A Supervised Active Learning Method for Identifying Critical Nodes in IoT Networks

Behnam Ojaghi *[1], Mohammad Mahdi Dehshibi[2], and Angelos Antonopoulos[3]

[1]Dept. of Signal Theory and Communications, Polytechnic University of Catalunya (UPC), Spain
[2]Dept. of Computer Science and Engineering, Universidad Carlos III de Madrid, Leganés, Spain
[3]Nearby Computing S.L., Barcelona, Spain

Abstract

The energy efficiency of wireless sensor networks (WSNs) as a key feature of the Internet of Things (IoT) and fifth-generation (5G) mobile networks is determined by several key characteristics, such as hop count, user's location, allocated power, and relay. Identifying important nodes, known as critical nodes, in IoT networks that involve a massive number of interconnected devices and sensors significantly affects these characteristics. However, it also requires a significant computational overhead and energy consumption. To address this issue, we introduce a novel supervised active learning method for identifying critical nodes in IoT networks aimed at enhancing the energy efficiency of WSNs in 5G environments. Our experimental results, designed to closely replicate varied and complex IoT network scenarios focusing on mission-critical multi-hop IoT applications, demonstrate the proposed method's capability to improve adaptability and computational efficiency. These results suggest a strong potential for mission-critical applications in real-world large-scale multi-hop WSN environments in 5G, as well as massively distributed IoT.

Keywords: Wireless Sensor Networks, Lifetime, IoT, Active Learning, Critical Nodes.

1 Introduction

Reasons such as gigantic data rate requirements, energy price, the ecological impact of carbon, and social responsibility for fighting climate change have inspired the telecommunication community to develop energy-saving techniques [1–5]. With the emergence of fifth-generation (5G) technology, the importance of adopting energy-efficient architectures has been realized more to meet the demands of increased capacity, improved data rate, and better quality of service (QoS) [6]. Moreover, the primary concern of improving energy efficiency (EE) without compromising on user experience makes green communication an urgent need. Green communication not only can meet these demands but also can observe the social responsibility to reduce the carbon footprint by reducing the power consumption in a wireless network.

In an era marked by soaring data rate requirements, escalating energy costs, and mounting ecological concerns, the telecommunication community has been galvanized into developing energy-saving techniques [1, 2, 4, 5, 7]. The advent of 5G technology has further amplified the imperative for energy-efficient architectures, as networks grapple with surging demands for increased capacity, enhanced data rates, and superior QoS [6]. Complicating this challenge is the dual mandate to bolster EE without sacrificing user experience, making the quest for green communication an immediate priority.

An apt choice of EE metrics can fundamentally transform the design and performance optimization of a network. EE metrics can be categorized based on various paradigms: either as the number of bits transmitted per joule or as the system throughput per unit energy consumed [8, 9]. These metrics assume

*bojaghi@uoc.edu

distinct interpretations depending on the application context: (i) In information theory, the focus is on minimizing energy per bit; (ii) In wireless sensor networks (WSNs), the aim is to prolong the network lifetime, where lifetime is normally measured from the moment the network starts to the point that the first sensor node runs out of energy [10]; and (iii) In Cellular Networks, the goal is to extend the standby time of mobile stations while also maximizing EE under QoS constraints. This paper narrows its focus to the improvement of EE in the context of IoT networks, where its topology is usually composed of sensor nodes, and data is relayed to the base station (BS) by cooperating sensor nodes with nearby nodes. In the intricate framework of multi-hop IoT networks, meticulous path determination between nodes and the sink is paramount for streamlined data transmission. However, the vulnerability of IoT networks to various threats, including denial of service (DoS) attacks and natural disasters such as earthquakes and landslides, poses significant challenges. These risks can incapacitate sensor nodes, disrupting data flow irreparably. For instance, in intelligent transportation systems, sensor nodes positioned at intersections play a vital role in real-time data collection on traffic patterns and pedestrian movement, making them prime targets for malicious actors aiming to disrupt traffic management. It's imperative to recognize that not all nodes within the network hold equal significance; some are more *critical* than others. Identifying and evaluating these critical nodes in IoT networks holds paramount importance for two key reasons [11]. Firstly, it may not be practical to safeguard every sensor against potential threats, but protecting critical sensors could present a viable solution. Secondly, understanding the maximum level of network degradation resulting from the removal of specific nodes is instrumental in network design and maintenance, aiding in adjusting service tolerance levels effectively. Thus, the identification and evaluation of critical nodes serve as fundamental steps in fortifying the resilience and efficiency of IoT networks.

Authors in [12] elucidate the influence of factors such as hop-count, user location, and allocated power on EE within WSNs. They highlight that relays, due to their susceptibility to fading conditions, can be strategically utilized for energy-saving measures. Initially introduced in third-generation (3G) wireless communication, the concept of using relays for energy-saving has evolved and expanded in 5G. Specifically, 5G has advanced the use of relays in conjunction with coordination and support actions (CSA) as well as device-to-device (D2D) communication.

However, finding the most suitable relay nodes in large WSNs, especially in Massive IoT, can be a challenging and time-consuming task that leads to significant computational overhead and energy consumption. Numerous studies have attempted to tackle this issue, but they have encountered various difficulties. For example, Parzysz et al. [13] found that even maximizing the output signal-to-noise ratio (SNR) in relay networks cannot address the problem without compromising performance due to the uncertainties inherent in global Channel State Information (CSI). Other studies [14, 15] have considered selection schemes based on the least distance, highest transmission rate, and SNR. However, they often overlook additional parameters like sensor coordinates, connectivity, and distance to the BS, which could potentially miss out on a more holistic approach to energy-efficient IoT network design.

In this paper, we make a significant observation about network performance: *not all nodes contribute equally to the stability and efficiency of the network*. We introduce and formally define a new concept called the *first critical node*, which is the node whose disablement results in the most substantial decline in network performance. Identifying these critical nodes allows network administrators and operators to take proactive measures to protect them, thereby enhancing the network's resilience against system failures and potential attacks. It is important to note that our definition of critical nodes is different from the ones that are typically used in WSN literature, where the removal of such nodes causes the network to become disconnected [16].

According to our definition of critical nodes, we argue that their removal should not necessarily lead to network disconnection, especially in the context of massive IoT networks, which are generally characterized by robust connectivity. In fact, we assume that the network remains strongly connected even in the event of eliminating certain critical nodes. In our definition, the key metric affected by removing a critical node is network latency rather than connectivity. We further extend our definition to incorporate additional parameters such as sensor coordinates, connectivity, distance to BS, and latency in identifying a critical node. To mitigate the computational overhead associated with identifying critical nodes in IoT networks, we propose using an active learning method. This method will allow us to make meaningful comparisons with other techniques, particularly those that may face computational constraints in seeking an optimal solution.

