Fig.1a) and LoRa wins at 1400m (Fig.1e). At 500m from the gateway, Zigbee achieves higher throughput most of the time. Whenever Zigbee's throughput is falling, LoRa is able to back up Zigbee to provide better throughput. The difference in throughput of Zigbee and LoRa is considerably higher at 500m. Fig. 1c shows the throughput fluctuations at 800m. Zigbee mostly wins, but the throughput of Zigbee is

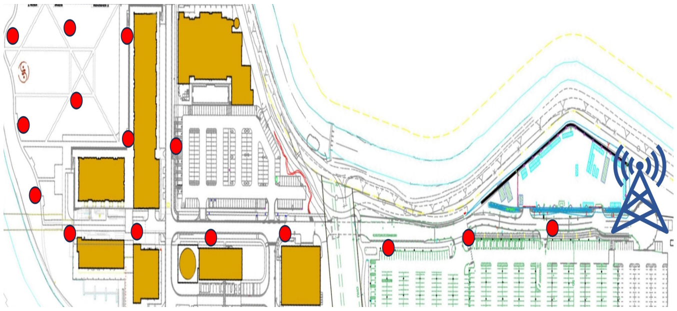


Figure 3: Mesh topology - nodes populated at the gray region.

highly fluctuating as packets are experiencing multiple hops. The average throughput of Zigbee and LoRa is very close to each other. Fig. 1d shows that Zigbee experiences low E2E throughput. So, it mostly underperforms at this longer distance of 1200m from the gateway. The fundamentally surprising information here is that two radios with an order of magnitude difference in theoretical throughput, are achieving competitive performance in the gray region. From the above throughput fluctuations, it is evident that using a single radio IoT network leads to throughput loss even on a simple line topology. While most of the real-world IoT applications will employ the more complex mesh topology, this throughput loss will get further amplified in real-world mesh topology applications.

IV. Why multi-radio for mesoscale IoT apps?

Line topology is seldom used in real-world applications. Hence, it will be interesting to explore the throughput fluctuations of LoRa and Zigbee in a mesh topology.

Experimental setup: The multi-radio end-devices shown in Figure 2 are populated in the gray region to form a mesh topology as shown in Fig. 3. A total of one thousand 29-byte packets are transmitted by each node in the network destined for the gateway. The R Pi host commands both radios every 3 seconds to transmit a packet concurrently. The Zigbee radios employ a Distance-Vector protocol [30] for multi-hop routing.

Results: The throughput achieved by both radios of the network is plotted as a CDF in Fig. 4. This figure shows that LoRa achieves higher throughput for 59% of the transmissions and Zigbee achieves higher throughput for 41% of the transmissions. The throughput is calculated as the fraction of total bits transmitted over the incurred latency. It is to be noted that the throughput has an inverse relationship with latency. According to 5G America's report [31], the average required latency for mesoscale IoT applications is 55ms. The average latency achieved by Zigbee-only and LoRa-only networks is 62.32 ms and 66.55 ms, respectively. This is higher than the average required latency, 55ms, for mesoscale apps. So, the single-radio networks are not useful for mesoscale IoT apps.

A trace-driven simulation is conducted to understand the throughput gain of a multi-radio network system comprising both Zigbee and LoRa radios. This trace-driven simulation mimics the performance of a multi-radio network that can choose a high-throughput radio for every transmission. This is plotted as the dashed golden line in fig. 4. This dashed golden line perfectly traces LoRa radio until the first 59% of the transmissions and follows Zigbee radio for the next 41% of the transmissions. The average latency achieved by this simulated multi-radio network, 50.62ms, falls within the average required latency bounds. This shows the necessity of an intelligent multi-radio system that can choose a higher throughput radio at the time of transmission. While it is convenient to choose a high-throughput radio based on traces, it is tedious to predict a high-throughput radio in real-world deployments, as the end-to-end throughput fluctuates over time, as shown in Figs.1a - 1e. If an end node can predict these throughput fluctuations, a high-throughput radio can be selected at the time of transmission to maximize throughput.

