

RESEARCH

Open Access



# Cognitive radio spectrum allocation using genetic algorithm

Jamal Elhachmi\* and Zouhair Guennoun

## Abstract

This paper presents the problem formulation, development, and use of a robust dynamic genetic algorithm (GA) for channel allocation in cognitive radio. This approach offers an efficient way to access available spectrum for both primary and secondary users. The proposed dynamic genetic algorithms based on the new sophisticated crossover and mutation operators ensure the validity of channels and the fast convergence to the best solution in a highly dynamic environment. Compared with existing methods, simulation results demonstrate that our approach algorithm produces satisfactory results with reduced network interference and enhance efficiently the spectrum throughput.

## 1 Introduction

With the recent growing increases of wireless devices, radio spectrum scarcity occurs and mandates the establishment of methods for developing technical description to access efficiently the available radio spectrum. For this, cognitive radio (CR) techniques [1] can be used to provide a promising solution to increase the spectrum utilization. The principle of the CR, included in the IEEE 802.22 and IEEE 802.16h norms [2], needs an alternative spectrum management, when each secondary/unlicensed user (SU) is permitted to sense and access the spectrum when the spectrum is unoccupied by primary/licensed users (PUs) [3, 4].

When a PU requests to access its own spectrum, the SUs using the same spectrum opportunistically should switch to other unoccupied spectra to protect the transmission of the PU and continue their own data delivery [4]. However, severe throughput degradation may occur in this situation [5].

In the literature, several algorithms have been proposed for channel allocation in cognitive radio, whether they are local or distributed search methods.

In this context, our work consists of presenting a robust genetic algorithm to solve the channel assignment problem in cognitive radio systems.

The objective of this problem concerns minimizing the channel interference to the primary radio users

meeting those requirements for opportunistic spectrum utilization. The use of genetic algorithm is quite appropriate in the context of cognitive radio to control and avoid interference in the channel assignment problem.

The rest of this paper is organized as follows: section 2 gives a brief analysis of existing methods for spectrum allocation used in cognitive radio. Section 3 concerns the representation and the construction of the objective function for channel allocation. In section 4, we propose the solution of the problem by using a sophisticated genetic algorithm. Then, in section 5, experimental results are presented and discussed. We conclude the paper in section 6.

## 2 Approaches for cognitive radio

Cognitive radio offers a promising solution for an efficient and full use of radio channel resources. It has attracted much research attention, and both distributed and centralized schemes have been proposed to facilitate the spectrum sharing between SUs and PUs [5]. Then, the concept of machine learning was applied to maximize capacity and dynamic spectrum access. Different learning algorithms can be used in CR networks such as Fuzzy Logic, Neural Networks, Hidden Markov Model, Genetic Algorithms, or Classification Algorithms [6, 7].

The multilayered neural networks were used to model and estimate the performances of IEEE 802.11 networks. They come in the form of a set of interconnected elementary processors that can perform the entire processing information chain. Each neuron adapts its parameters

\* Correspondence: Jamal\_elhachimi@hotmail.com  
Laboratory of Electronics and Telecommunications, Mohammadia School of Engineers (EMI), Mohamed V—Agdal University (UM5A), Rabat, Morocco

with its neighbors to achieve the objective for which they have been designed. Therefore, the neural networks can be considered as a suitable model for a cognitive radio network, and where a prompt response to the changing radio environment is required from an unlicensed user [8, 9].

Fuzzy logic is often combined with neural networks that can adapt to the environment during the evolution of a CR system. It can be applied to obtain the solution to a problem having imprecise, noisy, and incomplete input information.

Instead of complicated mathematical formulations, the fuzzy logic uses human understandable fuzzy sets and inference rules to obtain the solution that satisfies the desired system objectives. The main advantage of fuzzy logic is its simplicity. It is more suitable for real-time cognitive radio applications in which the response time is critical to system performance [7].

