Chapter 2
Cognitive Radio Network- A Review

As discussed in Chap. 1, the quality of service and energy efficiency of a cognitive radio system depends upon various parameters, including spectrum selection, media access scheme and spectrum sensing order. In this chapter, we provide a review on technologies which facilitates/improves the parameters responsible for QoS and energy management. At the end of this chapter, we will discuss about different cognitive radio platforms and their evolution.

2.1 Spectrum Management

Channel selection (spectrum management) forms an integral part of quality of service framework for cognitive radio network. Since different channels have different local policy restriction, their usage offer different grade of services depending upon the operating frequency [1]. Channel usage behavior of incumbents can be characterized as deterministic or stochastic, leading a secondary user to utilize different etiquettes for efficient spectrum utilization.

2.1.1 Ant Colony Optimization Based Spectrum Management

A channel selection technique is discussed in [2] which enables each secondary user (SU) to detect appropriate channels over which it can communicate without any coordination with the other SUs. The overall aim is to increase the spectrum utilization in the system. The authors discussed a biologically-inspired Spectrum Sharing (BIOSS) algorithm based on the adaptive task allocation model of insect colonies. Here the problem of channel allocation is reduced to ant colony optimization. Table 2.1 shows the mapping between CRN and insect colony.

An ant performs efficiently if the suited tasks are allocated to them individually and are allowed to simultaneously complete the task with other ants. Thus the col-

V. Mishra et al., *QoS and Energy Management in Cognitive Radio Network*,
Signals and Communication Technology, DOI 10.1007/978-3-319-45860-1_2

Table 2.1 A mapping between ant colony and cognitive radio

Insect colony	Cognitive radio network
Insect	Cognitive radio
Task	Available channel
Task associated stimuli (s)	Permissible power to channel (P_j)
Response threshold (θ)	Required transmission power (p_{ij})

Table 2.2 A mapping between ant colony and cognitive radio

Condition	Events
$P_j \gg p_{ij}$	Probability of cognitive radio i to select channel j increases
$P_j \ll p_{ij}$	Probability of cognitive radio i to select channel j decreases
For constant p_{ij}, if P_j increases	Probability of cognitive radio i to select channel j increases
For constant P_j, if p_{ij} increases	Probability of cognitive radio i to select channel j decreases

laborative behavior of the ants optimizes the overall job of ant colony. This division if labour is based on the response threshold θ of each ant, which is the probability of reacting to a task associated stimuli(s); whereas task stimuli(s) is the intensity of activator associated with the task. The task is performed iff $s \gg \theta$.

The probability of performing a task is given by:

$$T_\theta(s) = \frac{s^n}{s^n + \theta^n} \tag{2.1}$$

Here, $n > 1$ determines steepness of the threshold.

Each cognitive radio i in the system calculates the channel selection probability for every available channel c as:

$$T_{ij}^{csp} = \frac{P_j^n}{P_j^n + \alpha p_{ij}^n + \beta L_{ij}^n} \tag{2.2}$$

where, L_{ij}: learning factor to enable CR i to learn or forget channel j and α, β: positive constants.

The channels are selected on the basis of conditions as shown in Table 2.2.

The spectrum selection is done by Algorithm 2.1 which executes over each cognitive radio node to provide spectrum selection based on QoS requirements.

Algorithm 2.1: Algorithm for load balancing in a given category of channels

```
Determine available spectrum bands;
Estimate P_j values for every channel j;
Set learning factor (L_ij) value to be same for each channel and learning coefficients (ξ_0, ξ_1)
  to appropriate values [2].
Computes T^csp for all available channels.
Selects channels with max T^csp.
if Selected channel meets QoS requirements then
  | L_ij = L_ij − ξ_0 if selected channel meets QoS requirements.
else
  | L_ij = L_ij + ξ_1 if selected channel does not meet QoS requirements.
end
```

Some of the observation worth to note is that, as the interference increase, P_j for a channel decrease. The channel with small P_j due to the interference can be preferred by the cognitive radio node which needs low transmission power. This will enable the cognitive radio node to control the interference in the channels. Also, the spectrum sharing becomes more efficient as the power demand of CR increases which inturn would prefers the channel with more transmission power.

2.1.2 Non-linear Optimization Based Spectrum Management

In [3], the authors have considered a spectrum assignment problem for cognitive radio sensor network via a Mixed Integer Non-linear Programming problem formulation. The spectrum assignment problem is considered under coverage, interference, minimum data rate and power budget constraints. The problem is transformed to Binary Linear Programming problem and further reduced to random, greedy and two-stage (decoupled) algorithms.

The network model comprises of a set of N cognitive radio-enabled wireless sensor nodes distributed within a set C of cells and coexisting with PUs operating in an up-link network on the same band, and transmitting on each channel with power P_{PU}. Both systems operate on a band comprising a set of K channels, each with bandwidth $B_k, \forall k \in K$. Only one node is $active$ to report the measured data while others are in $sleep$ mode.

$|h^k_{n,SU}|^2$ is the channel gain between node n and the cognitive radio base station receiver on channel k. $|h^k_{PU}|^2$ is the the channel gain between the primary user transmitter and the secondary user base station receiver on channel k. N_o is the power spectral density of a single sided additive white Gaussian noise. S is the set of active nodes.

Each of the *active sensors* from different cells is assigned channels in order to minimize transmit power with the following constraints:

(i) Interference at primary network receivers is less than given threshold. That is, $P_n^k|h_{n,PU}^k|^2 \leq I_{th}^k, \forall k \in K, \forall n \in N$, where P_n^k is the transmission power of node n on channel k, $|h_{n,PU}^k|^2$ is the channel gain between node n and the primary user base station receiver on channel k, and I_{th}^k is the interference threshold at PU on channel k.

(ii) Mutual exclusion of channels among cognitive radio sensor nodes (A channel k is used by at most one node). That is, $\sum_{n=1}^{N} Y_n^k \leq 1, \forall k \in K$, where Y_n^k is a binary variable identifying whether a node n is assigned channel k.

(iii) Transmission power is limited by a power budget. That is, $\sum_{k=1}^{K} Y_n^k P_n^k \leq P_n^{max}, \forall n \in N$, where P_n^{max} is the maximum allowed power of node n.

(iv) Secondary network must achieve a minimum data rate (for QoS). That is,

$$\sum_{k=1}^{K} Y_n^k.B_k.log_2(1 + P_n^k * \gamma_n^k) \geq R_{th} \tag{2.3}$$

where γ_n^k is the channel to interference noise ratio of node n on channel k, and R_{th} is the minimum rate required required at the cognitive radio base station receiver.

(v) A coverage constraint where the geographical area is divided into multiple cells, each covers a certain event, and each cell must be covered by one active sensor node. That is, $\sum_{n=1}^{N} A_n^c Y_n = 1, \forall c \in C$, where A_n^c is a binary variable identifying whether a cognitive radio sensor node n is assigned a channel k, and $Y_n = \sum_{k=1}^{K} Y_n^k, \forall n \in N$, indicates whether a node is active or not.

The problem of choosing *active* sensors and assigning channels to these sensors can be formulated as:

$$\begin{aligned}
&\min_{Y_n^k, P_n^k} \sum_{n=1}^{N}\sum_{k=1}^{K} Y_n^k P_n^k \\
\textbf{subject to: } & P_n^k|h_{n,PU}^k|^2 \leq I_{th}^k, \forall k \in K, \forall n \in N, \\
& \sum_{n=1}^{N} Y_n^k \leq 1, \forall k \in K, \\
& \sum_{k=1}^{K} Y_n^k P_n^k \leq P_n^{max}, \forall n \in N, \\
& \sum_{k=1}^{K} Y_n^k.B_k.log_2(1 + P_n^k * \gamma_n^k) \geq R_{th}, \forall n \in N, \\
& \sum_{n=1}^{N} A_n^c Y_n = 1, \forall c \in C
\end{aligned} \tag{2.4}$$

where,

$$A_n^c = \begin{cases} 1, & \text{if sensor } n \text{ cover area } c, \\ 0, & \text{otherwise} \end{cases}$$

$$Y_n^k = \begin{cases} 1, & \text{if sensor } n \text{ is assigned channel } k, \\ 0, & \text{otherwise} \end{cases}$$

$$\gamma_n^k = \frac{|h_{n,SU}^k|^2}{N_o * B + P_{PU} * |h_{PU}^k|^2}$$

The problem in Eq. 2.4 being NP-hard cannot be solved in polynomial time. Since cognitive radio sensor nodes are resource constrained, a realistic assumption regarding single transceiver on each node can be made. This would reduce the above multi-integer linear program to binary linear program as shown in Eq. 2.5.

$$\begin{aligned} & \min_{Y_n^k, P_n^k} \sum_{n=1}^{N} \sum_{k=1}^{K} Y_n^k P_n^k \\ \textbf{subject to: } & P_n^k |h_{n,PU}^k|^2 \le I_{th}^k, \forall k \in K, \forall n \in N, \\ & \sum_{n=1}^{N} Y_n^k \le 1, \forall k \in K, \\ & P_n^k \le P_n^{max}, \forall k \in K, \forall n \in N, \\ & B_k.log_2(1 + P_n^k * \gamma_n^k) \ge R_{th}, \forall k \in K, \forall n \in N, \\ & \sum_{n=1}^{N} A_n^c Y_n = 1, \forall c \in C \end{aligned} \tag{2.5}$$

Algorithm 2.2: Algorithm for calculation of P_n^k

1 Calculate γ_n^k, $|h_{n,PU}^k|^2, \forall k \in K, \forall n \in N$;
2 Calculate P_n^k:
3 **for** $n = 1$ **to** N **do**
4   **for** $k = 1$ **to** K **do**
5         $P_n^k = P_n^k|_{min} = \frac{2^{R_{th}/B_k} - 1}{\gamma_n^k}$;
6   **if** $P_n^k|_{min} > min\left(\frac{I_{th}^k}{|h_{n,PU}^k|^2}\right)$ **then**
7             $P_n^k \Leftarrow \infty$;
8   **end**
9   **end**
10 **end**

Using Algorithm 2.2, the transmission power of each node n over all channels available in the network can be calculated. This can be utilized to generate a matrix Q, which provides transmission power of cognitive radio nodes in each cell. This can be given as $Q = P^k_{n_c}, \forall n \in N, \forall c \in C, \forall k \in K$ and can be calculated as given in [3]. The minimization problem given in Eq. 2.5 is reduced to a simpler minimization problem (Eq. 2.6) by utilizing the matrix Q.

For given $P^k_n, \forall n, \forall k$

$$\begin{aligned} &\min_{Y^k_n} \sum_{n=1}^{N} \sum_{k=1}^{K} Y^k_n P^k_n \\ \textbf{subject to:} \; &\sum_{n=1}^{N} Y^k_n \leq 1, \forall k \in K \\ &\sum_{n=1}^{N} A^c_n * Y_n = 1, \forall c \in C \end{aligned} \tag{2.6}$$

The discussed spectrum selection scheme utilizes following algorithms to solve the optimization problem in Eq. 2.6.

2.1.2.1 Minimum Power Algorithm

The aim of the Minimum power algorithm is to jointly optimize the selection of nodes and channels in such a way that the total transmission power is minimized. The transmission powers P^k_n from Algorithm 2.2 is utilized in Eq. 2.6 as shown in Algorithm 2.3.

Algorithm 2.3: Minimum Power Algorithm

1 Calculate $\gamma^k_n, |h^k_{n,PU}|^2, \forall k \in K, \forall n \in N$
2 Calculate P^k_n from Algorithm 2.2;
3 Calculate matrix Q from P^k_n;
4 **Node/Channel Selection:**
5 Solve Eq. 2.6 with the values of Q;

Assuming total number of bits in a binary representation for each of the optimal variables to be 1, the overall complexity of this algorithm is $O((NK)^{3.5})$.

2.1.2.2 Random Algorithm

The random algorithm choses a random node out of each cell and then assigns channels to it randomly. This can be achieved by the method discussed in Algorithm 2.4.

Algorithm 2.4: Random Algorithm

```
Calculate γ_n^k, |h_{n,PU}^k|^2, ∀k ∈ K, ∀n ∈ N;
Calculate P_n^k from Algorithm 2.2;
Calculate matrix Q from P_n^k;
Node Selection:
for c = 1 to C do
    S(c) ⇐ select a random node;
end
Channel Selection:
for c = 1 to C do
    S(s)_k ⇐ select a random channel k ∈ K;
    K ⇐ K − k;
end
```

The complexity of this algorithm is given as $O(1)$ for random selection of node/channel.

2.1.2.3 Greedy Algorithm

The greedy algorithm performs search over the matrix of P_n^k obtained from Algorithm 2.2 to find the node/channel pairs that has minimum transmission power. This can be achieved via the following Algorithm 2.5.

Algorithm 2.5: Greedy Algorithm

```
Calculate γ_n^k, |h_{n,PU}^k|^2, ∀k ∈ K, ∀n ∈ N;
Calculate P_n^k from Algorithm 2.2;
Calculate matrix Q from P_n^k;
Node/Channel Selection:
for c = 1 to C do
    S(c)_k = min(P_n^k), ∀n ∈ N∀k ∈ K;
    Q ⇐ Q − Q(:, k);
    Q ⇐ Q − Q(c, :);
    K ⇐ K − k;
end
```

The total complexity of this greedy algorithm is of $O(C(NK)^2)$.

2.1.2.4 Two-Stage (Decoupled) Algorithm

In this method, the problem of spectrum allocation is broken into two sub-problems: *node-selection* and *channel-assignment*. In node-selection problem, life of node is extended by selecting nodes with higher residual energy as cells, while channel-assignment problem is solved as a standard linear program.

Algorithm 2.6: Two-Stage Algorithm

Calculate γ_n^k, $|h_{n,PU}^k|^2$, $\forall k \in K, \forall n \in N$;
Calculate P_n^k from Algorithm 2.2;
Calculate matrix Q from P_n^k;
Node Selection:
for $c = 1$ **to** C **do**
  $S(c) = max(n_{E_{res}}), \forall n \in N_c$;
end
Channel Selection:
Solve Eq. 2.6 with Q_s as input.

The overall complexity of this two-stage scheme is $O((CN^2) + (CK)^{3.5})$.

2.1.3 Game Theory Based Spectrum Management

A solution to address the problem of efficient spectrum pricing in cognitive radio network is given in [4]. The inefficiency of the Nash equilibrium (equilibrium pricing adopted by primary service providers) is discussed, in the sense that the profit of the primary service providers is not maximized. Here an alternate solution to this problem is discussed.

The problem is modelled as oligopoly market, in which a small number of firms compete with each other non-cooperatively and independently to achieve their objectives while dominating the overall market. The objective is determined in terms of controlling the price of products/quantity, and such actions are influenced by the decisions of other firms too. To achieve an equilibrium pricing scheme Bertrand game model may be utilized. Similarly, pricing for primary users' spectrum can also be calculated in a similar way, which are willing to share with secondary users.

A wireless system with total number of primary services N is assumed in which all primary services operate over different frequency spectrum $F_i, \forall i \in N$. Each of the primary service here wants to sell a portion of the available spectrum F_i to secondary users for price p_i as shown in Fig. 2.1.

An adaptive modulation is utilized to dynamically adjust the transmission rate. The spectral efficiency k of transmission by a secondary user is given by Eq. 2.7.

$$k = log_2(1 + K\gamma), \text{ where } K = \frac{1.5}{ln\left(\frac{0.2}{BER^{tar}}\right)} \tag{2.7}$$

where γ is the SNR at the receiver and BER^{tar} is the target bit-error-rate.

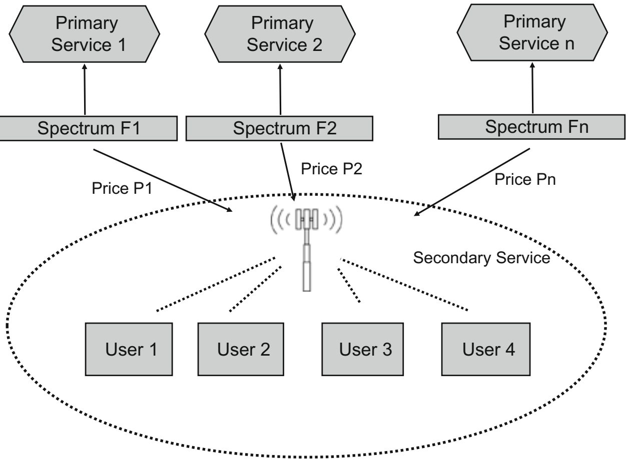

Fig. 2.1 Spectrum sharing model

Utility of the shared frequency spectrum is used to calculate the spectrum demand of the secondary users. A quadratic utility function is used to quantify the demand or utility of the spectrum.

$$U(b) = \sum_{i=1}^{N} b_i k_i^{(s)} - \frac{1}{2}\left(\sum_{i=1}^{N} b_i^2 + 2\nu \sum_{i \neq j} b_i b_j\right) - \sum_{i=1}^{N} p_i b_i \tag{2.8}$$

where b is the set consisting of the size of shared spectrum from all the primary services, i.e., $b = b_1, \ldots, b_N$; p_i is the price offered by primary service i, $k_i^{(s)}$ denotes the spectral efficiency of wireless transmission by a secondary user using primary users' frequency spectrum F_i. ν is the spectrum substitutability factor $\nu \in [-1.0, 1.0]$.

