Wireless Personal Communications (2023) 130:757–775
https://doi.org/10.1007/s11277-023-10308-4

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

Shailendra Shukla[1]

Accepted: 25 February 2023 / Published online: 20 March 2023

Abstract

Node failure in the Wireless Sensor Networks (WSN) topology may lead to economic loss, endanger people, and cause environmental damage. Node reliability can be achieved by adequately managing network topology using structural approaches, where the critical nodes are precisely detected and protected. This paper addresses the problem of critical node detection and presents two-phase algorithms (ABCND). Phase-I, a $2D$ Critical Node (C-N) detection algorithm, is proposed, which uses only the neighbor's Received Signal Strength Indicator ($RSSI$) information. In Phase II, a correlation-based reliable RSSI approach is proposed to increase the node resilience against the adversary. The proposed algorithms ($ABCND$) require $\mathcal{O}(\log(N))$ time for convergence and $\mathcal{O}(\delta(\ logN))$ for Critical Node detection, N represents the number of IoT devices, and δ is the cost required to forward the message. We compare our algorithm (ABCND) with the current state-of-the-art on C-N detection algorithms. The simulation result shows that the proposed $ABCND$ algorithm consumes 50% less energy to detect C-N with 90% to 95% accurate Critical Nodes (C-N).

Keywords Critical Node Detection · Reliability · Industrial Wireless Sensor Networks(IWSN) · Energy

1 Introduction

According to the Compound Annual Growth Rate (CAGR), industries are estimated to grow by 9.9% every year even after covid-19 impact [4]. In recent years WSN and IoT-based industrial equipment have been frequently used in the Industry, and it is anticipated to increase by 5 to 6 times in five years [1]. WSN and IoT improve process control and monitoring applications [2], [5] in production. The recent growth of WSN and IoT in Industry is due to the new protocol standards(802.15.4/Zigbee standards [6, 7]), power efficiency [11], small size [8], and embedded computing capability [9]. Another reason for the growth of WSNs is the adaptability and reliability of protocols and standards for most industrial uses and the easy automation process [10].

✉ Shailendra Shukla
ss@mnnit.ac.in

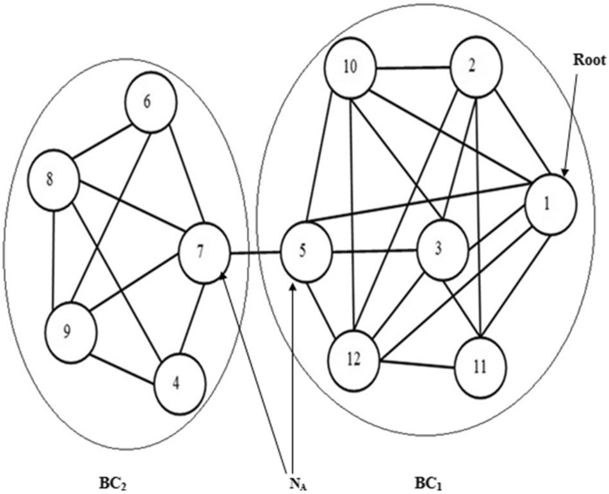
[1] Computer Science and Engineering Department, MNNIT Allahabad, Prayagraj, India

Industrial applications typically rate reliability, fault tolerance, and security at the top of the list while adopting any IWSN-based solutions. Industrial Wireless Sensor Networks (*IWSN*) based solution faces various challenges such as real-time data communication [15], robustness [16], energy [11], reliability [12], and fault tolerance [13] due to the inherent limitations of WSN. The modus operandi of IWSN is that the sensors collect data from the indoor or outdoor environment and deliver it to a central/sink or controller for processing, decision-making, and control [14]. Nowadays, IWSN-based solutions use IoT-based communications [17] [18] and IEEE standards algorithms (RFC 6550:RPL-based Industrial Networks [19]) for communication and Internet connectivity. However, due to the scalability, random deployment of sensor nodes, multi-hop communication [20], and limited battery lifetime [21] can hinder data communications. A trivial way to overcome this problem is to deploy a centralized node within the network topology of ISWN. The centralized node performs various computation tasks on collected data for various operation management [22]. Compared to other nodes, the central node's failure 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 impacts the energy-resilient industrial applications. The limitations of a centralized-based (IWSN) network [23] are 1. Low power and Lossy Networks (LLN) of IWSN and continuous participation of individual nodes in the data collection and message dissemination. In a large network (e.g., agriculture industries [2, 3]), there is the chance of message congestion [24] and latency [25], as all the nodes participate in the data transmission. 2. In large networks, a failure of particular nodes could impact an asymmetrically high number of different sensor nodes' connectivity, affecting data availability. Critical Node (*C-N*) failures [26] may disconnect all paths between the sink and other sensor nodes. It is costly to maintain the nodes for the networks' robustness frequently. However, if the critical node (important nodes) is detected, it may reduce the cost of maintenance. Detection of a Critical node (*C-N*) is crucial for the networks as it helps discover the unknown geometry and topology of an IWSN, which provides salient information for underlaying its environment and the efficient operation of networks. If a particular critical node stops working, many nodes may be separated from the network.

The reported work on critical node detection [37, 38, 41] proposes an algorithm for critical node detections. However, they are limited to only cut-vertex (articulation vertex detection). A Cut vertex [28] are nodes or link 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. The algorithm mostly uses breadth-first search (BFS) and works in forward and backward steps to detect the bridge in the network. The*ABIDE* requires the $\mathcal{O}(\delta N)$ 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+*) [27] use the Connected Dominant Set (*CDS*) (*C*) for Cut vertex(*C-V*) detection. 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*.