In the dynamic environment of IoT networks, the majority of nodes are, in fact, non-critical. This

presents unique challenges for the design of learning algorithms, given the inherent imbalances and dynamic nature of node importance. To address these issues, we introduce an active learning algorithm to identify critical nodes. Our approach not only minimizes bias in identifying non-critical nodes but also significantly reduces the need for retraining to adapt to network dynamics. In contrast to traditional optimization-based solutions [11, 17], our active learning approach offers enhanced flexibility, making it particularly well-suited for large-scale WSN environments. The algorithm comprises two interconnected modules: clustering and classification. In the clustering module, we employ local gravitation clustering (LGC) [18] to mitigate the effects of the imbalanced dataset. The classification module employs a support vector machine (SVM) [19] to classify nodes into critical and non-critical categories. The synergy between these modules enables our algorithm to iteratively reduce the dataset size in a standard supervised learning context, thereby handling uninformative examples.

The remainder of this paper is structured as follows: Section 2 provides a comprehensive review of existing studies in the field. Section 3 introduces the system model. In Section 4, we delve into the details of our active learning algorithm, which encompasses both clustering and classification modules, and discuss how the complexity of the feature space can impact the accuracy of the classification task. Section 5 proposes an optimization approach to identify critical nodes. Section 6 focuses on the empirical evaluation of our proposed methods in various applications. Finally, Section 7 offers concluding remarks and outlines potential avenues for future research.

2 Related Work

Various methods have been proposed to enhance EE in current and future IoT network architectures. Li [20] devises a heuristic energy-saving relay selection scheme aimed at maintaining minimum QoS for users. This scheme also involves the reallocation of surplus subcarriers through a bandwidth exchange algorithm, ultimately achieving both maximum EE and minimum spectral efficiency compared to other algorithms. Parzysz et al. [13], on the other hand, focuses on a relay power allocation method designed to optimize the SNR. Their results underscored the risk of significant performance degradation when uncertainties in channel state information are not accounted for.

Cheng et al. [21] explore a stochastic optimal power control method, balancing EE and fairness. Their approach prioritizes relay nodes based on power cost and channel conditions, using an algorithm akin to least mean square (LMS) for power control. Their results indicate that a convex optimization problem that considers fairness criteria for each relay could maximize the network's lifetime. Farooq et al [22] study a probabilistic weight-based energy-efficient cluster routing for large-scale WSNs protocol, where the results of this work show that the proposed protocol gives better performance as compared to the benchmarks for the metrics of energy consumption, network lifetime, latency, throughput, packet delivery ratio, and routing distance efficiency. Momnale et. al. [23] explore node utilization index-based data routing and aggregation protocol. In this work, they calculate the shortest path from the source node to the sink node based on the sensors' consumed bandwidth, minimum hop count, node's residual energy, and data aggregation factor. The result of this work shows that their proposed protocol outperforms the existing benchmark protocols.

Authors in [24] propose a multiuser intelligent reflecting surface (IRS)-aided multiple-input single-output (MISO) system with simultaneous wireless information and power transfer. The main objective of this work is to maximize the EE of the system and jointly optimize the BS transmit beamforming vectors, the IRS reflective beamforming vector, and the power splitting ratios. Their results demonstrate the effectiveness of the proposed algorithm as well as the benefit of deploying IRS for enhancing the EE performance compared with the benchmark schemes. Yang et al. [25] propose a grouped-adaptive-connected (GAC) hybrid precoding structure, to study the energy-efficient hybrid pre-coding in mm-Wave massive MIMO systems. Through simulation results, they show the advantages of the proposed hybrid precoding scheme.

Diverging from the focus on relay nodes, numerous research endeavors have expanded the scope to explore the impact of other network features on EE. For instance, Mobile VCE focused on BS hardware, architecture, and operations, achieving remarkable energy savings of 75–92% in simulated scenarios [26]. EARTH adopted a technological approach, introducing innovations like low-loss antennas, micro direct transmission, antenna muting, and adaptive sectorization. These technologies adapt to traffic fluctuations

and have been shown to achieve energy gains of 60–70%, with less than a 5% degradation in throughput [27]. On the industry side, operators are moving towards green technology [28]. They are adopting green BSs powered exclusively by renewable energies and are implementing collaborative cloud-based solutions like the Cloud/Collaborative/Clean Radio Access Network (C-RAN) [29–31]. C-RAN involves a soft and virtualized BS where multiple baseband units are integrated as virtual machines on the same server, supporting a range of radio access technologies and contributing to 5G goals of both spectral and EE. Recent studies have also focused on the concept of 'critical nodes,' whose removal has a pronounced effect on network performance metrics like lifetime and latency [32]. Kahjogh et al. [10, 11] proposed algorithms for identifying these critical nodes using both Sequential and Bulk approaches. The former involved a meticulous analysis of the repercussions of removing individual nodes, while the latter used optimization techniques for collective identification. However, the computational complexity of these methods limited their applicability in massive IoT environments. Complementing these efforts, other research has examined node states (*i.e.*, critical and non-critical) by running limited distributed Depth-First Search (DFS) algorithms. In these approaches, the failure or removal of nodes directly impacts network connectivity [33–36]. These approaches, although insightful, still face computational challenges, particularly in massive IoT settings where the timely identification of critical nodes is crucial.

While several research studies have focused on EE and node importance in WSNs and IoT, computational complexity remains a significant barrier to practical implementation, especially in large-scale, dynamic environments. Moreover, most existing algorithms are either too narrowly focused on specific network features or too computationally intensive for real-world applicability. Our work takes a novel approach by introducing an active learning algorithm that efficiently identifies critical nodes while adapting to the dynamic nature of IoT networks. The proposed method offers a scalable, computationally efficient solution that has the potential to significantly advance the field, especially in the context of massive IoT settings.

3 System Model

We assume a network with a single BS, and sensor nodes capable of transmitting their own generated data and also relaying data of other sensor nodes to BS. The data packets from all the sensors have the BS as the destination. In our system model, we consider G as our network topology that the network is strongly connected and can survive connectivity in the case of eliminating some critical nodes among V. We define a critical node as a node whose removal increases network latency. Fig. 1 illustrates the overall schematic view of our considered network model. The network topology is represented by a directed graph, $G = (V, E)$, where V is the set of all sensor nodes and the BS is defined as node n_0. We also define a set $W = V \setminus \{n_0\}$, which includes all nodes except n_0. $E = \{(i, j) : i \in W, j \in V \setminus \{i\}\}$ is the set of arcs in which $i \neq j$.