The quadratic utility function is used because of the following reasons:

(i) Quadratic function being concave, is able to represent the saturation of user satisfaction as the transmission rate increases.
(ii) A linear bandwidth demand function can be derived via differentiating the quadratic utility function. This eases the subsequent analysis for stability condition.
(iii) The spectrum substitutability factor ν, control the switching of secondary users onto its channels. If $\nu = 1$, a secondary user can switch freely among operating frequency spectra; while if $\nu = 0$, a secondary user can not switch freely.

The demand function for spectrum F_i is given by differentiating $U(b)$ with respect to b_i as shown in Eq. 2.9.

$$\frac{\partial U(b)}{\partial b_i} = 0 = k_i^{(s)} - b_i - v\sum_{i \neq j} b_j - p_i \quad (2.9)$$

The spectrum demand function is obtained from Eq. 2.9 by substituting the value of the prices of the primary service as shown in Eq. 2.10.

$$D_i(\mathbf{p}) = \frac{(k_i^{(s)} - p_i)(v(N-2)+1) - v\sum_{i \neq j}(k_j^{(s)} - p_j)}{(1-v)(v(N-1)+1)} \quad (2.10)$$

Price charged by the primary users can also be used to calculate the spectrum demand of the secondary users. The cost of the primary service in sharing the spectrum is computed from the degradation of the QoS for the primary service. The cost is only incurred if the required bandwidths for the primary connections cannot be provided due to bandwidth sharing with secondary services. For primary service i, the revenue function R_i and the cost function C_i, can be defined as follows:

$$\begin{aligned} R_i &= c_1 M_i \\ C_i(b_i) &= c_2 M_i \left(B_i^{req} - k_i^{(p)} \frac{W_i - b_i}{M_i} \right)^2 \end{aligned} \quad (2.11)$$

Here c_1 and c_2 denote the constant weight for revenue and cost functions. M_i is the number of primary connections, B_i^{req} is the bandwidth requirement and W_i is the size of spectrum of the the primary service i.

In the above given oligopoly market model, Bertrand game model can be utilized to formulate the game. Primary services act as the *players*, modifying the price per unit of the spectrum (p_i) act as the *strategy*, while the profit of sharing the spectrum with secondary services act as the *payoff*. Therefore the profit of each primary service can be calculated as shown in Eq. 2.12.

$$P_i(\mathbf{p}) = b_i p_i + R_i - C_i(b_i) \quad (2.12)$$

where $\mathbf{p}$ denotes the set of prices offered by all players in the game, that is $\mathbf{p} = \{p_1, p_2, p_3, \ldots, p_N\}$.

The best response function (best strategy of one player for given strategies of other players) of primary service i, given a set of prices offered by the other primary services $\mathbf{p}_{-i}$ is defined as follows:

$$B_i(\mathbf{p}_{-i}) = arg\max_{p_i} P_i(\mathbf{p}_{-i} \cup \{p_i\}) \quad (2.13)$$

An equilibrium is reached iff

$$p_i^* = B_i(\mathbf{p}_{-i}^*), \forall i \quad (2.14)$$

where $\mathbf{p}^* = \{p_1^*, p_2^*, \ldots, p_N^*\}$ is the set of prices when Nash equilibrium will be reached. $\mathbf{p}_{-i}$ is the set of best responses for player j such that $j \neq i$.

From Eqs. 2.12, 2.10 and 2.11, the profit function can be derived as shown in Eq. 2.15.

$$P_i(\mathbf{p}) = p_i D_i(\mathbf{p}) + c_1 M_i - c_2 M_i \left(B_i^{req} - k_i^{(p)} \frac{W_i - D_i(\mathbf{p})}{M_i} \right)^2 \tag{2.15}$$

The demand function in Eq. 2.10 can be expressed as: $D_i(\mathbf{p}) = D_1(\mathbf{p}_{-i}) - D_2 p_i$. Where $D_1(\mathbf{p}_{-i})$ and D_2 are given by Eq. 2.16.

$$\begin{aligned} D_1(\mathbf{p}_{-i}) &= \frac{k_i^{(s)}(v(N-2)+1) - v \sum_{i \neq j}(k_j^{(s)} - p_j)}{(1-v)(v(N-1)+1)} \\ D_2 &= \frac{(v(N-2)+1)}{(1-v)(v(N-1)+1)} \end{aligned} \tag{2.16}$$

For a Nash equilibrium to occur,

$$\frac{\partial P_i(p)}{\delta p_i} = 0, \tag{2.17}$$

From Eq. 2.15, 2.16 and 2.17, the Linear Eq. 2.18 can be obtained.

$$0 = 2c_2 k_i^{(p)} D_2 \left(B_i^{req} - k_i^{(p)} \frac{W_i - (D_1(\mathbf{p}_{-i}) - D_2 p_i)}{M_i} \right) + D_1(\mathbf{p}_{-i}) - 2D_2 p_i \tag{2.18}$$

For given parameters, the Eq. 2.18 provides the solution p_i^* which provides Nash equilibrium. From this the size of the shared spectrum can be calculated from Eq. 2.10 (i.e. $D_i(\mathbf{p}^*)$).

In reality, primary services may not be able to observe the profit gained by other primary services. In such scenario, for a Nash equilibrium to occur, a distributed form of price adjustment is required. Let the price offered by primary service i at iteration t be given as $p_i[t]$. If the strategies of the primary services in the previous iteration are observable by each other, then $p_i[t]$ is given as shown in Eq. 2.19.

$$p_i[t+1] = B_i(\mathbf{p}_{-i}[t]), \forall i \tag{2.19}$$

If the strategies used by other primary services are not known, then the relationship between the strategies in the current and the future iteration is to maximize the profit, which can be expressed as follows:

$$p_i[t+1] = p_i[t] + \alpha_i \left(\frac{\partial P_i(p)}{\partial p_i} \right) \tag{2.20}$$

where α_i is the learning rate.

The marginal profit of a primary service can be observed by the marginal spectrum demand for small variation in price ε as shown in Eqs. 2.21 and 2.22.

$$\frac{\partial P_i(\mathbf{p})}{\partial p_i} \approx \frac{P_i(\mathbf{p}_{-i}[t] \cup \{p_i[t] + \varepsilon\}) - P_i(\mathbf{p}_{-i}[t] \cup \{p_i[t] - \varepsilon\})}{2\varepsilon} \tag{2.21}$$

$$\begin{aligned} P_i(\mathbf{p}_{-i}[t] \cup \{p_i[t] \pm \varepsilon\}) &= p_i D_i(\mathbf{p}_{-i}[t] \cup \{p_i[t] \pm \varepsilon\}) + c_1 M_i \\ &\quad - c_2 M_i \left(B_i^{req} - k_i^{(p)} \frac{W_i - D_i(\mathbf{p}_{-i}[t] \cup \{p_i[t] \pm \varepsilon\})}{M_i} \right)^2 \end{aligned} \tag{2.22}$$

To check if the above developed dynamic game will reach Nash equilibrium at the steady state, stability of the game is evaluated. Stability can be evaluated by utilizing the eigenvalues of the Jacobian matrix of a self-mapping function. The self mapping function is stable if the eigenvalues λ_i are inside the complex unit circle. Here the self mapping function utilized is from Eq. 2.20. Therefore the Jacobian matrix is defined as:

$$J = \begin{bmatrix} \frac{\partial p_1[t+1]}{\partial p_1[t]} & \frac{\partial p_1[t+1]}{\partial p_2[t]} & \cdots & \frac{\partial p_1[t+1]}{\partial p_N[t]} \\ \frac{\partial p_2[t+1]}{\partial p_1[t]} & \frac{\partial p_2[t+1]}{\partial p_2[t]} & \cdots & \frac{\partial p_2[t+1]}{\partial p_N[t]} \\ \vdots & \vdots & & \vdots \\ \frac{\partial p_N[t+1]}{\partial p_1[t]} & \frac{\partial p_N[t+1]}{\partial p_2[t]} & \cdots & \frac{\partial p_N[t+1]}{\partial p_N[t]} \end{bmatrix}$$

It can be observed that, if the strategies of other primary services are observable, then there is no control parameter to stabilize the price adaptation. If the strategies of other primary services are not observable, then the control parameter depends upon the learning rate.

For a repeated game, a trigger strategy allows a primary service to maintain collusion as long as other services agree to do so. In the case that the collusion is maintained forever, the long-term profit of primary service i can be expressed as follows:

$$P_i^0 + \delta_i P_i^0 + \delta_i^2 P_i^0 + \cdots = \frac{1}{1 - \delta_i} P_i^0$$

where P_i^0 is the profit due to the optimal price, and δ_i is the weight ($0 \leq \delta_i \leq 1$) which multiplies the profit in next stage.

For a game where one primary service deviates from the optimal price, it will experience the profit at the Nash equilibrium during the rest of the stages. The long term profit in this case can be expressed as:

$$P_i^d + \delta_i P_i^n + \delta_i^2 P_i^n + \cdots = P_i^d + \frac{\delta_i}{1 - \delta_i} P_i^n$$

where P_i^d and P_i^n are the profit due to price deviation and profit due to price at the Nash equilibrium respectively.

Collusion will be maintained if:

$$\frac{1}{1-\delta_i} P_i^0 \geq P_i^d + \frac{\delta_i}{1-\delta_i} P_i^n$$

2.1.4 Learning Automata Based Spectrum Selection

A spectrum selection scheme is discussed in [5], which minimizes the overall switching of the spectrum done by secondary users. This is done so as to mitigate the problem of aggravated delay and deteriorated packet-loss ratio which occurs due to constant channel switching of sporadically varying channel in cognitive radio environment. This is further complicated by the governing guarantee on maximum interference that can be caused to primary users by secondary users. Therefore a secondary user has to vacate a channel within a given time threshold whenever it detects the presence of primary users.

To minimize channel switching, the discussed stochastic channel selection algorithm utilizes learning automata which maximizes the number of successful transmissions. The algorithm asymptotically converges to an optimal channel which has higher availability in comparison to other channels.

In the given system, a secondary user can tolerate interference up to a threshold value; once the threshold value is reached, the secondary user has to leave the system. Whenever the channel is selected the secondary user transmits over consequent time slots, until the channel ceases to be free, Thus, a transmission is successful, if:

(i) The primary user is absent throughout the time slot.
(ii) Any of the neighbouring secondary user (the secondary users whose actions may potentially cause a transmission failure of the given secondary user) isn't operating at the same frequency.

The cognitive radio system is modelled as learning automata based control system, where the channel availability is modelled as random environment while the finite actions are modelled as the spectrum decisions. Each of these deterministic finite action will induce a random output from the random environment, which could be either favorable or unfavorable (i.e. whether selected channel has high availability or not). A learning automata algorithm can be utilized over this output to provide a control over the decision making of the finite actions. The learning automata algorithm essentially utilizes past results to make decisions about the next action.

During initialization of the secondary network, a secondary user selects any available channel at the beginning of each time slot for transmission and the output is denoted as β ($\beta = 0$ for successful and $\beta = 1$ for unsuccessful transmission). The system comprises of M primary users operating over C channels on which secondary users coexist in a hierarchical way. The system is viewed from the perspective of a secondary user denoted by k. The starting time of each slot is denoted by T_i. Each secondary user maintains a probability vector denoted by $\mathbf{P}_k = [p_1, \dots, p_m]$ where p_i is probability of choosing channel i. Each secondary users also maintains an estimation vector denoted by $\mathbf{D}_k = [d_1, \dots, d_m]$, where d_i is the estimated possibility of successful transmission by selecting channel i. $H(n)$ denotes the number of channels having higher values in the estimation vector D than the current selected channel. R and W are the resolution parameter and the initialization parameter respectively. δ represents the step size of adjusting the probability vector where, $\delta = 1/R$. $S_i(n)$ is the number of slots where the transmissions with selected channel i are successful, up to T_n. $C_i(n)$ is the number of slots where the channel i are selected to transmit packets, up to T_n.

The Algorithm 2.7 shows the learning automata based channel selection technique which executes over each node k with B as a predefined convergence threshold.

This channel selection technique is ε-optimal for any cognitive radio networks which is stationary. Therefore for any arbitrarily small $\varepsilon > 0$ and $\gamma > 0$, there exists a n' satisfying:

$$P_r\{|1 - p_z(n)| < \varepsilon\} > 1 - \gamma, \forall n > n'$$

where z is the index of the optimal channel in terms of probability of successful transmissions.

Algorithm 2.7: Algorithm for Stochastic Channel Selection

Initialize:
$P(0) = [p_1, \dots, p_M]$ where $p_i = 1/M$ for all $1 \le i \le M$;
do
  Select available channel from $P(0)$;
  Evaluate and record β ;
while *Until each channel is selected W times*;
Update $S_i(0)$ and $C_i(0)$ according to value of βs for each channel i ;
Initializes $D(0) = [d_1, \dots, d_M]$ where $d_i = S_i(0)/C_i(0)$;
do
  Select an available transmission channel i according to the probability vector $P(n)$;
  Update the probability vector $P(n)\ as$:
  $p_j(n+1) = min\big(p_j(n) + \frac{\delta}{H(n)}, 1\big)\forall j$ if $d_j(n) > d_i(n)$
  $p_j(n+1) = max\big(p_j(n) - \frac{\delta}{M-H(n)}, 0\big)\forall j$ if $d_j(n) < d_i(n)$
  $p_i(n+1) = 1 - \sum_{j \ne i} p_j(n+1)$
  After T_n, the secondary user adjusts the value of D as:
  $S_i(n+1) = S_i(n) + (1 - \beta)$
  $C_i(n+1) = C_i(n) + 1$
  $d_i(n+1) = \frac{S_i(n+1)}{C_i(n+1)}$
while *$(max(P(n)) > B)$*;

2.1.5 Spectrum Selection in Varying Channel Bandwidth Environment

In [6], a scheme for cognitive radio spectrum hand-off is discussed where spectrum switching is done between channels of varying bandwidth. The concept of delay bandwidth product (DBP) is utilized to prioritize the channels with varying bandwidth. Here the delay bandwidth product is defined as the difference between the maximum tolerable delay threshold of the secondary user and the average occupation time of the primary user over a given channel. Based on this a secondary user can make spectrum selection to reduce power, increase range, increase throughput, load balance and guarantee its QoS requirement.

Theoretically delay bandwidth product refers to product of a data link's capacity and its round-trip delay time. Intuitively, this indicates the maximum amount of data that can exist in a network at any given time. Here the delay is considered as the difference between maximum tolerable delay of a secondary user and average occupation time of a primary user. As shown in Fig. 2.2, the total delay time (D_i) is the elapsed time until the secondary user gets the opportunity to transmit again. The total delay time is dependent on sensing time W_j, handoff time t_0 and the transmission time of secondary user T_k, where k is the current selected channel and j is candidate sensed channel. A secondary user can transmit again on the same selected channel or it can switch to another channel depending upon the DBP index.

The system comprises of N variable bandwidth channels each with bandwidth $B_i, \forall i \in (1, 2, \ldots, N)$. The current channel is defined as k while the candidate channel is defined as j. The total delay time D_i of secondary user i can be expressed as shown in Eq. 2.23.

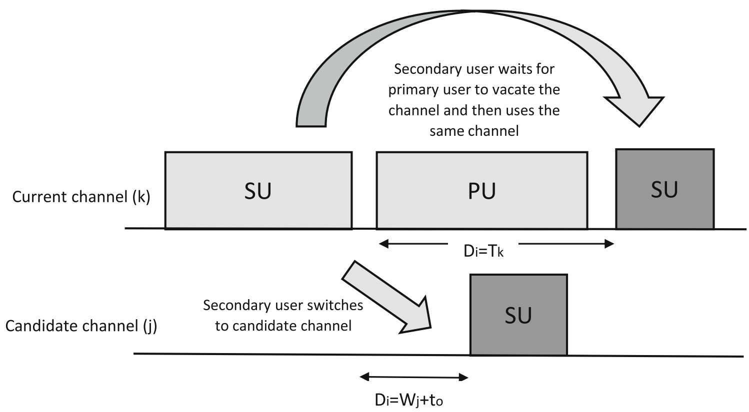

Fig. 2.2 Channel selection by secondary user

$$D_i = \begin{cases} t_0 + W_j, & \text{if } 1 \leq j \leq N, j \neq k; \\ T_k, & j = k \end{cases} \tag{2.23}$$

If the sensing time is less, the transmission time increases. Also for a higher bandwidth channel, there is an increase in throughput. Therefore an optimization is required between the bandwidth of the channel and the effective delay required by the channel itself. The values of T_k and W_j are dependent on what kind of traffic model is utilized in the system.