The algorithms, *CDSCUT+*, and *E-CDSCUT* [43] also depend upon the connected dominating set(CDS) and require broadcast $(dept_u + 1)$ for critical node detection. Mostly the critical node (*C-N*) selection algorithms focused only on detecting the articulation points. However, Industrial places like nuclear thermal plants, oil fields, refineries, and coal mines require detecting boundary nodes (*B-N*) [30, 39, 40], since the peripheral node is vital in surveillance and perimeter activity detection. So the

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



limitation of cut-vertex is that they only consider the network topology from the inside and cannot work correctly for any activity occurring at the periphery of networks.

None of the papers specifically address the critical node detection problem, either focused on boundary node detection or cut vertex detection. However, one of the naive approaches could be to use the existing Cut-Vertex *C-V* algorithms like [29] and combine it with Boundary Node (*B-N*) detection algorithms [30] for critical node *C-N* detection. The drawback of using two different algorithms for Critical node detection is increased computation and communication costs. Industrial Wireless Sensor Networks (IWSN) are Low power and Lousy Networks (LLN). Hence it will increase the overhead on energy and buffer size for individual nodes [17] [18]. If the nodes have mobility [31] and churn rate [32], it becomes difficult to implement two different algorithms in the IWSN environment. So, instead of finding a critical node using a separate algorithm. This paper proposes a distributed algorithm “Angle Based Critical Node Detection for *C-N* Detection (*ABCND*). The proposed ABCND algorithm includes both the Cut Vertex (*C-V*) detection [29]) and Boundary Nodes (*B-N*) detection [30]using only 1 hop information. The significant contributions of this paper are as follows: 1) A hybrid approach for *C-N* detection is proposed for 2*D*, which can be extended for 3*D* 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 the algorithm’s 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 to suit the IoT-based IWSN. The strength of this work is that it is practical as it requires only *RSSI* information. It is distributed in nature, so no global information is required.

The algorithm detects the reliable *C-N* in less energy with reduced false positivity. The complexity of *ABCND* algorithm is the time complexity of $\mathcal{O}(\log(N))$ and computation cost of $\mathcal{O}(\delta(\log N))$, N represents the number of nodes in networks, and δ is the total number of the messages.

The paper is organized as follows. Sect. 2 briefly summarizes the related work. Sect. 3 presents the system model and problem formulation. The proposed algorithm

(*ABCND* algorithms) is discussed in Sect. 4. Section 5 presents the result and performance evaluation of our proposed algorithm. Finally, Sect. 6 concludes the work with future work.

2 Related Work

None of the reported works simultaneously considers the boundary nodes and cut vertex as a critical node detection problem. Because of the above limitations, the related work section is divided into two subsections. a) Cut-Vertex *C-V* 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) [33]. Cut-Vertex is those nodes node or link that disconnects the graph into two (or more) separate components if removed (Fig. 1 Node(7,5) and edge(7,5)). The network is always at risk of being split into several separate components due to node failure (existence of articulation points in the networks) or topology change(mobility of nodes).

The algorithm in Ancestral Knowledge-Based Bridge Detection Algorithm *ABIDE* [37] proposes an ancestral knowledge-based bridge detection algorithm (articulation vertex detection-based approaches). The algorithm uses Breadth-First Search (BFS), which requires forward and backward steps. The forward step creates spanning BFS by the broadcasting packet to all the nodes. In the backward step, bridges are detected by converging backward casting messages to the sink nodes and terminates. ABIDE only detects the bridge in the network. It works on broadcasting and requires $O(\delta N)$ message and $O(\delta D \log_2(N))$ transmitted byte. Partition detection and recovery algorithm (*PADRA*) and (*PADRA*+) [27] uses a connected dominating set to detect Cut-Vertex. The working of the algorithm depends on two CDS rules i) ordinary nodes are noncritical, and b) the dominator with degree 1 is a critical node. Connected Dominating Set-based Cut detection *CDSCUT*+ and its extended version *E-CDSCUT* [29] proposes an approach to detect the cut vertex detection in the network. The approaches also use the connected dominating set(CDS), and Depth First Search(DFS) approaches to determine the cut vertex. CDS, in itself, is an NP-hard 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.

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. None of the existing works address the cut-vertex detections in 3-dimensional topology. However, some seminal work on boundary detection (*B-N*) is used to detect the critical nodes in WSN. Boundary detection (*B-N*) algorithm can be classified as a) 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) [44] requires high energy consumption due to extra hardware. Paper [34] 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 [35], 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 [35] 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 [34] uses the feature extraction techniques with 2*D* embedding [35] and distance information. In 2*D* embedding, it converts the 3*D* graph into 2*D* planer graph. Algorithm surfing on 3D surfaces *SURF* [42] constructs *isocontour*, *Reeb graph*, and *maximum cut-set* to divide the 3D network into different regions, then the connectivity-based algorithm is used to detect *B-N*. Node connectivity is used in *BLOW-UP* [36] to build a unit tetrahedron cell in the network. If any node lies in any tetrahedron cells, it is considered *B-N*. The limitation of [42] and [36] is the real-world implementation, which assumes a reliable network.