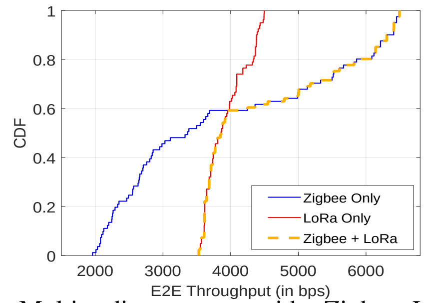


Figure 4: Multi-radio system with Zigbee+LoRa radios achieves higher throughput than single-radio systems.

Why ML? Throughput varies in real deployments (Figures 1a–1e) due to dynamic link conditions. ML enables real-time prediction of these fluctuations for radio selection.

V. Building a machine learning model

This section explains the process of building a Machine-Learning (ML) model for instantaneously selecting a high-throughput radio at the time of transmission. **Problem formulation:** This high-throughput radio selection problem is formulated as a classification problem. This classification problem takes the End-to-End (E2E) path quality estimations of both radios as input to output the instantaneous high-throughput radio. The input feature vector becomes:

$$Input_i = [E2E - PQ_{LoRa}, E2E - PQ_{Zigbee}] \quad (1)$$

The output of theML model is the radio predicted to have higher instantaneous throughput, expressed as:

$$Output_i = [Zigbee|LoRa] \quad (2)$$

Feature Selection: The classification problem formulated above needs E2E-PQ_{LoRa} and E2E-PQ_{Zigbee} to predict the instantaneous high throughput radio. The most common E2E path quality metrics used for multi-hop Zigbee networks are Hop Number (HN_Z), Packet Reception Ratio (PRR_Z), Expected Transmission Count (ETX_Z) [12], and Required Number of Packets (RNP_Z) [32]. HN_Z is the node's distance from the gateway in terms of hops. It is a discrete value ranging from [5,12] inclusively. It is obtained via the distance vector routing protocol run by Zigbee radios. The high data rate Zigbee Radio frequently transmits short beacon packets to estimate the path quality metrics. PRR_Z is a well-known metric calculated as the ratio of the total number

Table IV: Training and Testing accuracy of different ML models and optimizations for all the topologies

ML Models and Optimizations	Location A				Location B	
	Line		Mesh		Mesh	
	Training Accuracy (in %)	Testing Accuracy (in %)	Training Accuracy (in %)	Testing Accuracy (in %)	Training Accuracy (in %)	Testing Accuracy (in %)
SVM	83.59$\pm$1.60	83.37$\pm$2.51	82.15$\pm$0.89	80.87$\pm$3.50	78.31$\pm$1.90	76.62$\pm$4.65
LR	83.65$\pm$1.01	83.87$\pm$3.566	82.59$\pm$0.83	81.25$\pm$3.46	81.21$\pm$0.57	80.50$\pm$2.21
CART	**93.00$\pm$0.49**	88.00$\pm$3.40	**92.53$\pm$0.52**	83.75$\pm$1.97	90.56$\pm$0.25	82.37$\pm$2.94
TAO-CART	93.00$\pm$0.56	**88.00$\pm$2.33**	89.375$\pm$0.26	**85.625$\pm$4.12**	**94.71$\pm$0.38**	**93.87$\pm$1.69**

of packets received over the total number of packets sent. ETX_Z considers both forward and backward link qualities to calculate the metric. In our case, only the forward link quality is required to estimate E2E path quality estimation from an end device toward the gateway. So, the ETX_Z becomes $1/PRR_Z$, making this a redundant metric in the presence of PRR_Z. RNP_Z has the unique characteristic of capturing the underlying distribution of packet losses [32]. So the HN_Z, E2E PRR_Z, and E2E RNP_Z path quality metrics are considered for E2E-PQ_{Zigbee}. These metrics are calculated with frequent beacon packets since Zigbee is a high-data-rate radio. We calculate PRR_Z, RNP_Z, and ETX_Z over a window of size α. Through trial and error, we identified that α=10 gave us better results during our experiments.