For classification and clustering, we divided our problem space ($\mathcal{X}$) into two subsets referred to as (1) labelled ($\mathcal{L}$) and (2) unlabeled ($\mathcal{U}$) sets. A node is represented by a 5-tuple $(c_x, c_y, Con, Dist, Rel)$ with a corresponding label $\mathcal{Y} = \{0, 1\}$, where '1' is critical and '0' otherwise. In this representation, (c_x, c_y) refer to the node's coordinate, and $Con, Dist$, and Rel are connectivity, distance to the BS, and relay, respectively.

4 Problem Statement

In a wireless communication network, removing a critical node can result in network disconnection and a steep rise in latency. Considering the dynamic nature of wireless communication, finding critical nodes is not an offline problem, and the computational overhead of using optimization methods [37, 38] can substantially decrease the lifetime of the network. Although substituting classification algorithms with optimization methods can confront computational complexity, the imbalance problem (where only 15% of nodes are critical ones) can result in a high false-positive rate.

In this study, we propose employing an active learning approach to identify critical nodes. To resolve the imbalance problem, we used the LGC method [18], where the cluster with the maximum size is then

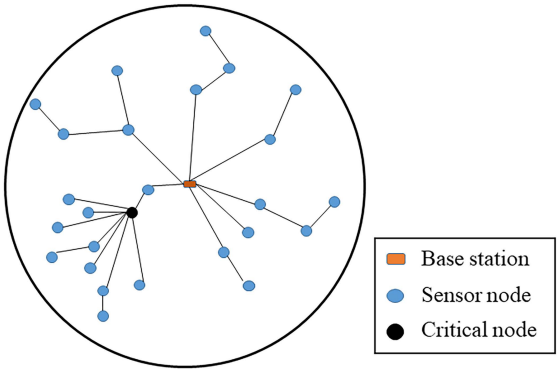


Figure 1: A schematic view of the network.

selected as a new problem space. In this way, the new problem space contains nodes, which 30% of them are critical ones. Compared to the original problem space, we observed a 50% increase in critical nodes' cardinality and a 15% decrease in non-critical's. The advantage of LGC over clustering approaches like k-means is that it is not dependent on the initial seed and can observe the problem characteristics. We classified nodes by using a linear SVM [19] with the Leave-One-Out cross-validation configuration to identify critical nodes. The proposed policy that links the clustering and classification modules could decrease the computational complexity of re-training, bring flexibility, and increase generalizability. Details of the proposed method are as follows.

As shown in Algorithm 1, the learning model $\mathcal{M}$ maps the input to output, and a loss function $Loss(\mathcal{M})$ measures the model's error. At each learning iteration, the learner selects a query $q \in \mathcal{U} \subset \mathcal{X}$ and obtains the label $l \in \mathcal{Y}$. Adding q to the previously labelled instances $\mathcal{L}$ and re-training the model $\mathcal{M}$ with this new pair (q, l) results in reducing the loss value. We used the entropy regularization instead of minimizing the expected error to satisfy the self-learning condition. In this way, we could also use the latent structure to improve $\mathcal{M}$ (see Algorithm 1).

Algorithm 1: Active Learning Algorithm.

Input : $\mathcal{L}$: Labeled set,
$\mathcal{U}$: Unlabeled set,
m: Training set size.
Output: Model $\mathcal{M}$

while *training size* $\leq m$ **do**
  $\mathcal{M} \leftarrow$ learn a model based on $\mathcal{L}$
  $\mathcal{U} \leftarrow \mathcal{X} \setminus \mathcal{L}$
  foreach $q_i \in \mathcal{U}$ **do**
    $u_i \leftarrow Loss(q_i)$
  $u^* \leftarrow \arg\min_i(u_i)$
  $\mathcal{L} \leftarrow \mathcal{L} \cup u^*$
  $\mathcal{U} \leftarrow \mathcal{U} \setminus u^*$
  $\mathcal{U} \leftarrow$ update the model based on $\mathcal{L}$
return $\mathcal{M}$

The active learning algorithm consists of an outer loop that runs until the training set size reaches m. Within this outer loop, an inner loop iterates over the unlabeled set $\mathcal{U}$ with size n_1 to calculate the loss for each element. The dominant time complexity in this algorithm is determined by this nested loop structure, resulting in an overall time complexity of $O(m \times n_1)$.

We use LGC to create $\mathcal{L}$ and $\mathcal{U}$ sets. We selected two clusters with the largest number of samples. It was observed in the course of experiments that the first cluster ($\mathcal{L}$) comprises 80-83% of critical nodes, and the second cluster ($\mathcal{U}$) contains 15-17% of the critical nodes, with the ratio of 30:70 for critical to

non-critical nodes in each set. If the union of both sets does not contain all critical nodes, we count this difference in calculating the final error. In the LGC, physical laws of gravitation were used to define the relationship between data points and their neighborhoods. In this algorithm, the local resultant forces are first calculated for each data point. Data points are then classified as interior, boundary, and unlabeled points. Afterward, the SoftConnection routine is applied to those points that were labeled as interior points. Finally, the data points labelled as boundary points are assigned to the clusters. This procedure is described in Algorithm 2, where details of the SoftConnection routine can be found in [18].

Algorithm 2: LGC Algorithm. [18]

Input : $\mathcal{X}$: Data set,
k: Number of neighbors,
$\widehat{CE}$:Centrality threshold,
IM: Initial momentum

Procedure: $\mathcal{L}$: Labeled set

foreach $x_i \in \mathcal{X}$ **do**
  $m_i = \frac{1}{\sum_{j=1}^{k} D_{ij}}$
  $\overrightarrow{F_i} = \frac{1}{m_i} \sum_{j=1}^{k} \hat{D}_{ij}$
foreach $x_i \in \mathcal{X}$ **do**
  $CE_i = \frac{\sum_{j=1}^{k} \cos\left(\overrightarrow{F}_j, \overrightarrow{D}_{ji}\right)}{k}$
  $CO_i = \sum_{j=1}^{k} \left(\overrightarrow{F}_i \cdot \overrightarrow{F}_i\right)$
SoftConnection($\mathcal{X}, \widehat{CE}$, **IM**)
foreach *un-clustered boundary point* x_i **do**
  if $CO_i \geq 0$ **then**
    Add x_i to its nearest cluster it points at.
foreach *un-clustered boundary point* x_i **do**
  Add x_i according to its neighbors' cluster labels.
return $\mathcal{L}$

The LGC algorithm involves several steps to perform clustering based on local gravity principles. It has two primary loops that iterate $O(n_2 \times k)$, where n_2 is the size of the data set $\mathcal{X}$ and k is the number of neighbours. Additionally, the SoftConnection routine is estimated to have a worst-case time complexity of $O(n_2^2)$. Summing up these individual complexities, the overall time complexity of the LGC algorithm is $O(n_2^2 + n_2 \times k)$.

