Chapter 5

Assignment Algorithms of Sensing Task in Cognitive Radio Networks

5.1 Introduction to Cooperative Sensing and Assigning The Sensing Task to Nodes

Spectrum sensing performance is affected by many uncertainty problems [27], [28]. This degradation in performance can be solved by cooperative spectrum sensing [29], [30] where the secondary users (SUs) share the sensing information with each other to reach a unique solid global decision among the cooperating nodes.

Cooperation between SUs can be performed in two ways: Centralized or Distributed [31]. In the centralized scheme, there exists a central entity called Fusion Center (FC) which communicates with nodes via a Common Control Channel (CCC). FC is responsible for controlling and scheduling the sensing process among cooperating nodes and determining the time interval needed for the sensing time slot. After performing the sensing process by the SUs, they report their local sensing data to the FC through a reporting channel. Sensing nodes may report hard decisions, where nodes send their local decisions of the presence (send 1) or absence (send 0) of the primary user (PU). Alternatively, sensing nodes can report soft decisions, where nodes send their observation -which results from the sensing operation- directly to the FC. Finally, the FC combines all sensing data to reach a global decision about the presence or absence of PUs and then this decision is sent back to SUs through the CCC.

In the distributed Scheme [31], cognitive nodes share information among each other so they make their own decisions concerning which part of the spectrum they can use.

obeikandl.com

In applications that depend on wide spectrum sensing, one node will not be able to carry out the sensing process in one sensing cycle (time slot). Due to hardware constraints, a SU will probably be able to sense a limited bandwidth (say k channels). Additionally, sensing multiple channels will result in a relatively large sensing time and overhead which is not recommended for efficient spectrum utilization.

Our work is concerned with the centralized approach. The objective is to distribute the sensing task of wide spectrum over cooperating nodes where each node has access to different set of channels with some overlap with other nodes. The sensing task is distributed among the secondary nodes to allow each node to sense only a subset of the whole frequency range and then local decisions are aggregated at the FC to reach a global decision about the availability of every channel. This way, the time required to find out about the activity in each channel is reduced. Instead of randomly distributing the sensing task among nodes, two approaches are presented in this chapter. They aim to reduce the overall probability of sensing error by choosing the best candidate nodes for each channel.

In this chapter, the main contribution is to present and compare between two approaches aiming to distribute the sensing task among the cognitive nodes in a centralized, hard-decision based CRN such that the global probability of sensing error is minimized. Hard decisions method is adopted in many applications where hardware constraints or bandwidth limitations of the CCC may not allow sending the value of the observation. The first approach uses a fuzzy-based algorithm while the second approach applies the well known Hungarian Algorithm on a task-agent assignment problem. Candidate nodes for channels sensing are determined and the sensing task is assigned to them. Cooperative spectrum sensing is applied among nodes that participate in the sensing process which are selected via the presented assignment algorithms. Afterwards, the sensing process takes place and nodes report their local decisions to the FC which combines them to reach a global decision regarding the availability of each channel.

obeikandi.com

5.2 Formulating The Problem of The Sensing Task Assignment

Given a Cognitive Radio Network (CRN) consists of N SUs and M licensed channels, it is required to assign the task of sensing these M channels to the N SUs in a way that leads to reducing the sensing error. The objective is to choose best candidates of SUs for each channel (to minimize the global sensing error) subject to the following constraints:

1. Each SU can sense only k channels at a time from the M channels (possibly due to hardware constraints, CCC bandwidth limitations or to reduce the sensing overhead to increase the overall network lifetime).

2. Cover all channels.

5.3 Preliminaries

In this section, an overview about the generalized assignment problem, the Hungarian algorithm and the metrics for choosing good candidate nodes is given. The mapping between these definitions and the assignment problem is presented in later sections.

5.3.1 Generalized Assignment Problem

Assignment problems deal with the question of how to assign m items (e.g. jobs) to n machines (or workers) in the best possible way [32]. The generalized assignment problem consists of assigning, at a minimum total cost, a set of tasks to a set of agents with limited resource capacity [33].

This generalized assignment problem can be described as follows:

There are a number of agents and a number of tasks. Any agent can be assigned to perform any task, incurring some cost that may vary depending on the agent-task assignment. Moreover, each agent has a budget and the sum of the weight result from assigning some tasks to

obeikandi.com

it cannot exceed this budget. The goal is to find an assignment in which all agents do not exceed their budget and total cost of the assignment is minimized.

The mathematical form of the assignment problem :

Minimize $\sum_{i=1}^{m} \sum_{j=1}^{n} C(i,j) x_{ij}$

Subject to $\begin{cases} \sum_{i=1}^{m} w_{ij} x_{ij} \leq w_j & for\ j = 1, ..., n \\ \sum_{j=1}^{n} x_{ij} = 1 & for\ i = 1, ..., m \end{cases}$

where,

- $C(i,j)$ is the cost results from assigning task i to agent j.
- n is the total number of agents.
- m is the total number of tasks.
- w_j is the budget of agent j.
- w_{ij} is the weight results from assigning task i to agent j.
- $x_{ij} = 1$ if task i is assigned to agent j, otherwise $x_{ij} = 0$.