The geometrical approach requires the nodes to have an exact position or distance between nodes. In work, [45], the embedding of a 3*D* graph is projected on the 2*D* surface. The approach misinterprets nodes as a *B-N* due to projection, so it uses the *Isolated Fragment Filtering* (*IFF*) to remove false positives. The limitation of paper [45] 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 [46] a *convex-hull* based algorithm [47] is proposed for 3D *B-N* detection. Both [46] and [47] have constraint-like nodes equipped with *GPS* and known residual energy of other nodes. *LBDP* ignores the coverage and connectivity of boundary regions, which is essential for any Industrial based monitoring system. Chao, et al. in [48] 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 [48] is it requires the uniform distribution of nodes and known euclidean distance between the nodes.

The major limitations of the current reported work are: None of the critical node detection algorithms includes cut-vertex and Boundary node detection at the same time. Seminal works have been done in critical node detection for 3D topology. Most of the work has limitations like embedding of 3D to 2D, some hardware dependencies like GPS, known global coordinates, anchor node(Landmarks) requirement, and bootstrap. These dependencies make it impractical to be used directly in the IWSN.

3 System Model and Assumption

We assume that the nodes in IWSN are dynamic (mobile) and randomly placed in 2D area $\in \mathcal{R}^2$. Nodes and devices in the 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 vertices $V = \{v_1, v_2, ..., v_n\} \in \mathcal{R}^2$ represents the IWSN device and E represents edges. Edges between the nodes v_i,v_j are represented as E_{v_i,v_j}: $dis(v_i, v_j) \leqslant R_c$ (communication range) where $dis(v_i, v_j)$ is the eucledian distance between the nodes/devices v_i and v_j. Without the loss of generality, it assumed that sensing range R_s, and communication range R_c have the same spherical range.However, the communication range R_c follows the Quasi unit disc Graph. Let the graph G consist of a set of nodes V in 3D Teritrian plane $\mathfrak{R}^3$ with $d \in [0, 1]$ [49]. 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 but not equal to 1. Some nodes may not be in communication range of each other, so some may receive the signal, and some may not. We also consider the problem of the hidden terminal problem hidden terminal problems, signal blocks due to walls, etc.

The set of neighbors of sensor node v_i is defined as $Nbr(v_i)$. It is assumed that the node does not have 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 Industrial environment. At the data link layer, IWSN devices are configured with IEEE 802.15.4 communication [51], 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* [52] ($P_r = P_t$ x $(\frac{1}{d})^n$) is considered. The definitions of *B-N*, *C-N* and *C-V* used throughout the paper 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) [50]).

The boundary nodes (*B-N*) can be defined as a set of nodes that lies at the perimeter and follows the following properties: a)A node $'E'$ is a boundary node if its degree is less than 2 ($\forall v \in$B$-$N$\Longrightarrow$deg(E)$\leq$ 2). b) a node 'E' is a boundary node iff its degree deg'E' $\leq$ 3 and neighbor of 'E' are noncolinear points A, B, $C \in$ V such that a, b, $c \in$ neighbor of (E) and c) the nodes A, B, C form a triangle that encloses the node E($\forall E \in$ B$-$N$\iff$deg(E)$\leq$ 3 $\wedge$ *neighborofE* noncollinear, illustrated in Fig. 3a).

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 $\underline{\text{Node 'v' is } C\text{-}N \text{ if } \exists \text{ 'v' } \in B\text{-}N \vee \text{'v' } \in C - V}$.

The nodes apart from Critical nodes are Interior Node (*I-N*). Interior Node (*I-N*) are formed when at least three nodes enclose the node. Node E is considered as *I-N* such that set $\{A, B, C, E\} \in$ V and A, B, $C \in Nbr(E)$. Set $\{A, B, C\}$ forms a triangle that encloses the node E (illustrated in Figs. 2a and 3b). We define *Interior Node criteria* as being enclosed in a triangle formed by three nodes in its neighborhood ($\forall$ E $\in$ *I-N* $\Longrightarrow \exists A, B, C \mid A$, B, $C \in$ Nbr(E) $\wedge$ A, B, C forms triangle inclose E or $\neg$ *B-N*=*I-N*, as illustrated in Fig. 2b). Throughout the paper, we considered the interior node as those nodes which are not a boundary node (*B-N*)(e.g., 'E' is an internal node (*I-N*) as illustrated in Fig. 2) [50].

