SN Computer Science (2024) 5:451
https://doi.org/10.1007/s42979-024-02819-z

ORIGINAL RESEARCH

Mitigating Security Attacks in Spectrum Allocation in Cognitive Radio Networks

Wangjam Niranjan Singh[1] · **Ningrinla Marchang**[2]

Received: 14 January 2024 / Accepted: 21 March 2024

Abstract
Current channel assignment techniques rely on the nodes's decent nature and lack of malicious intent. The accuracy of the information disseminated by the nodes is not confirmed because it is based on an assumption of trust. Furthermore, in distributed channel assignment methods, each node chooses their spectrum allocation independently and notifies its neighbours of the choice, which is also unreliable. Due to its autonomous decision-making and lack of validation, a node is susceptible to three different kinds of channel assignment attacks: Channel Ecto-Parasite Attack (CEPA), Network Endo-Parasite Attack (NEPA), and LOw cost Ripple effect Attack (LORA). In this paper, we present a detection technique for distributed Cognitive Radio Network named "Smart Neighbour Mechanism" that detects attacks that occur in spectrum assignment, such as CEPA, NEPA and LORA. The suggested security technique is based on the idea of using neighbour monitoring to detect malignant users within the system, protecting it from security risks that take advantage of channel assignment problems. Additionally, we conduct simulation-based studies to illustrate the efficacy of our proposed technique.

Keywords Spectrum allocation · CEPA · NEPA · LORA · Security · Cognitive radio

Introduction

Due to the increasing usage of smartphones, the popularity of a variety of online services (e.g., social media), and the decline in subscription costs, the emergence and relentless expansion of wireless networks have greatly boosted spectrum demand. This tendency will persist, and the need for bandwidth will increase as more wireless technologies and gadgets are deployed in the future. The few license-free radio frequencies now in use are also regularly overused in densely populated areas. This circumstance causes disagreement and interference, and as a result, a major performance decrease. Regardless of location or time, the great majority of licensed radio spectrum remains unused or underutilised, resulting in multiple empty spectrum bands. This poor spectrum utilisation is a direct result of the current spectrum regulation policies, which divide the spectrum between licensed and unlicensed frequencies. It is anticipated that there may be an urgent spectrum crisis, where the wireless capacity will soon be overwhelmed by the quickly growing demand for spectrum. The challenge is that wireless data carriers lack access to new spectrum. The formulation of more adaptable regulatory regulations and the advancement of related and novel technology will change this paradigm. To solve poor spectrum usage, Cognitive Radio (CR) [1] was developed. Joseph Mitola III initially suggested the idea of CR at a seminar at KTH in 1998. Mitola and Maguire [2] then wrote about the idea in an article in 1999. A more adaptable and effective use of the wireless resources is made possible by the relatively new network design known as "Cognitive Radio". Its main goal is to allow wireless equipments to access certain radio frequency bands without interfering with authorized users. Wireless regional area networks

This article is part of the topical collection "SWOT to AI-embraced Communication Systems (SWOT-AI)" guest edited by Somnath Mukhopadhyay, Debashis De, Sunita Sarkar and Celia Shahnaz.

✉ Wangjam Niranjan Singh
niranwang@gmail.com

Ningrinla Marchang
ningrinla@gmail.com

[1] Department of Computer Science and Engineering, Assam University, Dorkagona, Silchar 788011, Assam, India

[2] Department of Computer Science and Engineering, North Eastern Regional Institute Of Science And Technology, Nirjuli, Itanagar 791109, Arunachal Pradesh, India

Published online: 18 April 2024

(WRAN), which are based on IEEE 802.22 [3], provide the necessary requirements for using TV-free spectrum. The term "Cognitive Radio Network (CRN)" refers to a network that allows cognitive radio equipments to communicate wirelessly with one another. These technologies can detect their surroundings and adapt accordingly. They utilise the licensed frequencies in an effective and smart manner. A device has the ability to search for and use a free frequency band when it learns that the band it is now using is no longer available. As a result, it can opportunistically switch between bands rather than sticking to just one. Primary users (PUs) are authorized to operate within certain frequency bands, while secondary users (SUs) can opportunistically use a primary user's assigned frequency bands without affecting them.

Spectrum assignment maps the frequency to radio interfaces to enhance spectrum usage while minimising interference. In terms of interference, CRN spectrum allocation differs from ordinary wireless networks. It is possible for a single SU to significantly interfere with PUs that are using the permitted radio band, in addition to other SUs. Spectrum assignment for CRN has been a top research focus in order to address this issue. Spectrum allocation in CRNs involves mapping free licenced channels to SUs for optimal performance. Because it also handles a number of design considerations, such as fault tolerance, interference, stability, throughput, and connection, this is more challenging than channel allocation (CA) in traditional wireless networks.

Recently, there has been a lot of discussion in the literature on the problem of resource assignment in CRNs and its remedies. Spectrum assignment is often classified as fixed, dynamic, or hybrid based on the type of mapping involved. While a fixed channel assignment [4, 5] frequently has a fixed channel configuration, a dynamic channel assignment [6–8] necessitates periodic channel mapping modifications to account for evolving network conditions. Static and dynamic behaviour are both altered by a hybrid method [9]. Some parts are allocated statically, while the rest sections are set dynamically. The strategies of spectrum allocation may be classified depending on the following implementation techniques: cluster-based methods [13, 14], which combine distributed and centralised methods to avoid the shortcomings of both, conventional methods based on centralised control [8, 10], and distributed methods [11, 12] that do not require a central controller. The literature covers several resource allocation strategies, including Evolutionary Algorithms [16–18], Soft computing [19–21], Heuristics [9, 22, 23], Game Theory [24–27], Linear Programming [7, 28, 29], Network Graph Based [8, 10, 30, 31] and Non Linear Programming [5, 32, 33]. The spectrum allocation techniques described in [10, 11, 34] are intended for a single radio interface. Such methods are simple to implement and handle interference. However, if the channel is reclaimed by the PU, the ongoing data transmission will be disrupted. The methods listed in [35, 36] are designed for users who own two radios. For multi-radio users, the resource allocation strategies covered in [7, 9] apply. In this scenario, network partition does not occur when a PU reclaims a specific channel.

Assuming that users have no malicious intentions, numerous effective resource allocation techniques have been developed over the years. However, malicious users may alter the standard resource allocation process that might significantly impact network performance. Naveed and Kanhere investigated security concerns to Wireless Mesh Network frequency assignment in their study in [37]. The authors assess the effect of these vulnerabilities and conclude that malicious nodes can harm the overall network's radio bandwidth and functionality. Such CRN vulnerabilities are also discussed by the authors in [38, 39]. Similar studies have been published in CRN specifically in [40, 41], where the authors assess the impact of these attacks on both centralized and distributed spectrum allocation algorithms. In our work, we present a detection technique for distributed CRN named "Smart Neighbour Mechanism" that detects attacks that occur in spectrum assignment, such as CEPA, NEPA and LORA. The suggested security measure is built on the notion of neighbour monitoring to detect malignant SUs in the system, hence safeguarding the network from security threats that exploit spectrum allocation issues. This work, to the extent that we know, this is the first effort to address security problems with spectrum assignment in CR. Here are some highlights of this paper's significant contributions:

1. We present a detection technique for distributed CRN named "Smart Neighbour Mechanism" that detects attacks that occur in spectrum assignment, such as CEPA, NEPA and LORA.
2. We demonstrate how the suggested method can be used to detect and isolate attackers on the fly.
3. We present effectiveness of the detection mechanism based on the outcomes of simulation.

Here is a summary of the remaining portion of the paper. Section "System Model" presents the system model. Section "A Distributed Spectrum Allocation Method" then provides the distributed spectrum assignment method and the attacks based on the algorithm. Then the proposed detection technique is shown in section "Smart Neighbour Detection Approach". The numerical findings of the simulation are presented in section "Simulation and Performance Evaluation". Section "Conclusions" brings this research to a close.