Mapping of the generalized assignment problem to this work:

obeikandl.com

Parameter	Mapping
Agent	Node
Task	Sensing a group of k channels
Cost	Sum of costs of all channels inside the group
Channel cost	Probability of sensing error (statistically calculated)
Budget	k channels

5.3.2 Hungarian algorithm

The Hungarian algorithm is one of many that have been devised to solve the linear assignment problem within time bounded by a polynomial expression of the number of agents [34]. Input for the Hungarian algorithm is a matrix $A(N\text{x}M)$. Each element in this matrix represents the task-agent cost. Output from the Hungarian algorithm is a matrix $R(N\text{x}M)$. $R(i, j) = 1$ if task j is assigned to agent i and $R(i, j) = 0$ otherwise.

The methodology of the Hungarian algorithm depends on three main steps:

1. **Find the opportunity cost matrix:** by subtracting from each row the minimum number of this row and subtracting from each column the minimum number of this column.

2. **Test for the optimal assignment:** by counting the number of horizontal and vertical lines required to cover all zeros in the A matrix. If an optimal assignment can be made (the total number of lines required equals N) then the algorithm will be terminated, else go to step 3.

3. **Revise the opportunity cost matrix:** Determine the smallest uncovered entry. Subtract it from each uncovered row, and then add it to each covered column. Return to Step 2.

Later sections explain how the Hungarian algorithm is used in our work by modeling the channel-node assignment problem as task-agent assignment problem in which the cost resulting from assigning a channel to a node is represented by the statistical computation of the

obeikandi.com

sensing error probability of this node over this channel.

5.3.3 Metrics for determination of good candidate nodes

The sensing task is distributed among SUs by assigning k channels to sensing nodes in a way that minimizes the global probability of sensing error at the FC.

Two indicators are considered for the ability of the sensing node to sense a channel correctly and report an error-free decision to the Fusion Center (FC):

1. The computed individual probability of sensing error of each node-channel pair.

2. The SNR of the FC-SU reporting subchannel for each node-channel pair.

5.3.3.1 The individual probability of sensing error of each node-channel pair

The individual probability of sensing error of each node-channel assignment is used as a measure to the contribution of this node in the global probability of error. To calculate this measure statistically, the system model assumed in [35] is considered here with slight modifications.

Consider X_i as the received signal at each node, where i is the SU index. Two hypothesis are possible:

a) No primary user (Noise only is present)

$$H_o : X_i \sim N(0, \sigma_{0_i}^2) \tag{5.1}$$

obeikandi.com

b) Active primary user (A signal is received at the sensing node)

$$H_1 : X_i \sim N(0, \sigma_{1_i}^2) \tag{5.2}$$

where,

$\sigma_{0_i}^2$ is the additive white gaussian noise variance (noise power) at each node. $\sigma_{1_i}^2$ is the variance of the received signal at each node and

$$\sigma_{1_i}^2 > \sigma_{0_i}^2 \tag{5.3}$$

The decision metric of each node is the Log-Likelihood Ratio (LLR) which is the ratio of the probability of the observation given the presence of PU to that given the absence of PU.

$$LLR_i = \log \frac{\Pr(X_i|H_1)}{\Pr(X_i|H_o)} \tag{5.4}$$

For a single node, error occurs when the LLR is above the sensing threshold while no PU is present (False Alarm) or when the LLR is below the threshold while the PU is active (Misdetection). The individual probability of error at each node is computed as follows:

$$P_{e_i} = \pi_{o_i} P_{FA_i} + \pi_{1_i} P_{MD_i} \tag{5.5}$$

where,
π_{o_i} is the probability of idle PU, π_{1_i} is the probability of active PU ($\pi_{1_i} = 1 - \pi_{o_i}$). Both probabilities are assumed known for each node. P_{FA_i} is the probability of false alarm and given by: $P_{FA_i} = Pr(LLR_i \geq t_o|H_o)$ and P_{MD_i} is the probability of misdetection and given by: $P_{MD_i} = Pr(LLR_i \leq t_o|H_1)$, where t_o is the decision threshold.

To find P_{e_i} we need to find P_{FA_i} and P_{MD_i} which depend on the probability density function (pdf) of LLR_i.

obeikandl.com

Assuming Additive White Gaussian Noisy (AWGN) channel, we prove that the pdf of LLR_i is given as (Proof can be found in Appendix B):

$$f_Y(y_i) \;=\; |\frac{1}{b_i}| \, \frac{1}{\sqrt{2\ \pi\ \frac{1}{b_i}(Y_i - a_i)\ e^{\frac{1}{b_i}(Y_i - a_i)}}} \tag{5.6}$$

Where, Y_i is the LLR_i, $a_i = log\frac{\sigma_{0_i}}{\sigma_{1_i}}$, $b_i = \frac{\sigma_{1_i}^2}{2\sigma_{0_i}^2} - \frac{1}{2}$ in case of H_1 and $b_i = -\frac{\sigma_{0_i}^2}{2\sigma_{1_i}^2} + \frac{1}{2}$ in case of H_o.

