Angle Based Critical Nodes Detection (ABCND) for Reliable Industrial Wireless Sensor Networks

Shailendra Shukla (sshukla2000@gmail.com)
MNNIT Allahabad: Motilal Nehru National Institute of Technology https://orcid.org/0000-0002-8316-8460

Research Article

Keywords: Critical Node Detection, Reliability, Industrial Wireless Sensor Networks(IWSN), Wireless sensor network (WSN)

Posted Date: September 20th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-2068881/v1

Angle Based Critical Nodes Detection ($ABCND$) for Reliable Industrial Wireless Sensor Networks

Shailendra Shukla

Received: date / Accepted: date

Abstract Node failure in the Wireless Sensor Networks (WSN) topology may lead to economic loss, endanger people, and cause environmental damage. It is found that reliability and security are the two most essential concerns in IWSN. Reliability can be achieved by adequately managing network topology using the structural approaches, where the critical wireless sensor nodes are precisely detected and protected. In this paper, we address the problem of critical node detection and present two-phase algorithms. In Phase-I, a $2D$ critical node (C-N) detection algorithm is proposed, which uses only $RSSI$ information of the neighbor. This algorithm is also extended to work in $3D$ topology. In Phase II, a correlation-based reliable approach is proposed to increase the node resilience against node failure to achieve low overhead. The proposed algorithms ($ABCND$) requires *mathcalO*(*log*(N)) time for convergence and *mathcalO*(*delta*(*logN*)) for critical node detection , N represents the number of IoT devices, and *delta* is the cost required to forward and backward message. We compare our algorithm with the current state-of-the-art on C-N detection algorithms. The simulation results show that the proposed $ABCND$ consumes 50% less energy to detect C-N and 90% to 95% reliable C-N.

Keyword: Critical Node Detection, Reliability, Industrial Wireless Sensor Networks(IWSN), Wireless sensor network (WSN).

Computer Science and Engineering Department
MNNIT Allahabad
Email: ss@mnnit.ac.in

1 Introduction

According to the reported compound annual growth rate (CAGR), industries are estimated to grow by 9.9% even after covid-19 impact [4]. The application of WSN has also increased with the scalability of industrial equipment. It is anticipated to increase by 5 to 6 times in five years [1]. WSN has significant involvement in the Industries [5], as it improves the process control and monitoring applications [6], [7]. Recent development in 802.15.4/Zigbee standards [8], [9], power efficiency [10], small size, and embedded computing has given new dimensions to the Industry management. Another reason for the growth of WSN is the adaptability and reliability of protocols and standards for most industrial uses and the easy process of automation.
Unlike consumer applications, where cost is often the most critical system attribute, industrial applications typically rate reliability and security at the top of the list. Hence, Industrial Wireless Sensor Networks ($IWSN$) face various challenges such as real-time data communication[13], robustness [14], energy [10], reliability [11], and fault tolerance [12]. The modus operandi of IWSN is that the sensors collect data from the indoor or outdoor environment and deliver it to a central node or controller for processing, decision making, and control. IWSN usages the IoT based on communications [15][16] where the internet connection is either with randomly selected neighbor nodes or with some IEEE standards algorithms (RFC 6550:RPL-based Industrial Networks [18]). However, due to the random deployment of sensor nodes, multihop communication [19]. Limited battery lifetime [20] can cause node failure, higher latency [21], and power consumption [20]. A trivial way to overcome this problem is to deploy a centralized node within the network topology. The centralized Node performs

various computation tasks on collected data for various operation management [22]. In comparison to other nodes, the failure of the central Node is more likely to impact the reliability and capability of networks. The maintenance of such nodes is costly, and resurrection from failure or downtime requires lots of message exchange between the nodes, which is not good for energy resilient industrial applications. The significant limitations of the Industrial wireless sensor networks (IWSN) network are 1. Due to the Low power and Lossy Networks (LLN) nature and limited resources of IWSN, each Node has to participate in the data collection to reduce the burden on the single Node. In a large network like agriculture industries, there may be the chance of message congestion [23], latency, and end-to-end message delay, as all nodes, will transmit data to one point (center or sink Node). 2. In large networks, sometimes a failure of particular nodes can potentially impact an asymmetrically high number of different sensor nodes' connectivity, affecting data availability. So node failures[24] may disconnect all paths between the sink and other sensor nodes. If a particular critical node stops working, many nodes may be separated from the network. Critical Node (*C-N*) is crucial for the networks as it helps in discovering the unknown geometry and topology of an IWSN, which provides salient information for underlaying its environment and the efficient operation of networks.

The reported work on the critical node detection, like Ancestral Knowledge-Based Bridge Detection Algorithm $ABIDE$[33], proposes an algorithm for the critical node detections. However, they are limited to only cut-vertex(articulation vertex detection-based approaches). The algorithm is concatenated with the breadth-first search (BFS) and works in two steps, forward and backward steps. The purpose of the algorithm is to detect the bridge in the network. The algorithm requires the broadcasting of packets, $\mathcal{O}(\delta N)$ of the incoming message and $\mathcal{O}(\delta D log_2(N))$ of the number of bytes sent are transmitted during communications. Partition detection and recovery algorithms ($PADRA$) and ($PADRA+$) [25] use the Connected Dominant Set (CDS) (C) approach to Cut vertex(C-V) detection. A Cut vertex [26] are nodes or links whose removal can split the graph into two (or more) separate components. Figure 1 shows the cut vertices (node 5 or node 7), where removing them splits the network into two parts, BC1 and BC2, so 5 and 6 are cut vertices. However, the algorithm does not guarantee the retrieval of every *C-V* in the network. The algorithm depends on two rules a) Uncritical nodes and b) Dominator nodes with the degree of 1 as *C-V*

Algorithm in $CDSCUT+$ and E-$CDSCUT$ [35] also depends upon the connected dominating set(CDS) and requires broadcast ($dept_u + 1$) for critical node detection. The computational complexity of $ABIDE$ is $\mathcal{O}(\delta^2)$, and message complexity is $\mathcal{O}(clog_2 n)$ where δ is the maximum node degree, c is the critical node count, and n is the node count. Reported work on the critical node (*C-N*) selection algorithms can only detect the articulation points. However, the places like nuclear thermal plants, oil fields, refineries, and coal mines require detecting boundary nodes (*B-N*) since the peripheral node is important in surveillance and perimeter activity detection. The boundary nodes (*B-N*) can be defined as a set of nodes that lies at the perimeter or a)A node $'E'$ is a boundary node if its degree is less than 2 ($\forall v \in$B$-$N $\implies$ deg(E)$\leq$ 2). b) a node 'E' is boundary node iff its degree deg'E' ≤ 3 and neighbour of 'E' are non colinear points $A, B, C \in$ V such that $a, b, c \in$ neighbor of (E) and the nodes A, B, C forms a triangle that encloses the node $E(\forall E \in$B$-$N $\iff$ deg(E)$\leq$ $3 \wedge neighborofE$ non collinear,illustrated in Figure 3.a). So the limitation of cut-vertex is that they only consider the network topology from inside and cannot work properly for any activity occurring at the periphery of networks.