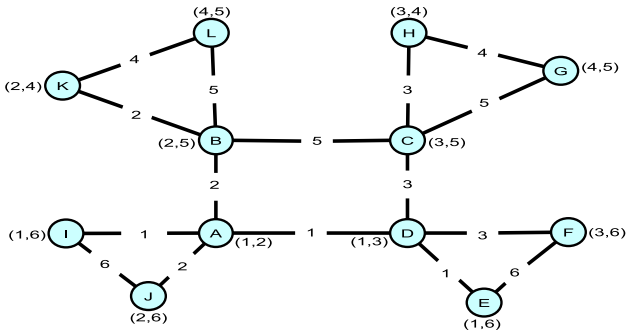

Fig. 1 Spectrum assignment under distributed normal allocation

System Model

We consider a CRN environment with N SUs, dual radios, and access to a set of $\mathcal{CV}$ resources (channels). Let $M = |\mathcal{CV}|$. It is represented as an undirected graph termed $G_{undirect}(V,E)$ with each node $p \in V$ representing an SU and an edge $e = (p,q) \in E$ if SUs represented by nodes p and q are within the transmission range. $G_{undirect}(V,E)$ is a connected graph in which any two SUs are linked by a direct connection (edge) or a route connecting several SUs. If two SUs are on the same resource, or channel, and are within transmission range of one another, they can interact with one other. A spectrum assignment or channel assignment $\mathcal{CA}$ generates a new undirected graph $G_{\mathcal{CA}}(V, E_{\mathcal{CA}})$, with $E_{\mathcal{CA}}$ containing the edges described below. If $(p,q) \in E$ and $m \in \mathcal{CA}(p) \bigcap \mathcal{CA}(q)$, then there is an edge $e = (p,q;m)$ on resource m. $\mathcal{CA}(p)$and $\mathcal{CA}(q)$ denote the sets of channels allotted to p and q, respectively. $\mathcal{RA}(p)$ denotes the number of radios at SU p, and $|\mathcal{RA}(p)| \leq M$. There could be more than one edge connecting two nearby SUs if they share more than one channel. All SUs are considered to have same transmission and interference ranges. We considered half-duplex communication, where each link between two SUs can only communicate one direction at a time.

A Distributed Spectrum Allocation Method

The distributed resource allocation used here is the same as the one used in [41] which is based on CRTCA.

Algorithm 1 Normal channel allocation rule

```
// This algorithm is executed by an SU to assign channels to each
   link connecting it to a neighbour
for each edge e incident on SU do
    if |CA(p)|<|RA(p)| and |CA(q)|<|RA(q)| then
        m← the least used resource among the resources in CV;
    else if |CA(p)|=|RA(p)| and |CA(q)|<|RA(q)| then
        m← the least used resource among the resources in CA(p);
    else if |CA(p)|<|RA(p)| and |CA(q)|=|RA(q)| then
        m← the least used resource among the resources in CA(q);
    end
    Assign m to e;
end
```

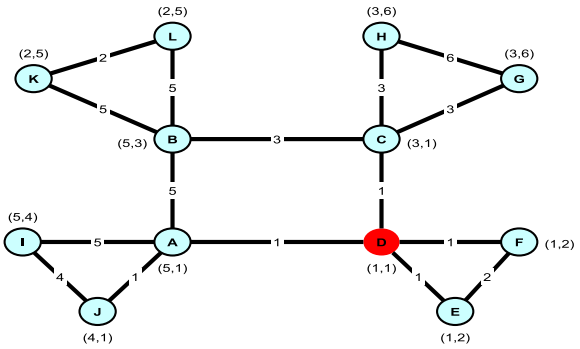

Fig. 2 Spectrum assignment under attack CEPA in distributed environment

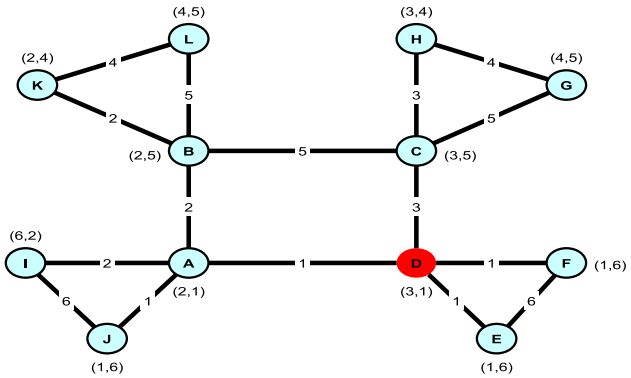

Fig. 3 Spectrum assignment under attack NEPA in distributed environment

Figure 1 depicts 12 SUs with two radios each, along with six freely accessible resources or channels. The resource allocation rule listed below is used to assign the resources. The group of channels shared by the two end SUs of a connection are indicated by the label at the link, and the group of channels assigned to an SU is indicated by the label along with the SU. According to the distributed resource allocation approach discussed above, Figure 1 depicts the resource allocation.

Algorithm 1 is executed by each SU u in the network. Let N be the number of SUs. The for statement at line 1 is executed as many times as the number of neighbors of u since it is equal to the number of edges incident on u. The maximum number of neighbors an SU can have is $N–1$. Thus, the for statement runs in time $O(N)$. For each iteration of the for statement, one of the statements in lines 3, 5 and 7 is executed. Among them, line 3 has the highest order of magnitude since the least used channel among all the channels in $\mathcal{CV}$ is selected whereas in lines 5 and 7, the least used channel is selected from $\mathcal{CA}$(p) and $\mathcal{CA}$(q) respectively. We note that $\mathcal{CA}(p) \subseteq \mathcal{CV}$ and $\mathcal{CA}(q) \subseteq \mathcal{CV}$. Thus, one run of line 3 takes $O(M)$ as it finds the minimum among M elements. Therefore the overall running time of Algorithm 1 is $O(MN)$.

Channel Ecto-Parasite Attack (CEPA) in Distributed Allocation

The CEPA's primary aim is to increase interference at the most utilised resource. Algorithm 2 shows how CEPA alters the channel allocation rule. An SU assigns the radio interfaces the least used resource using the normal distributed assignment algorithm. On the other hand, the malicious SU initiates CEPA by assigning its interfaces to the resource that is utilized the most. The resource assignment under the influence of CEPA is shown in Fig. 2. Following the same argument as for Algorithm 1, Algorithm 2 also takes $O(MN)$.

Algorithm 2 Distributed CEPA

```
   // This algorithm is executed by a malicious SU to assign
      channels to each link connecting it to a neighbour
 1 for each edge e incident on SU do
 2 |  if |CA(p)|<|RA(p)| and |CA(q)|<|RA(q)| then
 3 |  |  m← the highest used resource among the resources in CV;
   |  |
 4 |  else if |CA(p)|=|RA(p)| and |CA(q)|<|RA(q)| then
 5 |  |  m← the highest used resource among the resources in CA(p);
 6 |  else if |CA(p)|<|RA(p)| and |CA(q)|=|RA(q)| then
 7 |  |  m← the highest used resource among the resources in CA(q);
 8 |  end
 9 |  Assign m to e;
10 end
```

Network Endo-Parasite Attack (NEPA) in Distributed Allocation

A SU generally assigns its radio interfaces with low priority (least loaded) resources. However, in a NEPA attack, the malignant SU assigns its interfaces to highly desired resources in an attempt for enhancing interference at highly loaded, extensively used resources, without informing the neighbors of the change. We introduce a malignant SU D, represented by a red node within the network. As seen in the figure, the resource allocation is altered in conjunction with the launching of this attack (refer Fig. 3).

Algorithm 3 Distributed NEPA

```
   // This algorithm is executed by a malicious SU to assign
      channels to each link connecting it to a neighbour
 1 for each edge e incident on SU do
 2     if |CA(p)|<|RA(p)| and |CA(q)|<|RA(q)| then
 3         m← choose randomly among the r most used among the resources in
            CV;
 4     else if |CA(p)|=|RA(p)| and |CA(q)|<|RA(q)| then
 5         m← choose randomly among the r most used among the resources in
            CA(p);
 6     else if |CA(p)|<|RA(p)| and |CA(q)|=|RA(q)| then
 7         m← choose randomly among the r most used among the resources in
            CA(q);
 8     end
 9     Assign m to e;
10 end
```

In a normal spectrum assignment, an SU allocates the least utilised channel to radio interfaces. However, in NEPA, a malicious SU initiates this attack by allocating one out of r most used channels to the interfaces. Here, r is a predefined parameter. The resource distribution under the influence of the NEPA attack is depicted in Fig. 3. We introduce a malicious SU D, identified in the network by a red node.