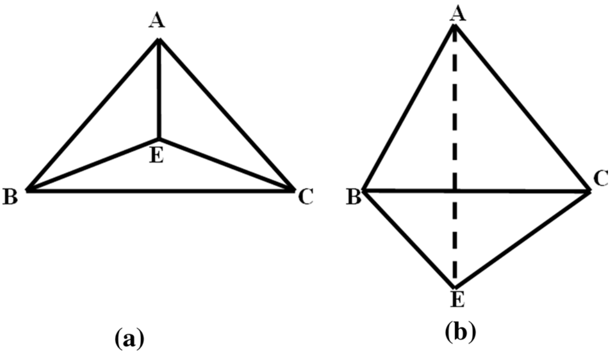


Fig. 2 Node *E* as boundary node (**a**) and interior node (**b**) in 2D

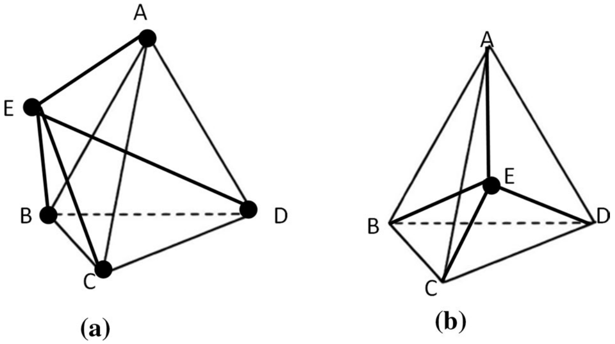


Fig. 3 Node *E* as boundary node (**a**) and interior node (**b**) in 3D

3.1 Contribution

This work proposes a lightweight RSSI-based distributed technique 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 is distributed in nature and depends only on single-hop information. The proposed algorithm is not limited to the detection of Critical- Node. However, it can identify Boundary Nodes and Cut-Vertex in IWSN as per the need and requirements of the industry.
- The algorithm uses two different approaches. The first approach is for the 2D topology, which can be extended to the 3D topology. The second approach addresses the reliability and accuracy of *C-N*. It uses the key ($K_{\aleph}$) and correlation function($\rho_{\aleph}$ [i], and $\rho(\aleph)$) between the nodes and angular property (cosine) to ensure accuracy.
- The proposed algorithm is independent of high-level protocol and uses only network layer (with RPL) and data link layer(RSSI) information. RSSI is prone to noise, so the algorithm uses the angular property(cosine theorem) to reduce the false positivity.

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

This section presents the modus operandi of the proposed algorithm for critical node selection in 2D and 3D scenarios [50]. 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 Fig. 2). Suppose a node E wants to test whether it is a critical node (C-N) or an Interior Node(I-N). If node E lies inside the triangle, it is an Interior 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 the 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 is known, then the angle formed between any two edges $\overline{\text{AB}}$ and $\overline{\text{BC}}$ ($\angle ABC$) 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 'E' is defined as the Interior Node I-N if it satisfies any of the given six relationships.*

1. $\measuredangle\, EAC > \measuredangle BAC \cos^{-1}\left(\frac{(d((E,A))^2+d((C,A))^2-d((E,C))}{2d((E,A))d((C,A))}\right)$
   $> \cos^{-1}\left(\frac{d((B,A))^2+d((C,A))^2-d((C,B))^2}{2d((B,A))+d((C,A))}\right) \mathit{Similarly},$
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 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 = $\measuredangle BCA$ (Fig. 2a) and if the point $'E'$ lies outside the triangle then $\measuredangle$ECA + $\measuredangle$ ECB > $\measuredangle BCA$. Hence, node $'E'$ is an Interior Node node according to the definition 1(if all the six relations satisfy). □

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), Fig. 2a.*

Proof Suppose node $'E'$ is not an Inter//ior Node. Interior Node ($'E'$) lies inside the triangle form by any three nodes △ ABC. The angle formed by node $'E'$ with other vertex $'A','B'$ $,'C'$ satisfy Lemma 1 hence $'E'$ is an interior node (I-N) (proof by contradiction). □

Lemma 3 *Minimum node required to make a node 'v' a boundary node(B-N) in QUDG: if deg(v) ≤ 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 $\measuredangle 60^o$ [Direct Proof]. □

Lemma 4 *Not all boundary nodes* (B-N) *are Cut-Vertex* (C-V) $[\underline{\exists \text{ 'v' } \in \neg B\text{-}N \Rightarrow \text{ 'v' } \in C\text{-}V]}]$.

Proof Let node $'v'$ belong to the connected component C_1 of graph G. Grow a maximal path P starting from $'v'$. Since $deg(v) = 1$(definition 1), vertex v is one of the two ends, so by maximal path, theorem C_1 -v is connected [53]. 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) $[\underline{\forall \text{ v } \in C\text{-}V \Leftrightarrow \text{ v } \in C\text{-}N}]$.

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 [19] for message communication. Each node 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). The 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 proposed algorithm is distributed in nature and uses the following header of RPL protocol for message dissemination a) DODAG information request(DIS), b) object of information of the DAG (DIO), and c) object of the update to the destination (DAO) [19]. Each node 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). The 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: 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 ℵ calculates signal strength using equation $P_r =$ - $10 \log \frac{\gamma^2}{(4\pi)2dis^2}$ of neighbour where dis← eucledian distance between devices and γ is the wavelength. Every device creates a 2D dataset *NbrN*(ℵ) contains the data: "node id" along with its "distance" (computed using RSSI), and a key $K_{\aleph}$ (each node has a key $K_{\aleph}$ which is also stored at the sink node). A node periodically updates its *NbrN*(ℵ) list and shares it with all its neighbors for the periodic 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 ℵ will compute *NbrN*(ℵ), a 2*D* 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 ℵ. Once *Nb*(ℵ) is created, node ℵ compute the Lemma 4.1 for *B-N* detection. If ℵ computes true for Boundary then update boolen *B-N* to 1 and send the information $[RII_{EN}(\aleph_i)$,*Time*,$B\text{-}N_{\aleph}]$ to sink node for reliability detection.

Algorithm 1 Algorithm to detect the $(C\text{-}V)$ and $(B\text{-}N)$ excuted at each node.

```
1: Intilize the node $N_{\aleph}$
2: Each node detects the $RSSI$ value of neighbour node.
3: $RI(N_{\aleph}[i]) \leftarrow RI(N_{\aleph}[i]) + \text{gain}$
4: $d(\aleph_i) \leftarrow P_r(RI(N_{\aleph}[i]))$
5: $RI_{\aleph}$ [i] $\leftarrow Quantization(d(\aleph_i))$ {Determine the similarity between the received node by the help of quantization to $RI_{\aleph}$ [i] }
6: $RII_{EN}(\aleph_i) \leftarrow \text{XOR}[RI_{\aleph}$ [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 ℵ as $I\text{-}N$ node, put $B\text{-}N_{\aleph}$=0.
10: else
11:     MArk the ℵ as $B\text{-}N$ node and put $B\text{-}N_{\aleph}$=1.
12: end if
13: Send MSG[$RI_{EN}(\aleph_i)$,$Timer$,$B\text{-}N_{\aleph}$] to Sink node
```