Two different primary user traffic models are considered for channel availability: *Pareto distribution model* and *Two-state Markov chain model*.

1. Pareto distribution model:
The probability density function (f(x)) and cumulative distribution function (F(x)) are described as shown in Eq. 2.24.

$$\begin{aligned} f(x) &= \frac{\lambda K^{\lambda}}{x^{\lambda+1}}, x \geq K \\ F(x) &= \begin{cases} 1 - (K/x)^{\lambda}, & x > 0 \\ 0, & \text{otherwise} \end{cases} \end{aligned} \tag{2.24}$$

where λ is the scale parameter and K is the shape parameter, $\lambda > 0$, $K > 0$.

2. Two-state Markov chain:
A two-state Markov chain model is utilized to model the primary user traffic behavior. Here P_I and P_B represent channel transition probability to being available and busy respectively. $1 - P_I$ and $1 - P_B$ represent transition probability from state of availability to unavailability and vice versa.

The steady state probability of channel i to be busy by sending primary user's traffic is given by:

$$\Pi_B(i) = p_B(1 - p_B)$$

The average sensing time of a SU W_i is given by:

$$W_i = \sum_{L=1}^{T_{Threshold}} LP(T_s = L) \tag{2.25}$$

where, $T_{Threshold}$ is the number of slot of secondary user's maximum channel sensing tolerance and T_s is the number of slots spend by secondary user for sensing the available channel.

A control parameter C_i is used to increase or decrease DBP index according to the channel conditions and the channel bandwidth ratio in reference to other channels' bandwidth. It is also used to track the difference between the data rates of different channels. C_i value starts from one for all channels and updates as primary user appears on channel i. The successful transmission probability P_i is calculated as the percentage of successful transmitted slots to the total number of transmitted slots on channel i. The short term updates of data rate (Ri) of channel i can be given as:

$$R_i = \begin{cases} R_i(1-\alpha), & \text{if } P_i \leq 0.9; \\ R_i(1+\alpha), & \text{if } P_i > 0.9; \end{cases}$$

where i varies from 1 to N. α is the rate smoothing parameter and its value is 0.001. R_i increases with increase in successful transmission probability $P_i > 0.9$. Similarly, R_i decreases for transmission probability $P_i \leq 0.9$ which indicates bad channel condition. The long term updating of C_i is done every 50 slots according to the following rule:

$$C_i = \begin{cases} C_i - \Delta C, & \text{if} \left[\frac{R_i}{R^*} - \frac{1}{N} \sum_{j=1}^{N} \frac{R_j}{R^*} < -\varepsilon \right] \\ C_i + \Delta C, & \text{if} \left[\frac{R_i}{R^*} - \frac{1}{N} \sum_{j=1}^{N} \frac{R_j}{R^*} > \varepsilon \right] \end{cases}$$

where R^* is the target rate, i is the current channel, j is the candidate channel, ε is threshold value, and ΔC is a fixed step parameter.

If T_{max} is the maximum delay allowed by secondary user and T_{avg} is the predetermined average time, then for i varying from 1 to N, the priority index η_i can be expressed as:

$$\eta_i = \begin{cases} (T_{max} - D_i)B_i, & D_i \leq T_{avg} \\ C_i(T_{max} - D_i)B_i, & \text{otherwise} \end{cases}$$

η_i represents the maximum capacity of channel i. This priority index represents the delay bandwidth product based scheme, therefore priority index increases as the DBP increases.

The channel selection when primary user appears on the current selected channel is given as:

$$Channel_i = \max\{\eta_i\}$$

This selected $Channel_i$ becomes the candidate channel k.

The performance of this DBP-based scheme is calculated to determine whether it meets the required service and reliability objectives. The effective data rate R_{eff} can be calculate from Eq. 2.26.

$$R_{eff} = \frac{\sum_{i=1}^{N} ts_i B_i}{t_{total}} \tag{2.26}$$

where ts_i is the successful transmission slot, B_i is the average bandwidth of channel i, while t_{total} is the total transmission time and can be calculated from Eq. 2.27

$$t_{total} = \sum_{i=1}^{N} ts_i + \sum_{i=1}^{N} D_i \tag{2.27}$$

where t_{si} is successful transmission slot and D_i is total delay time of secondary user i.

The total effective data rate will be maximized as long as the cognitive radio node stays over the channel with the highest DBP index.

2.2 Media Access Control

As discussed in Sect. 1.7, a media access scheme (MAC) is required among multiple secondary users who are sharing a channel so as to mitigate *access collision*. Media access control protocols can be broadly classified into two categories: *random* and *time-slotted* media access schemes, each having its pros and cons. Recently a new class of access scheme, known as *hybrid* schemes has been developed with better performance than that of the random and time-slotted access schemes.

2.2.1 QoS Aware Media Access Schemes

A QoS aware MAC protocol for cognitive radio with real time delay sensitive multimedia applications is discussed in [7]. The discussed scheme determine the set of channels for sensing and transmission of data which will satisfy the QoS requirements of the cognitive radio user. The authors also implements a priority based spectrum access scheme at the secondary user level to provide the QoS guarantee.

The protocol is utilized in a single hop cognitive radio network environment where all the secondary users can communicate with each other. The cognitive radio networks comprises of Nc data channel and 1 control channel, which is selected in the unlicensed UWB (ultra wide band) spectrum. A control channel is used to to synchronise and agree on a data channel for transmission and then tune to that data channel for communication. Each secondary user is equipped with a single transceiver and therefore can only sense one channel at a time. Each channel is modelled as an ON/OFF model. Secondary users keeps track of the slowly varying characteristics and usage patterns of the primary users.

As discussed in Sect. 1.5.1, the secondary user senses the channel for an arbitrary time, the arbitrary sensing period (ASP) which includes a basic sensing time to ensure sensing accuracy and a random slot selected from window [0, SW_i]. This is done to mitigate access collisions among all secondary users who are accessing the same channel. If a primary users shows up on the same channel utilized by the cognitive radio during the data transmission phase, the transmission fails and the secondary user has to vacate the channel within a given time threshold. If the secondary user wants to continue transmission of data, the secondary user has to move to some other channel for data communication.

The transmission time of the current frame of user i is given as t_i, then the probability that the current frame is transmitted over the channel k is

$$P_i^k = \frac{\beta_k}{\beta_k + \alpha_k} F_k(t_{AS_i} + t_i) \tag{2.28}$$

where $1/\alpha_k$ and $1/\beta_k$ are the mean idle and busy periods of channel k respectively, t_{AS_i} is the arbitrary sensing period selected by user i, t_i is the transmission time for the current frame, and $F_k(x)$ is the cumulative distribution function of the usage of channel k by the primary users. It is assumed that no more than one primary user will transmit at the same time on the given channel. Therefore, the transmission by a cognitive radio user is successful if no primary user appears during the $t_{AS_i} + t_i$ interval.

Once the transmission probabilities are computed, a channel sensing sequence can be designed. Two different channel sensing policies have been discussed: *greedy* and *ascending*. In the greedy technique, all the channels are sorted in the descending order and the secondary users selects channel in that order. This guarantees maximum probability of successful transmission for each secondary user which will result in increased selection of this channel by secondary users. This increase in selection will ultimately lead to increase in contention which will significantly reduce the throughput. The ascending approach allows different secondary users to select various channels in the order of the available channel probability. For a given available channel probability of channel k for user i (P_i^k), the expected transmission time over channel k is given as $E[T_i^k] = \frac{t_{AS_i}+t_i}{P_i^k}$. The secondary user selects the channels which will satisfy the condition $E[T_i^k] < \tau_i$, where τ_i is the maximum tolerable one hop delay which a secondary user can guarantee for QoS provisioning. Because of this, the delay performance can be guaranteed by the ascending policy.

To improve the QoS, service differentiation is applied in terms of size of sensing window. A cognitive radio user differentiate the traffic based on different sensing window size. A higher priority application with higher QoS constraints will have a shorter sensing period so that they have an higher chance of accessing the data channel when the opportunity appears. (i.e., $SW_{voice} < SW_{video} < SW_{data}$). Multiple levels of QoS provisioning can be defined by defining different sensing window sizes for different traffic types as shown in Table 2.3.

For a secondary user, the transmission fails for the current time slot on channel k if any of the following conditions are true:

(i) The channel is already occupied by a primary user. The probability of which can be given as:

$$P_{occupied}^k = \frac{\alpha_k}{\alpha_k + \beta_k} \tag{2.29}$$

Table 2.3 Sensing windows for different multimedia services

Type	Strict priority	Statistical priority	No priority
Voice	[0, 31]	[0, 31]	[0, 31]
Video	[32, 63]	[0, 63]	[0, 31]
Data	[64, 127]	[0, 127]	[0, 31]

(ii) A primary user occupies the channel in middle of channel sensing. The probability of which can be given as:

$$P_{PU}^{k} = \frac{\beta_k}{\alpha_k + \beta_k} \cdot F_k(t_{AS_i}) \tag{2.30}$$

(iii) The channel gets busy due to another secondary user transmission. This will depend upon the sensing window size. Here for a maximum sensing window size of W and N_k secondary users contending on channel k, the probability that the given secondary user wins the contention is given as:

$$P_r(\text{given secondary user wins contention}) = \sum_{j=1}^{W} \frac{1}{W} \left(\frac{W-j}{W} \right)^{N_k} \tag{2.31}$$

Since every secondary user is contending on the channel k, it implies that there were no primary users' detected on that channel. Therefore the probability that a secondary user's transmission fails due to another secondary user transmission is given as:

$$P_{SU}^{k} = \frac{\beta_k}{\alpha_k + \beta_k} \left(1 - F_k(t_{AS_i})\right) \left(1 - \sum_{j=1}^{W} \frac{1}{W} \left(\frac{W-j}{W} \right)^{N_k} \right) \tag{2.32}$$

Considering all three cases, the probability that the transmission from a secondary user is successful on channel k is given as:

$$P_{success}^{k} = 1 - \left(P_{occupied}^{k} + P_{PU}^{k} + P_{SU}^{k} \right) \tag{2.33}$$

With this, the average time a secondary user occupies a channel k is given as:

$$E[T^k] = P_{success}^{k}(t_{AS_i} + t_i) + (1 - P_{success}^{k}) t_{AS_i} \tag{2.34}$$

A transmission is successful only if no primary user shows up during the total sensing and transmission time of the secondary user. The probability of which can be given as:

$$P_{TS}^{k} = \frac{\beta_k}{\alpha_k + \beta_k} \sum_{j=1}^{W} \frac{1}{W} \left(\frac{W - j}{W} \right)^{N_k} F_k(t_{AS_i} + t_i) \tag{2.35}$$

The probability that a secondary user is able to succeed in transmission in r-th attempt on the channel k is given as:

$$P_{TS}(r) = P_{TS}^{k} \prod_{j=1}^{r-1} (1 - P_{TS}^{k})^{j} \tag{2.36}$$

The average transmission delay of a secondary user over a given channel k is given as:

$$E[T] = \sum_{r=1}^{\infty} E[T^k] P_{TS}(r) \tag{2.37}$$

The discussed media access scheme along with channel sensing scheduling scheme is QoS aware, which provides differentiated services for multimedia and voice. It provides real time service guarantees for traffic with higher priorities. Since it is utilized for single hop network, in multiple hop network, the protocol will suffer from the problem of hidden terminal.

In [8], two media access scheme are discussed for cognitive radio networks which provisions certain level of QoS to voice packets in the network. The two media access scheme are based on *contention* and *time-division*.

The *contention* based medium access protocol is similar to various protocols that have been already discussed. The channel time is partitioned into time slots, and a time slot is further divided into four phases: *sensing, contention, transmission* and *ACK*. The first phase (sensing phase), is used for all the secondary users to sense the activity of the primary users. If the channel is sensed busy, no secondary user contends for that slot. The second phase (contention phase), comprises of a number of minislots. All the secondary users have the same contention window size. If the channel is sensed idle in the sensing phase, a secondary user randomly chooses a backoff timer from the contention window and continues to sense the channel in the contention phase. If the channel is sensed to be continuously idle for a duration of the backoff timer (in units of minislots) in the contention phase, the secondary user will terminate during the contention phase and transmit its packet in the third phase of the time slot (transmission phase); otherwise, it will quit the contention for the current slot. Therefore, for each contention, the secondary user with the smallest backoff timer will win and transmit its packet in the transmission phase of the time slot. If more than one secondary user choose the same smallest backoff timer, it will result in a collision. To determine whether a packet has successfully been transmitted, the receiver sends an acknowledgment (ACK) at the ACK phase of each slot to the sender upon a successful packet reception.

The *time-division* based media access depends on dividing the time into indexed mini slots. Each mini slot is assigned to a secondary user in a deterministic way such that the secondary user can only transmit if the channel is idle from minislot 0 to $i-1$. Therefore a secondary user has to sense the channel from 0 to $i-1$ and can only transmit if the channel is found idle. This scheme also means that the CR users with higher indexed mini slots will have a lower chance of getting a transmission opportunity and hence the indexing is rotated periodically. If the number of users in the CR network are always same, mini slot assignment can be done at initialization. If instead the number of users change dynamically, the protocol can utilize control packets for joining and leaving the network.

A 3-dimensional Markov chain is utilized to do numerical analysis where each state of the Markov chain is represented by (n, t, s), where $s = 0$ is the idle state and $s = 1$ is the busy state. When $n > 0$, t is the queuing delay experience by the queued packet, and when $n = 0$, t is the time required for the arrival of the next voice packet where n indicates the number of packets in the queue.

The probability of success for the contention based protocol in independent channel-state model is derived as:

$$P_s = \theta^{N_p} \sum_{i=0}^{N_s-1} \binom{N_s-1}{i} \rho^i (1-\rho)^{N_s-i-1} \left[\sum_{j=1}^{CW} \frac{1}{CW} \left(\frac{CW-j}{CW} \right)^i \right] \tag{2.38}$$

where θ^{N_p} is the probability that a channel is vacated by individual primary users and ρ is the queue utilization of the secondary user. N_s, is the total number of secondary users and CW is the contention window size of secondary users.

The probability of success for the contention-free (time-division) protocol in independent channel-state model is derived as:

$$P_s = \theta^{N_p} \sum_{i=1}^{N_s} \frac{1}{N_s} (1-\rho)^{i-1} \tag{2.39}$$

The packet drop probability in independent channel-state model is calculated as:

$$P_{drop} = \frac{\sum_{n>0} \pi(n,t)(1-P_s)^{D-t}}{\sum_{n>0} \pi(n,t)} \tag{2.40}$$

where $\pi(n, t)$ is the steady state probability of state (n, t). Then, the capacity for the voice secondary users N_s^* can be obtained by finding the maximum integer value of N_s that satisfies $P_{drop} \leq P_l$.

The method ensures the packet loss if the number of users are within the allowed limits along with the provisioning of certain level of QoS the the voice users. The protocol has very high overhead due to maintaining indices and dynamically changing indices in the time-division based media access scheme. The performance of the network would suffer when nodes leave and rejoins the networks.

In [9], a media access control scheme for multi-hop cognitive radio network is discussed. This MAC scheme aims at increasing the throughput in a distributed cognitive radio environment while utilizing the power saving mode operation for conserving energy.

The media access control utilizes CSMA/CA based channel access with distribution coordination function for distributed multi-channel access. The media access scheme also utilizes a power saving mode (PSM) operation for conserving power. The idea of PSM is to let the nodes enter a low powered dozing state if no communication is required. As discussed in Sect. 1.7.1, time in the system is divided into superframes, where each superframe is divided into $ATIM$ window and $Data$ window. The $ATIM$ window also comprises of a beacon interval. At the start of each beacon interval, any node A which has to send a packet to a node B sends an ad-hoc traffic indication message ($ATIM$) to node B within this window. When B receives the packet, it sends an ATIM-ACK frame and A and B stay awake for the rest of communication duration. Nodes that did not send or receive a $ATIM$ frame enter a doze state till the next beacon period. The $ATIM$ window can be represented as contention window as discussed in Sect. 1.7.1 and shown in Fig. 1.9. Each of the cognitive radio user utilizes a dedicated sensing hardware for sensing purposes. Therefore a cognitive radio user has the ability to perform *out-of-band* sensing. We will discuss about *out-of-band* sensing later in Sect. 3.4.2.