In addition to real labels, the hidden cluster's label $k \in \{1, 2, \cdots, K\}$ is also introduced where K is the number of clusters. k indicates that the sample belongs to the k^{th} cluster, where all information about the class label y is already encoded in k. Therefore, by knowing k, $x \in \mathcal{X}$ and $y \in \mathcal{Y}$ can be considered independent, and the joint distribution is written as Eq. 1:

$$p(x, y, k) = p(y|x, k)p(x|k)p(k) = p(y|k)p(x|k)p(k) \tag{1}$$

In our model, $p(y|k)$ is defined as a logistic regression for all clusters with the same parameters (Eq. 2).

$$p(y|k) = \frac{1}{1 + e^{-y(c_k.a+b)}} \tag{2}$$

where c_k is the representative of the k^{th} cluster, $a \in \mathbb{R}^d$ and $b \in R$ are the logistic regression parameters. Therefore, once all the parameters of $p(y|k)$ are determined, this likelihood can be used to determine the label of samples belong to this cluster. Given $p(y|k)$, the posterior probability of sample label can be calculated by Eq. 3

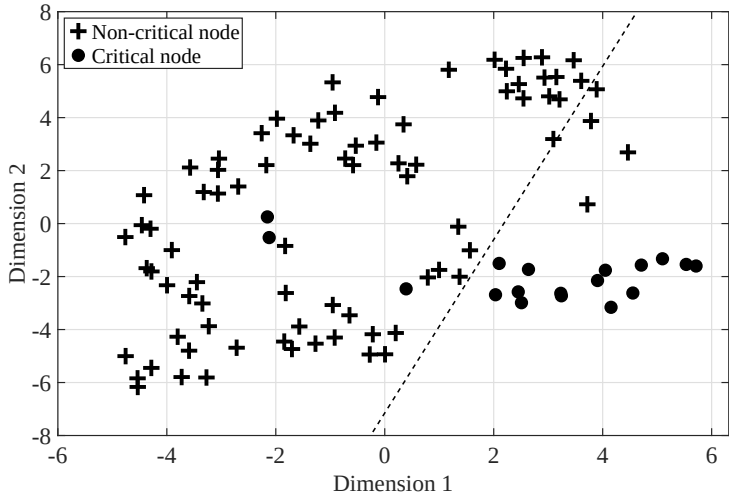
$$p(y|x) = \sum_{k=1}^{K} p(y, k|x) = \sum_{k=1}^{K} p(y|k)p(k|x) \tag{3}$$

where $p(k|x) = p(x|k)p(k)/p(x)$. The new label for data is subject to clustering and can be defined by using the Bayes decision rule (4):

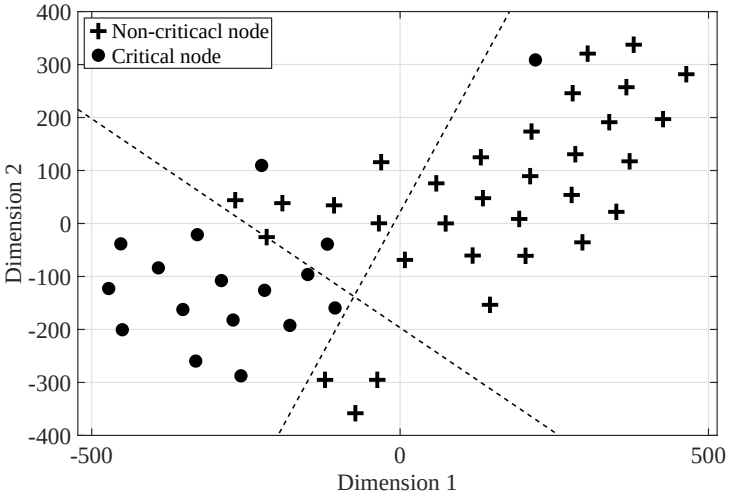
$$\hat{y}(x) = \begin{cases} 1 & \text{if } p(y=1|x;\hat{a},\hat{b}) > p(y=0|x;\hat{a},\hat{b}) \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

where $\hat{a}, \hat{b}$ denote the current estimates of the parameters. Equation 3 shows that the weighted combination of the posterior probability is used for the representatives. Moreover, the same label will be assigned to well-clustered instances as the nearest representative.

We visualized the feature space using t-distributed stochastic neighbor embedding (t-SNE) [39] to show the impact of clustering. t-SNE is a nonlinear dimensionality reduction technique that is used for visualizing high-dimensional latent space in two or three dimensions. In this approach, similar objects are modeled by nearby points and distant points with high probability model dissimilar objects. Figure 2(a) shows the distribution of the original feature space considering all nodes. Figure 2(b), however, shows the distribution of feature space where the most informative clusters are considered.



(a)



(b)

Figure 2: Distribution of latent space by applying t-SNE (a) original network configuration, (b) selected nodes by applying LGC.

In Figure 2, we synthetically added dashed lines to show how a classifier can discriminate feature

space using support vectors. It is apparent that classifying $\mathcal{X}$ will result in a high false-positive rate due to the biasing problem that is resolved after clustering feature space.

Following the clustering process, we used a SVM [19] with a linear kernel to identify critical nodes. This classification algorithm tries to minimize the objective function of Eq. 5 to find the maximum margin hyperplane parameters dividing the two classes.

$$\begin{aligned} \min_{\overrightarrow{w},b,\xi,C} \frac{1}{2} \|\overrightarrow{w}\|^2 + C \sum_{i=1}^{l} \xi_i \\ \text{subject to:} \\ \forall_{i=1}^{l} : y_i[\overrightarrow{w}.\Phi(\overrightarrow{q_i}) + b] \geq 1 - \xi_i \\ \forall_{i=1}^{l} : \xi_i > 0 \end{aligned} \tag{5}$$

where $\overrightarrow{w}$ and b are the parameters defining the maximum margin hyper-plane, $\Phi(\cdot)$ is the kernel function, C is the parameter defining the trade-off between the margin size and mis-classified examples, q_i is a sample from $\mathcal{L}$, y_i is the associated label to q_i, and ξ is the slack variable.