Therefore,

Probability of false alarm:

$$P_{FA_i} = \int_{t_o}^{\infty} f_Y(y_i)\, dy \tag{5.7}$$

where, $f_Y(y_i)$ is as given in equation (5.6) and the constants "a_i" and "b_i" are as given in the assumption of H_o.

Probability of misdetection:

$$P_{MD_i} = \int_{0}^{t_o} f_Y(y_i)\, dy \tag{5.8}$$

where, $f_Y(y_i)$ is as given in equation (5.6) and the constants "a_i" and "b_i" are as given in the assumption of H_1.

By substituting in equation (5.5) each node can get its individual probability of error which is then reported to the FC to run the fuzzy fusion.

obeikandl.com

5.3.3.2 SNR of the reporting sub-channel of each node-channel pair

The FC receives the local hard decisions of each node about the availability of the channels assigned to it to combine them and reach a solid global decision regarding each channel in the band of interest. One of the factors that might affect this operation is the SNR of the reporting subchannel between the FC and the sensing node. A node might have a very good sensing capability, however, a low SNR of the linking subchannel (of the CCC) between it and the FC can cause erroneous decisions to reach the FC affecting the global decision and causing a global sensing error.

That's why the SNR of the FC-SU subchannel is chosen to be a metric on which selecting a good sensing candidate node depends. It is assumed that by means of spectral estimation techniques, the FC is able to determine an estimation of this SNR [36].

5.4 Proposed Assigning Algorithms

5.4.1 Fuzzy-based algorithm

5.4.1.1 Fuzzy fusion of metrics

Consider a fuzzy inference system (FIS) that runs for each node-channel pair. It takes two inputs, which are the individual probability of sensing error (denoted as $P_{e_{individual}}$) and the SNR of the reporting subchannel (denoted as $SNR_{reporting}$), and results in one output which can be called *candidate possibility*. The output represents a value between 0 and 1 that is considered a measure to what extent a node can be a good candidate for the sensing process of a certain channel. i.e. A measure for the node-channel pair to be a good assignment.

Three Gaussian membership functions for each input and output are considered. $P_{e_{individual}}$ and $SNR_{reporting}$, each has three membership functions called (low, medium and high) and the output "candidate possibility" has three membership functions called (No, Probably Yes and Yes). Membership functions are shown in Fig. 5.1, 5.2 and 5.3.

obeikandl.com

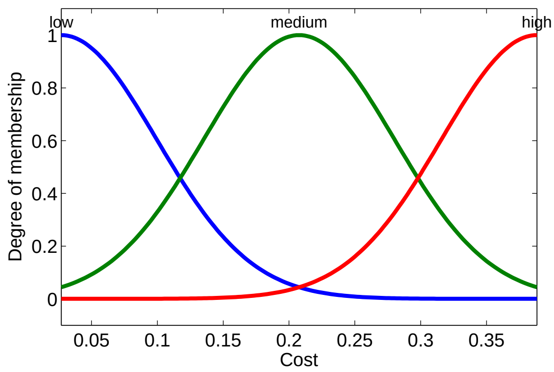

FIGURE 5.1: Membership functions of $P_{e_{individual}}$

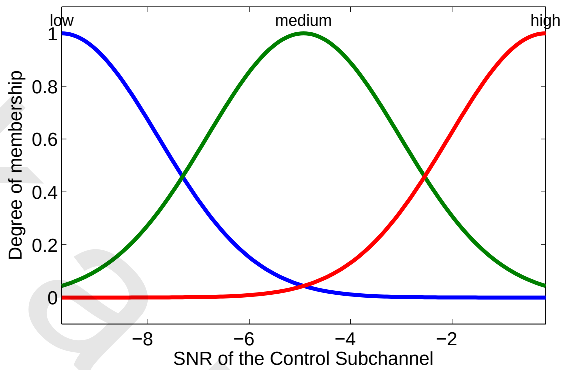

FIGURE 5.2: Membership functions of $SNR_{reporting}$

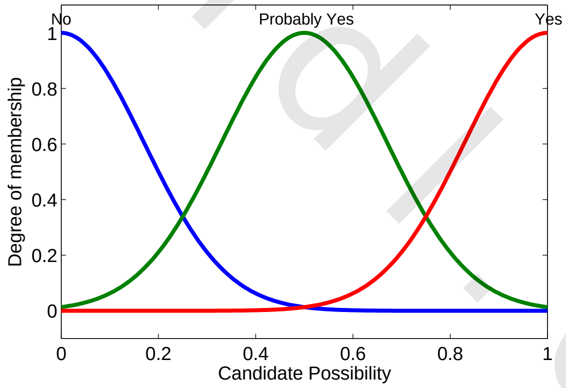

FIGURE 5.3: Membership functions of Candidate Possibility

Using a group of IF-THEN rules, shown in Table 5.1, the FC fuses the two inputs to determine to what extent a node-channel pair can be a good assignment. These IF-THEN rules are chosen based on human knowledge of the system. For instance, in rule number 3, if *low* individual probability of error is expected and *high* SNR of the reporting channel is estimated, the node is considered a good candidate to sense the corresponding channel. On

obeikandi.com

the contrary, in rule 7, if the individual probability of error is expected to be *high* and the estimated reporting SNR is *low*, the node is not considered a good candidate for sensing the corresponding channel.