On the other hand, the communication channels of low data-rate LoRa radio will be clogged if frequent beacon packets are sent. Hence the RSSI of the ACK sent by the gateway for the previous data packet is considered for estimating the end-to-end path quality of the LoRa radios (E2E-PQ_{LoRa}). After finalizing all the input features, eq. 1 becomes:

$$Input_i = [HN_Z, RSSI_L, PRR_Z, RNP_Z] \tag{3}$$

Data collection is done on three different topologies. "Location A - Line" is the multi-hop line topology described in §III and "Location A - Mesh" is the mesh topology described in §IV. "Location B - Mesh" is the mesh topology mentioned in §VII. We develop separate ML models for each topology. Each topology consists of 15 end nodes and one gateway. Since our region of interest is in the gray region, nodes are populated in this region to form a mesh topology. Data packets transmitted by nodes in the gray region are considered for model training and testing. During this data collection experiment, the host Raspberry Pi will command both radios to simultaneously transmit a 29-byte data packet. The throughput of each transmitted data packet is recorded along with path quality estimations of each radio at the time of transmission. A total of 25,500 data packets were recorded. This dataset is manually labeled by a human to identify the high-throughput radio for each transmitted packet. Zigbee radios sent beacon packets to estimate the local link qualities of each link in the network. These local link qualities were used to manually calculate the path quality metrics.

Prediction Methods and Results: The three widely used classification models, namely Logistic Regression (LR), Support Vector Machine (SVM), and CART Decision Trees (CART) were trained using the trace-driven data set obtained from large-scale real-world experiments on different topologies. An ML model is built offline for each location. MARS will deploy the chosen ML model in all the nodes in that specific topology. Their training and testing accuracy are averaged over 5-fold cross-validation on a dataset based on real-world experiments. CART Decision Tree Classifier [13] is a classical and one of the most popular algorithms to train a DT. The TAO algorithm [18], takes an initial tree, either generated randomly or induced by traditional algorithms (e.g. CART), and optimizes it jointly over the parameters of all the nodes in the tree. TAO works in an alternating optimization fashion by cycling over different depths of a tree. At a given depth, TAO optimizes all nodes in that specific depth in parallel while guaranteeing a monotonic decrease of the desired objective function, such as misclassification errors.

The training and testing accuracies of the widely used classification models, namely SVM, LR, and CART, are tabulated in Table IV. From these results, it is clear that CART can achieve higher accuracy than SVM and LR. Compared with CART, TAO-CART achieves similar accuracy for the simple "Location A - Line topology" which is seldom used in real-world applications, and higher accuracy than CART for the "Location B-Mesh" topology. Also, the difference between the Training and Testing accuracies of TAO-CART is comparably smaller than that of CART for all the topologies. This means TAO-CART is highly generalized and suitable for unseen data.