In Algorithm 3, the for statement at line 1 is executed as many times as the number of neighbors of u since it is equal to the number of edges incident on u. The maximum number of neighbors an SU can have is $N–1$. Thus, the for statement runs in time $O(N)$. For each iteration of the for statement, one of the statements in lines 3, 5 and 7 is executed. Among them, line 3 has the highest order of magnitude since the r most used channel among all the channels in $\mathcal{CV}$ are selected whereas in lines 5 and 7, the r most used channel are selected from $\mathcal{CA}(\mathrm{p})$ and $\mathcal{CA}(\mathrm{q})$ respectively. We note that $\mathcal{CA}(\mathrm{p}) \subseteq \mathcal{CV}$ and $\mathcal{CA}(\mathrm{q}) \subseteq \mathcal{CV}$. To execute line 3, first the r most used channels in $\mathcal{CV}$ have to be selected first. So, it needs $O(M)$ to find the first most used channel. Then, after removing it from the list, it needs $O(M\text{-}1)$ for selecting the second most used channel and so on. Thus, it needs time of the order of $M+(M\text{-}1)+\cdots+(M\text{-}(r-1))$. Thus, one run of line 3 takes $O(Mr)$. Therefore the overall running time of Algorithm 3 is $O(MNr)$.

LOw cost Ripple effect Attack (LORA)

LORA occurs in distributed CRNs when a malicious SU forces other SUs to reallocate their channels by spreading incorrect and misleading resource information around the network. Since the impact of LORA extends beyond the neighbours of the impacted node to a considerable portion of the network, it is relatively more worse than the prior

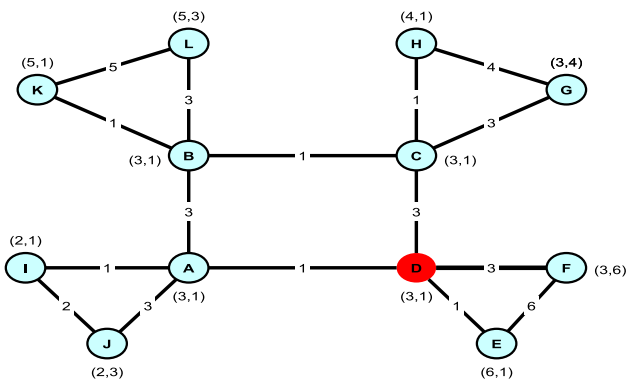

Fig. 4 Spectrum allocation under attack LORA in distributed environment

Fig. 5 Message format for CHNL_USAGE, CHNL_CHANGE, and MONITOR_REQUEST

CHNL_CHANGE MESSAGE FORMAT

NODE ID	INTERFACE ID	CHANNEL

CHNL_USAGE MESSAGE FORMAT

NODE ID	INTERFACE ID	CHANNEL

MONITOR_REQUEST MESSAGE FORMAT

SUSPICIOUS NODE ID	INDENTIFIED DISCREPENCY INTERRFACE ID	CHANNEL

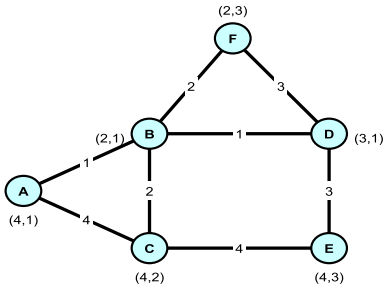

Fig. 6 Spectrum allocation under distributed normal assignment (topology 1 when N = 6 and M = 4)

two attacks. This is due to the fact that the attack disables the traffic forwarding capabilities of numerous nodes for an extended period of time.

Algorithm 4 Distributed LORA

```
// This algorithm is executed by an SU to assign channels to each
   link connecting it to a neighbour
for each edge e incident on SU do
    if |CA(p)|<|RA(p)| and |CA(q)|<|RA(q)| then
        m← the least used resource among the resources in CV;
    else if |CA(p)|=|RA(p)| and |CA(q)|<|RA(q)| then
        m← the least used resource among the resources in CA(p);
    else if |CA(p)|<|RA(p)| and |CA(q)|=|RA(q)| then
        m← the least used resource among the resources in CA(q);
    end
    Assign m to e;
end
```

An SU assigns the least utilized channel to the radio interfaces using the normal resource allocation technique. The SUs will assign using the normal method in LORA as well. However, as soon as the assignment is finished, a compromised SU starts this attack by providing false information about *M*'s channel utilisation, which forces the other SUs to reallocate their resources. Following the same argument as for Algorithm 1, Algorithm 4 also takes *O(MN)*.

Smart Neighbour Detection Approach

To detect the malignant SUs in a system, the proposed Smart Neighbor Mechanism is built on the notion of neighbour monitoring. Here, each SU keeps track of a Bad_counter, initialised to 0, for each of its neighbouring SUs. One hop neighbours validate the channel assignment information and the decision made by each SU regarding its spectrum allocation (connected nodes only). The neighbours of a suspicious SU are notified of the irregularities found in the disseminated information and the spectrum allocation of that SU, so labelling it as suspicious. Individually, the neighbours of a node verify the accuracy of any identified abnormalities. If the abnormalities are confirmed, the Bad_counter of the suspect SU is increased. When it reaches a predetermined limit, such SU's information can no longer be relied upon. After some time, when the suspicious SU begins to behave appropriately,

the Bad_counter will be decreased till it reaches 0 and neighbours will trust that SU's information again (Fig. 4).

The mechanism uses three messages namely CHNL_USAGE, CHNL_CHANGE, and MONITOR_REQUEST message and their formats are given in Fig. 5. Using the CHNL_USAGE messages, SUs routinely share information regarding spectrum allocation and its usage with their neighbours. Upon receiving the message, the neighbouring SUs recalculate and, if necessary, change their channel assignments in order to minimise interference. The CHNL_CHANGE message is used to change the channel assignment for a link between two SUs that share the link. The name of the mechanism is derived from the fact that neighbouring SUs detect the misbehaviour of the parent SU as the parent SU's neighbours receive the CHNL_USAGE message from the parent SU. For instance, in Fig. 6, SU B is called the parent SU of A, C, F and D as SU B is one-hop distance from them and also A, C, F and D are the neigbours SU of parent SU B. In order for a neighbour SU to be connected to the parent SU, one of their interfaces must share the same channel. Therefore, the neighbouring nodes can independently validate the truthfulness of the CHNL_USAGE message's contents. Using a single MONITOR_REQUEST message, identified anomaly information is transmitted.

Smart Neighbor Detection Mechanism

Algorithm 5 Smart neighbor Mechanism to detect CEPA, NEPA and LORA.

```
1  Channel assignment to a node using distributed channel allocation
   CHNL_USAGE message is disseminated to neighbour nodes
   CHNL_CHANGE message is disseminated to neighbour nodes
2  while Node = NeighnourNode do
3      if CHNL_USAGE message has discrepancies then
4          create MONITOR_REQUEST message and broadcast.
           if Bad_counter < H then (H = no of neighbour nodes) then
5              increment Bad_counter
6          else
7              if Bad_counter = H then
8                  declare Node as malicious.
                       // find type of maliciousness.
9                  if CHNL_USAGE message is not modified then
10                     if all interface IDs contain same channel and channel =
                        Highest used Channel then
11                         Declare CEPA attack.
12                     else
13                         if some interface ID contain same channel and channel
                            = Highly used Channel then
14                             Declare NEPA attack.
15                         end
16                     end
17                 end
18                 else if CHNL_CHANGE message is not modified then
19                     Declare LORA attack
20                 end
21             end
22             else if Bad_counter! = 0 then
23                 Decrement Bad_counter.
24             end
25         end
26     end
27 end
```

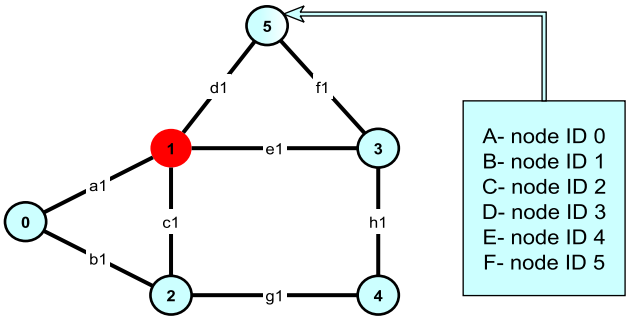

Fig. 7 Illustrative example showing working of "Smart Neighbor Mechanism" under CEPA/NEPA attack