No.	$P_{e_{individual}}$	$SNR_{reporting}$	Candidate Possibility
1	low	low	Probably Yes
2	low	medium	Yes
3	low	high	Yes
4	medium	low	No
5	medium	medium	Probably Yes
6	medium	high	Probably Yes
7	high	low	No
8	high	medium	No
9	high	high	No

TABLE 5.1: IF-THEN RULES

Fig. 5.4 shows the fuzzy reasoning process.

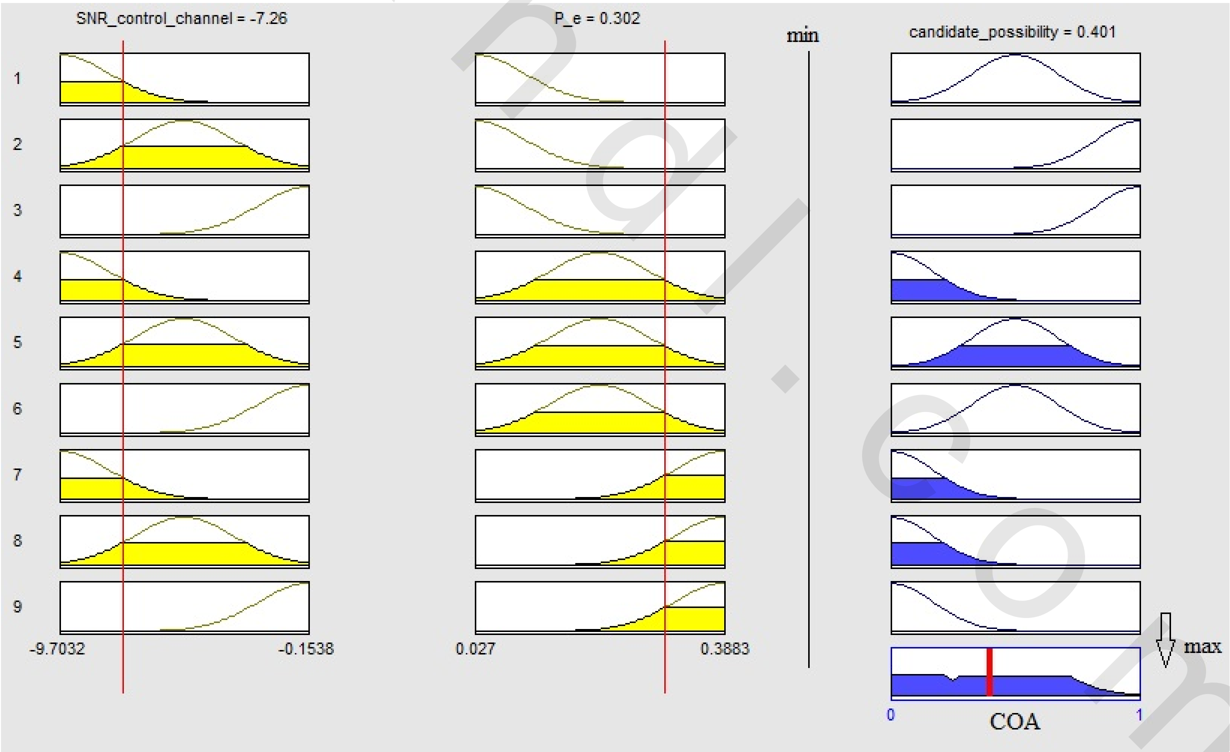

FIGURE 5.4: Fuzzy Reasoning Process

obeikandl.com

5.4.1.2 Channel assignment

The FC runs the fuzzy fusion on each node-channel pair resulting in a matrix (NxM) containing the values of candidate possibility for each possible assignment. Afterwards, each channel is assigned to its best k candidates (those with the highest candidate possibility values). This can be shown in the flow chart of Fig. 5.5.