None of the papers in specific address the critical node detection problem either they focused on the boundary node detection or cut vertex detection. However, one of the naive approaches could be to use the existing *C-V* algorithms like [27] and combine it with boundary node (*B-N*) detection algorithms [28] for critical node *C-N* detection. The drawback of using two different algorithms for Critical node detection is increased computation and communication costs. IWSN are LLN [15][16] hence it will increase the overhead on energy and increase in buffer size for individual nodes. If the nodes have mobility and churn rate, it becomes difficult to be implemented in an IWSN environment. So, instead of finding a critical node using a separate algorithm. In this paper, we propose a distributed algorithm "Angle Based Critical Node Detection for *C-N* Detection ($ABCND$), which includes both the Cut Vertex (*C-V*) detection[27]) and Boundary Nodes (*B-N*) detection[28] in IWSN. The significant contributions of this paper are as follows: 1) A hybrid approach for *C-N* detection is proposed for $2D$ and then extended for $3D$ IWSN topology. The *C-N* detection algorithm is free from any costly hardware, graph embedding, graph planarization, and known initial anchors (or bootstrap nodes) for initialization. 2) To ensure the coverage and reliability of the critical node, a correlation-based approach is proposed 3) The proposed algorithm is implemented over the RPL protocol. The strength of this work is that it is practical as it requires only $RSSI$). It is distributed in nature (no global information required)

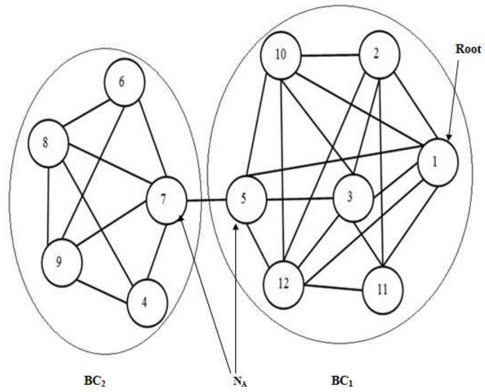

Fig. 1 Set N_A={5,7} as cut vertex.

and detects a reliable C-N in less energy with reduced false positivity. The complexity of proposed algorithms $ABCND$ has a time complexity of $\mathcal{O}(\log(N))$ and computation cost of $\mathcal{O}(\delta(\log N))$ where N is the number of nodes in networks and δ is the total number of forward and the backward message. The remainder of this paper is organized as follows. We have briefly summarized the related work in Section 2. Section 3 presents the system model, problem formulation, and the proposed algorithm ($ABCND$ algorithms) is proposed in Section 4. Section 5 covers the performance evaluation of our work. Finally, the proposed work concludes with future work in Section 6.

2 Related work

Reported work on the critical node detections is limited to either cut vertex or boundary node detection algorithms. None of the papers simultaneously considers the boundary nodes and cut vertex as a critical node detection problem. The related work section is divided into two subsections a) Cut-Vertex detection and b) Boundary node detection for C-N detection. Cut-Vertex detection (aka articulation vertex, often referred to in the literature as the cut vertex). Cut-Vertex is those nodes node or link that disconnects the graph into two (or more) separate components if removed. The network is always at risk of being split into several separate components due to node failure (existence or articulation points in the networks) or topology change(mobility of nodes). Boundary node detection is essential to be discovered the network's topology. The boundary node helps monitor the topological perimeters of strategic and sensitive sites like nuclear sites, oil fields, coal mines, agriculture, and other large industrial sites.

The algorithm in Ancestral Knowledge-Based Bridge Detection Algorithm $ABIDE$ [33]proposes an ancestral knowledge-based bridge detection algorithm (articulation vertex detection-based approaches). The algorithm is integrated with the Breadth-First Search (BFS), and its operation requires two steps, forward and backward steps. The forward step creates spanning BFS by broadcasting between all the nodes in a WSN. In the backward step, bridges are detected by converging backward casting messages to the sink nodes and terminates. The limitation of this work is that it only detects a bridge in the network, and it requires broadcasting in the networks with $O(\delta N)$ received and overheard message and $O(\delta D log_2(N))$ transmitted byte count. Partition detection and recovery algorithm ($PADRA$) and ($PADRA$+) [25]is a distributed approach to find the Cut-Vertex(C-N. The connected dominating set is modified to detect Cut-Vertex in this approach. The algorithm does not guarantee the detection of all the Cut-Vertex in the networks. It depends only on two CDS rules a) ordinary nodes are noncritical and b)dominator with degree 1 is a critical node. Connected Dominating Set-based Cut detection $CDSCUT$+ and its extended version E-$CDSCUT$ [27] proposes an approach to detect the cut vertex detection in the network. The approaches use the connected dominating set(CDS), and Depth First Search(DFS) approaches to determine the cut vertex. CDS, in itself, is an NP-complete problem. The limitation of this work is that Nodes cannot change their location, the dominating set is in prior known by the neighbor nodes, and ECDSCUT depends on broadcast ($dept_u$+1). The computational complexity of the algorithm is $\mathcal{O}(\delta^2)$. Message complexity is $\mathcal{O}(clog_2 n)$ where δ is the maximum node degree, c is the critical node count, and n is the node count.

None of the existing reported works address the cut-vertex detections in 3-dimensional topology. However, some work on boundary detection (B-N) can be used to detect the important nodes in WSN. Boundary detection (B-N) algorithm can be classified as a) connectivity-based approach and b) Geometric based approach. The connectivity-based approach considers the known topological information (connectivity between neighbors), flooding, and broadcasting for B-N detections. The Least Polar-Angle Connected Node Algorithm (D-LPCN) [36] requires high energy consumption due to extra hardware put up. The complexity of D-LCPN does not i) include the starting node computation cost, ii) leader election algorithm computation, and extra bits for fields like AC(ask for coordinate), CS(send coordinate), and SN(select node). Paper [29] identifies the 3D B-N using connectivity, neighbor information and frequent flooding. $CABET$ considers the B-N nodes with fewer neighbors than their internal counterparts. The algorithm $CABET$ assumes that network nodes are evenly distributed. If the nodes are deployed randomly, then $CABET$ encounters false positivity.

In [30], constructs a 3D $Voronoi$ mesh in the form of a *tetrahedral* structure to detect the *B-N* in the 3D sensor network. The algorithm is distributed and does not rely on any specific communication model. The limitation of [30] is that it requires pre-known *landmarks* information and frequent use of shortest path computations. The algorithm involves complicated probing procedures to infer the boundary information and takes $\mathcal{O}(N)$ messages to compute the 3D *B-N*. Paper [31] uses the $feature\ extraction$ techniques with $2D$ embedding [30] and distance information. In $2D$ embedding, it converts the $3D$ graph into $2D$ planer graph. In algorithm surfing on 3D surfaces $SURF$ [34] constructs *isocontour*, *Reeb graph*, and *maximum cut-set* to divide the network into different regions then the connectivity-based algorithm is used to detect *B-N*. Node connectivity is used in *BLOW-UP* [32] to build a unit tetrahedron cell in the network. If any node lies in any unit tetrahedron cells, it is considered 3D *B-N*. The limitation of [34] and [32] is their implementation in the real world and the assumption of reliability of networks.