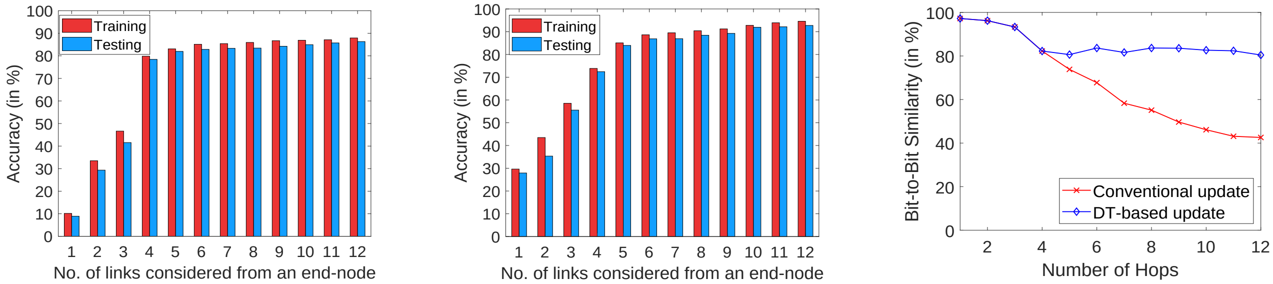
VI. Realizing TAO-CART on end-devices

The TAO-CART radio selector is found to achieve higher accuracy. However, realizing and deploying TAO-CART on an IoT end device entails the following challenges:

(i) *Model size and Inference Latency:* Deploying TAO-CART with a minimal memory footprint on resource-constrained IoT devices is crucial. TAO-CART outputs a tree-like structure for predicting high-throughput radio.

The tree-based model, when converted into a series of if-else statements, occupies 36KB of disk memory. On a Raspberry Pi platform, the model achieves an inference latency of 0.008ms. Owing to its low computational and memory footprint, this deployment is well-suited for resource-constrained IoT edge devices, with further room for optimization.

(ii) *Instantaneous path quality estimations:* The TAO-CART radio selector needs instantaneous end-to-end path quality as an input to accurately predict the high-throughput radio. The models trained and tested in the previous section used a trace-driven data set, whose multi-hop end-to-end path qualities of Zigbee radios were manually calculated. In real-world deployments, the local link qualities should be propagated throughout the network for path quality estimation. However, this network-wide multi-hop link quality propagation may not



(a) RP_n=4 - "Location A - mesh"
(b) RP_n=5 - "Location B - mesh"
(c) DT-based PQ estimation

Figure 5: DT-based PQ estimation. In figures 5a, 5b, the x-axis denotes the number of links considered to calculate PQ metrics.

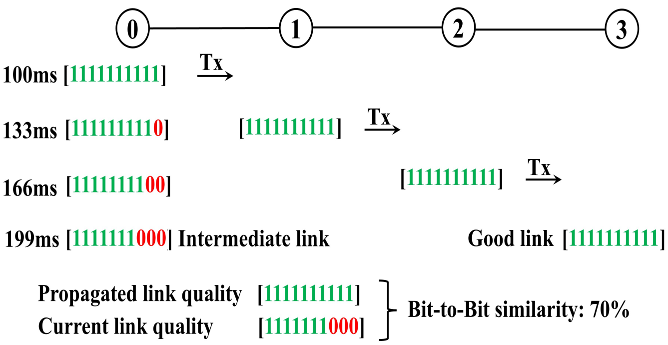


Figure 6: Traditional path quality propagation is not instantaneous at all distances from the gateway.

be instantaneous at all distances from the gateway due to the chaotic link quality variations, queuing and propagation delays. We solve this challenge by developing a DT-based instantaneous path quality estimator.