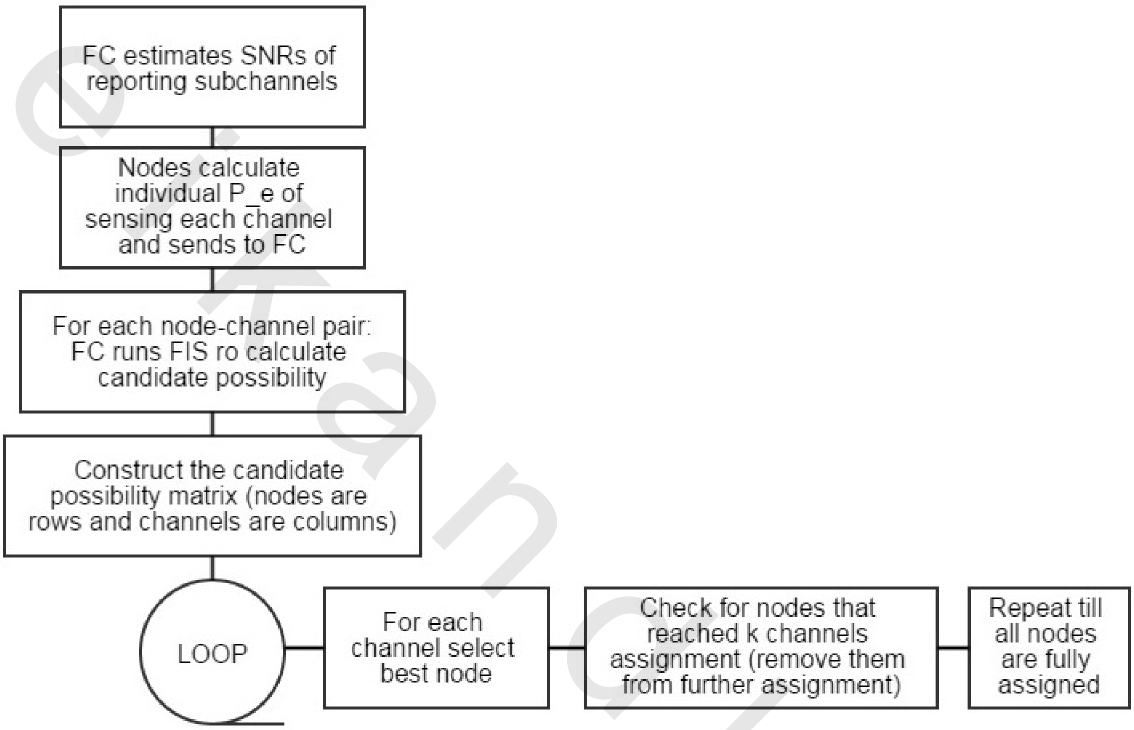

FIGURE 5.5: Flowchart of the assignment process based on fuzzy logic

5.4.2 Hungarian algorithm based task-agent assignment

According to the constraint which requires that each user senses only k channels, the assignment problem can be reduced where we have N SUs nodes and Y channel groups. Each group contains k channels and it is required to assign the Y groups to the N nodes with the objective of minimizing the global sensing error and under the constraint of not exceeding a certain cost budget for each group assignment. The cost of each group is the sum of the individual probability of error calculated for each channel in this group.

Channels are divided to overlapping groups. To ensure covering all channels, the total number of groups should be less than or equal the total number of nodes.

The assignment phase is described by algorithms 1 and 2.

Algorithm 1 Calculating Local Probability of Spectrum Sensing Error: Each node

Define I: Node applying the algorithm
Output V: Cost For Each Group Containing k Channels
for each channel **do**
  Calculate individual probability of error statistically
end for
for each group X **do**
  Calculate the cost for group X by summing the sensing error of k channels
end for
Node I sends to the fusion center the cost values of all groups

Algorithm 2 Channel Assignment Algorithm: Fusion Center

Define A : Cost Matrix for each node-group pair
Output R: Assignment matrix for each node-group pair where, $R(i, j) = 1$ if group j is assigned to user i and $R(i, j) = 0$ otherwise
Initialization The fusion center collects the individual cost values of all groups from all nodes in the network
Fusion center applies Hungarian Algorithm
User - group assignment is processed
for each node **do**
  Inform each node about its assigned k channels
end for
Fusion center receives confirmation message from all nodes

The flow of this process can be shown in the flowchart of Fig. 5.6.

5.5 Periodic re-assignment

Due to network changes, the following scenarios trigger the re-run of the assignment process:

- If a node leaves the region of potential interference of the PU and this resulted in one or more uncovered channels.

- If a node joins the region of potential interference of the PU and there is a need to update the assignment by adding the new sensing information received from the new