A genetic algorithm is a robust evolutionary algorithm that models biological processes to solve a highly complex computational problem to find optimal solutions. Crossover and mutation are two basic operators of GA. Performance of GA depends tightly on these operators, and new solutions are found based on old solutions through the use of crossover and mutation processes. Crossover and mutation perform two different roles. Crossover is a convergence operation; it is intended to pull the population toward a local minimum/maximum. Mutation is a divergence operation; it is intended to occasionally break one or more members of a population out of a local minimum/maximum space and potentially discover a better minimum/maximum space.

GA has been applied to spectrum optimization in cognitive radio networks. For example, genetic algorithms have been investigated through CR test-beds under certain controlled radio environments [9–11]. In [12], a genetic algorithm has been used to optimize the Bit Error Rate (BER) performance in cognitive radio. Unlike the traditional GA, Kaur et al., in [13], proposed an Adaptive Genetic Algorithm (AGA) to optimize QoS parameters in a cognitive radio system. The algorithm uses different crossover and mutation rates [14]. In [15], the genetic algorithm is used with two criteria: (1) maximizing the probability of detection (i.e., the capability of a SU to determine if a PU is using a certain portion of spectrum) and (2) minimizing the probability of false alarm (i.e., sensing the presence of a PU while it is not), for an optimal space allocation. An improved genetic spectrum assignment model with the consideration of interference constraints is proposed in [16], in order to reduce computational complexity, where the population of genetic algorithm is divided into two sets: the feasible spectrum assignment and the randomly updated spectrum assignment. In [17], the genetic algorithm is

compared to a Particle Swarm Optimization (PSO) algorithm, and authors conclude that in a dynamic environment, the GA requires more operations to perform than PSO and hence takes longer time.

In [14] a Chaotic Genetic Algorithm (CGA) was developed, where a chaotic sequence is used to generate the initial population and incorporate chaos through the crossover and mutation processes. In [18], A CGA based on the new chaotic map is proposed having a better distribution.

We would like to stress that each of the learning techniques has its own advantages and disadvantages, and none of them can be considered on its own to fully accomplish the realization of cognitive radio systems. All of the proposed approaches introduced so far can find solutions for radio resource management but may not produce the best ones as they have limitations in adapting parameters in a highly varying radio environment.

Fuzzy logic is a far powerful and flexible method, based on learning in transmission rate and prediction. It has potential in either specific problem-solving areas or as a part of cognitive radio system, to reduce its complexity. Fuzzy logic can approximate the solutions independently for certain input, but it does not provide accurate solutions. Other parameters could be included to predict the best radio configuration; it should establish a rule related to the specific situation in which it is used, and these rules may involve some limitations in programming.

Neural networks have a lesser need for prior knowledge; they can be used in any phase of cognition. They include extensive training to generate observed behavior, but they become unstable when constraints are necessary to account for. Their application to different scenarios needs to be analyzed for a CR; each individual node has to be provided with a pre-trained network or a set of training examples mapping observations to correct actions. For the latter case, each CR also has to know the parameters used for training the neural network so that all radios can create and reproduce the same neural network deterministically [19].

GA is well suited for multi-objective performance and non-mathematical optimization problems in cognitive radio networks as it can search for multiple sets of solutions over a large search space and can enforce constraints [20]. Fast convergence, ease of implementation, and optimization of discrete and continuous radio parameters render GA as an excellent optimization tool for making resource management decisions in cognitive radio networks [21].

One of the main issues involved in successful genetic algorithm behavior is the selection of the objective function(s) that is a non-closed form of constraints and may

be difficult to use in practice for some parameters introduced in cognitive radio algorithms.

The main advantages of the proposed algorithm in this paper regarding the other approaches might be perceived through the following criteria:

- (1) Implicit parallelism and robustness; such algorithm works on a set of solutions and can explore several parts of the solution space in parallel,
- (2) Distributed memory; the last solutions remain registered and the GA looks for idle channels to allocate new call requests, and
- (3) The algorithm can operate under the assumption of limited information exchange.