A. Tradition path quality estimations are not instantaneous

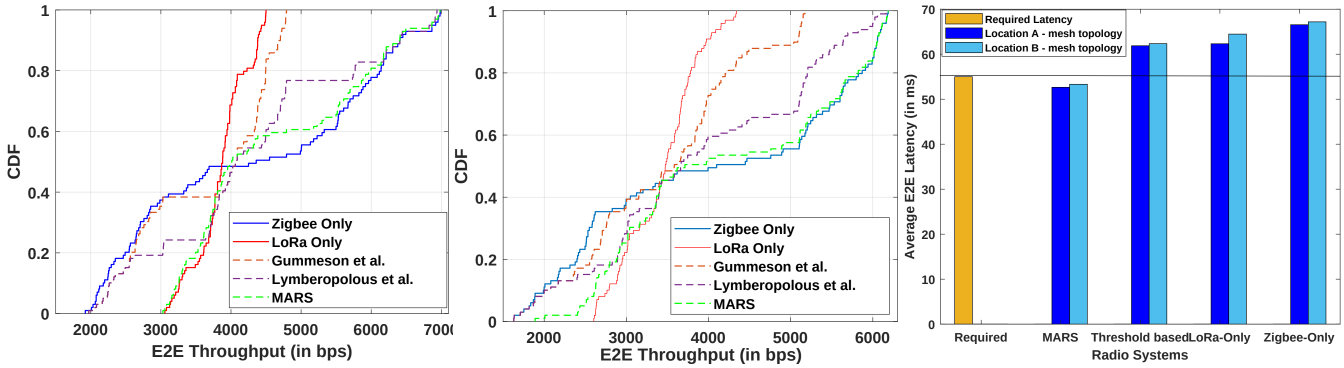
Traditional PQ estimation for a *multi-hop* ZigBee network may not be instantaneous at all distances from the gateway. Traditional PQ estimations for a *multi-hop* ZigBee network need the link quality estimation of all the links along the path. In section V, TAO-CART was trained with a trace-driven data set, whose path qualities were manually calculated based on the local link qualities. In practice, local link qualities along the path should be propagated throughout the network so that they can be used to estimate the path quality. The packet in which these link qualities are propagated will experience delays at different hops due to (i) wireless link quality variations and (ii) packet processing delays. Packet processing includes read/write delays to append link quality sequences of the intermediate nodes, link-layer Acknowledgment delay, CSMA delay, propagation, and queuing delays along the path in a multi-hop network. So, the propagated local link quality may expire when it reaches a node estimating the metric.

The problem of traditional PQ estimation is depicted in Fig. 6. In this figure, Node 1 sends a beacon packet to Node 0 every 30ms. Node 0 stores the reception and loss of the beacons as a binary bit-sequence. It is identified through experiments that propagating a packet containing this link quality sequence takes 33ms on average for a single hop transmission accounting for all the above-mentioned delays while the network has fully functional control and data planes. Figure 6 depicts that the link quality sequence of link 1-0 propagated by Node 0 becomes invalid when it reaches Node 3. The bit-to-bit similarity of the link sequence of Node 0 at 199ms is not the same as propagated by Node 0. The bit-to-bit similarity reduces to 70% at the third hop. This problem will further be amplified if all the nodes along the path append their link quality sequence to this packet for network-wide advertisement. Since the gray region is 500m-1200m from the gateway, estimating path quality with expired link quality sequences will not be instantaneous, unable to act as an indicator of E2E throughput fluctuations.

B. DT-based path quality estimation

DT-based path quality estimation is developed to mitigate the problem of conventional path quality estimation for multi-hop networks. While the conventional path quality is calculated with Link Quality (LQ) information from all the links along the path, DT-based path quality estimation requires LQ information only from a portion of the entire end-to-end path to compute the PQ metrics. This will highly reduce the delay incurred to propagate LQ information in a multi-hop Zigbee network. DT-based path quality estimation surfaces from the two important observations listed below: (i) The link quality of each link in the path may independently change based on the deployed environment, and (ii) Each end node runs MARS to select a radio for transmitting the packet. A path from an end node to the gateway consists of multiple links. While it is intuitive to understand that LQ information from a portion of the entire end-to-end path is enough to compute PQ metrics, the challenge here is to identify the required path length, RP_n, so that the TAO-CART radio selector can accurately predict the high-throughput radio. For example, say a path from the end node to the gateway consists of 10 links. Traditional PQ estimation uses LQ information from all 10 links to compute PQ metrics, whereas DT-based PQ estimation requires LQ information only from RP_n (<10) links to compute PQ metrics. The problem here is to define RP_n. We address this problem by training and testing decision trees with PQ metrics computed from different partial path lengths (RP_n) to understand the prediction accuracy of DT.

A DT takes the end-to-end path quality metric of both radios as input and chooses a radio for transmitting the packet. The following test is conducted to identify the Required Path length RP_n: The path quality metrics are calculated with LQ information from a different number of links, from an end node towards the gateway.