Due to the dependency of clustering and classification modules in the proposed method and imbalance nature of the classification task, we must define a weighted accuracy (Eq. 6) and report extra measures for a fair comparison. In addition to true positive and false negative rates, we calculated weighted precision, recall, and F1-score (Eq. 7). A weighted precision computes the precision independently for each class and then take the weighted average. Weighted recall is the fraction of the total amount of relevant instances that were actually retrieved weighted by the weight of each class. $F1_{weighted}$ score is a harmonic mean of precision and recall.

$$\text{weighted accuracy} = \sum_{k=1}^{|\mathcal{Y}|} w_i \sum_{x:y=k} I\left(y = \hat{y}\right) . \tag{6}$$

where I is the indicator function and returns 1 if the classes match and 0 otherwise. w_k is the assigned weight to each class such that $\sum_{k=1}^{|\mathcal{Y}|} w_k = 1$. The higher the value of w_k for an individual class, the greater is the influence of observations from that class on the weighted accuracy.

$$P = \frac{TP}{TP + FP},\ R = \frac{TP}{TP + FN}, \tag{7}$$

$$F1 = 2P\frac{R}{(P + R)},$$

$$P_{weighted} = \sum_{k=1}^{|\mathcal{Y}|} w_i P_i,\ R_{weighted} = \sum_{k=1}^{|\mathcal{Y}|} w_i R_i, \tag{8}$$

$$F1_{weighted} = \sum_{k=1}^{|\mathcal{Y}|} w_i F1_i. \tag{9}$$

where TP, FP, TN, and FN stand for true positive, false positive, true negative, and false negative, respectively.

5 Optimization Approach

Besides the active learning approach, we also proposed to employ a mixed-integer programming (MIP) approach to identify critical nodes with two goals: (1) providing ground-truth for data fed into the proposed active learning approach; (2) comparing the proposed learning-based approach with an optimization approach in terms of computational complexity and flexibility to adapt to the network changes.

In this approach, a MIP model is used to evaluate the criticality of each network's node. For the given topology, the MIP runs for N_S times, and the node that its removal results in the maximum latency adds to the critical nodes set. In our implementation, we used the summation of flows in the network as

a metric for calculating the total hop count. In algorithm 3, the network topology is represented by $G = (V, E)$, where V is the node, and G is the edge. This algorithm takes the number of critical nodes to be found (N_C) as an input and returns an ordered set $(C, \leq_{latency})$ of critical nodes associated with the latency. This algorithm iterates for N_C times and, in each iteration, finds a node whose removal from G maximizes the network latency.

Algorithm 3: Critical Node Identification

Input : G=(V, E): Network topology graph
N_C: Number of critical nodes
Output: $(C, \leq_{latency}) = \{(v_j, lt_j) | v_j \in V,$
$1 \leq j \leq n,\ lt_j$: Network latency by removing
(v_{BS}, v_j) from $G\}$

```
for k=1 to N_C do
    C_k.latency = 0
    C_k.latency = ∞
foreach v_i ϵ V, 1 ≤ i ≤ |V| do
    lt_i ← latency(G\v_i)
    if (lt_i > C_k.latency) then
        C_k.latency = lt_i
G ← G\C_k.criticalNode
/* Nodes that their removal give the maximum latency are removed from G */
return C.
```

The critical node identification algorithm begins by initializing latency variables for N_C critical nodes. This process requires $\binom{N_S}{N_C}$ iterations, thereby resulting in a time complexity of $O\left(\binom{N_S}{N_C}\right)$ for this segment. Importantly, the worst-case scenario for $\binom{N_S}{N_C}$ occurs when N_C is closest to $\frac{N_S}{2}$, reaching its maximum value for either even N_S at $N_C = \frac{N_S}{2}$ or odd N_S at $N_C = \left\lfloor \frac{N_S}{2} \right\rfloor$ or $N_C = \left\lceil \frac{N_S}{2} \right\rceil$.

Following this initialization, the algorithm iterates over all vertices V in the network graph G to calculate the network latency for each node when removed. This operation has a time complexity of $O(T \times n_3)$, where n_3 is the total number of vertices and T is the time complexity for calculating latency. Summing these individual complexities, the overall time complexity for the algorithm is $O\left(\binom{N_S}{N_C} + T \times n_3\right)$.

6 Performance Evaluation

To prove the efficiency of the proposed method in identifying N_C critical nodes, we deploy a WSN with 100 nodes and run experiments over this scheme. For this evaluation, we deploy a disk-shaped network topology with a radius $G_r = 200$ m, including $N_S = 100$ nodes, where a single BS placed at the center and $N_S - 1 = 99$ sensors distributed within the disk. We use a uniform random distribution to determine the positions of sensor nodes which have a maximum transmission range of $r_{max} = 82.92$ m as defined in [10]. In particular, we use Mica2 motes' energy dissipation characteristics for the energy consumption calculations in our model [10]. As represented in Table 1, energy consumption for transmit power level l is denoted as $E_{tx}(l)$ and the transmission range at this level is indicated as $R(l)$. If the distance between n_i and n_j (d_{ij}) is greater than maximum transmission range ($r_{max} = 82.92$ m), no data can be sent from n_i to n_j. Each sensor node chooses its optimal transmission energy from a finite set denoted as S_L (*i.e.*, there are just 26 power levels to choose from). In this topology, schematically shown in Figure 1, each sensor covers a target (*i.e.*, neighbor node) if the Euclidean distance between the sensor and the target is less or equal to r_{max}. To this aim, we employ GAMS[1] to construct a sensor network. Then, we pass '(c_x, c_y)-coordinates,' 'connectivity,' and 'distance to the BS' to the mixed integer programming

[1]General Algebraic Modeling System (GAMS). Available: `https://www.gams.com/`.

l	$E_{tx}(l)$	$R(l)$	l	$E_{tx}(l)$	$R(l)$
1 (l_{min})	671.88	19.3	14	843.75	41.19
2	687.50	20.46	15	867.19	43.67
3	703.13	21.69	16	1078.13	46.29
4	705.73	22.69	17	1132.81	49.07
5	710.94	24.38	18	1135.42	52.01
6	723.96	25.84	19	1179.69	55.13
7	726.56	27.39	20	1234.38	58.44
8	742.19	29.03	21	1312.50	61.95
9	757.81	30.78	22	1343.75	65.67
10	773.44	32.62	23	1445.31	69.61
11	789.06	34.58	24	1500.01	73.79
12	812.50	36.66	25	1664.06	78.22
13	828.13	38.86	26 (l_{max})	1984.38	82.92

Table 1: Transmission energy consumption ($E_{tx}(l)$ in nJ/bit) and transmission range ($R_(l)$ in m) at each power level (l) for the Mica2 motes as a function of power level [10]. Energy dissipation for reception of data is constant ($E_{rx} = 922nJ/bit$).

Param.	Description	Value
N_S	Number of nodes	100
G_r	Radius of network	200 m
r_{max}	Max. trans. range	82.92 m
E_{rx}	Energy dissipation for data reception	922 nJ
$\mathcal{X}$	Problem space	2
S_L	Set of trans. energy power levels	26

Table 2: Simulation Parameters

(MIP) framework to find 'relay' and 'latency' using CPLEX solver [2]. Table 2 summarizes the simulation parameters and terminologies used in our framework.

In the course of experiments, clustering and classification modules of the proposed active learning approach were compared to state-of-the-art ones. We show that the proposed method outperforms the optimization approach in identifying critical nodes in terms of accuracy and computational complexity. We also examined the importance of nodes' characteristics in the performance of the proposed method.