The geometrical approach requires the nodes to have an exact position or distance between nodes. In [37], the embedding of a $3D$ graph is projected on the $2D$ surface. The approach faces the misinterpreted as a *B-N* due to projection, so it uses the $Isolated\ Fragment\ Filtering$ (IFF) approach to remove false positives. The limitation of paper[37] is its assumption that nodes have prior global information about coordinates (by creating a *localized coordinate system*) and embedding a 3D network on the 2D plane. In Localized boundary detection and parametrization ($LBDP$) algorithm[38] a *convex-hull* based algorithm [39] is proposed for 3D *B-N* detection. Both [38] and [39] have the constraint-like nodes equipped with GPS and known residual energy. $LBDP$ ignores the coverage and connectivity of boundary regions, which is essential for any surveillance or monitoring system. Chao, et al. in [40] proposed a $Virtual\ Force$ based $Energy\ Hole\ Mitigation$ strategy ($VFEM$) to detect *holes* (*boundary* of nodes form due to void or space) within the networks. $VFEM$ divides the network into several *annuluses* with the same width. A "virtual gravity generated by annulus" algorithm is proposed to optimize the positions of nodes in each *annulus*. The limitation of [40] is it requires the uniform distribution of nodes and known euclidean distance between the nodes. The major limitations of the current reported work are i) None of the critical node detection algorithms includes both cut-vertex and Boundary node detection; ii) seminal works have been done in critical node detection for 3D topology; iii) most of the work has limitations like embedding of 3D to 2D, some hardware dependencies like GPS, known global coordinates and requirement of anchor node(Landmarks) to bootstrap and ignores.

3 System Model and Assumption

We assume that the nodes in IWSN are dynamic (mobile) and randomly placed in relief or area $\in \mathcal{R}^2$. IWSNs have the same communication range, storage space, processing speed, and battery life. The nodes in the network can be represented by the connected graph G= (V, E) , where the vertices $V = \{v_1, v_2, ..., v_n\} \in \mathcal{R}^2$ represents the IWSN device and E represents edges. Edges E_{v_i,v_j} : if $dis(v_i, v_j) \leqslant R_c$ (communication range) where $dis(v_i, v_j)$ is the eucledian distance between the devices v_i and v_j. To maintain the simplicity and convenience , throughout the paper it assumed that sensing range R_s and communication range R_c have the same spherical range just for the simplicity in analysis and to maintain the connectivity.

However, the communication range R_c follows the Quasi unit disc Graph Let graph G consist of set of nodes V in 3D Teritrian plane $\Re^3$ and $d \in [0, 1]$be a parameter. Two nodes (u,v) are considered as connected if and only if the distance between them is less than or equal to d(i.e $E_\{v, u\} \iff$ dis(u,v) $\leq d$). If the range is between 1 and d, then the connectivity between the node is unknown. If the range is greater than 1, there is no connectivity between the nodes. The range d is distance until the signal is robust to both the IWSN devices u and v. A QUDG is equivalent to a unit disc graph (UDG) when the value of d= 1. In the context of QUDG, we assume that the range $'d'$ is independent of the number of nodes. We also assume that the value of d is the same for all the nodes. We have considered that some nodes are not within the mutual communication range of each other. Some nodes may receive the signal, some may not, hidden terminal problems, signal blocks due to wall, etc.

Any device $v_i \in \mathcal{R}^3$ is connected to other nodes if it is within the R_c of other nodes and covered if it is within the sensing range R_s from at least one node. The neighbor set of a sensor node v_i is defined as $Nbr(v_i)$. It is assumed that nodes are unaware of their location information like coordinate and GPS positions, and the network is not equipped with any bootstrap or anchor node. Also, the sink has sufficient resources like energy and memory

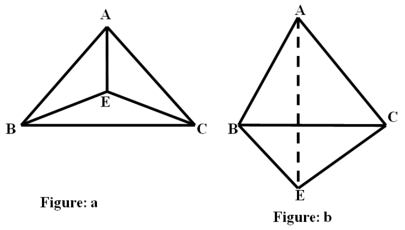

Fig. 2 Node E as boundary node(figure:a) and interior node(figure:b) in 2D

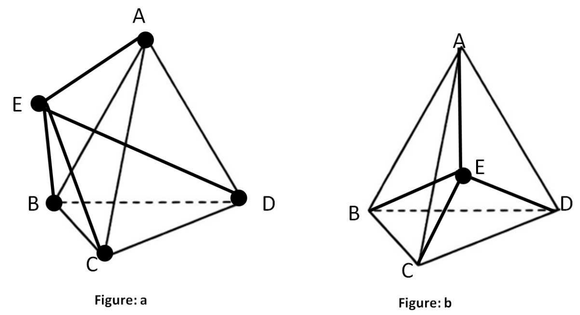

Fig. 3 Node E as boundary node(figure:a) and interior node(figure:b) in 3D

The set of neighbors of sensor node v_i is defined as $Nbr(v_i)$. It also assumes that the node does not know location information such as GPS coordinates and location. The network does not have a bootstrap or anchor node. Each IWSN unit is equipped with advanced sensors that collect information from the environment and transmit it to a receiver, controller or sink node.
Our IWSN devices are configured with IEEE 802.15.4 communication[42], which uses an omnidirectional antenna. An omnidirectional antenna does not form a clear-cut disc. So quasi unit disc communication model for our system is assumed. Our proposed algorithm($ABCND$) requires the distance between the 1-hop neighbor, so the cheaper and readily available distance metric, $RSSI$[43] ($P_r = P_t$ x $(\frac{1}{d})^n$) is considered, . The definitions required for our algorithm are as follows:

Boundary node (B-N): A device in IWSN is called a Boundary node (B-N) $\iff$ it does not lies inside the triangle (for 2D networks) and tetrahedron() volume formed by the at least four nodes(for 3D networks) [41].).

Cut Vertex(C-V): Cut-vertex is those devices in networks whose removal can disconnect the networks. C-V can also be defined as a vertex whose removal from the graph can create more than 1 component of the graph.

Critical Node (C-N): It is defined as the union of set Boundary nodes and Cut Vertex [C-N=B-N $\cup$ C-V]or Node 'v' is C-N if $\exists$ 'v' $\in$ B-N $\vee$ 'v' $\in$ C-V.

The nodes apart from Critical nodes are considered as Interior Node (I-N). Interior Node (I-N) is defined as a node $t \in V$, when at least three nodes $\{A, B, C\} \in$ V such that $A, B, C \in Nbr(t)$ and the nodes $\{A, B, C\}$ forms a triangle that encloses the node t (illustrated in Figure 2.a). We define *interior node criteria* as being enclosed in a triangle formed by three nodes in its neighborhood($\forall t$ $\in$I$-$N $\implies$ $\exists A, B, C | A, B, C \in Nbr(t) \wedge A, B, C$ forms triangle inclose E or $\neg$ B-N=I-N, as illustrated in Figure 2.b) . Throughout the paper we considered the interior node as those node which are not a boundary node (B-N)('E' is a internal node (I-N) as illustrated in Figure 2) or $\neg$ 3D B-N=3D I-N [41].
In this work, a lightweight RSSI-based novel distributed technique is proposed for critical node detection. The proposed algorithm uses only 1-hop neighbor information. The salient feature of our work is as follows:

- The proposed approach only depends on one hop information and is fully distributed. The proposed approach is not limited to the detection of Critical-Node. However, it can also identify the boundary nodes and cut-vertex in IWSN.
- This work proposed two different approaches. The first approach can be used with the 2D topology, which can be extended to the 3D topology. The second approach addresses the reliability and accuracy of detected C-N. Algorithm uses the key ($K_{\aleph}$) and correlation function($\rho_{\aleph}$ [i], and $\rho(\aleph)$) between the nodes for reliability and the angular property (cosine) for accuracy.
- The proposed algorithm is independent of high-level protocol. It requires only network layer (with RPL) and data link layer(RSSI) information. RSSI is prone to noise, so the algorithm uses the angular property to reduce the false positivity.