(a) Throughput - Location A (b) Throughput - Location B (c) Latency of MARS

Figure 7: Throughput and Latency performance of MARS

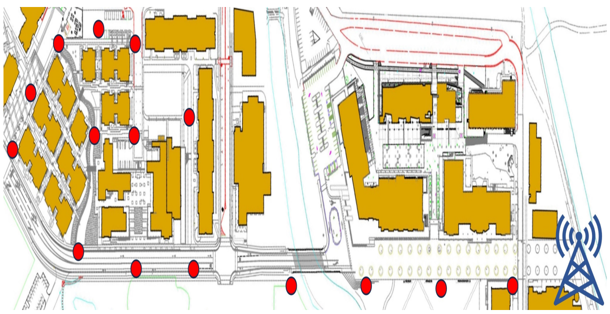


Figure 8: Location B - mesh topology

The training and testing accuracy of the models using LQ information from the different number of links along the path for mesh topologies at Locations A and B are depicted in Figures 5a and 5b respectively. From these figures, it is inferred that there is a very gradual increase in accuracy after the fourth and fifth hops for the "LocationA-mesh" and "LocationB-mesh" topologies, respectively. These decision trees are further optimized by the TAO algorithm. The above values are set to RP_n for our large-scale experiments. Figure 5c shows that the bit-to-bit similarity of conventional PQ estimation decreases with increases in hops while the bit-to-bit similarity of DT-based PQ estimation averages to 81%.

VII. Large-scale experimental evaluations

MARS is evaluated through large-scale real-world, mesh topology experiments conducted on our campus. Two mesh topologies are set up in two different locations as shown in Figs. 3 and 8. These topologies are deployed at locations with complex environments with different building materials and heavy human influx. A total of 10,400 data packets were transmitted at both locations for this real-world evaluation.

Benchmarks: MARS is compared with the single radio systems formed by (i) Zigbee-only and (ii) LoRa-only radios. MARS is also compared with two Multi-radio systems: (i) Q-learning-based radio selector [16] and (ii) Threshold-based radio selector [17], optimizing for energy efficiency. We made our best effort to adopt these to optimize throughput. The threshold-based algorithm [17] is based on the break-even points identified through experiments similar to Fig. 1f.

Table V: Thresholds identified to achieve higher throughput in the gray-region based on Lymberopoulos et al. [17]

	500-700m	700-1000m	1000-1200m
Zigbee RR	≥77%	≥80%	≥83%
LoRa RSSI	≥-72 dBm	≥-71 dBm	≥-71 dBm
Fallback radio	Zigbee	LoRa	LoRa

The thresholds are set based on a node's distance from the gateway and the instantaneous end-to-end path quality of the radios. End-to-end reception ratio and RSSI are used as the instantaneous path-quality indicator for the multi-hop Zigbee and the single-hop LoRa radios, respectively. These thresholds are identified from the traces obtained from real-world experiments. This threshold-based algorithm divides the gray region into three sub-regions 500-700m, 700-1000m, and 1000-1200m based on grouping similar indicators, achieving higher throughput. Thresholds are set based on these sub-regions for a fair comparison. Thresholds tabulated in Table V are identified via experiments to achieve higher throughput in each sub-region. A radio having end-to-end reception ratio greater than 77%, 80%, and 83% achieves higher throughput in the sub-regions 500-700m, 700-1000m, and 1000-1200m respectively while LoRa's RSSI ≥-72 dBm, ≥-71 dBm and ≥-71 dBm achieve higher throughput in the sub-regions 500-700m, 700-1000m, and 1000-1200m respectively. So, in the case of one radio performing better than the other, it will be indicated by the thresholds, eventually selected for transmitting the packet. From the experimental traces, the fall-back radio is identified to achieve better throughput if both radios fall inside or outside the defined threshold region.