Algorithm 2 is implemented at the sink. Sink nodes are capable of performing high computations. So once the sink node receives the MSG from the Step:13 of algorithm 1, it then starts the separates of $RI_{EN}(\aleph_i)$ and *Time*,$B\text{-}N_{\aleph}$ from MSG. Sink node check for $B\text{-}N_{\aleph}$ value, the reverse compute the $RII_{EN}\aleph[i]$ using $K_{\aleph}$ key will start if the $B\text{-}N_{\aleph} = 1$ and store in $RI_{\aleph}$ [i]. Similarly, in reverse computation, all the neighbors of ℵ $RII_{EN}i[\aleph]$ decrypt 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(ℵ). 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 an inaccurate RSSI value or an unreliable node. The parameter ϱ varies according to the implementer or scenario.

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

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

4.2 Complexity Analysis

The complexity of the *ABCND* algorithm depends 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 (combining both *C-V* and *B-N*). In the second subsection, we have covered the impact of the proposed algorithm in a faulty network and IoT environment with RPL protocol [19].

5.1 Comparision of *ABCND* with *B-N* and *C-V* algorithms

Proposed algorithm Angle Based Critical Node Detection (*ABCND*) is compared with *B-N* (*ABIDE* [37]+*D-LPCN* [44], *CDSCUT+* [43]+*D-LPCN* [44]) 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 environment and network parameters. We have implemented the simulation using Probabilistic Wireless Network Simulator (Prowler) running under MATLAB [54]. The nodes are deployed in an area of 1000x1000 m^2, and nodes are increased gradually to analyze the density impact on send and received byte count and power consumption. Nodes in IWSN are configured with the TDMA channel with IEEE 802.15.4 MAC. The transmission range is considered 50 ms. The parameters used to evaluate and compare: a) sent byte counts and received byte counts, b) energy consumption, and c) 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 an Increase in Nodes in Networks

The density of the networks is increased gradually from 50 to 250 nodes. The proposed algorithm (*ABCND*) is compared with the existing *C-V* and *B-N* algorithms (*ABIDE* [37] + *D-LPCN* [44], *CDSCUT* + [43] + *D-LPCN* [44]). The theoretical result shows that if the number of nodes increases, the number of the message will also increase. The result is illustrated in Fig. 4. The result indicates that our proposed algorithm(*ABCND*) produces 50 % fewer number messages than the *ABIDE*+ *D-LPCN* and *CDSCUT*+ +*D-LPCN* algorithms. The number of messages in the ABCND algorithm drops because C-V and B-N are detected using a single approach. It can be concluded from the result that the number of sent-byte counts increases with the increase in node density. Hence, it is unsuitable for the low-power devices of IWSN as it increases the network overhead.

The result in Fig. 5 shows that as the nodes are increased in the network, the received byte count of all three algorithms increases gradually and linearly. However, there is a significant fall in ABCND compared to other state-of-the-art algorithms because the *ABIDE* + *D-LPCN* and *CDSCUT* + *D-LPCN* execute two different algorithms for the *C-N* detection. The received message (byte count) of our proposed algorithm (*ABCND*) is approximately .5 times better than the *ABIDE* + *D-LPCN* and 2 times better than the *CDSCUT* + *D-LPCN*. The result confirms the validity of simulation results with the theoretical result

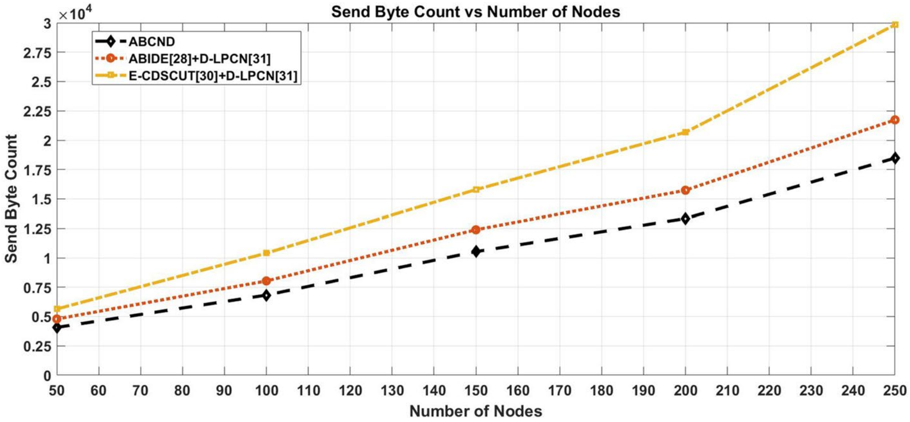


Fig. 4 Comparision of sent byte count w.r.t increase in number of node in network