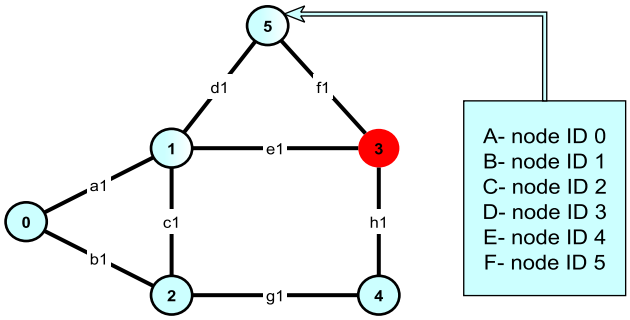

Fig. 8 Illustrative example showing working of "Smart Neighbor Mechanism" under LORA attack

When an SU sends the CHNL USAGE message and upon receipt of it, its neighbouring SUs execute the algorithm described in algorithm 5. Each SU keeps track of the Bad_ counter of its neighbours. Bad_counter value shows the severity of the SU's misbehaviour. An SU is said to be acting correctly when the value is 0, while acting incorrectly when the value is H. Intermediate values indicate that the node is acting suspiciously. The algorithm demonstrates that the neighbour SUs initially look for discrepancies after receiving the CHNL_USAGE message i.e., if neighbour SU− > interface id− >Channel (from CHNL_CHANGE message) is not equal to parent SU− >Interface id− >Channel (from CHNL_USAGE message) then a discrepancy is found. If a disparity is found, a MONITOR_REQUEST message is generated that includes the information about the suspect SU and the discrepancy. The neighbour SUs disseminate the message to their interference domain neighbours on all available resources.

The CHNL_USAGE message is handled normally if it arrives at the neighbour SU without the discrepancy and the parent SU is behaving properly (Bad_counter= 0). If MONITOR_REQUEST message(s) are received and the request is validated, Bad_counter is incremented by the no. of messages and CHNL_USAGE is discarded. In addition, neighbouring SUs can validate requests by inspecting the CHNL_ USAGE message of the SU that sent the MONITOR_REQUEST. Comparing the interface information of the neighbouring SU to the discrepancy listed in the MONITOR_REQUEST message one can validate the message's truthfulness. The validation of the MONITOR REQUEST message safeguards the parent SU in the event that a neighbour SU behaves inappropriately by wrongly accusing the parent.

Algorithm 5 is run by each SU when it receives a CHNL_ USAGE message from its neighbors (while statement of line 2). The statements inside the while statement take constant time, i.e., $\theta(1)$. The maximum number of neighbors an SU can have is *N-1*. Thus, the time taken by the algorithm is *O(N)*.

Illustrative Examples

In order to understand the working of this algorithm, we take 6 SUs A, B, C, D, E and F with Node_id 0, 1, 2, 3, 4 and 5 respectively and we consider Node_id 1 as malicious node under the influence of CEPA and as shown in Fig. 7. SU 1 is called the parent SU of 0, 2, 3 and 5 as SU 1 is one-hop distance from them and also 0, 2, 3 and 5 are the neigbours SU of parent SU 1. There are 4 available channels that are available for use i.e., {101, 102, 103, 104} and the interface IDs are given as a1, b1, c1, d1, e1, f1, g1 and h1. Information on channel usage is obtained by deploying a counter. Initially, the assignment is not started yet, which can be indicated as 101-0, 102-0, 103-0, 104-0. For illustration purpose we have consider only four fields in an SU i.e., {Node_id, Neighbors_no, Neighbor_Degree_table, Bad_Counter}. The Node_id field specifies the identifier (ID) of the SU. The number is unique to each SU. The neighbor_no field stores the numbers of neighbours of the SU. The Neighbor_Degree_table contains the degree information, or the total number of neighbours for each SU in the network. The value in the Neighbor_Degree_table is calculated by (Neighbors_no + (Node_id/100)). For example, for SU A, Neighbors_no is 2 and Node_id is 0, therefore (2+(0/100)) is 2.0 (first value of third field). Correspondingly the other values are calculated. So, following are the values of the fields of the SUs: SU A − {0, 2, (2.0, 4.10, 3.20, 3.30, 2.40, 2.50), 0}, SU B − {1, 4, (2.0, 4.10, 3.20, 3.30, 2.40, 2.50), 0}, SU C − {2, 3, (2.0, 4.10, 3.20, 3.30, 2.40, 2.50), 0}, SU D − {3, 3, (2.0, 4.10, 3.20, 3.30, 2.40, 2.50), 0}, SU E − {4, 2, (2.0, 4.10, 3.20, 3.30, 2.40, 2.50), 0}, SU F − {5, 2, (2.0, 4.10, 3.20, 3.30, 2.40, 2.50), 0}. Here, in this distributed algorithm assignment each SU check whether its (Neighbors_no + (Node_id/100)) value is the highest value in its neighbor_ degree_table or not. Here, Node_id 1 has the highest value. So, we start the assignment with Node_id 1. Since, the node is malicious therefore interface id a1, c1, d1, e1 will be infected. Channel number 101 is assigned to interface id a1 since $|\mathcal{CA}(1)| < 2$ and $|\mathcal{CA}(0)| < 2$, the highest used channel

is picked (any of the other channels can potentially be chosen because all counter values are zero). The information on channel usage is then updated to 101-1, 102-0, 103-0, and 104-0. Next, interface id c1 is selected for assignment and as $|\mathcal{CA}(1)| < 2$ and $|\mathcal{CA}(2)| < 2$ and also it is infected edge, therefore the highest used resource among the resources in M is chosen, i.e., channel number 101 is selected. Next we update the channel usage information to 101-2, 102-0, 103-0, 104-0. Similarly, interface id d1 and e1 are assigned with 101 each and we update the the channel usage information to 101-4, 102-0, 103-0, 104-0. Next, node 1 disseminates CHNL_USAGE message to nodes having hop count=1 i.e., neighbors node in the topology [refer Fig. 7]. So nodes with IDs 0, 2, 3 and 5 gets this message. CHNL_USAGE messages {1, a1,101}, {1, c1,101}, {1, d1,101} and {1, e1,101} are supposed to be disseminated to Node_id 0, 2, 3 and 5 but due to malicious intent of node 1, it sends {1, a1,0}, {1, c1,0}, {1, d1,0}, {1, e1,0} and channel info is 0 means interfaces a1, c1, d1, e1 are not assigned any channel. In that instance, CHNL_CHNGE message possessed by neighbour node 0 is {1, a1,101}. Here, it checks for discrepancies in CHNL_USAGE message received (non updated CHNL_USAGE message). That is, if neighbour SU− >interface id− >Channel (from CHNL_CHANGE message) is not equal to parent SU− >Interface id− >Channel (from CHNL_USAGE message) then a discrepancy is found. Here, we see that node 0.a1.101(channel) ≠ node 1.a1.0 (channel) so, discrepancy is found. So, it creates MONITOR_REQUEST message and broadcast. Here, channel 101 should be allocated to interface a1, but instead false channel 0 is allocated. Next, it checks whether *Bad_counter* H (No. of Neighbor nodes of node 1 is 4). Since Bad_counter is initially 0 and $0 < 4$, it increments Bad_counter of node 1 to 1. Similarly, CHNL_CHNGE message possessed by neighbour node 2 is different to CHNL_USAGE message parent node 1, so discrepancy is found. So, increments Bad_counter of node 1 to 2, creates MONITOR_REQUEST message and broadcast. Similarly discrepancy is found both by neighbour node 3 and 5. So, Bad_counter of node 1 becomes 4. Since *Bad_counter* = 4, declare the node 1 as malicious.

Every node will maintain one CHNL_USAGE message (previous message) and another CHNL_USAGE message (latest). It checks if CHNL_USAGE message (previous)= CHNL_USAGE message (latest) and all interface IDs contain same channel and it is the highest used channel, then it declares as CEPA attack. And If CHNL_USAGE message (previous)= CHNL_USAGE message (latest) and some interface IDs contain same channel and it is a highly used channel, then it is a NEPA attack. In our example, CHNL_USAGE message (previous)= CHNL_USAGE message (latest) and all interface ID a1, c1, d1, e1 contain same channel i.e., channel no 101 and it is the highest used channel, so it is declared as CEPA attack.