In Figure 3, we show that utilizing LGS can reduce feature space complexity. In the deployed network, the local gravity clustering method partitioned the problem spaces into four clusters, as is shown in this figure. We observed that Cluster 1 contains $N_C - 2 = 18$ critical nodes and Cluster 2 has the remaining critical nodes. Therefore, these two clusters are selected as the labeled ($\mathcal{L}$) and unlabeled ($\mathcal{U}$) set. The cardinality of the new problem space is reduced in a way that 30% of nodes are critical ones. Compared to the original problem space, we observed a 1.33 times increase in critical nodes' cardinality and a 15% decrease in non-critical ones, which helps in solving the imbalance problem. It is worth mentioning that there is a probability of appearing one critical node in other clusters. However, the error of overlooking this instance is much less than the error of biasing in the classifier as a result of increasing the number of non-critical nodes in the training set. To show the efficiency of utilizing LGC, we also clustered problem space with the k-means [40], where the initial seed was set to 2, 3, and 4, respectively. Table 3 shows the distribution of nodes in each cluster. It is obvious that applying k-means brings subjective results, which makes it inappropriate to be used in the proposed active learning approach.

[2]C Programming Language EXtended (CPLEX). Available: `https://www.gams.com/latest/docs/S_CPLEX.html`.

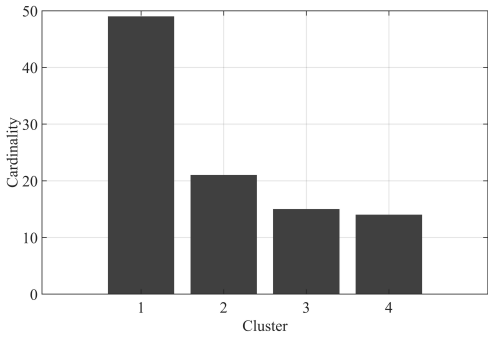


Figure 3: Results of applying LGC to the deployed network. Four clusters are formed, where Cluster 1 contains $N_C - 1$ critical nodes and Cluster 2 contains another critical node.

	Cluster 1	**Cluster 2**	**Cluster 3**	**Cluster 4**
seed = 2	12/76	8/34	NA	NA
seed = 3	5/21	9/54	6/25	NA
seed = 4	0/30	6/25	2/15	12/30

Table 3: Distribution of critical nodes in different clusters as the result of applying the k-means clustering approach.

The classification module of the proposed active learning approach is equipped with a linear SVM [41]. To demonstrate the efficiency of the proposed active learning approach in identifying critical nodes, we compared it with the case that a linear SVM solely applied to the total problem space $\mathcal{X}$. Table 4 tabulates confusion matrices for both proposed and compete methods.

			Predicted class		**TP rate**	**FN rate**
			Critical	**Non-critical**		
Active learning	**True class**	**Critical**	15	4	78.94%	21.06%
		Non-critical	1	69	98.58%	1.42%
Linear SVM		**Critical**	8	12	40.00%	60.00%
		Non-critical	2	78	97.5%	2.5%

Table 4: Results of utilizing the proposed active learning approach and a linear SVM in identifying critical nodes.

Table 5 shows the comparison of both the proposed active learning approach and linear SVM methods in terms of evaluation measures. In this study, we set $w_0 = 0.7, w_1 = 0.3$ as the assigned weight to each class.

	$P_{weighted}$	$R_{weighted}$	$F1_{weighted}$
Proposed method	84.83%	61.11%	91.70%
Linear SVM	57.25%	41.63%	69.14%

Table 5: The comparison of Proposed method and Linear SVM in terms of evaluation measures

In our experiments, we observed that LGC assigned one critical node to Cluster 3, resulting in a new type of error. We plotted nodes according to their (c_x, c_y)-coordinates to see if the coordinate can affect this assignment. Figure 4 shows the scatter plot of nodes that are clustered in Cluster 1 and Cluster 3. We selected these two clusters as Cluster 1 contains the majority of critical nodes, and the majority of instances in Cluster 3 are non-critical nodes. It is evident that the (c_x, c_y)-coordinate cannot solely change the distribution of clusters. Otherwise, there are points in both clusters that have an approximately equal Euclidean distance and could be misassigned.

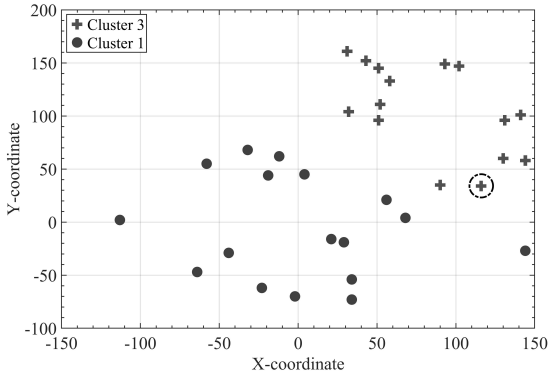


Figure 4: Scatter plot of nodes' location. • and + are used to point to Cluster 1 and Cluster 3 instances, respectively. The dashed circle highlights the node that is a critical node and was assigned to Cluster 3.

ID	Coordinate		Connectivity	Dist-to-BS	Relay
	x	y			
6	144	58	6	154.7094	1
18	32	104	16	109.0566	2
19	90	35	9	96.87996	3
21	43	152	12	157.748	1
22	31	161	13	164.1266	1
61	102	147	10	178.7874	1
71	131	96	10	162.1654	1
76	141	101	6	172.7776	1
78	52	111	13	123.1841	3
79	93	149	10	175.0799	1
85	51	96	15	108.8205	2
90	51	145	10	154.032	1
91	58	133	11	145.1135	1
93	130	60	6	143.6442	1
95	116	34	10	120.5214	4

Table 6: Cluster 3 instances, where the highlighted row shows the critical node.

We compared the features of this critical node with the features of nodes assigned to Cluster 1. Tables 6 and 7 show characteristics of sensor node instances belonging to Cluster 1 and Cluster 3, respectively. The highlighted row in Table 6 shows the critical node assigned to Cluster 3 and highlighted rows in Table 7 show nodes that could be candidates for assigning to Cluster 3. According to these tables, nodes' characteristics such as the node' coordinate, connectivity, distance to the BS (*i.e.,* Dist-to-BS), and relay have meaningful impacts to categorize the node role to be considered as a critical node or non-critical node. For example, Table 6 shows the non-critical nodes' characteristics, however, node 95 is listed among critical nodes after evaluation of its characteristics and we will discuss the reason below.