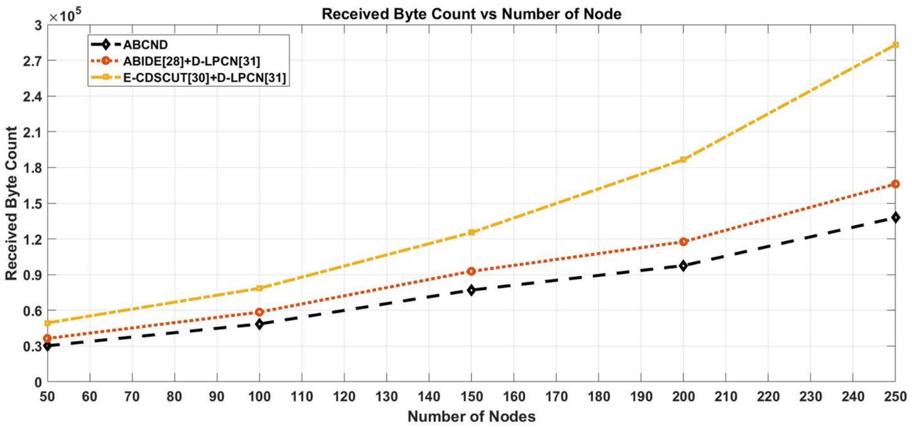


Fig. 5 Comparision of received byte count w.r.t increase in number of node in network

where the complexity of the received byte is equivalent to $Nbr(N))(\mathcal{O}(5N\log(N)+2E)$ & $\mathcal{O}(138Nbr(N)))$.

5.1.2 Power Consumption with an Increase in Nodes in Networks

Power consumption is a crucial parameter to be observed in WSNs, especially when deployed in outdoor environments, e.g., agriculture. The comparison is made against proposed *ABCND* and *ABIDE+ D-LPCN* and *CDSCUT+ D-LPCN* algorithms. The resulting Fig. 6 shows that power consumption increases as the network node count for all the algorithms increases. The result clearly shows that our *ABCND* algorithm requires almost 50% less energy than the *ABIDE+ D-LPCN* algorithm and three times lesser energy than the *CDSCUT++ D-LPCN* algorithms. The proposed *ABCND* algorithms significantly improve over battery-powered and can maximize the network lifetime in higher density. The reason for this improvement in *ABCDN* is the lesser message generation and algorithm-only dependence on one hop neighbor for RSSI value.

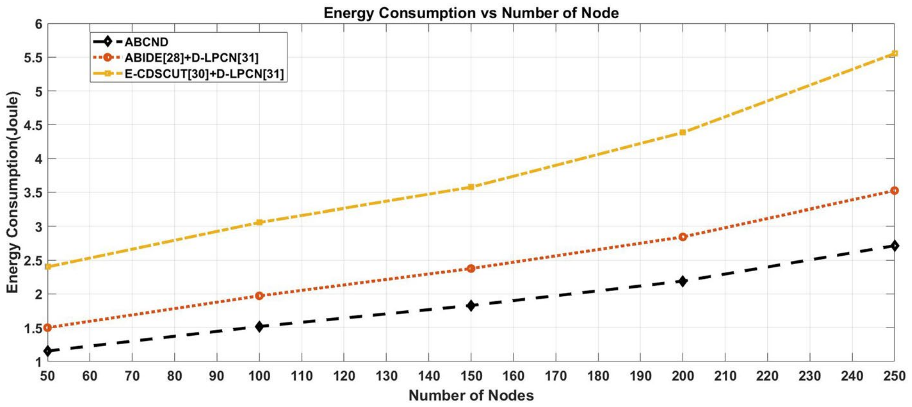


Fig. 6 Comparision of energy consumption w.r.t increase in number of node in network

5.1.3 Percentage of Correctly Detected Critical Nodes with an Increase in Nodes in Networks

The result shown in Fig. 7 illustrates the effect of node density on the accurate, critical nodes(*C-N*) detection. Again, *ABCND* algorithm is compared with the *ABIDE* with *D-LPCN* and *CDSCUT* with *D-LPCN* algorithms. The result 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 to *ABIDE+D-LPCN* and 15% more than *CDSCUT+ D-LPCN*. The reason for this improvement is the dependency of the algorithm only on the RSSI values.

5.2 Comparision of *ABCND* with RPL Protocol in Faulty Networks

We have used the Cooja simulator to evaluate the *ABCND* algorithm with other IoT-based algorithms. The Cooja simulator is a part of Contiki 2.7. Contiki is a lightweight, open-source operating system designed specifically for LLN-type WSN devices. To generate a real-time IWSN scenario, 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. We have used the mobility-based RPL [19] protocol for packet-based routing. To know the effectiveness of our algorithm in 3D, we 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 nodes (with faulty RSSI values) increased from 0 to 30. These unreliable nodes continuously randomly vary their *RSSI* value 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) ϱ =.7, b) ϱ =.5 and c) ϱ = 0. The simulation is performed for 1500 sec. The manipulation of *RSSI* starts after the 4th minute of stimulation so that the RPL protocol can create a DAG. The performance evaluation was based on the following

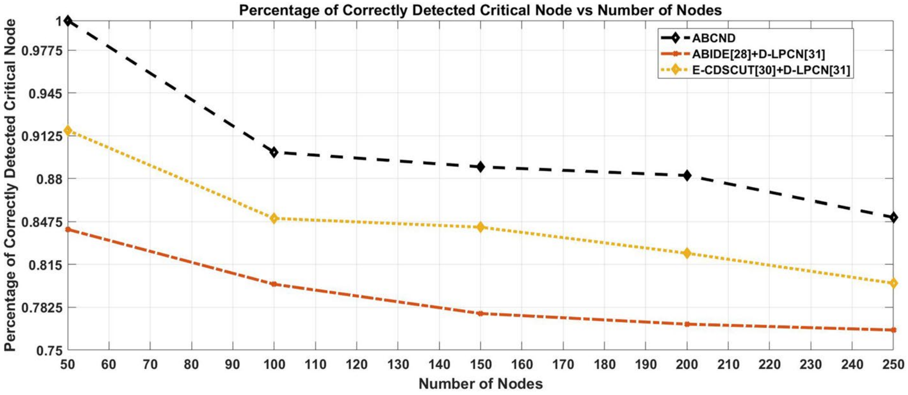


Fig. 7 Percentage of correctly detected critical nodes w.r.t the increase in number of nodes in networks

metrics a) Percentage of correctly detected Boundary nodes and b) Power Consumption (mW): and average power consumption at the network nodes.