To understand how this algorithm detects LORA, we consider 6 SUs A, B, C, D, E and F with Node_id 0, 1, 2, 3, 4 and 5 respectively and we consider Node_id 3 as malicious node under the influence of LORA and it is shown in Fig. 8. SU 1 is called the parent SU of 0, 2, 3 and 5 as SU 1

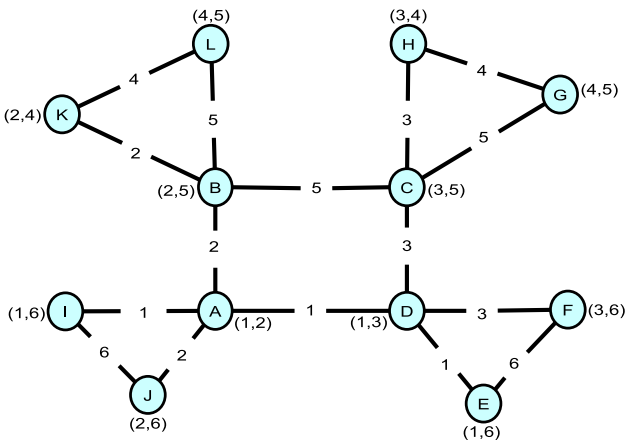

Fig. 9 Spectrum allocation under distributed normal assignment (topology 2 when N = 12 and M = 6)

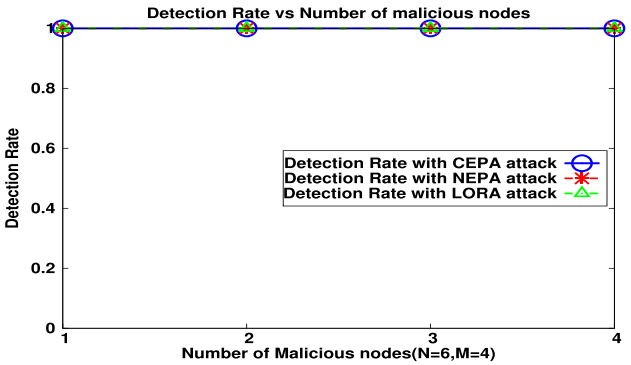

Fig. 10 Detection rate vs. number of malignant nodes (Topology 1)

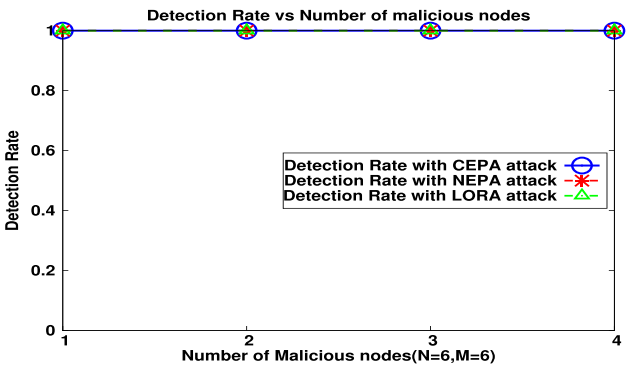

Fig. 11 Detection rate vs. number of malignant nodes (Topology 1)

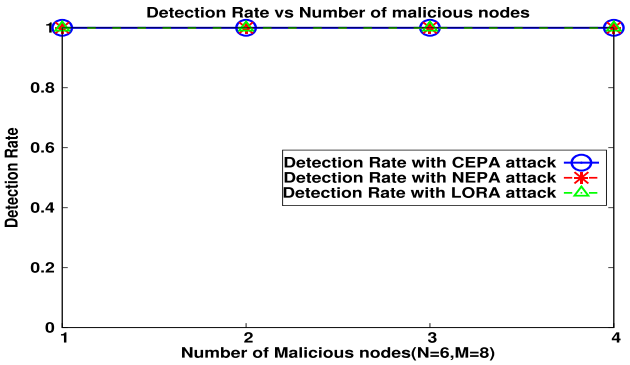

Fig. 12 Detection rate vs. number of malignant nodes (Topology 1)

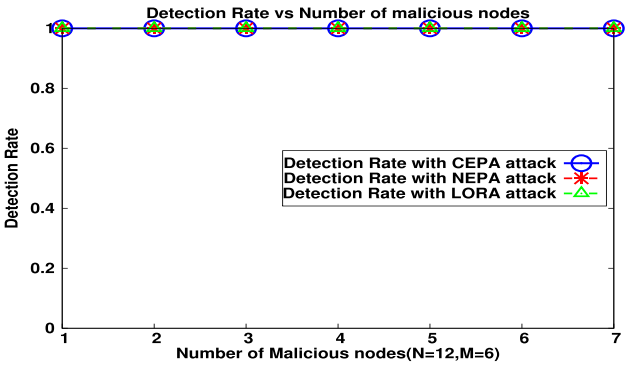

Fig. 13 Detection rate vs. number of malignant nodes (Topology 2)

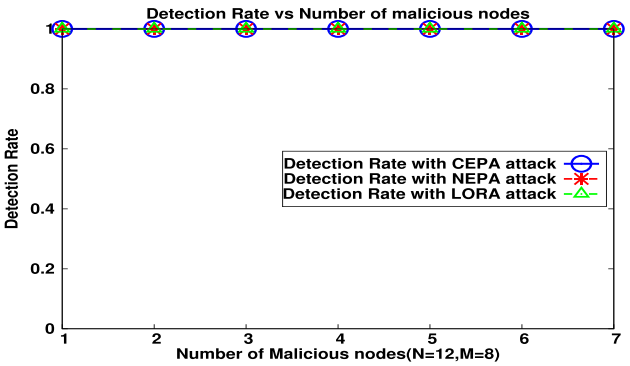

Fig. 14 Detection rate vs. Number of malignant nodes (Topology 2)

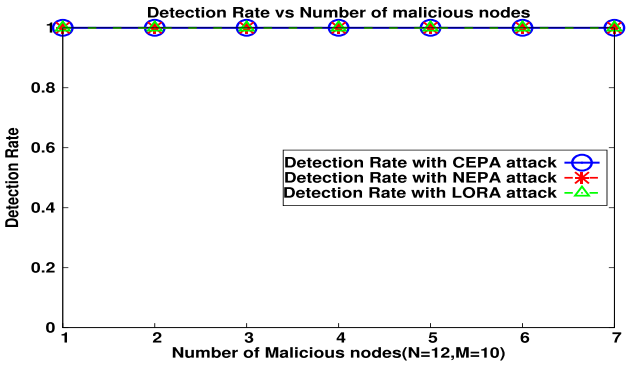

Fig. 15 Detection rate vs. number of malignant nodes (Topology 2)

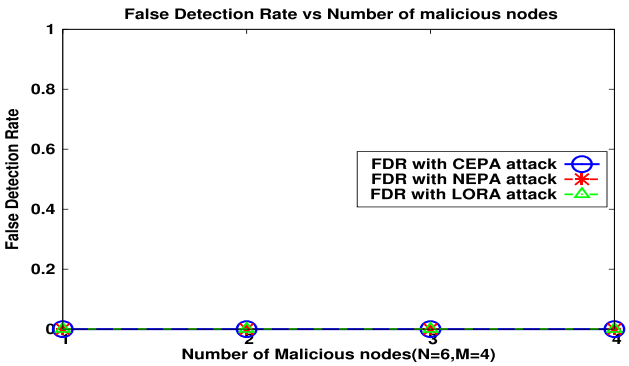

Fig. 16 False detection rate vs. number of malignant nodes (Topology 1)