ID	Coordinate		Connectivity	Dist-to-BS	Relay
	x	y			
7	-19	44	15	48.24708	5
13	-64	-47	12	79.78816	7
26	34	-73	14	80.31338	11
28	-58	55	16	80.23978	7
31	-23	-62	15	66.10476	9
36	-44	-29	15	52.63024	13
38	-12	62	14	63.59438	5
39	56	21	11	59.38667	8
59	68	4	9	68.59481	6
64	144	-27	11	146.8928	5
68	-32	68	15	75.38416	14
70	34	-54	14	63.92478	4
75	21	-16	12	26.56586	3
80	-113	2	13	113.0047	2
88	-2	-70	13	70.26103	3
99	4	45	14	45.28009	3
100	29	-19	12	34.84477	1

Table 7: Instances belong to Cluster 1 where the highlighted rows are the possible candidates to be assigned to Cluster 3.

In our system deployment, the maximum transmission range is set to 82 m which enables the relay of data to neighboring nodes. For nodes 59 and 95, their proximity to the BS falls within the range conducive to transmission via two hops, while node 59 can transmit data with just one hop. The connectivity among these three nodes exhibits notable similarities. However, concerning relay characteristics, node 95 demonstrates a lower value compared to nodes 64 and 59. The LGC algorithm initially prioritizes nodes with greater distances to the BS as non-critical node candidates. Subsequently, it evaluates relay and connectivity parameters, favoring nodes with lesser values as potential candidates for clusters predominantly comprising non-critical nodes. Consequently, the characteristics of node 95 inherently predispose it to a particular type of error. Nevertheless, it is apparent from the evaluation criteria that the proposed methodology not only addresses this issue but also can handle the imbalanced nature of data adequately.

7 Conclusion

The advent of 5G technology has brought the imperative of EE architectures into sharp focus, particularly to accommodate growing demands for increased capacity, faster data rates, and enhanced QoS. EE can be understood differently depending on the application domain, falling broadly into three categories: Information Theory, Sensor Networks, and Cellular Networks. In this paper, we concentrated on the EE metrics pertinent to IoT networks with the objective of prolonging network lifetime.

In the field of IoT networks, several key characteristics, including hop-count, user's location, allocated power, and relay, play a significant role in determining EE. However, determining these factors can be computationally intensive and energy-consuming. To address this issue, we used an active learning approach that can quickly identify critical nodes in IoT networks. We assumed that the network maintains strong connectivity even when critical nodes are removed, so we focused on network latency as the performance metric. Our proposed active learning framework effectively minimizes bias towards non-critical nodes and reduces the retraining effort required to adapt to the network's dynamic nature.

Our method synergizes clustering and classification modules to iteratively reduce data requirements in a standard supervised learning setup, thereby effectively tackling the challenge posed by uninformative examples. Experimental results indicated that our active learning-based approach not only offers a computationally efficient way to identify critical nodes in IoT networks but also provides the scalability and adaptability needed for large-scale, dynamic environments. Its robustness in handling the complex, evolving nature of IoT systems makes it a promising avenue for further research, particularly in integrating machine learning techniques to improve network resilience and EE.

Acknowledgement

This work has been partially supported by the BODYinTRANSIT project as part of the European Union's Horizon 2020 research and innovation programme under grant agreement No 101002711 and the projects AEON-ZERO (TSI-063000-2021-52), FREE-6G (TSI-063000-2021-144), AROMA3D (TSI-063000-2021-70/71), 6G-OASIS (TSI-063000-2021-24), SUCCESS-6G (TSI-063000-2021-39/40/41) and ADV5GTWN (TSI-063000-2021-112/113/114) under the UNICO5G-RPTR programme.

References

[1] Tao Yu, Shunqing Zhang, Xiaojing Chen, and Xin Wang. A novel energy efficiency metric for next generation green wireless communication network design. *IEEE Internet of Things Journal*, 2022.

[2] Praveen Gorla, Anuj Deshmukh, Sandeep Joshi, Vinay Chamola, and Mohsen Guizani. A game theoretic analysis for power management and cost optimization of green base stations in 5g and beyond communication networks. *IEEE Transactions on Network and Service Management*, 19(3):2714–2725, 2022.

[3] Xiaomin Qi, Shahid Khattak, Alam Zaib, and Imdad Khan. Energy efficient resource allocation for 5g heterogeneous networks using genetic algorithm. *IEEE Access*, 9:160510–160520, 2021.

[4] Hyun Jung Park, Hyeon Woong Kim, and Sung Ho Chae. Deep-learning-based resource allocation for transmit power minimization in uplink noma iot cellular networks. *IEEE Transactions on Cognitive Communications and Networking*, 9(3):708–721, 2023.

[5] Basman M Al-Nedawe, Raad S Alhumaima, and Wisam Hasan Ali. On the quality of service of next generation green networks. *IET Networks*, 11(1):1–12, 2022.

[6] Ericsson. Ericsson Mobility Report. Mobile data traffic outlook, 2020.

[7] Ishfaq Bashir Sofi and Akhil Gupta. A survey on energy efficient 5g green network with a planned multi-tier architecture. *Journal of Network and Computer Applications*, 118:1–28, 2018.

[8] Muhammad Usama and Melike Erol-Kantarci. A survey on recent trends and open issues in energy efficiency of 5g. *Sensors*, 19(14), 2019.

[9] Shie Wu, Rui Yin, and Celimuge Wu. Heterogeneity-aware energy saving and energy efficiency optimization in dense small cell networks. *IEEE Access*, 8:178670–178684, 2020.

[10] Huseyin Ugur Yildiz, Bulent Tavli, Behnam Ojaghi Kahjogh, and Erdogan Dogdu. The impact of incapacitation of multiple critical sensor nodes on wireless sensor network lifetime. *IEEE Wireless Communications Letters*, 6(3):306–309, 2017.

[11] Behnam Ojaghi Kahjogh, Ilker Demirkol, Davide Careglio, and Jordi Domingo Pascual. The impact of critical node elimination on the latency of wireless sensor networks. In *2017 Ninth International Conference on Ubiquitous and Future Networks (ICUFN)*, pages 182–187. IEEE, 2017.

[12] Kechen Zheng, Xiao-Yang Liu, Luoyi Fu, Xinbing Wang, and Yihua Zhu. Energy efficiency in multi-hop wireless networks with unreliable links. *IEEE Transactions on Network Science and Engineering*, 7(1):576–588, 2020.

[13] Fanny Parzysz, Mai Vu, and François Gagnon. Energy minimization for the half-duplex relay channel with decode-forward relaying. *IEEE Transactions on Communications*, 61(6):2232–2247, 2013.

[14] Yulong Zou, Jia Zhu, and Xiao Jiang. Joint power splitting and relay selection in energy-harvesting communications for iot networks. *IEEE Internet of Things Journal*, 7(1):584–597, 2020.