5.2.1 Percentage of Correctly Detected *C-N*

The result in Fig. 8 shows the correctly detected *C-N* with respect to the increase in unreliable and faulty nodes in the networks. We compare our algorithm at ϱ =.7, ϱ =.5 and $\varrho = 0$ while keeping unreliable nodes fixed at 10. The result in Fig. 8 shows that the detected *C-N* at ϱ =.5 is 18% more than the $\varrho = 0$ and ϱ =.7. Result also shows that increasing the correlation value does not guarantee the exact *C-N* will be detected.

5.2.2 Energy Consumption

The energy consumption of ISWN is obtained by concatenating the time cycle consumed by *CPU*, *LPM*, *TX*, and *RX*. However, the energy consumption of processors (*CPU*) dominates others. We have considered the power consumed by only *CPU* in this analysis. The result in Fig. 9 shows that our proposed algorithm slightly increases the energy consumption in the node when executed with the help of RPL [19]. 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 algorithm's angular feature helps calibrate 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

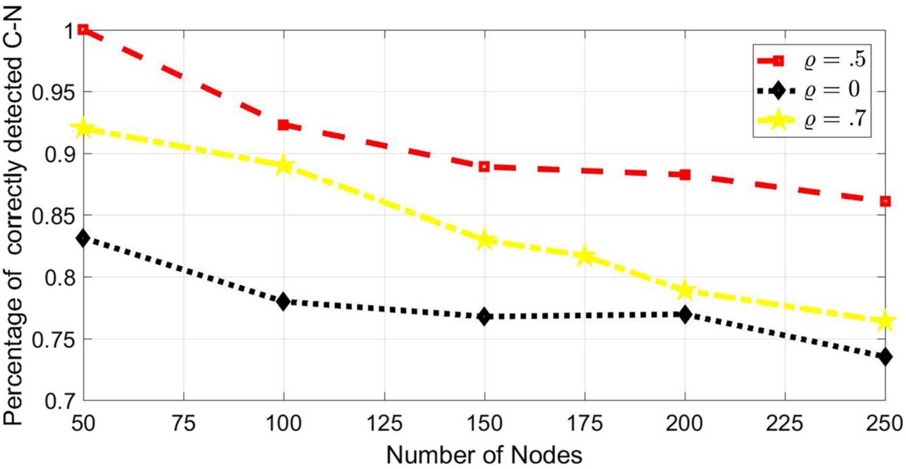


Fig. 8 Percentage of correctly detected *C-N* w.r.t total number of nodes

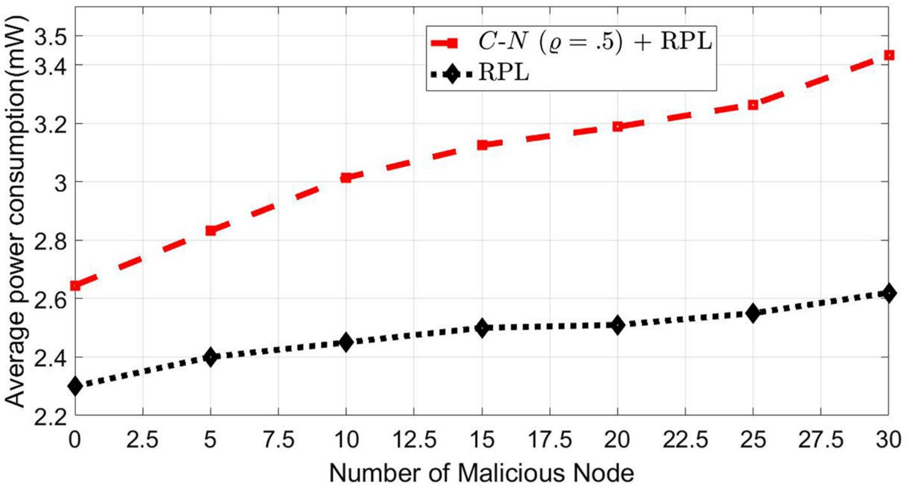


Fig. 9 Comparision of Energy consumption w.r.t unreliable node present in network

node ℵ. 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. To validate our algorithm, we performed various simulations and compared them with state-of-the-art and prominent IoT protocol (RPL). In future work, we plan to evaluate the *ABCND* algorithm for the large-scale IWSN and on real hardware. Moreover, we noticed that the algorithm has some false positivity when implemented in 3D topology, which we plan to address in the future.

Funding The research is supported by the MNNIT Allahabad.

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

Declarations

Conflicts of interest We have no conflicts of interest to disclose.

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. Keshtgari, M., & Deljoo, A. (2011). A wireless sensor network solution for precision agriculture based on zigbee technology.