4 Angle Based Critical Node Selection Algorithm ($ABCND$)

This section puts the modus operandi of critical node selection in 2D and its extension in 3D scenarios [41] . We have assumed that the nodes can detect the distance information using the RSSI value. Assume a smaller subgraph S of four $\{A, B, C, E\}$ non-collinear nodes with complete connectivity (as illustrated in Figure 2). Suppose node E wants to test whether it is a critical

node (C-N) or an interior/normal node(I-N). If node E lies inside the triangle, it is an interior/normal node; otherwise, it is a critical node. The stated Lemma 1 derives the relation to find the interior and critical nodes. Computation of relation is done using inverse cosine formula and the property of inscribing a center point in a triangle between the edges.
Given a triangle △ ABC, if the distance between the edges are known then the angle formed between any two edge $\overline{\text{AB}}$ and $\overline{\text{BC}}$ ($\measuredangle ABC$) can be computed (using inverse cosine formula) as:
$\cos^{-1}\left(\frac{d((A,B))^2+d((B,C))^2-d((A,C))^2}{2d((A,B))d((B,C))}\right)$ where d denotes the distance between any two neighbours.

Lemma 1 *A Node is Interior Node I-N if it satisfies any of the given six relationships.*

1. $\measuredangle$ *EAC* > $\measuredangle$*BAC*
   $\cos^{-1}\left(\frac{(d((E,A))^2+d((C,A))^2-d((E,C))}{2d((E,A))d((C,A))}\right)$
   $> \cos^{-1}\left(\frac{d((B,A))^2+d((C,A))^2-d((C,B))^2}{2d((B,A))+d((C,A))}\right)$
   Similarly,
2. $\measuredangle$ *EAC* > $\measuredangle$ *BAC*
3. $\measuredangle$ *EBA* > $\measuredangle$ *ABC*
4. $\measuredangle$ *ECA* > $\measuredangle$ *ACB*
5. $\measuredangle$ *ECB* > $\measuredangle$ *ACB*
6. $\measuredangle$ *EBC* > $\measuredangle$ *ABC*

else it is a Critical-Node (C-N).

Proof An angle divided into two-part will always be greater than the individual parts(Vacuous proof). Using Angle Addition Postulate: Angle Addition Postulate states that if a point $'E'$ lies in the interior of triangle ABC, then the $\measuredangle$ECA + $\measuredangle$ ECB = $\measuredangle BCA$ from Figure 2 a, but if the point $'E'$ lies outside the $\measuredangle$ECA + $\measuredangle$ ECB > $\measuredangle BCA$. So the above statement contradicts when E lies outside the triangle, and hence node $'E'$ is a boundary node according to the defination 1.

Lemma 2 *A node $'E'$ is said to be Interior Node(Not C-N) if it lies inside the triangle form by any three nodes (inside △ ABC).*

Proof If node $'E'$ is inside the triangle form by any three nodes △ ABC, it is a Normal Node. Suppose node $'E'$ is not a Normal Node, then it is a boundary node. The angle formed by node $'E'$ with other vertex $'A'$,$'B'$,$'C'$ will satisfy Lemma 1. Hence $'E'$ is an interior node (I-N).

Lemma 3 *Minimum node required to make a node 'v' a boundary node(B-N) in QUDG if $deg(v) \leq 6$ and separated by an angle $\measuredangle 60^o$ or they are within the distance d.*

Proof Consider a Graph [as illustrated in fig (3a)], the minimum neighbor required to 6 if the distance of communication is within $|d|$ distance and separated by $\measuredangle 60^o$.

Lemma 4 *Not all boundary nodes (B-N) are Cut-Vertex (C-V) [∄ 'v' ∈ ¬ B-N ⇒ 'v' ∈ C-V].*

Proof Let node $'v'$ belongs to the connected component of C_1 of G. Grow a maximal path P starting from $'v'$.Since $deg(v) = 1$(defination Boundary node), vertex v is one of the two ends, so by maximal path theorm C_1 -v is connected. Thus 'v' is a boundary node (B-N) but not a Cut-Vertex(C-V).

Lemma 5 *All Cut-Vertex (C-V) are boundary nodes (B-N) [∀ v ∈ C-V ⇔ v ∈ C-N].*

Proof Suppose $deg(v) \leq 2$ but v is not a Cut-Vertex. Let bridge $'e'$ of distance $||d||$ joining $'v'$ to neighbour $'u'$ and let C_1 and C_2 be connected component of G that consist of $'v'$ and $'u'$. Since $deg(v) \leq 2$, let $'v'$ be adjecent to node α and β different from $'v'$ and $'u'$ (as illustrated in fig (2a)). Using leema 3 if the $deg(v) \leq 5$ and seperated by disance $> d$ then node is boundary node. In C_1 u is connected to α via $'v'$. This says that the α,β,'v' and 'u' are in same component C_1. As assumed nodee 'v' is not a cot vertex of G, so it is not a Cut-Vertex of C_1 [C_1 -v] is connected. So all cut vertex in C_1 - $'v'$ are connected in same component hence u is also connected in C_1 -v to α and β via some path [contradict our intial assumption]. Since $'v'$ is a cut vertex hence it is a boundary nodes (B-N).

4.1 Description of the Algorithm

The proposed algorithm is distributed in nature and uses the DODAG information request(DIS), the object of information of the DAG (DIO), and the object of update to the destination (DAO) of RPL protocol [18] for message communication. Each nodes creates a neighbor list ($NbrN(\aleph_i)$) table where $\aleph_i$ is any node. The ($NbrN(\aleph_i)$) table contains the Node IP and distance between itself and the neighbor. It uses the $RSSI$ value to determine the distance. The limitation of RSSI is its accuracy, as it is influenced by environmental noise, which affects the measured distance. The inaccuracy can lead to node failure(as a node can block or due to energy depletion). Accuracy of RSSI is maintained by including the angular property. The reliability is achieved by using the key ($K_\aleph$), correlation ($\rho_\aleph$ [i], and $\rho(\aleph)$) are explored between any two nodes. The proposed algorithm works in two phases; phase-I: algorithm-1, a distributed algorithm for self B-N detection, is proposed. In phase II: a centralized algorithm