The system utilizes distributed collaborative sensing to detect primary user interference and to avoid the hidden node problem. The system comprises of N secondary users and utilizes $C + 1$ non-overlapping homogeneous channels, each with same bandwidth. A dedicated channel is utilized as common control channel for exchange of control signals. The size of the beacon-period is smaller than the size of the active period of primary users. A cognitive radio node can only perform *out-of-band* sensing on other channels.

Each cognitive radio user maintains the spectral image of the primary user in a vector defined by SIP. It also maintains a secondary channel load in a vector defined by SCL. The SIP provides the local view of the spectrum and has following meaning

(i) ($SIP[c] = 0$) indicates there is no primary user active on channel c.
(ii) ($SIP[c] = 1$) indicates that there is a primary user active on channel c.
(iii) ($SIP[c] = 2$) indicates that the activity of primary user on channel c is uncertain.

The SIP vector is used to determine if the network can user a certain channel for data communication. The SCL vector is used for choosing the communication channel, it contains the expected load of the cognitive radio communication. The spectral opportunity with the lowest SCL is picked when a node wishes to communicate.

During the $ATIM$ window, the nodes participate in the frame synchronization algorithm (for time synchronization), performing channel sensing, retrieving network wide spectral opportunities, and performing two-way control handshake for data exchange. The fast sensing is performed during the $ATIM$ window while $fast$

sensing is performed during the *Data* window. The *fast* sensing is used to update the *SIP* vector. If the *SIP* value of a channel is set to 2, that channel will be *fine* sensed. During the *ATIM* window, channel sensing results are also exchanged via scan result packets (SRP).

In a multi-hop scenario, certain operations are additionally required for the correct functionality of the cognitive radio system.

(i) A channel reservation is done during the *ATIM* window for utilization during *Data* window via a 3-way handshake. In the *ATIM* frame, the sender inserts its list of channel opportunities. Receiver compares this with his own list and selects the channel with the least *SCL* values. The selected channels for data communication is inserted into the $ATIM - ACK$ frame. To mitigate hidden terminal problem, the transmitter rebroadcast the selected channel in the $ATIM - RES$ frame.

(ii) To mitigate the problem of out-of-order frame delivery of *ATIM*, *SRP* and beacon, a distributed coordination function is utilized. Different backoff windows are applied for different type of traffic. Here traffic is divided into two classes, (high priority and low priority). High-priority packets can select a backoff timer $w \in [0, CW/2)$, and low- priority packets have to select a backoff timer $w \in [CW/2, CW)$. The value of backoff timer is reset each time a packet is transmitted successfully. The beacon and *SRP* utilizes the high priority backoff timer while the *ATIM* packet utilizes low priority backoff timer.

(iii) In a multi-hop scenario, the presence of a primary user should be detected by all cognitive radio user even if they are operating on different channels. If a secondary user sees the channel to be free during the current superframe and it appeared to be busy during the last superframe then a deep channel sensing is performed to check whether a channel is really free. This provides additional insurance over primary users mitigation to interference. The time required for vacating a channel is given as channel vacate time (CVT).

The miss detection probability of a secondary user is given by $p_{md}^{(i)}$. The false alarm probability of a secondary user is given by $p_{fa}^{(i)}$. The number of cognitive radio users within the detectability range of the primary user is given as M. The probability of randomly selecting a channel is given as $1/C$. The network wide miss detection and false alarm probabilities, $p_{md}^{(n)}$ and $p_{fa}^{(n)}$ can be calculated as.

$$p_{md}^{(n)} = \left(1 - \frac{1}{C}(1 - p_{md}^{(i)})\right)^M \tag{2.41}$$

$$p_{fa}^{(n)} = 1 - (1 - p_{fa}^{(i)})^{N/C} \tag{2.42}$$

A channel is only vacated if a cognitive radio detects at least one primary user on the channel i.e. a primary user appears inside the detectability range of the cognitive radio performing the channel sensing. The mean value of the CVT can calculated as:

$$E[CVT] = \frac{t_{BI}}{2} + \sum_{k=1}^{\infty}(p_{md}^{(n)})^k t_{BI}[s] = \frac{t_{BI}}{1 - p_{md}^{(n)}} - \frac{t_{BI}}{2}[s] \tag{2.43}$$

where t_{BI} is the length of the beacon interval.

A channel opening time is the time it takes to open up a channel on which a primary user has ceased its operation. To simplify the analysis, it is assumed that the false detection probability of cognitive radio is zero. If $s^{(i)}$ is the number of scans a single cognitive radio requires to detect a channel as free, then it can be calculated as:

$$p(s^{(i)} = x) = \frac{1 - p_{fa}^{(n)}}{C}\left(\frac{C-1}{C} + \frac{p_{fa}^{(n)}}{C}\right)^{x-1}, x \in \mathbb{N}_0^+ \tag{2.44}$$

Since, the system is utilizing a distributed sensing scheme, all cognitive radio need to sense the channel as free before it is opened for communication. The number of scans the entire cognitive radio network needs to detect the channel as free is denoted by $s^{(n)}$. Therefore,

$$\begin{aligned} p(s^{(i)} \leq x) &= \sum_{k=1}^{x-1} p(s^{(i)} = k), x \in \mathbb{N}_0^+ \\ p(s^{(n)} = x) &= \sum_{m=1}^{M}\left[\binom{M}{m} p(s^{(i)} = x)^m * p(s^{(i)} \leq x)^{M-m}\right], x \in \mathbb{N}_0^+ \end{aligned} \tag{2.45}$$

Hence the average channel occupancy time can be calculated as:

$$E[COT] = t_{BI}\left(\sum_{k=1}^{\infty} xp(s = x)\right) + \frac{t_{BI}}{2}[s] \tag{2.46}$$

The primary user activity is modelled as Poisson process where the ON and OFF periods are exponentially distributed with parameters μ_{ON} and μ_{OFF} respectively. The probability that a channel is free is calculated as:

$$p_{free} = \frac{\frac{1}{\mu_{open}^*}}{\frac{1}{\mu_{open}^*} + \frac{1}{\mu_{closed}^*}} \tag{2.47}$$

where μ_{open}^* and μ_{closed}^* are calculated as:

$$
\begin{aligned}
\mu^*_{open} &= \left(\frac{1}{\mu_{ON}} + E[COT] - E[CVT]\right)^{-1} \\
\mu^*_{closed} &= \left(\frac{1}{\mu_{OFF}} + E[COT] - E[CVT]\right)^{-1}
\end{aligned} \tag{2.48}
$$

If all primary users are assumed to be inside the detection range of secondary users, then the throughput of a single hop cognitive radio network can be calculated as:

$$
\begin{aligned}
S_t = \sum_{i=1}^{C+1} \Bigg[{}^{C}C_{i-1} \Big[p_{free}\Big(1 - p^n_{fa}\Big)\Big] \\
* \Big[1 - p_{free}\big(1 - p^n_{fa}\big)\Big]^{C+1-i} \left(\frac{t_{BI} - t_{ATIM}}{t_{BI}}\right) \\
* \Big(\big[L - mod(L,i)\big] S_c\big(\lfloor L \rfloor i\big) + mod(L,i) S_c\big(\lfloor Li \rfloor i + 1\big)\Big)\Bigg]
\end{aligned} \tag{2.49}
$$

where $L = N/2$, denotes the number of communication links, $\lfloor x \rfloor$ is the floor function, $mod(x, y)$ is the modulo operator, $S_c(x)$ is the throughput estimation of the given model with x as the number of links on the channel.

Increasing the number of cognitive radio nodes increases the false alarm probability and therefore there is a throughput loss. This can be compensate by increasing the number of channels. Increasing the number of the cognitive radio nodes in the vicinity of the primary users decreases the missed detection probability. This MAC scheme utilizes distributed sensing scheme and solve the problem of hidden terminal problem while providing the feature of broadcast and multicast. Since, this media access scheme is based on dedicated control channel, it suffers from control channel saturation problem. While contending over control channel the bandwidth is wasted on data channels.

2.2.2 High Throughput Media Access Schemes

A cognitive MAC protocol (C-MAC) for distributed multi-channel is discussed in [10], which utilizes distributed quiet periods for effective spectrum sensing. The multi-channel is synchronized by utilizing varying beacon periods in different channels. It supports group communication like broadcast and multicast packet dissipation through a common channel.

The protocol uses the concept of a rendezvous channel (RC) for coordination between the nodes of the network. All the nodes agree on using exactly one channel as RC when joining the network. Each channel time is divided into time-slots called as superframes, which in turn has a beacon period and a data transfer period. The beacon period across all channels are non-overlapping, which allows for efficient utilization of the channel. Also, this means that if any node wants to get information regarding other in all channels, the node has to switch channels taken in ascending order of beacon period start time (BPST). On the other hand, for getting information about a node and the channel it is located on, switching to the RC channel is adequate as all the nodes periodically transmit their information beacons on the RC.

The RC is utilized to manage the entire cognitive radio network in the following ways:

(i) Network wide group communication via broadcast and multicast packet transmission.
(ii) Cognitive radio nodes visit the RC regularly to keep their clocks synchronized so as they can adjust the channel beacon period start time. Also, this helps the cognitive nodes to keep track of the offset they required to shift their BPST when moving to other channels.
(iii) Each cognitive node visit the RC periodically and therefore can figure out the other nodes residing on other channels. This helps to dissipate the neighbor node information among all cognitive radio users.
(iv) During the exchange of beacons over RC, the number of beacons per channel can be utilized to figure out the load on that channel. Here load refers to the number of cognitive nodes residing on a channel.
(v) The RC also acts as entry point to any new cognitive radio node joining the network. Any new cognitive radio node which wants to join the network first searches the RC (with appropriate beacon header) and switches to it.

To join a network, a new node sends its beacon onto the RC letting other nodes know about its existence. After learning the network wide information from the RC, the new node switches to the appropriate channel and becomes part of that channel.

If no RC channel is found, the new node selects itself to make a decision on marking a channel as RC. Once the node makes a decision on particular channel as RC, it starts transmitting RC beacon on it, so that other cognitive users can join it. It is possible that two or more cognitive user starts the process of starting RC on some channels. In such case, convergence of RC is required. Cognitive nodes in such system do *out-of-band* measurements of the spectrum. This is done to detect the presence of other RCs and collecting channel quality information. If some other overlapping RC is detected then the detector cognitive node moves back to its original RC and transmit a RC-switch information in its beacon. This RC-switch information informs other users on that RC to switch to the new RC along with the designated time of moving. Other data channels are formed in a similar way by the cognitive nodes, except that they send an additional beacon information over RC regarding the channel switch information.

To dissipate the complete network information and mitigate hidden terminal problem, a form of distributed beaconing is employed in each channel by its occupants. At the beacon period of each frame, each cognitive node in its allocated slot transmits a beacon which contains information about its neighbours and neighbours of neighbours (information received in the past frames). This allows all nodes to keep track of other cognitive nodes on the same channel and allows mobility of cognitive nodes on the same channel.

All the nodes periodically (and in a non overlapping fashion) switch to RC and transmit their beacon for synchronisation purposes. This means that the information contained in the RC is almost always up to date. This also helps in channel selection. Any node in any of the channel can analyze the load on the channel from the number of node beacons transmitted in the channel. This is rebroadcasted at the RC when this node switches to RC during its resynchronization phase. Hence, other cognitive nodes comes to know about the loads on other channels too. This allows selection of appropriate channels by the cognitive nodes so as to decrease the contention and increase the throughput.

Group communication like broadcast and multicast is tackled in the protocol using the RC channel. Whenever a node has to perform a broadcast/multicast, it shifts to RC and send a beacon scheduling a group communication. The group communication is scheduled such that there is sufficient time in between for all other nodes to receive the information from RC during their resynchronisation phase and be able to schedule for group communication accordingly.

Another important aspect of this protocol is that it provide coexistence of secondary users with primary incumbents as well as other secondary users. The detection of primary users is done during quiet period (QP) defined for all channels (and non-overlapping for each). Once a primary user is detected, a beacon is transmitted to notify other on the channel about the incumbent and channel evacuation is facilitated. Coexistence with other secondary users (self-coexistence) is achieved via a beacon period merging scheme. If an overlapping foreign BP is detected within the BPST of the current cognitive node, the current cognitive node changes its BPST to the foreign BPST and adjust its beacon slot accordingly. If a non-overlapping foreign BP is detected, and its BPST lies within the first half of the current node's superframe, then the current node relocates its beacon to the foreign BP.

The scheme provides a robust coexistence mechanisms when incumbent nodes are present in the channel. It provides proper mitigation of hidden node problem, synchronization achieved in time, space and frequency via the beaconing approach. The utilization of RC provide multicast and broadcast facility.

The scheme is dependent on RC for coordination making it a bottleneck. The overhead for synchronisation and coordination becomes significant when data packets are small. Especially since a 3 way handshake is required in the data channel and for a beacon period synchronisation in the beacon period of the frame. Due to the requirement that the beacon period is non-overlapping, the frame length is a function of the number of nodes on a channel and the beacon length. This may cause a significant degradation in throughput if the packets are small compared to the transmitting window of the frame. The hidden terminal problem is addressed using neighbour and next neighbour information included in the beacons transmitted. Mobility may still cause nodes to go out of this extended awareness radius causing hidden node problems.

2.2.3 Self-coexistence Based MAC Protocol

If cognitive radio based wireless devices make a breakthrough in practical day to day life, then the opportunistic spectrum access will become more challenging. The increase in complexity will be attributed to coexistence of multiple cognitive radio networks in the vicinity of each other. Since one cognitive radio network will normally see other cognitive radio network as a foreign network or maybe as primary incumbent of the channel on which it is residing. Most of the existing MAC protocols in the literature takes a conservative approach in terms of spectrum sensing, i.e. when a spectrum is sensed as busy, the cognitive radio node ceases to transmit. In [11], a media access scheme is designed which considers the coexistence of multiple cognitive radio networks and ensures fair access of resources among all users.

The problem with such an approach is that even if the channels are occupied by other CR users from different networks, the node sensing the channel will still label it as busy and will simply exclude it. Instead of utilizing a two state model (idle or busy), a three-state sensing model for a channel is discussed.

In a system comprising of one primary user, M infrastructure based cognitive radio networks (denoted as $CR_1, CR_2, \ldots CR_M$) coexists. Each CR_i network comprises of N_i secondary users. The cognitive radio node utilizes two transceivers: one for communication and one is dedicated for spectrum sensing. Cooperative sensing is done by nodes of the CR_i network and a decision is made regarding the channel availability by the base station (BS) of the CR_i network. Let DT be the time required for cooperative spectrum sensing, therefore the channel informations are updated periodically every DT period of time. The channel sensed by the CR user (r_i) in a the three state model is given as:

$$r_i = \begin{cases} n_i & H_0 \\ x_p + n_i & H_1 \\ x_s + n_i & H_2 \end{cases}$$

where H_0, H_1 and H_2 are the state of channel being empty, channel occupied by a primary user and channel occupied by some secondary user, respectively. $x + p$ and x_s are the signals from the primary and secondary users, while n_i is the zero mean additive white Gaussian noise.

If a channel is busy, the cognitive node performs a fine sensing and analyze the received energy in the second stage via distance estimation technique to identify whether the channel is occupied by a primary user H_1 or by some other cognitive radio user H_2. When a cognitive radio user senses state H_2, it competes for the channel in a fair manner among all remaining cognitive radio nodes as ensured by the protocol. If instead it senses state H_1, it immediately vacates the channel to not cause interference to the primary user transmission.

Here primary users have strict higher priority over secondary users. If a primary user transmission is detected (H_1 state), a secondary user immediately tries to vacate the channel. This will require a certain duration of time, which appears as interference time to the primary user. There is a bound on maximum interference time that can appear to a primary user (T_{max}). If there is no primary user present on the channel (H_0 or H_2), the secondary users continue accessing the channel to determine the exact channel state (i.e. whether H_0 or H_2). If the channel state is determined as H_0, the secondary user access the channels like any other cognitive radio node. If the channel is determined in the H_2 state, then the conventional IEEE 802.11 [12] distributed coordination function (DCF) mechanism is utilized for contention and channel access as discussed in Sect. 1.5.1. If a secondary user has a packet for transmission, it waits for a DCF interframe space (DIFS) duration before transmitting the packet. If during this waiting period any other user is detected on the channel then a backoff mechanism is utilized. The backoff duration is randomly selected from $[0, W-1]$, where W is the size of contention window. To achieve the optimal fairness among all secondary users, this method does not utilize binary exponential backoff mechanism of IEEE 802.11.