6. 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.
7. 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.
8. Liu, Li, Guangjie Han, Zhengwei Xu, Lei Shu, Miguel Martinez-Garcia, and Bao Peng. Predictive boundary tracking based on motion behavior learning for continuous objects in industrial wireless sensor networks. IEEE Transactions on Mobile Computing (2021).
9. Ashjaei, M., Bello, L. L., Daneshtalab, M., Patti, G., Saponara, S., & Mubeen, S. (2021). Time-Sensitive Networking in automotive embedded systems: State of the art and research opportunities. *Journal of Systems Architecture, 117*, 102137.
10. Majid, M., Habib, S., Javed, A. R., Rizwan, M., Srivastava, G., Gadekallu, T. R., & Lin, J. C. W. (2022). Applications of wireless sensor networks and internet of things frameworks in the industry revolution 4.0: A systematic literature review. *Sensors, 22*(6), 2087.
11. 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.
12. 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.
13. Arapoglu, O., & Dagdeviren, O. (2020). A fault-tolerant and distributed capacitated connected dominating set algorithm for wireless sensor networks. Computer Standards & Interfaces, 103490.
14. Li, Jiwen, Jiapeng Dai, Alibek Issakhov, Sattam Fahad Almojil, and Alireza Souri (2021) Towards decision support systems for energy management in the smart industry and Internet of Things. Computers & Industrial Engineering 161, 107671
15. Kopetz, H. (1997). *Real-Time Communication* (pp. 145–170). US: Springer.
16. 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.
17. 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.
18. 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.
19. 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
20. 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.
21. 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.
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. Lin, C. C., Chin, H. H., Lin, W. X., & Lu, K. W. (2022). Dynamic energy-efficient surveillance routing in uncertain group-based industrial wireless sensor networks. Wireless Networks, 1-12.
24. 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.
25. Manikandan, S., & Chinnadurai, M. (2021). Effective energy adaptive and consumption in wireless sensor network using distributed source coding and sampling techniques. *Wireless Personal Communications, 118*(2), 1393–1404.
26. 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.
27. 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.

28. 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.
29. 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. 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.
31. Temene, N., Sergiou, C., Georgiou, C., & Vassiliou, V. (2022). A survey on mobility in wireless sensor networks. *Ad Hoc Networks, 125*, 102726.
32. Singh, S. K., Kumar, C., & Nath, P. (2021). Analysis and modelling the effects of mobility, Churn rate, node's life span, intermittent bandwidth and stabilization cost of finger table in structured mobile P2P networks. *Wireless Networks, 27*(2), 1049–1062.
33. Zrelli, A., & Ezzedine, T. (2021). A new approach of WSN deployment, K-coverage and connectivity in border area. *Wireless Personal Communications, 121*(4), 3365–3381.
34. Jiang, H., Zhang, S., Tan, G., & Wang, C. (2013). Connectivity-based boundary extraction of large-scale 3D sensor networks: Algorithm and applications. *IEEE Transactions on Parallel and Distributed Systems, 25*(4), 908–918.
35. 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.
36. 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.
37. 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).
38. Saeed, M. K., Mahmood, K., Shah, A. M., & Khan, J. (2021). Efficient solution for connectivity restoration (ESCR) in wireless sensor and actor-networks. *Wireless Personal Communications, 117*(3), 2115–2134.
39. Amutha, J., Nagar, J., & Sharma, S. (2021). A distributed border surveillance (dbs) system for rectangular and circular region of interest with wireless sensor networks in shadowed environments. *Wireless Personal Communications, 117*(3), 2135–2155.
40. Robinson, Y. H., Lawrence, T. S., Julie, E. G., & Vimal, S. (2021). Development of fuzzy enabled coverage hole detection algorithm in wireless sensor networks. *Wireless Personal Communications, 119*(4), 3631–3649.
41. Khalilpour Akram, V., Akusta Dagdeviren, Z., Dagdeviren, O., & Challenger, M. (2021). PINC: Pickup non-critical node based k-connectivity restoration in wireless sensor networks. *Sensors, 21*(19), 6418.
42. 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.
43. 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).
44. 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).
45. 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.
46. 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.
47. Renold, A. Pravin., & Chandrakala, S. (2017). Convex-hull-based boundary detection in unattended wireless sensor networks. *IEEE Sensors Letters, 1.4*, 1–4.
48. 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.
49. Kundu, S., & Das, N. (2022). A study on boundary detection in wireless sensor networks. Innovations in Systems and Software Engineering, 1-9.
50. Shukla, S. (2021). Reliable critical nodes detection for internet of things (IoT). *Wireless Networks, 27*(4), 2931–2946.
51. 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.

52. 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.
53. Ghorbanzadeh, H., Sheikhahmadi, A., Jalili, M., & Sulaimany, S. (2021). A hybrid method of link prediction in directed graphs. Expert Systems with Applications, 165, 113896. Simon, G., Volgyesi, P., Maróti, M., & Ledeczi, A. (2003, March).
54. Simulation-based optimization of communication protocols for large-scale wireless sensor networks. In IEEE aerospace conference (Vol. 3, pp. 31339-31346).

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Shailendra Shukla is Assistant Professor at the Computer Science and Engineering Department ofthe Motilal Nehru National Institute of Technology Allahabad. He received the MS-(Information Security) from the "Indian Institute of Information Technology Allahabad and then completed a Ph.D. from theIndian Institute of Technology Patna in computer science. His doctorial work is based on "On BoundaryDetection and Localization in Wireless Sensor Networks". In this work, he proposed a collection ofnetworking algorithms which address the security problems like routing in the Internet of Things,localization, boundary node detection (surveillance), virtual coordinate assignment (Geographyrouting/localizations), cyber-physical systems, monitoring, and surveillance. He has published articles invarious publication houses like in Elsevier, Springer, IEEE. Previously, he was associated with JaypeeUniversity for 5 years. Currently, he is supervising 2 Ph.D. students and guided 5 master students. Hehas been involved in TEQIP-III research projects in the area of IoT and Telemedicine.