is proposed to reduce the impact of RSSI error and unreliability.
The node $\aleph$ calculates signal strength using equation $P_r = 10\log\frac{\gamma^2}{(4\pi)2dis^2}$ of neighbour where dis$\leftarrow$ eucledian distance between devices and γ is the wavelength. Every device creates a 2D dataset $NbrN(\aleph)$ contains the data: "node id" along with its "distance" (computed using RSSI), and a key $K_\aleph$ (each node have it's key $K_\aleph$ which is also stored at the sink). Node periodically update its $NbrN(\aleph)$ list and share the list with all its neighbors for update. Computed distance $dis(\aleph_i)$ is then quantized using $RI_\aleph$ [i] $=\frac{1}{8} * \frac{8dis(\aleph_i)}{1}$ (quantization is done using 8 bit). After quantization, shared key $K_\aleph$ is then XOR with $RI_\aleph$ [i] . Now node $\aleph$ will compute $NbrN(\aleph)$, a $2D$ data structure which is consist of *neighbors nodes* computed after $\{NbrN(\aleph_1) \cap NbrN(\aleph_2)...NbrN(\aleph_n)\}$ where $NbrN(\aleph_1)$ to $NbrN(\aleph_n)$ are neighbour list of node $\aleph$. Once $Nb(\aleph)$ is created, node $\aleph$ compute the Lemma 4.1 for B-N detection. If $\aleph$ computes true for Boundary then update boolen B-N to 1 and send the information $[RII_{EN}(\aleph_i)$,$Time$,B-$N_\aleph]$ to sink node for reliability detection.

Algorithm 1 Algorithm to detect the (C-V) and (B-N) excuted at each node.

1: Intilize the node $N_\aleph$
2: Each node detects the $RSSI$ value of neighbour node.
3: $RI(N_\aleph[i]) \leftarrow RI(N_\aleph[i])$ + gain
4: $d(\aleph_i) \leftarrow P_r(RI(N_\aleph[i]))$
5: $RI_\aleph$ [i] $\leftarrow Quantization(d(\aleph_i))$ {Determine the similarty between the recived node by the help of quantization to $RI_\aleph$ [i] }
6: $RII_{EN}(\aleph_i) \leftarrow$ XOR[$RI_\aleph$ [i] , $K_\aleph$]
7: Determine the $Nb(\aleph) = \{NbrN(\aleph_1) \cap NbrN(\aleph_2)...NbrN(\aleph_n)\}$ in sorted order according to length.
8: **if** $Nb(\aleph) \geq 3$ && Lemma 4.1 = $FALSE$ (2D Scenario) **then**
9: Mark the $\aleph$ as I-N node, put B-$N_\aleph$=0.
10: **else**
11: MArk the $\aleph$ as B-N node and put B-$N_\aleph$=1.
12: **end if**
13: Send MSG[$RI_{EN}(\aleph_i)$,$Timer$,B-$N_\aleph$] to Sink node

---

Algorithm 2 is implemented at sink, Sink nodes is capable to perform high computations. So once sink node receives the MSG from the Step:13 of algorithm 1 it then start the separates of $RI_{EN}(\aleph_i)$ and $Time$,B-$N_\aleph$ from MSG. Sink node check for B-$N_\aleph$ value, the reverse compute the $RII_{EN}\aleph[i]$ using $K_\aleph$ key will start if the B-$N_\aleph$ = 1 and store in $RI_\aleph$ [i]. Similarly in reverse computation all the neighbour of $\aleph$RII$_{EN}i[\aleph]$ decreapt it using key K_i] and store it in corresponding $RI_i[\aleph]$. Spearman correlation for $\rho_\aleph$ [i] is calculated using variables $RI_\aleph$ [i],$RI_i[\aleph]$. The $\rho_\aleph[i]$ is computed as $\frac{\text{cov}(\text{R}\,II_\aleph[i],\text{R}\,I_i[\aleph])}{\sigma_{\text{R}\,II_\aleph[i]}\sigma_{\text{R}\,I_i[\aleph])}}$ where covariance is represented by the cov, standard deviations σ is the rank variables and i range from 1 to Nbr($\aleph$). Finally sink computes $\rho(\aleph)$ as $\frac{\sum_{i=1}^{NbrN(\aleph)} \rho_\aleph[i]+\sum_{i=1}^{NbrN(\aleph)} \rho_i[\aleph]}{2NbrN(\aleph)}$, if $\rho(\aleph)$ is high i.e between ϱ and 1, then it means that node is reliable and can be considered as B-N. However if the ρ is between ϱ and -1 represents the low similarity and the node ($\aleph$) can be considered to have inaccurate RSSI value or unreliable node. The parameter ϱ varies according to implementer or scenario.

Algorithm 2 Algorithm to detect the reliability, of the C-V to be executed at the sink node.

1: **Requires:** $RII_{EN}(\aleph_i)$,$Time - stamp$,B-N
2: Separate $RII_{EN}(\aleph_i)$,$Time - Stamp$,C-V
3: **if** B-N ($\aleph$) = 1 **then**
4: $RI_\aleph$ [i] $\leftarrow$($RII_{EN}\aleph[i]$, $K_\aleph$)
5: $RI_i[\aleph] \leftarrow [RII_{EN}i[\aleph]$,$K_i]$
6: $\rho_\aleph$ [i] $\leftarrow \rho(RI_\aleph$ [i],$RI_i[\aleph])$
7: Compute $\rho(\aleph)$
8: **if** $\varrho \leq \rho(\aleph) \geq 1$ **then**
9: Return $\aleph$ is B-N
10: **else if** $-1 \leq \rho(\aleph) \geq \varrho$ **then**
11: Received RI_EN for $\aleph$ has error may be $RSSI$ values is not correctly measured or may be $\aleph$ is untrusted.
12: **else**
13: Node $\aleph$ is trusted node and internal node(I-N).
14: **end if**
15: **end if**

4.2 Complexity Analysis

The complexity of the $ABCND$ algorithm relies on the intersection of the neighbor list and 2D dataset, which is $\mathcal{O}(Nb(N_{\aleph})$. Since the algorithm is using the 2D data set hence the complexity of the intersection will be $\mathcal{O}(N^2)$, and $\mathcal{O}(\frac{(N)!}{3!})$ (Lemma-1). So the overall computation cost is $\mathcal{O}(N^2)+\mathcal{O}(\frac{(N)!}{3!}) = \mathcal{O}(N^3)$. The message size of $ABCND$ algorithm is $\mathcal{O}(NbrN(\aleph))$ bits.

5 Performance Evaluation and Results

The performance evaluation section is divided into two subsections: The first subsection compares the proposed algorithm with the existing state-of-the-art. In the second subsection, we have covered the impact of the proposed algorithm in a faulty network and with RPL protocol[18].

5.1 The Performance of $ABCND$

Proposed algorithm Angle Based Critical Node Detection ($ABCND$) is compared with B-N ($ABIDE$[33]+D-$LPCN$ [36], $CDSCUT$+[35]+D-$LPCN$ [36]) and Cut-Vertex (C-V) approaches. The 50-250 nodes with IWSN capability were deployed randomly in the networks. We have taken one sink node to analyze the parameters network over like send and received byte count. We have implemented the simulation using PROWLER. The nodes are deployed in an area of 1000x1000 m^2, and nodes are increased gradually to analyze the density impact. Nodes in IWSN are configured with the TDMA channel access method with IEEE 802.15.4 MAC. The transmission range is considered as 50 meters. The parameters used to evaluate and compare a) sent byte counts and received byte counts, c) energy consumption, and d) the percentage of correctly detected critical nodes. The simulation is performed for the 20 iterations so that the standard deviation is smaller than the computed mean.