If X represent a random variable denoting the time period during which a primary user continuously transmits on a channel, its probability density function is represented as $f_X(x)$. Therefore the duration during which the channel is not accessible (Y) by any secondary user is given as:

$$\begin{aligned} Y &= \left(\lceil \frac{X}{DT} + 1 \rceil\right) DT \\ &\approx X + DT \end{aligned} \tag{2.50}$$

The cumulative distribution of Y can be calculated as:

$$\begin{aligned} F_Y(y) &= P_r\big((x + DT) \leq y\big) \\ &= \int_0^{y-DT} f_X(x)dx \end{aligned} \tag{2.51}$$

Therefore, the mean of Y can be calculated as:

$$E(Y) = \int_{DT}^{DT+t} y F_Y'(y) dy \tag{2.52}$$

If X follows an exponential distribution with parameter μ, the mean value of Y can be calculated as:

$$\begin{aligned} E(Y) &= \int_{DT}^{DT+t} y \mu e^{-\mu(y-DT)} dy \\ &\approx \frac{1}{\mu} + DT \end{aligned} \tag{2.53}$$

For a primary user following Poisson traffic with mean arrival rate as λ, the probability that the channel is occupied during the time period t can be calculated as:

$$P_{occupied} = \frac{\lambda t E(Y)}{t} = \left(\frac{1}{\mu} + DT\right)\lambda \tag{2.54}$$

Therefore the probability that the channel is free for secondary users to use is given as $P_{free} = 1 - P_{occupied}$.

A Markov chain model is developed to do the analysis of throughput saturation of the network. Where each state in the Markov chain represents the state of the backoff stage of the secondary user's transmission. Each time slot is considered either a period of primary user occupancy, successful secondary user transmission, idle or collision among multiple secondary user. The channel access probability P_α is calculated as:

$$P_\alpha = 1 - \frac{W}{P_{free}(1 - P_\alpha)^N + P_{occupied} + W} \tag{2.55}$$

Therefore the probability that a transmission by secondary user over a given time slot would be successful is given as:

$$P_s = P_{free} N P_\alpha \left(1 - P_\alpha\right)^{N-1} \tag{2.56}$$

The probability that a given time slot is idle slot is given as:

$$P_{idle} = P_{free}(1 - P_\alpha)^N \tag{2.57}$$

The probability that a collision happens in a given time slot is given as:

$$P_{cs} = P_{free}\left[1 - (1 - P_\alpha)^N - N P_\alpha.(1 - P_\alpha)^{N-1}\right] \tag{2.58}$$

If the average time of a successful secondary user transmission slot, the average time of collision slot caused by other secondary users and primary users are represented by T_s, T_{cs} and T_{cp}, then the throughput of secondary user is calculated as:

$$\theta = \frac{P_s T_s}{P_{idle}\sigma + P_s T_s + P_{cs} T_{cs} + P_{occupied} T_{cp}} \tag{2.59}$$

where σ is the duration of an idle time slot.

The saturation throughput (θ^*) is calculated by optimizing the Eq. 2.59 as:

$$\theta^* = max\{\theta\} \tag{2.60}$$

This work primarily focused on coexistence technique including self-coexistence which many other work fail to do. A kind of fairness scheme is utilized which provides fairness to all cognitive radio nodes in the network. Extending this scheme to an ad-hoc cognitive radio network would be tedious, as the system would have to make distributed decision over channel state. These channel states may appear different for different cognitive users within the same network.

A conceptual idea about an opportunistic spectrum access (OSA-MAC) [13] scheme was discussed in Sect. 1.7.1. This protocol utilizes a dedicated control channel for control packet interchange ans coordination between the nodes. Multiple opportunistic channels are utilized for data communication between secondary users which are owned by primary incumbents.

The time is divided into superframes, which in turn are divided into *ATIM* window and *transmission* intervals. During the *ATIM* window period, the secondary users exchange ATIM (announcement traffic indication message) on the control channel to indicate and agree on a channel for transmission. Once this has been done, the participating nodes switch to the agreed channel during *transmission* interval, where they compete with existing nodes using a RTS/CTS handshake and utilize the channel. A preemptive sensing is done on the data channel before any further action to make sure no primary users are present on it. If the case arises that primary users are present, then the secondary users employ backoff periods and wait for the primary users to vacate. The superframe period during second phase (where the secondary senses the channel for primary user activity) is such that none of the secondary users are allowed to transmit during this period (Quiet periods as discussed in Sect. 3.4.2).

The method utilizes two types of channel selection for data transmission: *random* and *opportunistic based*. In a random channel selection a random channel is selected with a probability of $1/L$, where L is the total number of data channels utilized in the system. In an opportunistic based channel selection, the spectrum availability probability is taken into consideration. If the number of secondary users is small as compared to the number of channels, it is better to allow the secondary users to exploit more available channels instead of selecting any random channel. For a threshold number of nodes at N_{th} and total number of users participating in communication N, if $N \leq N_{th}$ the opportunity based channel selection approach is employed or else the uniform channel selection method is employed. If $p_{i,j}$ is the probability that a channel j is available for secondary communication i and if $p_{i,k}$ is the probability that a channel k is available for secondary communication i, then the probability that a user i choses channel j is given as:

$$q_{i,j} = \frac{p_{i,j}}{\sum_{k=1}^{L} p_{i,k}} \tag{2.61}$$

The probability that k users selects channel j for data communication during transmission phase is given as:

$$C_{k,j} = {}^{N}C_k (q_j)^k (1 - q_j)^{N-k} \tag{2.62}$$

The probability that exactly one user wins the contention among k users on channel j with maximum window size W is given as:

$$S_{k,j} = \frac{k}{W} \sum_{l=1}^{W} \left(1 - \frac{l}{W}\right)^{k-1} \tag{2.63}$$

Therefore throughput on a channel j can be calculated as:

$$T_j = \frac{Rt_{TI}}{t_{SF}} \sum_{k=1}^{N} C_{k,j} p_j S_{k,j} \tag{2.64}$$

where R, t_{TI}, t_{SF} and p_j are the transmission rate, time needed to transmit data on each channel, length of the superframe and channel available probability of channel j, respectively.

The total throughput for L channels is the summation of throughput on each channel and can be calculated as:

$$T = \sum_{j=1}^{L} T_j \tag{2.65}$$

The probability of collision of a secondary user transmission with a primary user is calculated via utilizing the probability of miss detection. The probability of collision is thus given as:

$$P_j^{(c)} = \sum_{k=1}^{N} C_{k,j} \left[1 - (1 - P_m)^k\right](1 - p_j) \tag{2.66}$$

where P_m is the probability of miss detection by the given secondary user k, p_j is the probability of channel j to be free.

This opportunistic channel selection method provides a better saturation throughput for cases where the total number of secondary is comparable to the number of channels. There is an upper bound on the number of users a network can accommodate given a minimum QoS standard and the sensing error derived from it. The control channel must be owned by the secondary service provider which would incur cost. As the number of secondary users increase, the superframe period needs to increase and since no transmission occurs on other channels during this period among the secondary, this causes underutilization of resources.

2.3 Energy Management

Users in a cognitive radio network may be powered primarily by a portable battery which generally has limited endurance between recharges. Therefore, energy management is an important issue that needs to be dealt in a cognitive radio network for greener communication, extended operable life of a user between recharges and to mitigate the effects of intermediate node failures [14]. Operating at low energy forces a user to degrade its observed QoS, and hence a graceful degradation need to be observed by such systems.

Energy management in a cognitive radio system depends upon multitude of things. This can be performed at three different steps of cognitive radio network: *transmission power control, spectrum decision* and *media access scheme*. By doing transmission power control, a user limits the transmission power of a transmitter and hence reduces the energy consumption. This reduces the SNR/SINR at the receiver side and hence the observed QoS. Spectrum allocation based energy management techniques employ selection of channels which require lower energy for transmission. A media access scheme decides how a channel is shared among multiple users in the network and thus its etiquette determines the activity of a user and hence its power consumption. Depending upon cooperation between different secondary user in the network, the sensing in the cognitive radio network can be classified into two types: *cooperative* and *non-cooperative*.

2.3.1 Cooperative Sensing Based Energy Efficient Spectrum Sensing

An energy efficient cooperative spectrum sensing scheme is discussed in [15], which optimizes the sensing time in a given superframe satisfying the constraints of sensing accuracy, throughput and latency. In a given superframe, there is a trade-off between sensing and transmission. Similarly, there is also a trade-off between waiting on a occupied channel to become free or switch onto another free channel.

If the sensing time is short then the sensing result is not accurate and hence detection of idle channel becomes impossible. Therefore carrying the transmission on current channel without accurate sensing may result in throughput loss and interference to primary incumbents. In other case, if the sensing time is too long then the transmission duration becomes small. If the channel is sensed to be busy then the secondary user can either switch to other channel or wait for the next time slot. If the secondary user waits, it will conserve energy but it will also increase the transmission delay and reduce throughput. Similarly, if it switches to another channel and performs data transmission, the the throughput is increased at the cost of increased energy consumption.

The time is divided into recurring superframes of equal length. Each superframe length is divided into sensing and transmission duration. The system comprises of M primary incumbent channels which are opportunistically utilized by the secondary users. Each secondary user performs *in-band* sensing to detect whether the current channel is idle or occupied. On the basis a decision is made on whether to stay on the current channel or move to another idle channel.

As given in Eq. 2.91, the received signal at i-th secondary user is given as r_i for m-th sample, where $s(m)$ is the primary users' transmitted signal with zero mean and σ_s^2 variance while $w(m)$ is noise which is modelled as a Gaussian process with zero mean and σ_w^2 variance ($\mathcal{N}(0, \sigma_w^2)$); while H_0 represents the hypothesis that the channel is available while H_1 represents the hypothesis that the channel is occupied by the primary incumbents.

If γ is the average SNR of primary user, then the probability of detection (P_d) and false alarm (P_f) for sensing time τ_s is given as:

$$
\begin{aligned}
P_d(\tau_s) &= Q\left(\frac{1}{\sqrt{2\gamma+1}}\left(Q^{-1}(\bar{P_f}) - \gamma\sqrt{\tau_s f_s}\right)\right) \\
P_f(\tau_s) &= Q\left(\frac{1}{\sqrt{2\gamma+1}}\left(Q^{-1}(\bar{P_d}) + \gamma\sqrt{\tau_s f_s}\right)\right)
\end{aligned}
\tag{2.67}
$$

where $Q(x)$ is the Gaussian tail probability function, while $Q^{-1}(.)$ is its inverse. f_s is the sampling frequency. $\bar{P_f}$ and $\bar{P_d}$ are target probability of false alarm and detection, respectively.

The minimum sensing time to achieve the desired target probability of false alarm and detection can be related as:

$$
\tau_s^{min} = \frac{1}{\gamma^2 f_s}\left(Q^{-1}(\bar{P_f}) - Q^{-1}(\bar{P_d})\sqrt{2\gamma+1}\right)^2 \tag{2.68}
$$

Here the minimum sensing time is related to the desired target probability of false alarm and detection. But this amount of sensing time may consume considerable energy and an optimization over it is required constrained to maximum energy consumption.

Based on the current channel sensing results, there exist three possibilities: current channel is *idle*, all channels are *busy*, and current channel is *busy* with at least one other channel is *idle*.

If the current channel is *idle*, then the secondary user makes a decision on to stay on the current channel. Switching onto another channel is inefficient and will result in increase energy consumption. If all M channels are *busy* (can be detected by *out-of-band* sensing), then no channel switching is done. If the current channel is *busy* and at least one another channel is *idle*, then a decision is made to minimize the energy consumption. If the secondary user switch to another channel, then there is increase in throughput at the cost of increased energy consumption. If the secondary user stays on the channel, then the energy consumption is minimized at the cost of throughput loss and increased latency.

If P_s is the probability that the secondary user stays on the current channel for the above discussed third case, then a joint optimization problem can be formulated with optimal sensing time (τ_s) as following:

$$\begin{aligned} &\min_{\tau_s, P_s} E(\tau_s, P_s) \\ \text{subject to } &P_d(\tau_s) \geq \bar{P_d} \\ &P_f(\tau_s) \leq \bar{P_f} \\ &W(\tau_s, P_s) \geq \mathbf{W} \\ &D(\tau_s, P_s) \geq \mathbf{D} \end{aligned} \tag{2.69}$$

where E is the total average energy required to transmit one packet, W is the average throughput, D is the average delay, **W** is the minimum average throughput and **D** is the maximum average delay.

If E_s and E_t represents energy of sensing and transmission in each time-slot respectively, then the total average energy consumption can be given as:

$$E(\tau_s, P_s) = N\tau_s E_s + N P_{sw} E_{sw} + T_{tr} E_t \tag{2.70}$$

where N is the total number of slots needed for one data packet transmission, E_{sw} is the energy cost for one channel switching, P_{sw} is the probability of switching to idle channel, T_{tr} is time required for one packet data transmission.

The average throughput W can be calculated as a function of τ_s and P_s with the time-slots of period T as:

$$W(\tau_s, P_s) = \frac{B_{st} + B_{sw}}{\tau_s + T} \tag{2.71}$$

where B_{stay} is the average number of bits that are transmitted in the duration $\tau_s + T$ when the secondary user chooses to stay on the same channel; while B_{switch} is the average number of bits that are transmitted in the duration $\tau_s + T$ when he secondary user chooses to switch to another channel.

B_{st} and B_{sw} can be given as:

$$\begin{aligned} B_{stay} &= P_{idle} B_T \\ B_{switch} &= P_{swidle} B_T (1 - P_{swbusy}) \end{aligned} \tag{2.72}$$

where P_{idle} is the probability that a channel is correctly sensed by the secondary user, P_{swidle} is the probability that a secondary user switches to an idle channel while P_{swbusy} is the probability that a secondary user switches to a busy channel.

The average delay can also be expressed as a function of τ_s and P_s with the time-slots of period T as:

$$D(\tau_s, P_s) = N\Big(\tau_s + T P_{wait}\Big) \tag{2.73}$$

where P_{wait} is the probability that the secondary user makes a decision of staying on the current channel and N is the number of frames required for completing the data transmission.

From Eq. 2.69, an optimal value of P_s can be achieved for energy minimization via setting, $W(\tau_s, P_s) = \mathbf{W}$ and $D(\tau_s, P_s) = \mathbf{D}$. Therefore the optimization problem in Eq. 2.69 is reduced from joint optimization to a single optimization variable τ_s.

The system assumes that all the channels are sensed at the same time with same accuracy which is not true and thus the probability of false detections and finding busy channels are affected. Especially in the heterogeneous environment where channel characteristics varies.

In conventional distributed sensing approach, the higher detection performance comes with a higher network energy consumption. In [16], a distributed sensing technique is discussed which reduces the network energy consumption by optimally choosing the sleeping and censoring parameters. The energy consumption is constrained and subjected to the the minimum detection performance and a maximum false alarm probability.

The system comprises of N cognitive sensor nodes and a fusion centre (FC) in a parallel distributed fusion configuration. In a parallel distributed fusion configuration, each cognitive sensor node makes a decision on the channel availability and sends the result to fusion centre. The fusion centre then fuses these local sensing data and makes a decision on the channel availability. Each cognitive radio node follows two policies: *sleeping policy* and *censoring policy*. The sleeping policy decides whether a cognitive radio node is awake or in sleep mode, while the censoring policy decides whether a sensed results is transmitted or not.