obeikandi.com

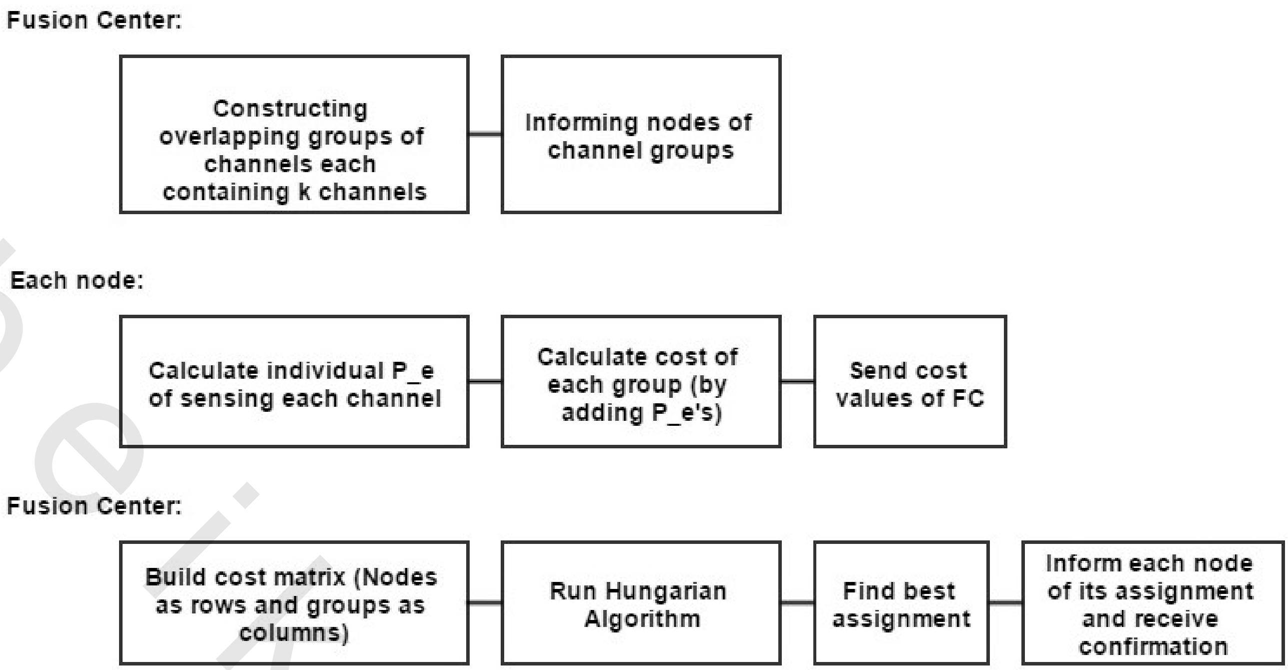

FIGURE 5.6: Flowchart of the assignment process based on the Hungarian algorithm

node.

5.6 Sensing Phase

Sensing takes place in an assumed Additive White Gaussian Noise (AWGN) environment. Each node takes one sensing sample from each of its assigned channels. All samples are applied to a Fast Fourier Transform (FFT) which enables the node to process them simultaneously and not sequentially [37]. Local decisions are made at each node and reported to the FC which combines them to reach a global decision regarding the availability of each channel.

obeikandi.com

5.7 Simulation Parameters

MATLAB is used as the simulation tool. A CRN is considered to be in an AWGN environment. It consists of 10 SUs, 80 channels available for sensing. The PU is free to access any of them at any time. Each node statistically calculates its probability of sensing error and reports it to the FC. Then, sensing task assignment takes place by either of the proposed methods. Finally, the sensing process takes place. Local hard decisions are obtained by nodes and reported to the FC which aggregates the nodes' local decisions using *Majority Fusion Rule* [38] and reach a global decision for the sensed channel activity.

Simulation parameters are shown in TABLE 5.2.

TABLE 5.2: Simulation Parameters

Parameters	Definition	Values
N	Number of secondary nodes	10
M	Number of channels that are to be sensed	80
k	Number of channels that a node can sense in one time slot	$\in$ [M/N M]
π_o	Probability of in-active PU	Uniformly distributed RV
π_1	Probability of active PU	$= 1 - \pi_o$
σ_o	Noise Standard Deviation	1
σ_1	Received signal Standard Deviation	Varying
r	Distance between SU and PU	Uniformly Distributed RV

Remarks:

- Number of channels a node can sense in one time slot (k) is supposed to be in the range [M/N M]. This assumption is necessary to make sure that all channels are covered.

- The received signal standard deviation (σ_1) which indicates the square root of signal power (signal variance) is set to be varying to cover all the scenarios of the network configurations (secondary nodes positions relative to the PU position). The probability distribution which σ_1 follows to vary can be obtained via the following derivation:

obeikandl.com

SUs' locations are assumed to be uniformly distributed around the PU because in most applications there is no restriction on the position of the SUs with respect to the PU. i.e. $f(r) = constant$, where $f(r)$ indicates the probability density function (pdf) of the distance r between SU and PU.

Received signal power (p) is inversely proportional with the squared distance between SU and PU (r). Therefore, $p = \frac{c}{r^2}$, where, c is the proportionality constant.

$$\begin{aligned} \sigma_1 &= \sqrt{p} \\ &= \frac{\sqrt{c}}{r} \end{aligned} \tag{5.9}$$

$$\therefore f(\sigma_1) = f(r)\, |\frac{dr}{d\sigma_1}| \tag{5.10}$$

From equation 5.9, $|\frac{dr}{d\sigma_1}| = \frac{\sqrt{c}}{\sigma_1^2}$.
Therefore, the probability distribution which σ_1 follows is:

$$f(\sigma_1) = f(r)\frac{\sqrt{c}}{\sigma_1^2} \tag{5.11}$$

5.8 Performance Evaluation

To evaluate the assignment algorithms, three performance metrics are considered, which are:

1. The **global probability of error** of each channel after reaching a global decision about the availability of each. It is plotted *vs* the average SNR of the M channels for both the Hungarian algorithm-based assignment method and the fuzzy-based assignment method. Additionally, It is compared to the *random assignment* case (where nodes are distributed randomly among nodes) and the *full assignment* case (where it is assumed that each node has the time and ability to sense the whole band of interest).

obeikandl.com

2. The **time delay** needed by the network before reaching the global decision. This includes the time of algorithm computations (assumed to be the same for both proposed techniques), the time required for sensing and processing the assigned channels and the time for reporting decisions to the FC. Normalized time delay is plotted *vs* the number of channels that a node can sense during a single time slot (k). Comparison takes place between the presented techniques and the full assignment case.

3. The **overall energy consumption** which is considered a direct measure of the network lifetime. It also includes the energy consumed for computations, sensing, processing and reporting channels. Similar to the time delay, normalized energy is plotted *vs* the number of channels that a node can sense during a single time slot (k). Comparison takes place between the presented techniques and the full assignment case.

Fig. 5.7 shows the global probability of sensing error for both presented algorithms in comparison to random assignment and full assignment. The presented algorithms almost give similar performance with a slight enhancement in favor of the Hungarian-Algorithm based technique. They both outperform the performance of the random and full assignment.

From the perspective of time delay and energy consumption, it is required to know how much enhancement do the presented systems offer rather than the full assignment (assuming that the full assignment can be possible). Fig. 5.8 and Fig. 5.9 show that the presented systems (if assumed to consume same computation time and energy) outperform the full assignment case. Note that in Fig. 5.8 and Fig. 5.9:

- Time delay of the algorithm-based assignment is increasing as k increases. This can be explained by the fact that as k increases, the time required to sense, process and report one group increases until it reaches $k = M$ (Full assignment).

- Time delay of the full assignment decreases as k increases. Although the time required to sense one group increases, but the rate of increase is much lower than the rate at

obeikandi.com

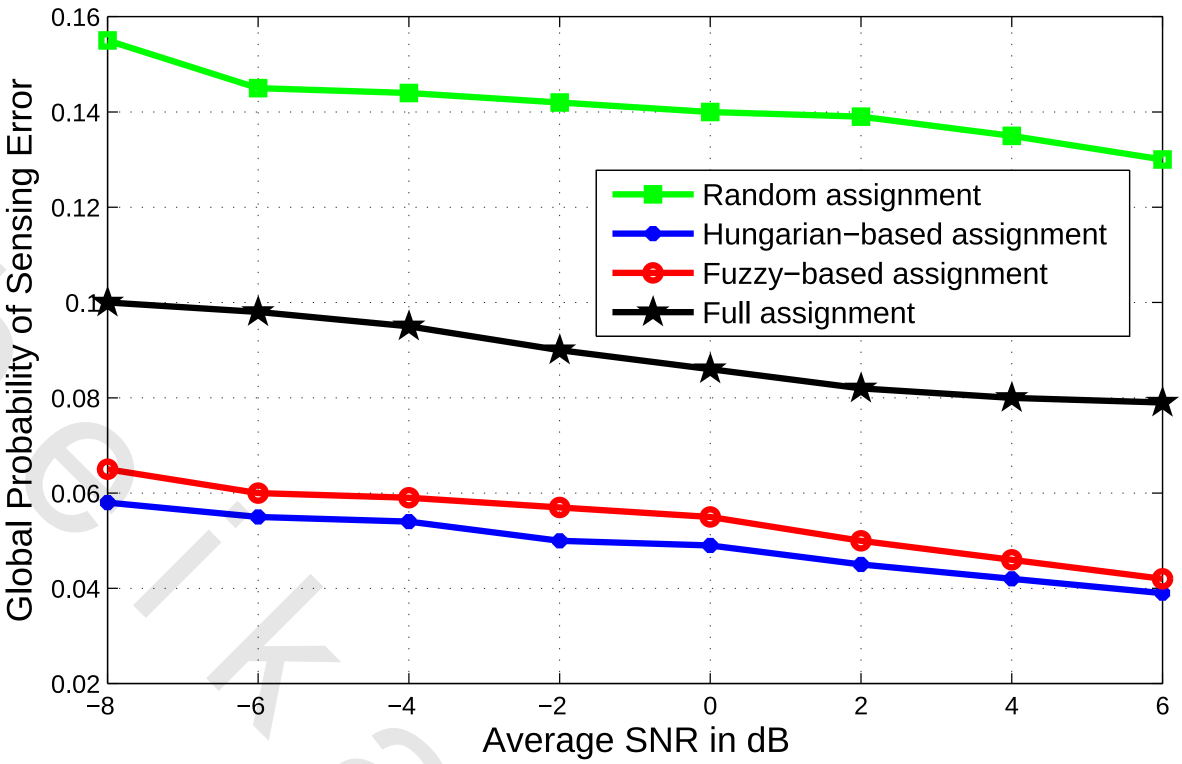

FIGURE 5.7: Global Probability of Sensing Error

which the number of groups $\frac{M}{k}$ decreases. i.e. The time required to finish all groups decreases and the total time delay decreases as well.