5.1.1 Sent and Received Byte Counts with respect to the Number of Nodes

In this simulation, the density (nodes increased from 50 to 250 nodes) of the networks is increased gradually. Our proposed algorithm's total byte count ($ABCND$) is compared with the $ABIDE$, $CDSCUT+$, and D-$LPCN$ algorithms. It is evident that if the number of nodes increases, then the amount of the message will also increase, as shown in message complexity (Table:1). The result illustrated in Figure 4 indicates that

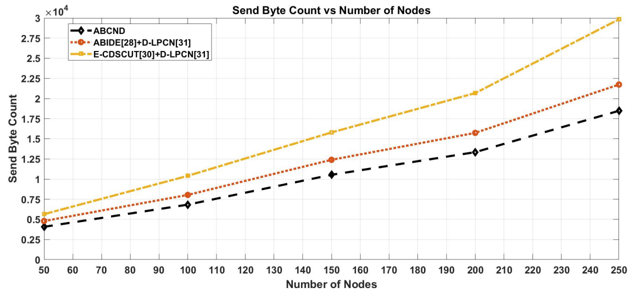

Fig. 4 Comparision of sent byte count w.r.t increase in number of node in network

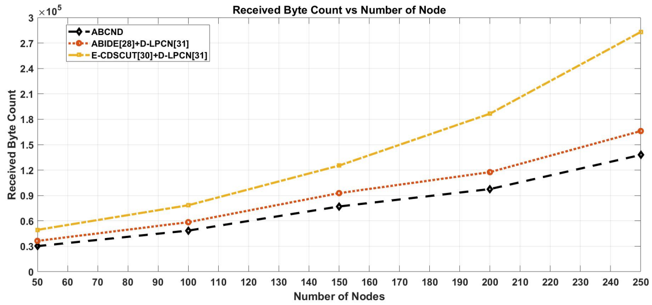

Fig. 5 Comparision of received byte count w.r.t increase in number of node in network

our proposed algorithm($ABCND$) produces 50 % less number of bytes as compared to the $ABIDE$+ D-$LPCN$ and $CDSCUT$+ +D-$LPCN$ algorithms. It can be concluded from the result that the number of sent byte counts increases with the increase in node density. Hence it is not suitable for the low power devices of IWSN as it is increasing the network overhead.

The total received byte count of our proposed algorithm ($ABCND$) is compared with the $ABIDE$, $CDSCUT+$, and D-$LPCN$ algorithms in the Figure 5. The result shows that as the nodes are increased in the network, the received byte count of all the three algorithms increases gradually and linearly. The result confirms the validity of simulation results with the theoretical result where the complexity of the received byte is equivalent to $Nbr(N))(\mathcal{O}(5N\log(N)+2E)$ & $\mathcal{O}(138Nbr(N)))$. The received byte count of our proposed algorithm($ABCND$) is approximately .5 times better than the $ABIDE$ + D-$LPCN$ and 2 times better then the $CDSCUT$ + D-$LPCN$.

5.1.2 Power Consumption

Power consumption is a crucial element to be observed in WSNs deployed in outdoor fields. However, is it essential in indoor ones considering that the indoor IWSN devices can have a continuous power supply. The comparison is made against proposed $ABCND$ and $ABIDE$+ D-$LPCN$ and $CDSCUT$+ D-$LPCN$ algorithms. The

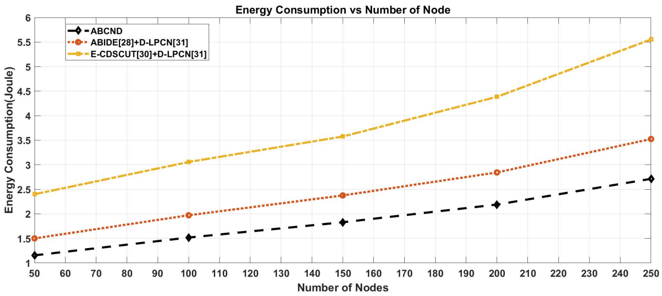

Fig. 6 Comparision of energy consumption w.r.t increase in number of node in network

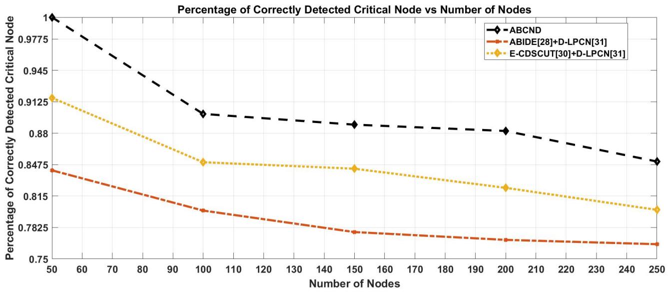

Fig. 7 Percentage of correctly detected critical nodes w.r.t the increase in number of nodes in networks

result illustrated in Figure 6 shows that power consumption increases with the increase in node count in networks due to an increase in send and receive bytes. The result also shows that our $ABCND$ almost requires 50% less energy than the $ABIDE$+ D-$LPCN$ algorithm and three times lesser energy than the $CDSCUT$++ D-$LPCN$ algorithms. The result concludes that our proposed $ABCND$ algorithms show significant improvement over battery-powered and can maximize the network lifetime in higher density.

5.1.3 Percentage of correctly detected critical nodes

The result shown in Figure 7 illustrates the effect of node density increase on the true critical nodes(C-N) detection. The $ABCND$ algorithm is compared with the $ABIDE$ with D-$LPCN$ and $CDSCUT$ with D-$LPCN$ algorithms. It is found the result illustrated in Figure 7 shows that as the density of the node is increased from 5 times the initially deployed nodes(50 nodes). The result clearly shows that the $ABCND$ algorithm loses only 16% of the critical node with the increase in density five times. The number of detected C-N nodes has sharply fallen. The result also shows that the $ABCND$ algorithm can effectively detect more than the 10% of correct C-N compared with $ABIDE$+D-$LPCN$ and 15% more than $CDSCUT$+ D-$LPCN$.

5.2 The Performance of $ABCND$ with RPL protocol in faulty networks

To evaluate our proposed algorithm with other IoT-based algorithms, we have used the Cooja simulator of Contiki 2.7. Contiki is a lightweight and open-source operating system designed specifically for LLN-type WSN devices. To make the realistic IWSN, we deployed Tmote Sky (MSP430-based board)motes. MSP430-based motes use the IEEE 802.15.4 compliant CC2420 radio chip over the MAC layer. For the packet-based routing we have used mobility-based RPL[18] protocol. To know the effective ness of our algorithm in 3D we have randomly (uniform random distribution) and independently placed nodes on a three-dimensional simulation area. The 250 randomly deployed node with 1 sink node in an area of 1000x1000 $units^2$. The number of unreliable/-faulty RSSI value nodes was increased from 0 to 30 nodes. To observe network performance changes, these unreliable nodes continuously vary their $RSSI$ value, randomly from 10% to 20% from the original RSSI. The unreliable nodes were not increased by more than 30. They created many control messages that are difficult to simulate in the cooja. The mean of 20 experiment results was taken with different seeds to get statistically valid results. The range of ϱ is adjusted and compared for three different ranges a) $\varrho = .7$, b) $\varrho = .5$ and c) $\varrho = 0$. The simulation is performed for 1500 sec. The manipulation of $RSSI$ starts after the 4th minute of stimulation so that the RPL protocol is first able to create a DAG. The performance evaluation was based on the following metrics Percentage of correctly detected Boundary nodes and Power Consumption (mW): average power consumption at the network nodes.