As given in Eq. 2.91, the received signal at the m-th sample for the i-th secondary user is given as r_i. Also, SNR at each cognitive radio node is considered to be same (γ). The cognitive radio node performs binary hypothesis on energy detection. H_0 represents the hypothesis that the channel is available while H_1 represents the hypothesis that the channel is occupied by the primary incumbents. $s(m)$ is the primary users' transmitted signal with zero mean and σ_s^2 variance while $w(m)$ is a Gaussian process with zero mean and σ_w^2 variance ($\mathcal{N}(0, \sigma_w^2)$). Therefore the received energy at the i-th cognitive radio node over T_0 observation samples is given as:

$$E_i = \sum_{k=1}^{T_0} r_i^2(k) \tag{2.74}$$

A form of thresholds (λ_1, λ_2) are applied to cognitive sensor nodes to enable censoring policy. The threshold bound $\lambda_1 < E_i < \lambda_2$, is called as the censoring region. Therefore by this rule, the local decisions made by the cognitive sensor nodes can be given as:

$$\text{result} = \begin{cases} \text{send 1 (implying } H_1), & \text{if } E_i \geq \lambda_2 \\ \text{no decision}, & \text{if } \lambda_1 < E_i < \lambda_2 \\ \text{send 0 (implying } H_0), & \text{if } E_i \leq \lambda_1 \end{cases} \tag{2.75}$$

The false alarm probability (P_f) and detection probabilities (P_d) can be given as:

$$P_f = \frac{\Gamma(T_0, \lambda_2/2)}{\Gamma(T_0)}$$
$$P_d = Q_{T_0}(\sqrt{2\gamma}, \sqrt{\lambda_2}) \tag{2.76}$$

where $\Gamma(n, x)$ is an upper incomplete gamma function given by $\Gamma(n, x) = \int_x^{\infty} t^{n-1} e^{-t} dt$, with $\Gamma(n, 0) = \Gamma(n)$, and $Q_u(n, x)$ is the generalized Marcum Q-function given by, $Q_u(n, x) = \frac{1}{n^{u-1}} \int_x^{\infty} e^{-\frac{t^2+n^2}{2}} I_{u-1}(nt)dt$. $I_{u-1}(.)$ is the Bessel function of the first kind with the order $u - 1$.

The cost function is modelled as the average energy consumed by each cognitive radio node in the system, which can be given as:

$$C_T = (1 - \mu) \sum_{i=1}^{N} (C_{s_i} + C_{t_i}(1 - \rho)) \tag{2.77}$$

where, C_{s_i} and C_{t_i} are the energy required for sensing and transmission by i-th cognitive sensor node, respectively. μ is the sleeping rate while ρ is the censoring rate and equals to the probability such that $\lambda_1 < E_i < \lambda_2$.

The constraints are applied over the global detection (Q_D) and false alarm (Q_F) probabilities. Therefore the optimization problem can be formulated as:

$$\min_{\mu, \lambda_1, \lambda_2} C_T$$
$$\text{such that } Q_D \geq \beta \tag{2.78}$$
$$Q_F \leq \alpha$$

where β and α are parameters that can be modified for different use case scenario.

Depending on the prior knowledge about the probabilities of hypotheses H_0 and H_1, two different cases are discussed.

2.3.1.1 Unknown Prior

In case of unknown prior, it can be assumed that $Pr(H_0) \lll Pr(H_1)$ (probability of hypothesis H_0 is very less than hypothesis H_1) implying that channel is mostly free from primary incumbents. The problem then can be formulated as blind Neyman–Pearson setup, which can be given as

$$\rho_{Neyman} = Pr(\lambda_1 < E_i < \lambda_2 | H_0) \tag{2.79}$$

From Eq. 2.76, the above equation can be rewritten as:

$$\rho_{Neyman} = \frac{\Gamma(T_0, \frac{\lambda_1}{2})}{\Gamma(T_0)} - \frac{\Gamma(T_0, \frac{\lambda_2}{2})}{\Gamma(T_0)} \tag{2.80}$$

From Eqs. 2.77 and 2.80, the Eq. 2.78 can be given as:

$$\min_{\mu,\lambda_1,\lambda_2} (1-\mu)\sum_{i=1}^{N}\left[C_{s_i} + C_{t_i}\left(1 - \frac{\Gamma(T_0, \frac{\lambda_1}{2})}{\Gamma(T_0)} + \frac{\Gamma(T_0, \frac{\lambda_2}{2})}{\Gamma(T_0)}\right)\right]$$
$$\text{such that } Q_D \geq \beta$$
$$Q_F \leq \alpha \tag{2.81}$$

The global probability of false alarm (Q_F) in the Neyman–Pearson setup can be calculated as:

$$Q_F = \sum_{k=1}^{N}\binom{N}{k}\mu^{N-k}(1-\mu)^k * \sum_{l=1}^{K}\binom{K}{l}\rho^{K-l}(1-\rho)^l[1-(1-P_f)^l] \tag{2.82}$$

Using binomial expansion theorem this can be reduced to

$$Q_F = 1 - \left[1 - P_f(1-\mu)(1-\rho)\right]^N \tag{2.83}$$

The global probability of detection (Q_D) in the Neyman–Pearson setup can be calculated as:

$$Q_D = \sum_{k=1}^{N}\binom{N}{k}\mu^{N-k}(1-\mu)^k * \sum_{l=1}^{K}\binom{K}{l}\delta^{K-l}(1-\delta)^l[1-(1-P_d)^l] \tag{2.84}$$

Using binomial expansion theorem this can be reduced to

$$Q_D = 1 - \left[1 - P_d(1-\mu)(1-\delta)\right]^N \tag{2.85}$$

Therefore by utilizing the value of Q_F and Q_D, the optimization problem in Eq. 2.81 can be solved to derive optimum sleeping rate (μ) and censoring thresholds (λ_1, λ_2).

2.3.1.2 Known Prior

If the probabilities of hypothesis H_0 ($Pr(H_0)$) and hypothesis H_1 ($Pr(H_1)$) are known in advance, then the problem can be formulated as knowledge aided Bayesian setup. Therefore the censoring rate can be given as:

$$\rho_{Bayesian} = \pi_0 Pr(\lambda_1 < E_i < \lambda_2|H_0) + \pi_1 Pr(\lambda_1 < E_i < \lambda_2|H_1) \tag{2.86}$$

where π_0 and π_1 are the probability of hypothesis H_0 and H_1, receptively.

From Eq. 2.76, we have:

$$\begin{aligned} Pr(\lambda_1 < E_i < \lambda_2 | H_0) &= \frac{\Gamma(T_0, \frac{\lambda_1}{2})}{\Gamma(T_0)} - \frac{\Gamma(T_0, \frac{\lambda_2}{2})}{\Gamma(T_0)} \\ Pr(\lambda_1 < E_i < \lambda_2 | H_1) &= Q_{T_0}(\sqrt{2\gamma}, \sqrt{\lambda_1}) - Q_{T_0}(\sqrt{2\gamma}, \sqrt{\lambda_2}) \end{aligned} \tag{2.87}$$

From Eqs. 2.77 and 2.87, Eq. 2.78 can be rewritten as:

$$\begin{aligned} &\min_{\mu,\lambda_1,\lambda_2} (1-\mu) \\ &\sum_{i=1}^{N}\Bigg(C_{s_i} + C_{t_i}\Bigg(1 - \pi_0\Big(\frac{\Gamma(T_0, \frac{\lambda_1}{2})}{\Gamma(T_0)} - \frac{\Gamma(T_0, \frac{\lambda_2}{2})}{\Gamma(T_0)}\Big) \\ &+ \pi_1\Big(Q_{T_0}(\sqrt{2\gamma}, \sqrt{\lambda_1}) - Q_{T_0}(\sqrt{2\gamma}, \sqrt{\lambda_2})\Big)\Bigg)\Bigg) \text{ such that } Q_D \geq \beta \\ &Q_F \leq \alpha \end{aligned} \tag{2.88}$$

The local probability of false alarm and detection in knowledge aided Bayesian setup can be given as:

$$\begin{aligned} P_f^B &= (1-\mu)(1-\delta_0)P_f \\ P_d^B &= (1-\mu)(1-\delta_1)P_d \end{aligned} \tag{2.89}$$

Therefore, the global probability of false alarm and detection in knowledge aided Bayesian setup can be derived as:

$$\begin{aligned} Q_F &= 1 - \Big[1 - P_f(1-\mu)(1-\delta_0)\Big]^N \\ Q_D &= 1 - \Big[1 - P_d(1-\mu)(1-\delta_1)\Big]^N \end{aligned} \tag{2.90}$$

where $\delta_0 = Pr(\lambda_1 < E_i < \lambda_2 | H_0)$ and $\delta_1 = Pr(\lambda_1 < E_i < \lambda_2 | H_1)$.

Therefore by utilizing the value of Q_F and Q_D, the optimization problem in Eq. 2.88 can be solved to derive optimum sleeping rate (μ) and censoring thresholds (λ_1, λ_2).

The blind Neyman–Pearson setup is shown to be an special case of Bayesian setup. As the transmission energy increases compared to the sensing energy, the total transmission energy has to be reduced more than the sensing energy thus the censoring rate increases and the sleeping rate decreases. It is shown that without sleeping or censoring, the energy consumed saturates to a level which is much lower to conventional energy consumption. Also, as the number of users increase, the optimal sleeping rate increases dramatically to keep the energy consumption of the system stable.

2.3.2 *Non-cooperative Sensing Based Energy Efficient Spectrum Sensing*

Traditionally usage of cooperative sensing provides efficient channel occupancy results in a resource constrained environment, but it requires exchange of control messages. This exchange of control messages for cooperation increases the energy consumption of cognitive radio network. In [17], a time division energy efficient (TDEE) non-cooperative spectrum sensing scheme for cognitive radio network is discussed. In this scheme, the sensing period is divided in time slots and each secondary user in the network is assigned a different channel to detect in each slot. This allows secondary users to accurately sense the network without exchanging control messages for cooperation and allowing to save considerable energy.

The system utilizes a central controller node (base station in infrastructure network or cluster head in case of ad-hoc network) which coordinates the cognitive nodes in the network. The central controller designates cognitive nodes in the network to perform sensing on given channels along with the given sensing order. The information is then received by the central controller from the cognitive nodes, which in turn perform fusion on the received spectrum sensing data to make a decision on the availability of channels. The system time is divided into superframes, where each superframe comprises of two parts: *sensing* and *transmission*. The duration of the superframe is given as T, in which T_s is the time duration of spectrum sensing whereas T_r is the time duration of transmission in the superframe. The time T_s is further divided in equal time slots during which the secondary user senses different channels in the order given by the central controller. After the expiration of sensing period all the SUs send the sensing results to the central controller. For given N secondary users and M channels in the system, the given scenario considers where U secondary users want to perform sensing, where $2 \leq U \leq min(N, M)$.

The steps required for a secondary user to perform the sensing in the discussed scheme can be given as:

(i) All secondary users join to the network controller by central controller by following their network join/leave protocol. Once joined, it waits for the control messages from the central controller.
(ii) The central controller provides the information on number of channels to be sensed by the secondary users. Depending upon that, the sensing duration is divided into mini time-slots during which the secondary user individually scans each channel. Once the sensing is performed on all channels, the secondary users will update the information to the central controller.
(iii) The central controller fuses the channel availability information received from all secondary users and based on that selects a channel.

Since, the control packets are only exchanged with central controller, the number of control packets exchanges is substantially reduced. The efficiency of the discussed can be analysed in two schemes: *Homogeneous* networks and *Heterogeneous* networks.

2.3.2.1 Homogeneous Networks

In a cognitive radio based homogeneous network environment, all secondary users have the same sensing time and same channel coefficient h from primary incumbent's transmitter. The sensing duration is divided in U time slots and length of each slot is τ such that $T_s = U\tau$.

Total energy spent in the sensing duration is sum of energy spent for sensing different channels and energy used for exchange messages. It is clear that the sensing/exchange energy will not only depend on the sensing/exchange power but also on sensing/exchange time. Although the exchange duration is very small the exchange energy can be neglected, but as the number of cooperative secondary users increase, the exchange energy will also increase (keeping in mind retransmission if error happens). Therefore the energy consumed via exchange of control packets is also taken into consideration. The received signal r_i at the m-th sample for the i-th secondary user can be given as:

$$r_i(m) = \begin{cases} w(m), & H_0 \\ s(m) + w(m), & H_1 \end{cases} \tag{2.91}$$

where H_0 represents the hypothesis that the channel is available while H_1 represents the hypothesis that the channel is occupied by the primary incumbents. $s(m)$ is the primary users' transmitted signal with zero mean and σ_s^2 variance while $w(m)$ is a Gaussian process with zero mean and σ_w^2 variance ($\mathcal{N}(0, \sigma_w^2)$).

The detection probability can be calculated by utilizing the central limit theorem over the Chi-square distribution and is given as:

$$P_i^d = Q\left(\frac{\lambda - \tau f_s(|h|^2\sigma_s^2 + \sigma_w^2)}{\sqrt{2\tau f_s}(|h|^2\sigma_s^2 + \sigma_w^2)}\right) \tag{2.92}$$

Similarly, the false alarm probability can be calculated as:

$$P_i^f = Q\left(\frac{\lambda - \tau f_s\sigma_w^2}{\sqrt{2\tau f_s}\sigma_w^2}\right) \tag{2.93}$$

where, $Q(x) = \frac{1}{\sqrt{2\pi}}\int_x^\infty e^{-\frac{t^2}{2}}dt$ is the tail probability of the standard Normal distribution function and f_s is the sampling frequency.

The detection probability and false alarm probability calculated by the central controller can be given as:

$$\begin{aligned} P^d(U) &= 1 - (1 - P^d)^U \\ P^f(U) &= 1 - (1 - P^f)^U \end{aligned} \tag{2.94}$$

If P_{tx} and P_{rx} are the transmitted power and received power at each secondary user, respectively, then the optimization problem of calculation of requirement of number of secondary user for energy minimization can be calculated as:

$$\begin{aligned} \min_{U} :& U\Big[T_s P_{tx} + T_e P_{rx}\Big] \\ \text{such that, } & P^d(U) \geq P^d_{Th} \\ & P^f(U) \leq P^f_{Th} \end{aligned} \tag{2.95}$$

where $T_s P_{tx}$ is the sensing energy consumed in the homogeneous network and $T_e P_{rx}$ is energy consumed by exchanging control messages in which T_e is the control message exchange time. $P_{d,Th}$ is the threshold of the cooperative detection and $P_{f,Th}$ is threshold of false alarm.

By applying bounds on the constraints of Eq. 2.95, the bounds on U can be given as:

$$\frac{ln(1 - P^d_{Th})}{ln(1 - P^d)} \leq U \leq \frac{ln(1 - P^f_{Th})}{ln(1 - P^f)} \tag{2.96}$$

Since, the optimization Eq. 2.95 is a decreasing function of the parameter U, the minimal value of U for minimal energy consumption is given as:

$$U = \frac{ln(1 - P^d_{Th})}{ln(1 - P^d)} \tag{2.97}$$

If P_s is the probability that a channel is successfully found then it can be given as:

$$P_s = p\Big[1 - P_f(U)\Big] \tag{2.98}$$

where p is the channel available probability in a 2-state Markov chain model of spectrum occupancy.

Therefore the probability that the available channels can be found in one sensing period ($P_{available}$) can be given as:

$$P_{available} = \sum_{u=1}^{U} \binom{U}{u} (1 - P_s)^{U-u} P_s^u \tag{2.99}$$

The throughput of a channel can be derived by utilizing the Eq. 2.99:

$$\mathbf{TH} = (T - U\tau) P_{available} R \tag{2.100}$$

where R is the bandwidth of the channel.

2.3.2.2 Heterogeneous Networks

In heterogeneous networks all secondary users have different sensing periods and different channel coefficients ($h_i, \forall i \in U$). In this case the total energy consumed will be dependent on the number of secondary users (U) along with the sensing time of each secondary user. If τ_i is the size of the time slot of i-th secondary user, then the optimization equation for minimizing the energy consumption can be given as

$$\begin{aligned} &\min_{U;T_s^1,\ldots T_s^U} \sum_{i=1}^{U} \left[T_s^i P_{rx} + T_e P_{tx} \right] \\ &\text{such that, } P^d(U) \geq P_{Th}^d \\ &\qquad P^f(U) \leq P_{Th}^f \end{aligned} \tag{2.101}$$

The Eq. 2.101 can be refined for the condition that $\sum_{i=1}^{U} T_s^i$ is minimized when all T_s^i are minimized and therefore can be rewritten as:

$$\begin{aligned} &\min_{U;\tau_i} T_s^i = U\tau_i \\ &such\, that,\ P^d(U) \geq P_{Th}^d \\ &\qquad P^f(U) \leq P_{Th}^f \end{aligned} \tag{2.102}$$

A bound on the value of U can be calculated as:

$$\frac{ln(1-P_{Th}^d)}{ln(1-P_{i,min}^d)} \leq U \leq \frac{ln(1-P_{Th}^f)}{ln(1-P_{i,max}^f)} \tag{2.103}$$

The optimization problem of finding the optimal sensing timeslots can be given as:

$$\begin{aligned} &arg \max_{\tau_i} \frac{Q\left(\frac{\lambda-\tau f_s(|h_{min}|^2\sigma_s^2+\sigma_w^2)}{\sqrt{2\tau f_s}(|h_{min}|^2\sigma_s^2+\sigma_w^2)}\right)}{\tau_i} \\ &\text{such that, } U \leq \frac{ln(1-P_{Th}^f)}{ln(1-P_{i,max}^f)} \end{aligned} \tag{2.104}$$

The optimal value of the sensing time slot is calculated as:

$$\tau_i^* = \frac{1}{2f_s}\left(\sqrt{\frac{\lambda^2}{4(|h_{min}\sigma_s^2+\sigma_w^2)+1}} - 1\right) \tag{2.105}$$

Similar to the homogeneous environment case, the throughput in heterogeneous environment can be calculated as:

$$\mathbf{TH} = \left(T - \sum_{u=1}^{U} \tau_i\right) P_{available} R \tag{2.106}$$

where, R is the bandwidth of each channel and $P_{available}$ is the probability that the available channels can be found in one sensing period.