The sophisticated RL-based radio switching protocol of Gummeson et al. [16] suffers from the below-described problems: (i) During data transfer between two radios in the communication range, the radio switching protocol, designed for energy efficiency, does a three-way handshake to switch radios. This incurs additional latency which will heavily degrade the throughput. (ii) A well-known issue of model-free RL is that it requires heavy training data to converge to an acceptable performance [33] and the amount of data samples used by Gummeson et al. [16] for training is obscure.

Performance evaluation: The throughput and latency performance of MARS are depicted in Figs 7a-7c. Figs. 7a,7b plots the throughput of different radio systems as CDF. From these figures, it is clear that the Q-learning-based radio selector [16] achieves the least throughput gain of all the multi-radio systems because of its three-way handshake. The threshold-based radio selector [17] converges towards the high-throughput Zigbee only after 60% of the transmissions. This is because the threshold-based radio selector is not able to identify the high-throughput radio when both radios fall inside the threshold region. The identified fallback radios do not achieve higher throughput all the time because of erratic link quality variations.

MARS closely follows the high-throughput radio as it instantaneously identifies the throughput fluctuations. In Location-A, the threshold-based radio selector algorithm achieves an average throughput gain of 18.79% and 15.31% than Zigbee-only and LoRa-only networks respectively. Whereas, MARS achieves an average throughput gain of 55.93%, 57.22%, and 36.32% than Zigbee-only, LoRa-only, and Threshold-based radio selector networks respectively. Fig. 7b shows similar trends in Location B. In this location, all three multi-radio systems tend to cross the solid red line unlike Fig. 4, where the optimal performance was obtained offline through a trace-based evaluation. Hence, the simulated optimal multi-radio performance has to be chosen from any one of the available throughputs making the golden dashed lines of Fig. 4 to stay within the solid lines. The performance evaluations were conducted in real-world deployments. The channel conditions may not be identical when each system is evaluated. This change in channel conditions led to a small difference in throughput making the multi-radio systems cross the solid line. In Location-B, the threshold-based radio selector algorithm achieves an average throughput gain of 16.06%, and 19.57% than Zigbee-only and LoRa-only networks respectively. MARS achieves an average throughput gain of 53.77%, 58.34%, and 32.49% than Zigbee-only, LoRa-only, and Threshold-based multi-radio networks respectively.

Average required latency for mesoscale IoT. According to 5G America's report [31], the average required latency for mesoscale IoT applications is 55ms. Fig. 7c shows the average latency achieved by different radio systems at two different locations. LoRa-only and Zigbee-only networks are not able to achieve the required latency in the gray region. MARS is able to achieve an average latency of 51.48ms and 53.98ms in Locations A and B, respectively. MARS achieves the goal while the other radio systems fail.

VIII. Conclusion

To conclude, we identified that multi-radio architecture is inevitable for mesoscale IoT applications. The goal of MARS is to select a high-throughput radio to be used at any point in the network, using different network paths and link-layer metrics gathered from the radios. The radio selection is done using the TAO-optimized decision trees, which are easy to deploy in an IoT end device with limited computational power. In addition, we show that collecting local path metrics as input to our decision trees provides sufficient information to identify the high-throughput radio over the entire path. MARS is evaluated on a large-scale complex mesh topology at two different locations. The results show that MARS can identify the high-throughput radio at the time of transmission. This leads to an average throughput gain of 48.2% and 49.79% than the competing schemes at Locations A and B, respectively.