5.2.1 Percentage of correctly detected C-N

The result in Figure 8 shows the correctly detected C-N when the number of unreliable node in network were increased. We compare our algorithm at $\varrho = .7$, $\varrho = .5$ and $\varrho = 0$ while keeping unreliable nodes fixed at 10. The result is illustrated in Figure 8, shows that the detected C-N at $\varrho = .5$ is 18% better than the $\varrho = 0$ and $\varrho = .7$. Result also clearly shows that increasing the correlation value does not guarantee the exact C-N will be detected (lead to false positivity).

5.2.2 Energy Consumption

The energy consumption of ISWN is obtained by concatenating the time cycle consumed by CPU, LPM,

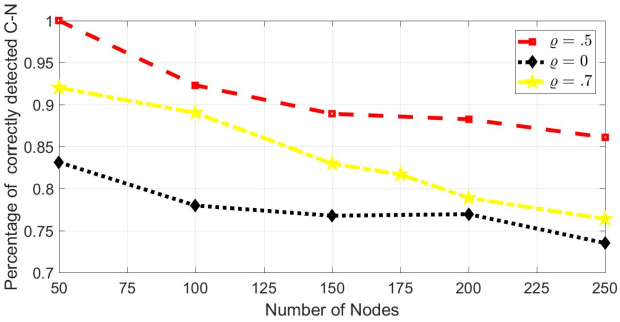

Fig. 8 Percentage of correctly detected *C-N* w.r.t total number of nodes

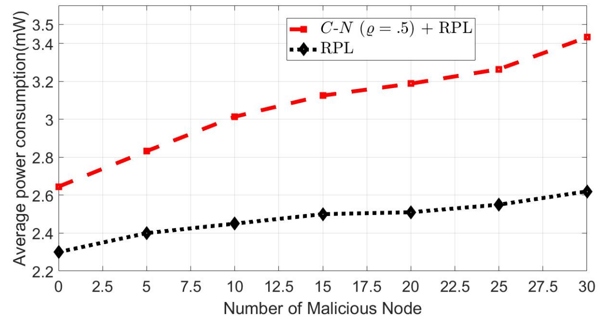

Fig. 9 Comparision of Energy consumption w.r.t unreliable node present in network

TX, and RX. However, the energy consumption of processors (CPU) dominates others. In this analysis, we have considered the power consumed by only CPU. The result in Figure 9 shows that our proposed algorithm slightly increases the energy consumption in the RPL node when executed with the help of RPL[18]. Our proposed algorithm requires 12% to 15% of extra energy to detect reliable 3D *C-N* when used with RPL.

6 Conclusion and Future Work

This paper investigates the problem of *C-N* detection in the IWSN and proposes a reliable and distributed algorithm $ABCND$ to detect the *C-N*. The salient feature of the proposed algorithm works on a geometric approach that is distributed in nature and uses the RSSI value and angular information for distance and angle computation. The angular feature of the algorithm helps in calibrating the RSSI value, which ultimately helps achieve accuracy and precision. The computation cost of the proposed algorithm is $\mathcal{O}(NbrN(\aleph) \log NbrN(\aleph))$ and communication cost as message complexity is $\mathcal{O}(1)$ where $NbrN(\aleph)$ is the neighbor of node $\aleph$. Our simulation results show that the proposed $ABCND$ algorithm can effectively outperform the other know algorithms of WSN like $ABIDE$ and $CDSCUT$ with D-$LPCN$ algorithms. We performed various simulations and compared them with state-of-the-art and prominent IoT protocol (RPL) to validate our algorithm. We plan to evaluate our algorithm for the large-scale IWSN and on real hardware in future work. Moreover, we noticed that the algorithm has some false positivity when implemented in 3D topology, which we plan to address in the future.

Data Availability

The result analysis was performed on the simulator and synthetic data.

References

1. Shen, X., Wang, Z., Sun, Y. (2004, June). Wireless sensor networks for industrial applications. In Fifth World Congress on Intelligent Control and Automation (IEEE Cat. No. 04EX788) (Vol. 4, pp. 3636-3640). IEEE.
2. Moysiadis, V., Sarigiannidis, P., Vitsas, V., Khelifi, A. (2021). Smart Farming in Europe. Computer Science Review, 39, 100345.
3. Haseeb, K., Ud Din, I., Almogren, A., Islam, N. (2020). An energy efficient and secure IoT-based WSN framework: An application to smart agriculture. Sensors, 20(7), 2081.
4. Covid-19 impact on digital agriculture market accessed 31 October 2020 [https://www.marketsandmarkets.com/Market-Reports/covid-19- impact-on-digital-agriculture-market-222616344.html]
5. Rodriguez, P. M., Lizeaga, A., Mendicute, M., Val, I. (2019). Spectrum handoff strategy for cognitive radio-based MAC for real-time industrial wireless sensor and actuator networks. Computer Networks, 152, 186- 198.
6. Ouyang, F., Cheng, H., Lan, Y., Zhang, Y., Yin, X., Hu, J., ... Chen, S. (2019). Automatic delivery and recovery system of Wireless Sensor Networks (WSN) nodes based on UAV for agricultural applications. Computers and Electronics in Agriculture, 162, 31-43.
7. Keshtgari, M., Deljoo, A. (2011). A wireless sensor network solution for precision agriculture based on zigbee technology.
8. Di Francesco, M., Anastasi, G., Conti, M., Das, S. K., Neri, V. (2011). Reliability and energy-efficiency in IEEE 802.15. 4/ZigBee sensor networks: An adaptive and cross-layer approach. IEEE Journal on selected areas in communications, 29(8), 1508-1524.
9. Koubaa, A., Cunha, A., Alves, M. (2007, July). A time division beacon scheduling mechanism for IEEE 802.15. 4/ZigBee cluster-tree wireless sensor networks. In 19th Euromicro Conference on Real-Time Systems (ECRTS'07) (pp. 125-135). IEEE.
10. Tang, Q., Yang, L., Giannakis, G. B., Qin, T. (2007). Battery power efficiency of PPM and FSK in wireless sensor networks. IEEE Transactions on Wireless Communications, 6(4), 1308-1319.
11. Jia, J., Chen, J., Deng, Y., Wang, X., Aghvami, A. H. (2021). Reliability optimization for industrial WSNs with FD relays and multiple parallel connections. Journal of Network and Computer Applications, 179, 102993.