is one-hop distance from them and also 0, 2, 3 and 5 are the neigbours SU of parent SU 1. There are 4 available channels that are available for use i.e., {101, 102, 103, 104} and the interface IDs are given as a1, b1, c1, d1, e1, f1, g1 and h1. Information on channel usage is obtained by deploying a counter. Initially, the assignment is not started yet, which can be indicated as 101-0, 102-0, 103-0, 104-0. The value in the Neighbor_Degree_table is calculated by (Neighbors_no + (Node_id/100)). For example, for SU A, Neighbors_no is 2 and Node_id is 0, therefore (2+(0/100)) is 2.0 (first value of third field). Correspondingly the other values are calculated. So, following are the values of the fields of the SUs: SU A − {0, 2, (2.0, 4.10, 3.20, 3.30, 2.40, 2.50), 0}, SU B − {1, 4, (2.0, 4.10, 3.20, 3.30, 2.40, 2.50), 0}, SU C − {2, 3, (2.0, 4.10, 3.20, 3.30, 2.40, 2.50), 0}, SU D − {3, 3, (2.0, 4.10, 3.20, 3.30, 2.40, 2.50), 0}, SU E − {4, 2, (2.0, 4.10, 3.20, 3.30, 2.40, 2.50), 0}, SU F − {5, 2, (2.0, 4.10, 3.20, 3.30, 2.40, 2.50), 0}. Here, in this distributed algorithm assignment each SU check whether its (Neighbors_no + (Node_id/100)) value is the highest value in its neighbor_degree_table or not. Here, Node_id 1 has the highest value. So, we start the assignment with Node_id 1. Channel number 101 is assigned to interface id a1 since $|\mathcal{CA}(1)| < 2$ and $|\mathcal{CA}(0)| < 2$, the lowest used channel is picked (any of the other channels can potentially be chosen because all counter values are zero). The channel usage information is then updated to 101-1, 102-0, 103-0, and 104-0. Next, interface id c1 is selected for assignment and as $|\mathcal{CA}(1)| < 2$ and $|\mathcal{CA}(2)| < 2$, therefore the lowest used resource among the resources in M is picked, i.e., channel number 102 is chosen. The information on channel usage is then updated to

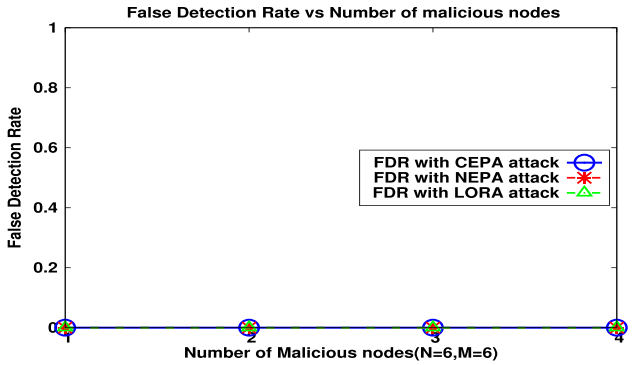

Fig. 17 False detection rate vs. number of malignant nodes (Topology 1)

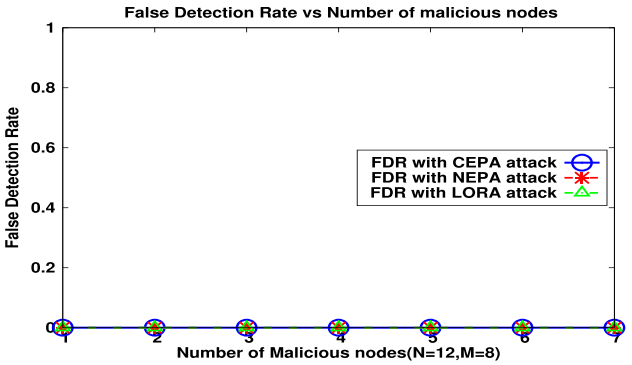

Fig. 20 False detection rate vs. number of malicious nodes (Topology 2)

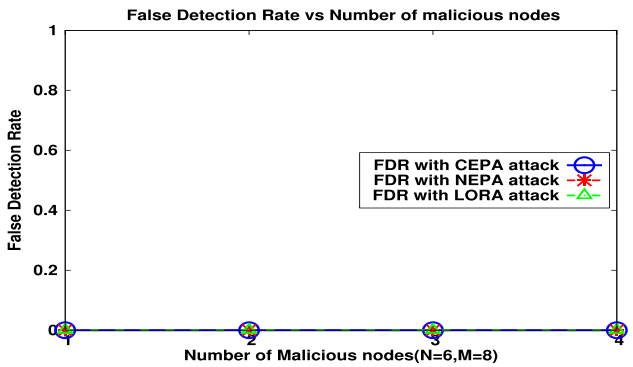

Fig. 18 False detection rate vs. number of malignant nodes (Topology 1)

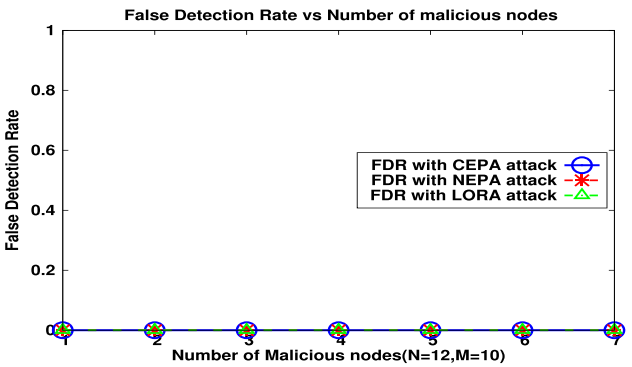

Fig. 21 False detection rate vs. number of malignant nodes (Topology 2)

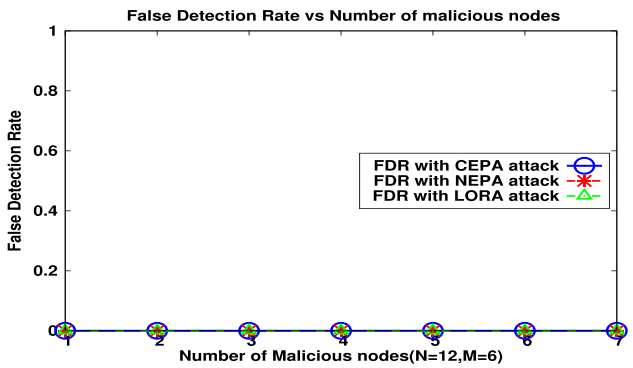

Fig. 19 False detection rate vs. number of malignant nodes (Topology 2)

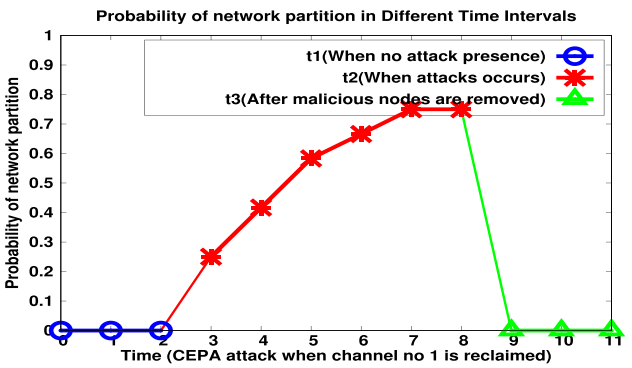

Fig. 22 Probability of network partition at different time intervals for CEPA attack (Topology 1)

101-1, 102-1, 103-0, 104-0. Next, interface id d1 is selected for assignment. Now, SU 1 has already been assigned with channel 101 and 102. Thus, $|\mathcal{CA}(1)| = 2$ and $|\mathcal{CA}(5)| < 2$. So the least used channel among the channels in $\mathcal{CA}(1)$ is chosen, i.e., channel number 101 and we update the channel usage information to 101-2, 102-1, 103-0, 104-0. Similarly, interface id e1 are assigned with 102 and we update the information on channel usage to 101-2, 102-2, 103-0, 104-0.