[15] Xiaojin Ding, Yulong Zou, Xiaoshu Chen, Xiaojun Wang, and Lajos Hanzo. Secrecy outage and diversity analysis of multiple cooperating source-destination pairs. *IEEE Transactions on Vehicular Technology*, 69(7):7648–7662, 2020.

[16] Orhan Dagdeviren, Vahid Khalilpour Akram, and Bulent Tavli. Design and evaluation of algorithms for energy efficient and complete determination of critical nodes for wireless sensor network reliability. *IEEE Transactions on Reliability*, 68(1):280–290, 2018.

[17] Behnam Ojaghi, Ferran Adelantado, Elli Kartsakli, Angelos Antonopoulos, and Christos Verikoukis. Sliced-ran: Joint slicing and functional split in future 5g radio access networks. In *ICC 2019-2019 IEEE International Conference on Communications (ICC)*, pages 1–6. IEEE, 2019.

[18] Zhiqiang Wang, Zhiwen Yu, CL Philip Chen, Jane You, Tianlong Gu, Hau-San Wong, and Jun Zhang. Clustering by local gravitation. *IEEE transactions on cybernetics*, 48(5):1383–1396, 2018.

[19] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[20] Yun Li, Chao Liao, Yong Wang, and Chonggang Wang. Energy-efficient optimal relay selection in cooperative cellular networks based on double auction. *IEEE Transactions on Wireless Communications*, 14(8):4093–4104, 2015.

[21] Yulun Cheng, Jun Zhang, Jing Zhang, Haitao Zhao, Longxiang Yang, and Hongbo Zhu. Small-cell sleeping and association for energy-harvesting-aided cellular iot with full-duplex self-backhauls: A game-theoretic approach. *IEEE Internet of Things Journal*, 9(3):2304–2318, 2022.

[22] Muhammad Umar Farooq, Xingfu Wang, Ammar Hawbani, Asad Khan, Adeel Ahmed, Saeed Alsamhi, and Bushra Qureshi. Power: probabilistic weight-based energy-efficient cluster routing for large-scale wireless sensor networks. *The Journal of Supercomputing*, 78(10):12765–12791, Jul 2022.

[23] Archana Bomnale and Avinash More. Node utilization index-based data routing and aggregation protocol for energy-efficient wireless sensor networks. *The Journal of Supercomputing*, Dec 2023.

[24] Jie Tang, Ziyao Peng, Daniel K. C. So, Xiuyin Zhang, Kai-Kit Wong, and Jonathon A. Chambers. Energy efficiency optimization for a multiuser irs-aided miso system with swipt. *IEEE Transactions on Communications*, 71(10):5950–5962, 2023.

[25] Yang Huang, Chenghao Zhu, Liang Wu, Zaichen Zhang, and Jian Dang. Energy-efficient hybrid precoding with a grouped-adaptive-connected structure. In *2023 21st International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, pages 95–102, 2023.

[26] Mobile VCE. Power amplifiers for 4g and beyond - managing the efficiency, bandwidth and linearity tradeoff. `http://www.mobilevce.com/green-radio`, Accessed October 2019.

[27] Rumeng Tan, Ying Shi, Yingying Fan, Wentao Zhu, and Tong Wu. Energy saving technologies and best practices for 5g radio access network. *IEEE Access*, 10:51747–51756, 2022.

[28] Line M. P. Larsen, Henrik L. Christiansen, Sarah Ruepp, and Michael S. Berger. Toward greener 5g and beyond radio access networks—a survey. *IEEE Open Journal of the Communications Society*, 4:768–797, 2023.

[29] Chathurika Ranaweera, Elaine Wong, Ampalavanapillai Nirmalathas, Chamil Jayasundara, and Christina Lim. 5g c-ran architecture: A comparison of multiple optical fronthaul networks. In *2017 International Conference on Optical Network Design and Modeling (ONDM)*, pages 1–6, 2017.

[30] Massimiliano Maule, Ojaghi Kohjogh, and Farhad Rezazadeh. *Advanced Cloud-Based Network Management for 5G C-RAN*, pages 371–397. Springer International Publishing, Cham, 2022.

[31] Behnam Ojaghi, Ferran Adelantado, and Christos Verikoukis. On the benefits of vdu standardization in softwarized ng-ran: Enabling technologies, challenges, and opportunities. *IEEE Communications Magazine*, 61(4):92–98, 2023.

[32] Wenwei Yue, Peiang Zuo, Wengang Li, Yao Zhang, Yunfeng Zhang, Changle Li, and Jun Huang. Critical nodes identification: A non-cooperative method for unknown topology information in ad hoc networks. *China Communications*, 20(7):217–232, 2023.

[33] Muhammed Cobanlar, Huseyin Ugur Yildiz, Vahid Khalilpour Akram, Orhan Dagdeviren, and Bulent Tavli. On the tradeoff between network lifetime and k-connectivity-based reliability in uwsns. *IEEE Internet of Things Journal*, 9(23):24444–24452, 2022.

[34] Zuleyha Akusta Dagdeviren, Vahid Khalilpour Akram, Orhan Dagdeviren, Bulent Tavli, and Halim Yanikomeroglu. k-connectivity in wireless sensor networks: Overview and future research directions. *IEEE Network*, pages 1–7, 2022.

[35] Vahid Khalilpour Akram and Onur Ugurlu. Detecting the most vital articulation points in wireless multi-hop networks. *IEEE/ACM Transactions on Networking*, 31(5):2389–2402, 2023.

[36] Guangjie Han, Juntao Tu, Li Liu, Miguel Martínez-García, and Yan Peng. Anomaly detection based on multidimensional data processing for protecting vital devices in 6g-enabled massive iiot. *IEEE Internet of Things Journal*, 8(7):5219–5229, 2021.

[37] Alireza Sepas-Moghaddam, Alireza Arabshahi, Danial Yazdani, and Mohammad Mahdi Dehshibi. A novel hybrid algorithm for optimization in multimodal dynamic environments. In *2012 12th International Conference on Hybrid Intelligent Systems (HIS)*, pages 143–148. IEEE, 2012.

[38] Mohammad Mahdi Dehshibi, Mohamad Sourizaei, Mahmood Fazlali, Omid Talaee, Hossein Samadyar, and Jamshid Shanbehzadeh. A hybrid bio-inspired learning algorithm for image segmentation using multilevel thresholding. *Multimedia Tools and Applications*, 76(14):15951–15986, 2017.

[39] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[40] Hao Yu, Hui Chen, Shengjie Zhao, and Qingjiang Shi. Distributed soft clustering algorithm for iot based on finite time average consensus. *IEEE Internet of Things Journal*, 8(21):16096–16107, 2020.

[41] Johan AK Suykens and Joos Vandewalle. Least squares support vector machine classifiers. *Neural processing letters*, 9(3):293–300, 1999.