12. Arapoglu, O., Dagdeviren, O. (2020). A fault-tolerant and distributed capacitated connected dominating set algorithm for wireless sensor networks. Computer Standards Interfaces, 103490.
13. Kopetz, H. (1997). Real-Time Communication (pp. 145-170). Springer US.
14. Keerthana, G., Anandan, P., Nandhagopal, N. (2021). Enhancing the Robustness and Security Against Various Attacks in a Scale: Free Network. Wireless Personal Communications, 117(4), 3029-3050.
15. Xu, W., Zuo, Z., He, C., Hu, R., Cui, S., Shao, S., ... Yang, N. (2021, July). Opportunities, challenges and feasibilities of Zero-Power IoT in 5G advanced. In 2021 IEEE/CIC International Conference on Communications in China (ICCC Workshops) (pp. 374-378). IEEE.
16. Orozco-Santos, F., Sempere-Payá, V., Silvestre-Blanes, J., Albero-Albero, T. (2021). TSCH Multiflow Scheduling with QoS Guarantees: A Comparison of SDN with Common Schedulers. Applied Sciences, 12(1), 119.
17. Sahoo, P. K., Thakkar, H. K., Hwang, I. (2017). Prescheduled and self organized sleep-scheduling algorithms for efficient K-coverage in wireless sensor networks. Sensors, 17(12), 2945.
18. Winter T., Thubert P., Brandt A., Hui JW, Kelsey R., “RFC 6550: RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks,” 2012. [Online]. Available: https://tools. ietf. org/html/rfc6550
19. Vinitha, A., Rukmini, M. S. S. (2019). Secure and energy aware multi-hop routing protocol in WSN using Taylor-based hybrid optimization algorithm. Journal of King Saud University-Computer and Information Sciences.
20. Qi, N., Dai, K., Yi, F., Wang, X., You, Z., Zhao, J. (2019). An adaptive energy management strategy to extend battery lifetime of solar powered wireless sensor nodes. IEEE Access, 7, 88289-88300.
21. Fu, X., Fortino, G., Li, W., Pace, P., Yang, Y. (2019). WSNs-assisted opportunistic network for low-latency message forwarding in sparse settings. Future generation computer systems, 91, 223-237.
22. Chamanian, S., Baghaee, S., Uluşan, H., Zorlu, Ö., Uysal-Biyikoglu, E., Külah, H. (2019). Implementation of energy-neutral operation on vibration energy harvesting WSN. IEEE Sensors Journal, 19(8), 3092-3099.
23. Jan, M. A., Jan, S. R. U., Alam, M., Akhunzada, A., Rahman, I. U. (2018). A comprehensive analysis of congestion control protocols in wireless sensor networks. Mobile networks and applications, 23(3), 456-468.
24. Fu, X., Yang, Y. (2020). Modeling and analysis of cascading node-link failures in multi-sink wireless sensor networks. Reliability Engineering System Safety, 197, 106815.
25. Akkaya, K., Thimmapuram, A., Senel, F., Uludag, S. (2008, March). Distributed recovery of actor failures in wireless sensor and actor networks. In 2008 IEEE Wireless Communications and Networking Conference (pp. 2480-2485). IEEE
26. Dagdeviren, O., Akram, V. K., Farzan, A. (2019). A Distributed Evolutionary algorithm for detecting minimum vertex cuts for wireless ad hoc and sensor networks. Journal of Network and Computer Applications, 127, 70-81.
27. Dagdeviren, O., Akram, V. K., Tavli, B. (2018). 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.
28. Liu, L., Han, G., Xu, Z., Shu, L., Martinez-Garcia, M., Peng, B. (2021). Predictive Boundary Tracking based on Motion Behavior Learning for Continuous Objects in Industrial Wireless Sensor Networks. IEEE Transactions on Mobile Computing.
29. Jiang, H., Zhang, S., Tan, G., Wang, C. (2013). Connectivity-based boundary extractionof large-scale 3D sensor networks: Algorithm and applications. IEEE Transactions on Parallel and Distributed Systems, 25(4), 908-918.
30. Zhou, Hongyu, Hongyi Wu, and Miao Jin. ”A robust boundary detection algorithm based on connectivity only for 3D wireless sensor networks.” 2012 Proceedings IEEE INFOCOM. IEEE, 2012.
31. Jiang, H., Zhang, S., Tan, G., Wang, C. (2013). Connectivity-based boundary extractionof large-scale 3D sensor networks: Algorithm and applications. IEEE Transactions on Parallel and Distributed Systems, 25(4), 908-918.
32. Wang, C., Wei, W., Lin, H., Jiang, H., Lui, J. C. (2016). BLOWUP: Toward distributed and scalable space filling curve construction in 3D volumetric WSNs. ACM Transactions on Sensor Networks (TOSN), 12(4), 1-20.
33. Akram, V. K., Dagdeviren, O. (2013). Breadth-first search-based single-phase algorithms for bridge detection in wireless sensor networks. Sensors, 13(7), 8786-8813 (ABIDE)
34. Wang, C., Jiang, H., Dong, Y. (2016). Connectivity-based space filling curve construction algorithms in high genus 3D surface WSNs. ACM Transactions on Sensor Networks (TOSN), 12(3), 1-29.
35. Dagdeviren, O., Akram, V. K., Tavli, B. (2018). 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.[30] (E-CDSCUT)
36. Saoudi, M., Lalem, F., Bounceur, A., Euler, R., Kechadi, M. T., Laouid, A., ... Sevaux, M. (2017). D-LPCN: A distributed least polar-angle connected node algorithm for finding the boundary of a wireless sensor network. Ad Hoc Networks, 56, 56-71 [31] (D-LPCN)
37. Zhou, H., Xia, S., Jin, M., Wu, H. (2014). Localized and precise boundary detection in 3-D wireless sensor networks. IEEE/ACM Transactions on Networking, 23(6), 1742-1754.
38. Li, F., Zhang, C., Luo, J., Xin, S. Q., He, Y. (2013). LBDP: Localized boundary detection and parametrization for 3-D sensor networks. IEEE/ACM Transactions on Networking, 22(2), 567-579
39. Renold, A. Pravin, and S. Chandrakala. ”Convex-hull-based boundary detection in unattended wireless sensor networks.” IEEE Sensors Letters 1.4 (2017): 1-4.
40. Sha, C., Ren, C., Malekian, R., Wu, M., Huang, H., Ye, N. (2019). A Type of Virtual Force-Based Energy-Hole Mitigation Strategy for Sensor Networks. IEEE Sensors Journal, 20(2), 1105-1119
41. Shukla, S. (2021). Reliable critical nodes detection for Internet of Things (IoT). Wireless Networks, 27(4), 2931-2946.
42. Molisch, A. F., Balakrishnan, K., Chong, C. C., Emami, S., Fort, A., Karedal, J., ... Siwiak, K. (2004). IEEE 802.15. 4a channel model-final report. IEEE P802, 15(04), 0662.
43. Benkic, K., Malajner, M., Planinsic, P., Cucej, Z. (2008, June). Using RSSI value for distance estimation in wireless sensor networks based on ZigBee. In 2008 15th International Conference on Systems, Signals and Image Processing (pp. 303-306). IEEE.