The energy consumed by this approach is always less than the traditional cooperative scheme. The throughput of this scheme is much higher than the traditional scheme as during the same sensing duration, the number of channels discovered correctly is always more.

A distributed resource allocation based energy efficient scheme is discussed in [18], for cognitive radio based wireless sensor network operating in multi-carrier environment. The discussed scheme allows subcarrier selection and power allocation among individual cognitive radio nodes. The overall goal of the scheme it to reduce the energy consumption per bit over all subcarriers while maintaining the certain level of data transmission rate.

Multiple users are allowed to share the same subcarrier as long as their Signal to Interference and Noise Ratio (SINR) is acceptable. The problem of subcarrier (channel) allocation is converted into an unconstrained optimization problem and then branch and bound method of solving optimization equations is utilized to find an optimal solution.

As discussed in Sect. 1.7.2, the system time is divided into time slots where the size of each time slot is T_s. During each time-slot the cognitive nodes exchange beacons which is utilized for synchronization purposes. The entire frequency spectrum is divided into M subcarriers of which each spectrum experience flat Rayleigh fading.

After each user senses the network and obtains the list of available subcarriers, the scheme will select subcarriers and allocate power to them such that the energy efficiency is maximized, and data rate and power constrains are satisfied. If $P_t^{(i)}$ is the transmission power allocated to sub carrier i, then the optimization problem of power allocation in distributed subcarrier can be given as:

$$\min_{P_t^{(i)}} \frac{\sum_{i=1}^{M} P_t^{(i)} + P_r}{B.\sum_{i=1}^{M} log_2\left(1 + \alpha_i P_t^{(i)}\right)} \tag{2.107}$$

Subject to,

$$\begin{aligned} &\sum_{i=1}^{M} R^{(i)} = B.\sum_{i=1}^{M} log_2\left(1 + \alpha_i P_t^{(i)}\right) \geq R_{req}, \forall i = 1, 2, \ldots, M \\ &\sum_{i=1}^{M} P_t^{(i)} = P_{max}, P_t^{(i)} \geq 0, \forall i = 1, 2, \ldots, M \end{aligned} \tag{2.108}$$

where P_r is the total circuit power consumption, B is bandwidth of each subcarrier, P_{max} is the maximal power bound, R_{req} is the target data rate, α_i is channel state information of subcarrier i and M is the number of available channels.

The optimization Eq. 2.107 is not quasi-concave/convex and therefore to solve it, the constraints from this Equations are removed initially. Once the optimal solution to unconstrained problem is known, the initial constraints of Eq. 2.108 can be examined again.

On removing the constrains from the Eq. 2.107, the optimal transmission power $P^* = P_t^{(i)^*}, \forall i = 1, 2, \ldots M$ can be given as [18]:

$$
\begin{aligned}
P_t^{(i)^*} &= \max\left[B.log_2^e.\zeta^* - \left(\frac{1}{\alpha_i}\right), 0\right] \\
\zeta^* &= \min_{P_t^{(i)^*}} \frac{\sum_{i=1}^{M} P_t^{(i)} + P_r}{B.\sum_{i=1}^{M} log_2\left(1 + \alpha_i P_t^{(i)^*}\right)}
\end{aligned}
\tag{2.109}
$$

where ζ^* is optimal energy per bit.

Multiple solutions exist for Eq. 2.109. The constraints of Eq. 2.108 are examined. For two constraints, a total of four possibilities exists:

(i) If $\sum P_t^{(i)^*} \leq P_{max}$ and $\sum R^{(i)} \geq R_{req}$, then Eq. 2.109 is satisfied.

(ii) If $\sum P_t^{(i)^*} \geq P_{max}$ and $\sum R^{(i)} \leq R_{req}$, it implies that even after allocating all the channels which exceeds the maximal power bound, the target data rate requirement is not met. Therefore in such condition, there exist no feasible solution to the problem.

(iii) If $\sum P_t^{(i)^*} \leq P_{max}$ and $\sum R^{(i)} \leq R_{req}$, it implies that the target data rate requirement is not met. Since power allocated to subcarrier still hasn't reached the maximal bound, power should be increased till the maximal power bound is reached. The minimal required additional power to achieve the data rate of R_{req} can be given as:

$$
\Delta P_{min} = M\beta\left[e^{\frac{R_{req} - \sum_{i=1}^{M} R^{(i)^*}}{Blog_2^e M}} - 1\right]
\tag{2.110}
$$

If even after this additional power, if $\sum P_t^{(i)^*} + \Delta P_{min} \geq P_{max}$, then there exist no feasible solution to the Eq. 2.107 with constraints in Eq. 2.108.

(iv) If $\sum P_t^{(i)^*} \geq P_{max}$ and $\sum R^{(i)} \geq R_{req}$, it implies that total power allocation and data rate both exceeds their upper bound. In this case, the power allocated in subcarrier is decreased so as to make it equal to or below the P_{max} value.

The optimal subcarrier selection and power allocation is done independently by all new users, which may lead to multiple users deciding to use the same sub-carrier and thus causing interference on it. Therefore a power control scheme based on distributed co-channel interference mitigation is required. If $N^{(i)}$ new users want to use the same subcarrier i at the same time, then the following distributed power control iteration will converge if a feasible solution exist.

$$P_n^{(i)}(k+1) = \frac{\gamma_n^{req,(i)}}{\gamma_n^{(i)}(k)} P_n^{(i)}(k) \tag{2.111}$$

Here $P_n(i)(k)$, $\gamma_n^{(i)}(k)$ are transmission power and measured SINR of new user n on subcarrier i in step k, respectively. $\gamma^{req,(i)}$ is the target SINR and is calculated from the optimal allocation data rate $R_n^{(i)*}$.

Each node only needs to know its received SINR at the receiver and use it to change its transmission power. The receiving node returns this as the feedback through a control channel. If the power control algorithm cannot resolve the interference issues then it should be resolved by the medium access control (MAC) scheme which needs to be developed. Each user can only detect interference from other user after they have started transmitting. Also it may take a lot of steps for the iterative algorithm to converge which can waste a lot of time.

In [19], an energy efficient source and channel sensing (JSCS) scheme is discussed for cognitive radio sensor networks. The overall work of the scheme is divided into two tasks: *ambient sensing* for vacant channels and *application sensing* for collecting application oriented information and delivering it to the access point/gateway. Both of these tasks consume ample amount of energy, therefore the objective of the discussed scheme is to reduce the total energy consumption while maintaining a bound on the distortion of the application-specific source information.

The ambient oriented channel sensing performs periodic channel sensing so as to detect the available channels for transmission. This helps to detect and determine the energy distribution of the spectrum and share the information with other cognitive devices in the network for cooperation. The application oriented source sensing involves invoking a sensor in the cognitive sensor node (sensors like temperature, humidity, luminosity etc.) and passing the obtained information to the access point/gateway.

The cognitive sensor network perform the two tasks of channel sensing and application sensing simultaneously. The objective of the system is to minimize total consumed energy in the system by doing a trade-off between the energy consumed while doing the channel sensing and while doing the application sensing. If excessive resources are given to application sensing then even after the collection of precise application information, it cannot be transmitted in due time due to the lack of channel sensing information. If channel sensing is given excessive resources then channel availability can be determined reliably but then the application related information might not be delivered at an acceptable rate to the access point. Thus there exist a trade-off between these two tasks.

The system time is divided into recurring superframes. The time period of each superframe is T and is short enough for a primary incumbent activity to remain unchanged in this timeframe. At the start of each superframe each node in the cognitive sensor node performs channel sensing via detection of N samples on the spectrum. The cognitive sensor network operates on carrier frequency f_c and bandwidth W and performs N sample of this channel at the starting of each superframe.

The received signal at any cognitive node can be expressed as shown in Eq. 2.91. Here $r(m)$ is the received energy at m-th sample, with $s(m)$ as the primary users' transmitted signal with zero mean and σ_s^2 variance while $w(m)$ as a Gaussian process with zero mean and σ_w^2 variance. Therefore the energy detector for N samples can be given as:

$$T(r) = \frac{1}{N}\sum_{m=1}^{N}|r(m)|^2 \tag{2.112}$$

For sufficient number of samples, the probability density function (T(r)) via central limit theorem can be expressed as:

$$T(r) = \begin{cases} \mathcal{N}(\mu_0, \sigma_0^2), & H_0 \\ \mathcal{N}(\mu_1, \sigma_1^2), & H_1 \end{cases} \tag{2.113}$$

As discussed in Eq. 2.91, H_0 and H_1 are hypothesis for the absence and presence of the primary user on the given channel, respectively. Here the primary user transmitted signal is modelled as MPSK complex signal.

If p_{false} and p_{detect} are the probability of false alarm and detection respectively, then the probability of missed detection is given as, $p_{miss} = 1 - p_{detect}$, and the probability that the cognitive node is allowed to transmit on a given superframe is calculated as:

$$p_t' = \Big(1 - p_{false}\Big)p(H_0) + \Big(1 - p_{detect}\Big)p(H_1) \tag{2.114}$$

where $p(H_0)$ and $p(H_1)$ are inactive and active probabilities of primary incumbents.

If collision probability is given as $p_{collision} = p_{miss}p(H_1)$, then the effective transmission probability for a cognitive node is given as:

$$p_t = p_t' - p_{miss}p(H_1) \tag{2.115}$$

If the energy consumed by the detection on each sample is denoted as E_{sample}, then the average channel sensing power consumption on a single superframe can be given as:

$$P_{ch-sensing} = \frac{E_{sample}N}{T} \tag{2.116}$$

The transmission power with respect to effective transmission probability p_t can be given as:

$$P_{ch-sensing}(p_t) = \left[\frac{Q^{-1}\left(1 - \frac{p_t}{p(H_0)}\right) - \sqrt{\frac{2\sigma_s^2}{\sigma_w^2} + 1}\,Q^{-1}(1 - p_{miss})}{\frac{\sigma_s^2}{\sigma_w^2}} \right]^2 \frac{E_{sample}}{T} \tag{2.117}$$

where Q^{-1} is the inverse tail probability of the standard normal distribution.

The application sensing power comprises of energy consumption in sensor activation, source-channel coding and transmission. The energy required for correct delivery of each bit can be given as:

$$E_{bit} = N_0 W \left[\frac{2^{(R_{channel}/W)} - 1}{R_{channel}} \right] \tag{2.118}$$

where $R_{channel}$, N_0 and W are maximum channel capacity, unilateral noise power spectral density and channel bandwidth, respectively.

Therefore the average application sensing power can be given as:

$$P_{app-sensing}(p_t) = p_t' E_{bit} R_{channel} \tag{2.119}$$

For a multi-terminal source coding system, the Eq. 2.119 can be rewritten as:

$$P_{app-sensing}(p_t) = N_0 W\Big(p_t + p_{miss}\,p(H_1)\Big)\left[\left(\frac{\left(\frac{\sigma_s^2}{D}\right)^{1/K}}{1 - \left(\frac{\sigma_w^2}{K}\right)\left(\frac{1}{D} - \frac{1}{\sigma_s^2}\right)}\right)^{\frac{L}{2p_t W}} - 1\right] \tag{2.120}$$

where K is number of nodes and D is distortion. σ_s^2 and σ_w^2 are the variance of the source and the variance of noise, respectively.

If the total power of a cognitive node is constrained then a trade-off is done between the application sensing and channel sensing. The effective transmission probability is the key that connects the two parameters. Therefore a joint transmission model is considered where total energy consumed can be given as:

$$P_{total}(p_t) = P_{app-sensing}(p_t) + P_{ch-sensing}(p_t) \tag{2.121}$$

If the probability of false alarm is kept below 0.5 then $P_{total}(p_t)$ forms a convex curve with respect to p_t and therefore an optimal solution exist for minimal total power consumption. By utilizing the Eq. 2.121, an optimization can be done over the application sensing and channel sensing power so as to satisfy a given distortion constraint of the source coding.

Table 2.4 Comparison of simulation, emulation and real-time test beds

	Simulation	Emulation	Real-time test beds
Implementation	Based on probabilistic and analytical modelling of components	Based on simulated and some real components	Based on realtime system components
Behavior	Results are macroscopic and moderate in terms of accuracy	Results are quite accurate and near-realistic	Results are realistic
Replication	Replication is improbable due to varying parameters	Replication to near value can be achieved	Replication can be achieved via utilizing the same components
Realization complexity	Minimal	Modest	High
Computation complexity	Very high	High	Minimal
Cost	Minimal	Modest	Expensive

2.4 Cognitive Radio Platforms

The performance of a cognitive radio system may be evaluated via either three method of implementations: simulation, emulation and real-time test beds. Simulation allows performance analysis of the overall system into an abstract level while providing macroscopic facts and figures. Emulation involves utilization of real-time (real-system) component integrated into a simulation system thereby providing a more accurate form of analysis. The small-scaled deployment of a real-time test-bed allows providing realistic results and accurate microscopic behavior of the overall system [20]. Each of these techniques have their pros and cons. Table 2.4 provides pros and cons of each of these techniques.

In wireless environment, simulation model normally misses the complex channel characterization. The wireless channels are complex in terms of shadowing, fading, noise,[1] interference and multi-path propagation. Collective modelling of these parameters poses deviation from realistic behavior which can only be solved via using realistic components in the evaluation system. Emulation tend to solve this problem by utilizing realistic components instead of complex models in the evaluation system. This provides more realistic results at the cost of utilization of realistic components. To provide accurate realistic results, real-world platform need to be deployed as test-beds [21, 22]. Therefore development of platforms for test-bed is important for evaluation of cognitive protocols.

[1]Here we refer noise as intrinsic noise, which is generated by the communication device itself; while we refer interference as the extrinsic noise, which the communication device receive from other unintended signal sources.

There has been various advances in realizing a real-time, at-scale validations of the cognitive radio technologies [22–24]. To realize such platform, generally a frequency agile flexible radio is preferred which is capable of generating multitude of waveforms controlled via software. This is normally achieved via a software defined radio. These platforms tend to accommodate design and development specifications that cover up the current scenario and scenarios yet to come. Most of the platforms are designed to be modular, so as to allow ease in development, debugging and multiple iterations of verification. The modular design also allows the platform to be structured and clean slate for easy learning and additional component building.

2.4.1 From FPGAs to Software Defined Radio

One of the requirements is that cognitive radio systems change its behaviour according to its context autonomously, based on experience and learning. Therefore the system should be able to interact with the reconfigurable network node. Also, since real time sensing is computationally intensive, it requires for high demanding digital signal processing algorithms.

Until recently, reconfigurable logic and Field Programmable Logic Array (FPGA) chips have been used primarily for prototyping. However, it is now being realized that reconfigurable architectures are a separate family in their own right, just as processors and Application specific Integrated Circuits (ASIC) are with unique properties that can be taken advantage of. General Purpose Processors are highly programmable platforms, however the performance is limited. The basic building block of FPGA is a reconfigurable cell which is implemented using Look-Up-Table (LUT), Programmable Logic Array (PLA) or Memory. LUT or memory-based cell is configured by writing the result of the combinational logic to the cell [25]. That is some modules need to be selectively reconfigured or reassigned for new mode of operation in software defined radio. Therefore, reconfigurable architectures will give greater flexibility in such an environment [24].

2.4.2 From Software Defined Radio to Cognitive Radio

Cognitive radio was first conceptualized from software defined radio by J. Mitola while his work on the field of cognitive radio [26]. The capabilities of SDR to tune to any frequency, selecting any band filter, generation of any modulation via software allows the realization of cognitive radio. It is well known that CR are computationally intensive systems that implement highly demanding digital signal processing algorithms on different platforms. Design of mutli-mode platforms using FPGA based SDR would be ideal for cognitive radio architectures. The possibility of such a system has been investigated in detail [27]. A comparative study of SDR based cognitive radio platform is done in [23].