Next allocation will be performed by SU 3 (malicious SU) since it has the next highest value in the Neighbor_Degree_table. Here, SU 3 does not change channel assignment (thus no CHNL_CHNGE message modification), but transmits wrong CHNL_USAGE message (using normal allocation only). Also here malicious SU 3 updates the channel usage information in such a way that the least used

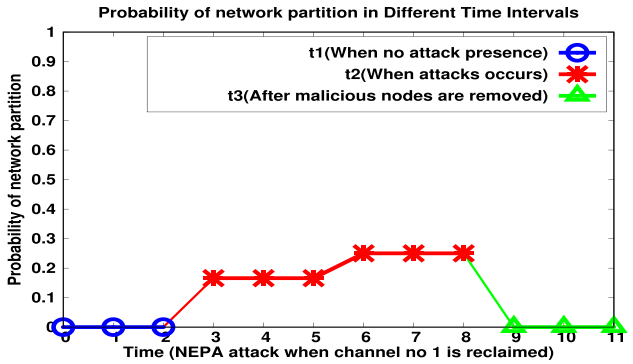

Fig. 23 Probability of network partition at different time intervals for NEPA attack (Topology 1)

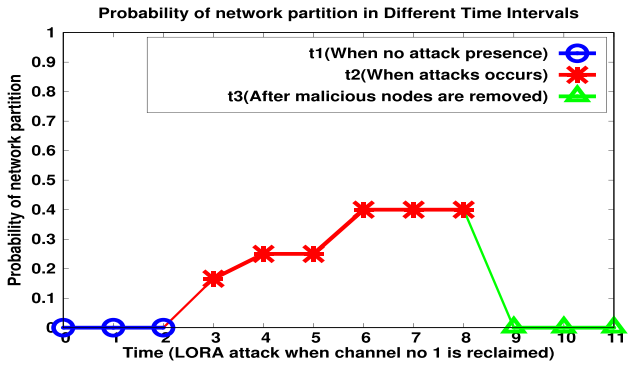

Fig. 24 Probability of network partition at different time intervals for LORA attack (Topology 1)

channels becomes mostly used and vice versa. Here, SU 3 changes the channel usage information to 101-0, 102-0, 103-2, 104-2. Then it disseminates CHNL_USAGE message to nodes having hop count=1, i.e., neighbors node in the topology [refer Fig. 8]. So nodes with IDs 4, 5 and 1 gets this message. CHNL_USAGE messages {3, e1,102}, {3, h1,0} and {3, f1,0} are supposed to be disseminated to Node_id 4, 5 and 1 but due to malicious intent of node 3 (tampering channel usage information), it sends {3, e1,103}, {3, h1,103} and {3, d1,104}. At this instance, CHNL_CHNGE message possessed by neighbour node 4 is {3, h1,0}. Here, it checks for discrepancies in CHNL_USAGE message received (non updated CHNL_USAGE message). If neighbour SU− >interface id− >Channel (from CHNL_CHANGE message) is not equal to parent SU− >Interface id− >Channel (from CHNL_USAGE message) then a discrepancy is found. Here, we see that SU 4.h1.0 (channel) ≠ SU 4.h1.103 (channel) so, discrepancy is found. Then it creates MONITOR_REQUEST message and broadcasts. Next, a neighbor (node 4) checks whether $Bad\_counter < H$ (No. of Neighbor SUs of SU 3 is 3). Since Bad_counter is initially 0 and $0 < 3$, it increments Bad_counter to 1. Next, CHNL_CHNGE message possessed by neighbour node 5 is different from CHNL_USAGE message parent node 3. So discrepancy is found, so it increments Bad_counter to 2, creates MONITOR_REQUEST message and broadcasts. Similarly discrepancy is found by neighbour node 1 and the Bad_counter updated to 3 and creates MONITOR_REQUEST message and broadcasts it. Since $Bad\_counter$=3, node 3 is declared as malicious.

Every node will maintain one CHNL_USAGE message (previous message) and another CHNL_USAGE message (latest). Here, in this scenario, CHNL_USAGE message is modified, so it should not be CEPA nor NEPA attack and so it checks for LORA attack. Also, it sees that CHNL_CHANGE message is not modified, it is declared as LORA attack.

Simulation and Performance Evaluation

We evaluate the effectiveness of the proposed security mechanism using simulation-based experiments. Without taking into account path loss or fading, our simulation is carried out under the premise of a stationary and noiseless radio network. There are two different performance indicators that are employed: detection rate and false detection rate. Detection rate(DR) is the ratio of the no. of detected malignant SUs to the total no. of malignant SUs present. False detection rate(FDR) is the ratio of the no. of honest SUs detected as malignant to the no. of malignant SUs (Fig. 9).

Figures 10, 11, 12, 13, 14 and 15 present the detection rate vs. number of malignant SUs for the Topology 1 and Topology 2 respectively. Here in Topology 1, we observe that we able to achieve perfect detection for all cases when M is 4, 6 and 8 for varying no. of malignant SUs. Similarly for Topology 2 also, we are able to achieve perfect detection for all cases when M is 6, 8 and 10 for varying no. of malignant SUs. Similarly, we performed simulation for other topologies in the same fashion and we also got similar good results. Since, the results are similar; we put only the results obtained from topology 1 and topology 2 to save up space. In order to mitigate the aforementioned attacks, the “Smart Neighbour Detection Mechanism" use the neighboring nodes of the malicious node use their own channel assignment and usage information to identify irregularities. A single MONITOR_REQUEST message is used for disseminating the information regarding the observed anomalies. The neighboring nodes of the malicious node solely rely on their channel assignment and usage information to confirm the accuracy of the CHNL_USAGE message and develop their own conclusions about the particular node. Since, the algorithm basically works on the message passing technique and also we have also considered half duplex communication, where each link between two SUs can only communicate one direction at a time. Hence, changes of messages colliding

are negligible in such scenario. Moreover, we have also considered noiseless environment, so chances of reaching the messages to the targeted destination are very high. Because of the above reasons, the method works exceptionally well in detecting the malicious nodes from the network and hence we get good results in terms of detection rate.

Figures 16, 17, 18, 19, 20 and 21 illustrate the false detection rate vs. number of malignant SUs for Topology 1 and Topology 2. Here in Topology 1, we observe that we able to achieve perfect zero false detection for all cases when M is 4,6 and 8 for varying number of malignant SUs. Similarly for Topology 2 also, we are able to achieve perfect zero false detection for all cases when M is 6, 8 and 10 for varying number of malignant SUs.

Next, we used probability of network partition as a performance parameter to evaluate the attack's efficacy. The probability of network partition can be defined as the probability that the network is partitioned when a channel is reclaimed by the PU. It is calculated as the ratio of the number of disconnected components if a channel is redeemed by PU in the network to the total number of possible disconnected components in the network (i.e., Total number of SUs). Figures 22, 23 and 24 illustrate the probability of network partition for CEPA, NEPA and LORA attacks at different time intervals. Here, first the malicious nodes are detected by "Smart Neighbor Detection Mechanism". After that identified malicious nodes are ignored (or isolated) by the other honest nodes. Then, the channel reassignment is done by the normal assignment algorithm as in Algorithm 1 to the remaining non malicious nodes. We assume channel no 1 is reclaimed by PU. Initially at time t1, the probability of network partition is zero as in this point of time no malicious nodes are present, so normal assignment will be performed resulting in zero probability of network partition. At time t2, we introduce malicious nodes causing network partition to occur. As we increase the no. of malignant nodes in the system, we can see the probability of network partition increases. At time t3, the proposed detection mechanism is applied and the detected malicious nodes removed from the network. After that reassignment of the channels has been performed using the normal assignment algorithm to the remaining non malicious nodes. At this point of time, we examine probability of network partition and we get zero probability of network partition. This is due to the fact that malignant nodes have been removed by now.

Conclusions

In this paper, we present an algorithm called "Smart Neighbour Mechanism " which detects the attacks that can be arise in distributed CRN resource allocation, viz., CEPA, NEPA and LORA. Through numerical simulations, it was found that the method has a high detection rate and a low false detection rate. We also analyze the probability of network partition for CEPA, NEPA and LORA attacks at different time intervals. Furthermore, we demonstrate how the suggested technique successfully detects malicious nodes. After removing the detected malicious nodes, the network behaves in a normal way. A future work may involve the development of intrusion detection systems and automated response systems to guarantee CRNs will be self-healing. The suggested strategy can potentially be expanded to address more realistic scenarios that account for mobility, path loss, and fading.

Funding None reported.

Data Availability During the investigation, no datasets were generated or examined.

Declarations

Conflict of interest None.