References

[1] V. et al., "{FarmBeats}: an {IoT} platform for {Data-Driven} agriculture," in *USENIX NSDI'17*, 2017.
[2] D. et al., "Constrained broadcast with minimized latency in neighborhood area networks of smart grid," *IEEE Transactions on Industrial Informatics*, 2019.
[3] Y. et al., "Energy efficient multiple target tracking in wireless sensor networks," *IEEE Transactions on Vehicular Technology*, 2007.
[4] O. et al., "Throughput maximizing and fair scheduling algorithms in industrial internet of things networks," *IEEE Transactions on Industrial Informatics*, 2018.
[5] Z. et al., "Throughput optimization with delay guarantee for massive random access of m2m communications in industrial internet of things," *IEEE IoT Journal*, 2019.
[6] P. et al., "Recent advances in local energy trading in the smart grid based on game-theoretic approaches," *IEEE Transactions on Smart Grid*, 2017.
[7] S. et al., "Impact of distributed energy resources in smart homes and community-based electricity market," *IEEE Transactions on Industry Applications*, 2022.
[8] L. et al., "Intraday residential demand response scheme based on peer-to-peer energy trading," *IEEE Transactions on Indus. Informatics*, 2019.
[9] "Smart meter system," http://tinyurl.com/4z2dy6nc, 2024.
[10] "Large-capacity battery control system," http://tinyurl.com/2ryvbcxv.
[11] S. et al., "Industrial internet of things: Challenges, opportunities, and directions," *IEEE transactions on industrial informatics*, 2018.
[12] D. C. et al., "A high-throughput path metric for multi-hop wireless routing," in *MobiCom '03*, 2003.
[13] Praagman, "Classification and regression trees," 1985.
[14] A. I. et al., "Backpacking: Deployment of heterogeneous radios in high data rate sensor networks," in *IEEE ICCCN*, 2011.
[15] K. et al., "Radio diversity for reliable communication in sensor networks," *ACM TOSN*, vol. 10, no. 2, pp. 1–29, 2014.
[16] G. et al., "An adaptive link layer for range diversity in multi-radio mobile sensor networks," in *IEEE INFOCOM*, 2009.
[17] L. et al., "Towards energy efficient design of multi-radio platforms for wireless sensor networks," in *ACM/IEEE IPSN*, 2008.
[18] C.-P. et al., "Alternating optimization of decision trees, with application to learning sparse oblique trees," *In NeurIPS '18*, 2018.
[19] "Sigfox system description," http://tinyurl.com/45tbwraf.
[20] L. et al., "Wifi halow for long-range and low-power IoT: System on chip development and perf. evaluation," *IEEE Comm. Magazine*, 2021.
[21] T. L. et al., "Bluetooth® Low Energy and the automotive transformation," *TI*, 2017.
[22] "Cc2520 datasheet," http://tinyurl.com/pf6ecmr6.
[23] P. et al., "Evaluation of lora technology for indoor remote health and wellbeing monitoring," *Intnl. Journal of Wireless Info. Networks*, 2017.
[24] M. et al., "An environment for runtime power monitoring of wireless sensor network platforms," in *In IEEE SSST'05.*, 2005.
[25] S. et al., "A survey on LoRa networking: Research problems, current solutions, and open issues," *IEEE Comm. Surveys & Tutorials*, 2019.
[26] P. L. et al., "Tinyos: An operating system for sensor networks," *Ambient intelligence*, pp. 115–148, 2005.
[27] R. LLC, "LoStik," https://github.com/ronoth/lostik.
[28] R. Boorstyn and et al., "Throughput analysis in multihop csma packet radio networks," *IEEE Transactions on Communications*, 1987.
[29] B. et al., "Radio link quality estimation in wireless sensor networks: A survey," *ACM TOSN*, vol. 8, no. 4, pp. 1–33, 2012.
[30] C. et al, "A loop-free extended bellman-ford routing protocol without bouncing effect," *ACM SIGCOMM CCR*, 1989.
[31] G. Americas. (2019) 5G - The future of IoT. [Online]. Available: http://tinyurl.com/26y5epsp
[32] C. et al., "Temporal properties of low power wireless links: modeling and implications on multi-hop routing," in *In MobiHoc '05*, 2005.
[33] D. et al., "Mb2c: Model-based deep reinforcement learning for multi-zone building control," in *ACM BuildSys'20*, 2020.