- The energy consumed in case of algorithm-based assignment increases as k increases. This is due to the fact that the energy required to sense, process and report one group increases until it reaches $k = M$ (Full assignment).

- Similarly, the energy consumed in case of full-assignment increases as well.

5.9 Conclusion

This chapter presents two assignment scheme for cooperative spectrum sensing in cognitive radio networks. It is required to reduce the sensing overhead and time delay that result from

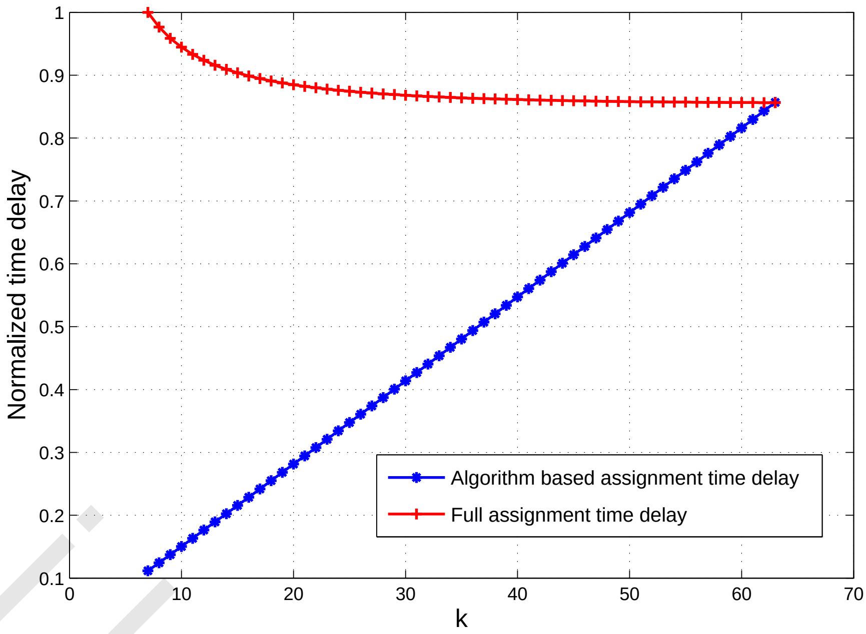

FIGURE 5.8: Normalized Time Delay

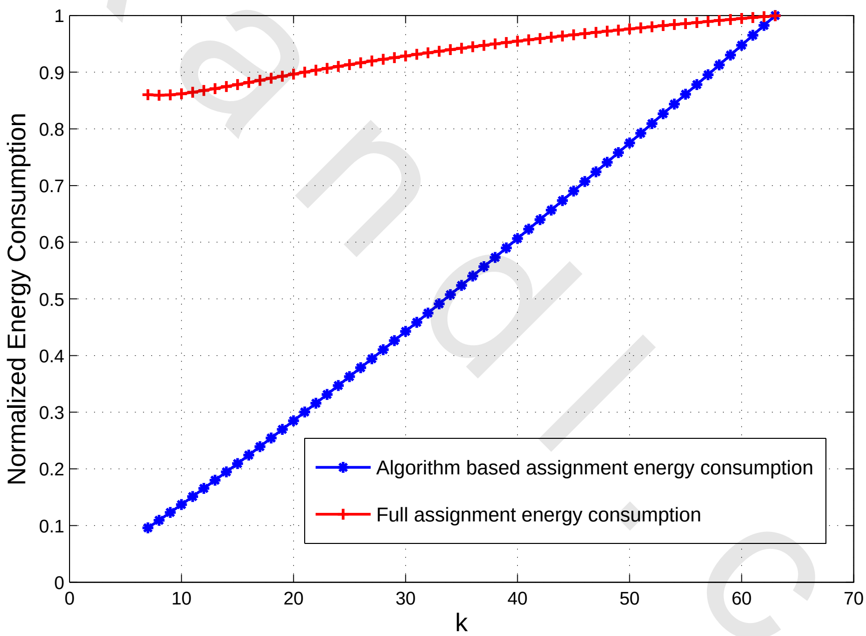

FIGURE 5.9: Normalized Energy Consumption

sensing a wide banded spectrum by distributing the sensing task among nodes of a CRN such that the global probability of sensing error is minimized. The first scheme applies the Hungarian Algorithm on a task-agent assignment problem to select the best candidate nodes to perform the sensing task. The second scheme is a fuzzy-based approach which depends on the calculated sensing error of each node over every channel and the SNR of the reporting

obeikandl.com

subchannels. MATLAB simulation results have shown that the performances of the proposed algorithms outperform the random assignment case and the full assignment case with respect to global probability of sensing error in addition to time and energy consumed reduction.

obeikandl.com