References

1. Mitola J. Cognitive radio: an integrated agent architecture for software defined radio. PhD thesis in Royal Institute of Technology (KTH); 2000.
2. Mitola J, Maguire GQ. Cognitive radio: making software radios more personal. IEEE Pers Commun. 1999;6(4):13–8.
3. Cordeiro C, Challapali K, Birru D, Shankar S. IEEE 802.22: the first worldwide wireless standard based on cognitive radios. In: First IEEE international symposium on new frontiers in dynamic spectrum access networks, DySPAN; 2005. pp. 328–337.
4. Ghosh C, Agrawal DP. Channel assignment with route discovery (CARD) using cognitive radio in multi-channel multi-radio wireless mesh networks. In: 1st IEEE workshop on networking technologies for software defined radio networks; 2006. pp. 36–41.
5. Chen-li D, Guo-an Z, Jin-yuan G, Zhi-hua B. A route tree-based channel assignment algorithm in cognitive wireless mesh networks. In: International conference on wireless communications and signal processing, Nanjing; 2009. pp. 1–5.
6. Canberk B, Oktug S. Xpec, a cross-layer spectrum assignment in cognitive radio networks. In: Advanced networks and telecommunication systems (ANTS), IEEE 4th international symposium; 2010. pp. 67–69.
7. Irwin RE, MacKenzie AB, DaSilva LA. Resource-minimized channel assignment for multi-transceiver cognitive radio networks. IEEE J Sel Areas Commun. 2013;31(3):442–50.
8. Zhao J, Cao G. Robust topology control in multi-hop cognitive radio networks. In: Proceedings IEEE INFOCOM; 2012. pp. 2032–2040.
9. Kim W, Kassler AJ, Felice M-Di, Gerla M. Urban-X, Towards distributed channel assignment in cognitive multi-radio mesh networks. In: IFIP wireless days; 2010. pp. 1–5.
10. Xin C, Ma L, Shen C-C. A path-centric channel assignment framework for cognitive radio wireless networks. Mob Netw Appl. 2008;13(5):463–76.

11. Tan LT, Le LB. Channel assignment with access contention resolution for cognitive radio networks. IEEE Trans Veh Technol. 2012;61(6):2808–23.
12. Hashem M, Barakat SI, AttaAlla MA. Distributed channel selection based on channel weight for cognitive radio network. In: 10th international computer engineering conference (ICENCO); 2014. pp. 115–120.
13. Alsarahn A, Agarwal A. Channel assignment in cognitive wireless mesh networks, advanced networks and telecommunication systems (ANTS). In: IEEE 3rd international symposium; 2009. pp. 1–3.
14. Alam S, Malik AN, Qureshi IM, Ghauri SA, Sarfraz M. Clustering-based channel allocation scheme for neighborhood area network in a cognitive radio based smart grid communication. IEEE Access. 2018;6:25773–84.
15. Hongshun Z, Xiao Y. Advanced dynamic spectrum allocation algorithm based on potential game for cognitive radio. in: information engineering and electronic commerce (IEEC), 2nd international symposium; 2010. pp. 1–3.
16. Elhachml J, Guennon Z. Cognitive radio spectrum allocation using genetic algorithm. EURASIP J Wirel Commun Netw. 2016;133(1):1–11.
17. Morabit YEl, Mrabti F, Abarkan EH. Spectrum allocation using genetic algorithm in cognitive radio networks. In: Third international workshop on RFID and adaptive wireless sensor networks (RAWSN); 2015. pp. 90–93.
18. Zhu L, Xu Y, Chen J, Li Z. The design of scheduling algorithm for cognitive radio networks based on genetic algorithm. In: IEEE International conference on computational intelligence & communication technology; 2015. pp. 459–464.
19. Chowdhury SA, Benslimane A, Akhter F. Throughput maximization of cognitive radio network by conflict-free link allocation using neural network. IEEE International Conference on Communications (ICC). 2017;2017:1–6.
20. Huang X, Du J, Kuang S. A channel assignment algorithm of CRSNs based on FOA. In: 12th international conference on natural computation. Fuzzy systems and knowledge discovery (ICNC-FSKD), Changsha; 2016. pp. 680–685.
21. Maheshwari P, Singh AK. A fuzzy logic based approach to spectrum assignment in cognitive radio networks. In: IEEE international advance computing conference (IACC), Banglore; 2015. pp. 278–281.
22. Saleem Y, Bashir A, Ahmed E, Qadir J, Baig A. Spectrum-aware dynamic channel assignment in cognitive radio networks. In: International conference on emerging technologies, Islamabad; 2012. pp. 1–6.
23. Alsarhan A, Agarwal A. Cluster-based spectrum management using cognitive radios in wireless mesh network. In: Computer communications and networks, proc. 18th international conference; 2009. pp. 1–6.
24. Xie X, He L, Yang H, Ma B. An efficient and unbiased power control algorithm based on game theory in cognitive radio. J Comput. 2014;9:1990–8.
25. Wu Y, Wang B, Ray Liu KJ, Clancy TC. A scalable collusion-resistant multi-winner cognitive spectrum auction game. IEEE Trans Commun. 2009;57(12):3805–16.
26. Ji Z, Liu KJR. Multi-stage pricing game for collusion-resistant dynamic spectrum allocation. IEEE J Sel Areas Commun. 2008;26(1):182–91.
27. Zhou X, Zheng H. TRUST: a general framework for truthful double spectrum auctions. In: IEEE INFOCOM, Rio de Janeiro; 2009. pp. 999–1007.
28. Yu L, Liu C, Hu W. Spectrum allocation algorithm in cognitive ad-hoc networks with high energy efficiency. In: The international conference on green circuits and systems, Shanghai; 2010. pp. 349–354.
29. Tabassum M, Razzaque M, Hassan M, Almogren A, Alamri A. Interference-aware high-throughput channel allocation mechanism for CR-VANETs. EURASIP J Wirel Commun Netw. 2016;2:1–15.
30. Zhao C, Zou M, Shen B, Kim B, Kwak K. Cooperative Spectrum allocation in centralized cognitive networks using bipartite matching. In: Glob telecommunications conference, IEEE GLOBECOM; 2008. pp. 1–6.
31. Sohan TA, Haque HH, Hasan A, Islam J, Alim Al Islam ABM. A graph coloring based dynamic channel assignment algorithm for cognitive radio vehicular Ad Hoc networks. In: International conference on networking systems and security (NSysS), Dhaka; 2016. pp. 1–8.
32. Pareek U, Lee DC. Resource allocation in bidirectional cooperative cognitive radio networks using swarm intelligence. In: IEEE symposium on swarm intelligence, Paris; 2011. pp. 1–7.
33. Salah A, El-Atty HA, Rizk RY. Cross-layer routing optimization for centralized multi-hop cognitive radio networks. In: 11th International computer engineering conference (ICENCO), Cairo; 2015. pp. 25–31.
34. Lee DH, Jeon WS. Channel assignment and routing with overhead reduction for cognitive radio-based wireless mesh networks. In: International conference on wireless communications and signal processing (WCSP), Nanjing; 2011. pp. 1–5.
35. Wang J, Yuqing H. A cross-layer design of channel assignment and routing in Cognitive Radio Networks. In: Proceedings of the 3rd international conference on computer science and information technology, Chengdu; 2010. pp. 542–547.
36. Anifantis E, Karyotis V, Papavassiliou S. A Markov random field framework for channel assignment in cognitive radio networks. In: IEEE international conference on pervasive computing and communications workshops, Lugano; 2012. pp. 770–775.
37. Naveed A, Kanhere SS. NIS07-5: Security vulnerabilities in channel assignment of multi-radio multi-channel wireless mesh networks In: IEEE Globecom; 2016. pp. 1–5.
38. Singh WN, Marchang N. A review on spectrum allocation in cognitive radio network. Int J Commun Netw Distrib Syst. 2019;23(2):172–93.
39. Sazia Parvin, Hussain FK, Hussain OK, Han S, Tian B, Chang E. Cognitive radio network security: a survey. J Netw Comput Appl. 2012;35(6):1691–708.
40. Singh WN, Marchang N. Security vulnerability in spectrum allocation in cognitive radio network. In: Kamal R, Henshaw M, Nair P., editors. International Conference on advanced computing networking and informatics. Advances in intelligent systems and computing. Springer, Singapore; 2019. pp. 215–224.
41. Singh WN, Marchang N. Impact of security attacks on spectrum allocation in cognitive radio networks. In: Mishra M, Kesswani N, Brigui I, editors. Applications of computational intelligence in management and mathematics. ICCM 2022, Springer Proceedings in Mathematics and Statistics, vol 417. Springer, Cham; 2022.