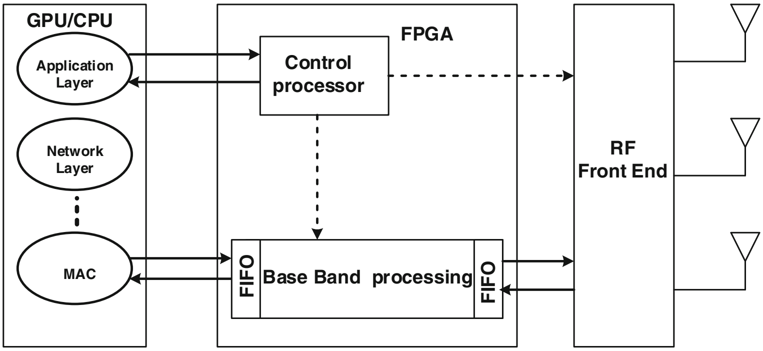

Fig. 2.3 Block schematics of SDR

The component therefore which translate SDR into Cognitive Radio is Cognition. Apart from cognition, other software components like protocol stacks are required to realize the full functioning cognitive radio network. In the next section we will discuss about different SDR control software that are utilized to emulate the cognition behavior.

Figure 2.3 shows a basic block schematic of Software defined Radio (SDR) architecture. Fundamentally, the idea of SDR is to replace most of the analog signal processing in the transceivers with digital signal processing in order to provide the advantage of flexibility through reconfiguration or reprogramming. This will enable different air-interfaces to be implemented on a single generic hardware platform. A multi-mode SDR handset with dynamic reconfigurability has the promise of integrated services and global roaming capabilities. In order to become reality, the SDR technology requires more research not only from the perspective of wireless systems, but also in the areas of novel computing architectures, embedded systems, and design methodologies to realize an all-digital reprogrammable radio.

2.4.3 SDR Software

Software controls are required for providing control over the SDR hardware. These software controls can be utilized to program a SDR to behave like a radio with cognition. Some of these SDR software are: GNU Radio Software, Pothosware, IRIS, OSSIE [23].

Figure 2.4 shows the block diagram of an SDR with an application software controlling it. The application software controls the radio behavior including frequency selection, modulation, bandwidth, gain, noise filtering etc.

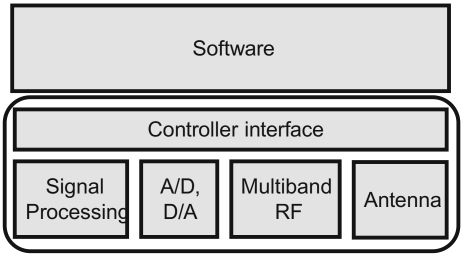

Fig. 2.4 Application software on top of SDR

2.4.3.1 GNU Radio

GNU Radio is a free, open source development toolkit for SDR [28]. It is licensed under GNU GPL version 3. It is widely used in academia and in commercial ecosystem. The toolkit comprises of different modules each representing an individual step in signal processing. The overall functionality of the toolkit is achieved by applications which renders themselves as a 'flowgraphs'. These flowgraphs interconnect different signal processing modules via a vertex which originates from a source module and terminates at a sink module, thus representing the flow of the data. The application flowgraph can be written in either C++ language or python language, while the core signal processing modules are written in C++ language. The GNU Radio comes with a graphical UI and called as GNU Radio Companion. The GNU Radio Companion provides drag-n-drop functionality to configure a particular application, thus allowing a novice user without having any programming knowledge to utilize this toolkit. GNU Radio is natively available for Linux, but other alternate build methods are available for Windows and MAC OS.

2.4.3.2 Pothosware

Pothosware is an open source software for interconnected processing blocks [29]. It provides multiple toolkits for different purposes. Pothos provide toolkit for interfacing with SDR hardware, provides another toolkit for real-time signal analysis and multiple toolkits signal processing blocks. Pothos also allows to integrate different components from other SDR softwares into the pothosware. The Pothos SDR toolkit provides a SDR source block for receivers and SDR sink block for transmitters. For interfacing with different hardwares, the Pothos project wraps SoapySDR project. SoapySDR is an open source C/C++ library for interfacing with different types of SDR hardware. The library is vendor neutral, implying that it is compatible with different types of hardware.

There exist multitude of software defined radio platforms, including Universal Software Radio Peripheral (USRP), eFalcon, Microsoft Research Software Radio (Sora), Cognitive Baseband Reconfigurable Radio (COBRA), Berkeley Emulation Engine (BEE2), and Wireless open-Access Research Platform (WARP). To keep it short and interesting, we will discuss one platform from traditional SDR: *WARPnet* and one unconventional SDR hacked platform: *RTL-SDR*.

2.4.4 WARPnet

The platform is build from ground up by integrating individual blocks at the lower layer to realize a whole cognitive radio functional platform [30]. The platform allows the developers to fine control the lower layer functionalities while allowing conventional users to deploy the platform for field testing. It utilizes two wide band radio boards to support desired transmission characteristics. One is MAX2829 which is dual band 2.4 and 5 GHz radio, while the other radio utilized is AD9352 which is a 2.5 GHz band radio. The radio board utilizes multiple dedicated control interfaces (channels) for coordination between different boards.

The WARPnet board utilizes a Xilinix Virtex-4 FX series FPGA. This FPGA provides the processing power required for doing complex computation over software for the working on real-time algorithms for wireless communication. The board has Marvell Alaska 88e1111 Gigabit Ethernet transceiver for bridging wireless to wired network infrastructure. The board is capable of supporting 2 GB of RAM. The board also has two 6.5 Gbs SATA bus for connecting hard drives for data storage. The board also support other hardware features which are not discussed in the book. Readers are suggested to go through the WARPnet documentation to get the detailed specifications.

Maxim MAX2829 RF transceiver provide analog baseband interfaces and handles RF translations. The transmitter has 16 bit dual input Analog Devices' AD9777 DAC. The receiver has dual 14bit output AD9248 ADC. The transceiver provide software control over transmit power, input and output filter etc. via SPI interface.

Analog device's AD9352 transceiver chip integrates the ADC and DAC along with the RF transceiver, filters and amplifiers. A dedicated Spartan-3 AN FPGA is utilized for providing control and data interface along with the processing algorithms for radio.

The board also utilizes a backdoor board which provides dedicated control channel and a control hub. This allows as a central controller to control all boards in the network. This includes remote programming of deployed boards in the network. The backdoor interface utilizes Ethernet and 900 MHz long range wireless interface for communication. The board utilizes GPS module to synchronize time with other boards for the purpose of MAC layer algorithms.

WARPnet architecture utilizes three functional cores: FPGA logic, the PowerPC and the Axis Etrax SoC. The high sped FPGA implements the physical layer of the cognitive radio via utilization of complex designs in HDL. The MAC layer is implemented in C/C++ language via the PowerPC as it has direct access to the lower physical layer. The overall cognitive control of the WARPnet node is done via the Axis Etrax SoC which provide control channel and monitoring of the network. In a WARPnet network, each node is connected and managed by a central entity called as WARPnet controller. The WARPnet controller monitors the system characteristics and provides a system wide view of the whole deployment. This allows easy calculation of the performance of the grid.

2.4.5 RTL-SDR

RTL-SDR is a software defined radio that is commonly found in the DVB-TV tuner dongle which utilizes Realtek RTL2832U chipset [31]. It was found that the RTL2832U chipset allows transferring raw I/Q samples to the host, thus allowing demodulation to be done in the software. The DVB-TV tuner is a cheap dongle which acts as a receiver, therefore with proper tools it can be utilized as a SDR receiver. The dongle utilizes a combination of a tuner, sampler and USB interface modules. The RTL2832U supports tuners at Intermediate Frequency of 36.125 MHz, low-Intermediate Frequency of 4.57 MHz, or Zero-Intermediate Frequency.

The RTL2832U provides 8 bit ADC sample with maximum theoretical sampling rate of 3.2 MS/s along with USB interface for pumping data to the host machine. Although the maximum bandwidth is 3.2 MHz, there are sample losses. The largest sample rate successfully without sample loss is 2.8 MHz. Different tuners provide different types of signal reception in different range of frequencies. Table 2.5 provides the list of tuners which is usually utilized with RTL2832U [32].

Although this hardware does not provide any transmission capability, it provides sensing capability which can be utilized as a starting point for people entering into cognitive radio field. The hardware does not provide accuracy to industry standards, but with proper noise/EMI protection, heat sinks and stable oscillator, a high level of accuracy can be obtained. The librtlsdr library provides generic driver interface for RTL2832U chipset [32]. Therefore most of the software defined radio software can be interfaced with this dongle via the help of librtlsdr library. The applications include FM radio, GSM listening, ADSB receiver, GPS receiver, AIS receiver etc.

Table 2.5 Different tuners with supported frequency bands

Tuner	Frequency band
Rafael Micro R820T	24–1766 MHz
Rafael Micro R828D	24–1766 MHz
Fitipower FC0013	22–1100 MHz
Fitipower FC0012	22–948.6 MHz
FCI FC2580	146–308 MHz and 438–924 MHz (gap in between)
Elonics E4000	52–2200 MHz with a gap from 1100 to 1250 MHz

2.5 Discussion

In this chapter various cognitive radio network technologies for QoS provisioning and energy management related works were reviewed and background materials were described which are essential to appreciate the following chapters. First, various spectrum management strategies were discussed and requirements of media access control techniques were introduced. Towards the end of the chapter, various energy management techniques were also discussed. Further a brief overview on different cognitive radio platforms was also described. The capabilities of these platform indicates that realization of the cognitive radio technologies. In the following chapters we focus exclusively on a framework which provides QoS provisioning and energy management for cognitive radio network, encompassing the technologies of spectrum management, media access scheme, energy management and also self-coexistence.

References

1. FCC. (2003). Revision of parts 2 and 15 of the commissions rules to permit unlicensed national information infrastructure (U-NII) devices in the 5 GHz band. *Memorandum opinion and order* (Vol. 21).
2. Atakan, B., & Akan, O. B. (2007). Biologically-inspired spectrum sharing in cognitive radio networks. In *2007 IEEE Wireless Communications and Networking Conference* (pp. 43–48).
3. Eletreby, R. M., Elsayed, H. M., & Khairy, M. M. (2014). Optimal spectrum assignment for cognitive radio sensor networks under coverage constraint. *IET Communications*, *8*(18), 3318–3325.
4. Niyato, D., Member, S., Hossain, E., & Member, S. (2008). Competitive pricing for spectrum sharing in cognitive radio networks: Dynamic game, inefficiency of nash equilibrium, and collusion. *IEEE JSAC*, *26*(1), 192–202.
5. Song, Y., Fang, Y., & Zhang, Y. (2007, November). Stochastic channel selection in cognitive radio networks. In *Global Telecommunications Conference, 2007. GLOBECOM '07. IEEE* (pp. 4878–4882).
6. Talat, S., & Wang, L.-C. (2009). Qos-guaranteed channel selection scheme for cognitive radio networks with variable channel bandwidths. In *International Conference on Communications, Circuits and Systems, 2009. ICCCAS 2009* (pp. 241–245).
7. Cai, L. X., Liu, Y., Shen, X., Mark, J. W., & Zhao, D. (2010). Distributed QoS-aware MAC for multimedia over cognitive radio networks. In *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE* (pp. 1–5).
8. Wang, P., Niyato, D., & Jiang, H. (2010). Voice-service capacity analysis for cognitive radio networks. *IEEE Transactions on Vehicular Technology*, **59**, 1779–1790.
9. Timmers, M., Pollin, S., Dejonghe, A., Van der Perre, L., & Catthoor, F. (2010). A distributed multichannel MAC protocol for multihop cognitive radio networks. *IEEE Transactions on Vehicular Technology*, *59*, 446–459.
10. Cordeiro, C., & Challapali, K. (2007). C-MAC: A cognitive MAC protocol for multi-channel wireless networks. In *2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2007. DySPAN 2007* (pp. 147–157).
11. Zhao, Y., Song, M., Xin, C. (2013). FMAC: A fair MAC protocol for coexisting cognitive radio networks. In *INFOCOM, 2013 Proceedings IEEE* (pp. 1474–1482).
12. IEEE 802.11 Standard (2012). IEEE 802.11 working group, wireless LAN medium access control (MAC) and physical layer (PHY) specifications.

13. Le, L., Hossain, E. (2008). A MAC protocol for opportunistic spectrum access in cognitive radio networks. In *Wireless Communications and Networking Conference, 2008. WCNC 2008. IEEE* (pp. 1426–1430).
14. Mishra, V., Tong, L. C., Chan, S., & Mathew, J. (2011). MAC protocol for two level QoS support in cognitive radio network. In *2011 International Symposium on Electronic System Design (ISED)* (pp. 296–301).
15. Wang, S., Wang, Y., Coon, J. P., & Doufexi, A. (2012). Energy-efficient spectrum sensing and access for cognitive radio networks. *IEEE Transactions on Vehicular Technology*, *61*, 906–912.
16. Maleki, S., Pandharipande, A., & Leus, G. (2011). Energy-efficient distributed spectrum sensing for cognitive sensor networks. *IEEE Sensors Journal*, *11*, 565–573.
17. Liu, Y., Xie, S., Zhang, Y., Yu, R., & Leung, V. C. M. (2012). Energy-efficient spectrum discovery for cognitive radio green networks. *Mobile Networks and Applications*, *17*(1), 64–74.
18. Gao, S., Qian, L., & Vaman, D. R. (2008). Distributed energy efficient spectrum access in wireless cognitive radio sensor networks. In *2008 IEEE Wireless Communications and Networking Conference* (pp. 1442–1447).
19. Zhang, H., Zhang, Z., Chen, X., & Yin, R. (2011). Energy efficient joint source and channel sensing in cognitive radio sensor networks. In *2011 IEEE International Conference on Communications (ICC)* (pp. 1–6).
20. Zimmermann, A., Günes, M., Wenig, M., Ritzerfeld, J., & Meis, U. (2006). Architecture of the hybrid MCG-mesh testbed. In *Proceedings of the 1st International Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization, WiNTECH '06* (pp. 88–89). New York: ACM.
21. Tan, K., Wu, D., Chan, A., & Mohapatra, P. (2010). Comparing simulation tools and experimental testbeds for wireless mesh networks. In *2010 IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks (WoWMoM)* (pp. 1–9).
22. Thompson, J. J., Hopkinson, K. M., & Silvius, M. D. (2015). A test methodology for evaluating cognitive radio systems. *IEEE Transactions on Wireless Communications*, **14**, 6311–6324.
23. Masonta, M. T., Mzyece, M., & Mekuria, F. (2012). A comparative study of cognitive radio platforms. In *Proceedings of the International Conference on Management of Emergent Digital EcoSystems, MEDES '12* (pp. 145–149). New York: ACM.
24. Gustafsson, O., Amiri, K., Andersson, D., Blad, A., Bonnet, C., Cavallaro, J. R et al. (2010) Architectures for cognitive radio testbeds and demonstrators 2014; an overview. In *2010 Proceedings of the Fifth International Conference on Cognitive Radio Oriented Wireless Networks and Communications* (pp. 1–6).
25. Lotze, J., Fahmy, S. A., Noguera, J., Doyle, L., & Esser, R. (2008). An FPGA-based cognitive radio framework. In *Signals and Systems Conference, 2008. (ISSC 2008). IET Irish* (pp. 138–143).
26. Mitola, J., III, & Maguire, G.Q., Jr. (1999). Cognitive radio: Making software radios more personal. *IEEE Personal Communications*, *6*, 13–18.
27. Kuon, I., Tessier, R., & Rose, J. (2008). FPGA architecture: Survey and challenges. *Foundations and Trends in Electronic Design Automation*, *2*, 135–253.
28. Blossom, E. (2004). Gnu radio: Tools for exploring the radio frequency spectrum. *Linux Journal*, *2004*, 4.
29. SoapySDR Pothosware. (2016) Retrieved April 04, 2016, from http://www.pothosware.com.
30. Gupta, S., Hunter, C., Murphy, P., & Sabharwal, A. (2009). Warpnet: Clean slate research on deployed wireless networks. In *Proceedings of the Tenth ACM International Symposium on Mobile Ad Hoc Networking and Computing, MobiHoc '09* (pp. 331–332). New York: ACM.
31. Wickert, M. A., & Lovejoy, M. R. (2015) Hands-on software defined radio experiments with the low-cost RTL-SDR dongle. In *Signal Processing and Signal Processing Education Workshop (SP/SPE), 2015 IEEE* (pp. 65–70).
32. RTL-SDR. Retrieved November 01, 2016, from http://www.rtl-sdr.com/about-rtl-sdr.

http://www.springer.com/978-3-319-45858-8

QoS and Energy Management in Cognitive Radio Network
Case Study Approach
Mishra, V.; Mathew, J.; Tong, L.C.
2017, XII, 202 p. 60 illus., 39 illus. in color., Hardcover
ISBN: 978-3-319-45858-8