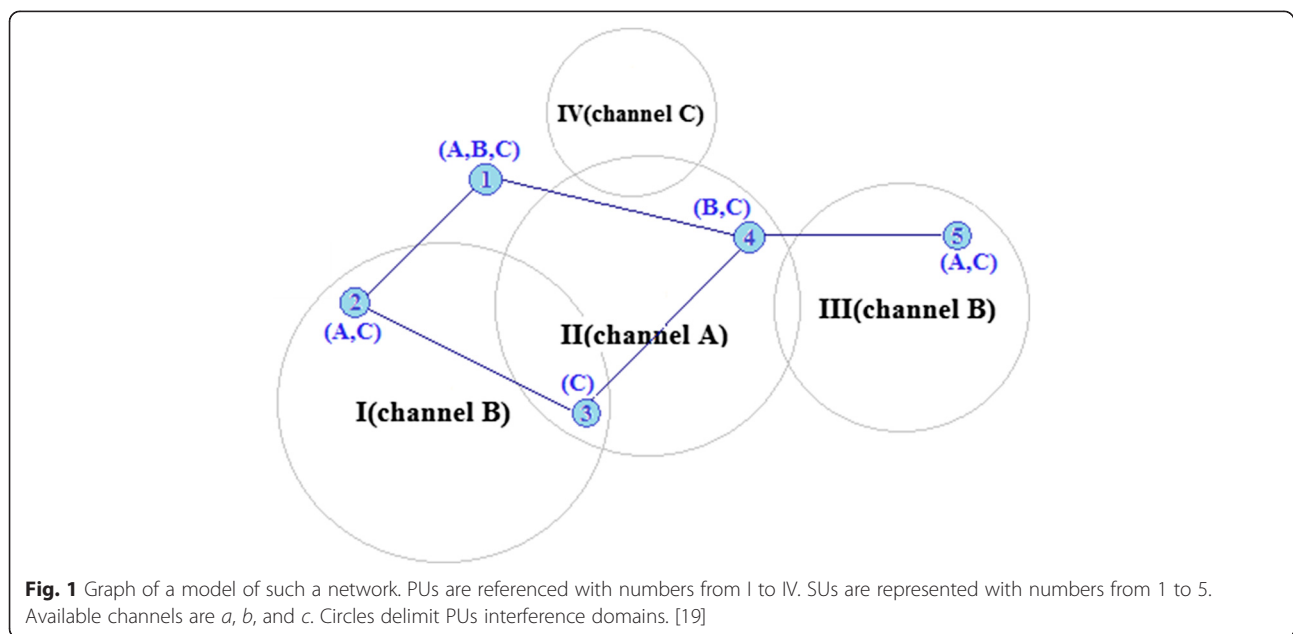
### 3 System model

We will use the model described in [5] where the channel (i.e., frequency band) allocation problem is formulated as a graph coloring problem. As in Fig. 1, the network is abstracted as an undirected graph  $G = (V, E, L)$ , where vertices ( $V$ ) represent users, edges ( $E$ ) represent interference, so that no channels can be assigned simultaneously to any adjacent nodes, and  $L$  represents the availability of channel bands at vertices of  $G$ . For the sake of simplicity, the interference graph is assumed the same for all channel bands. This can be generalized to the case where each channel band has its own interference graph, a possible scenario due to the different propagation properties in the environment associated with individual bands. Furthermore, let  $K$  be the number of available channels in  $G$ . Although it is possible that different

channels have different bandwidths, the model treats all channels with the same bandwidth [19].

The available channel band is divided into orthogonal channels of the same bandwidth using the FDMA method. It is assumed that there is a mechanism that enables wireless devices to use multiple channels to communicate at the same time.

Let  $N = |V|$  denotes the total number of secondary users. Edges will be represented by the  $N \times N$  matrix  $E = \{e_{ij}\}$ , where  $e_{ij} = 1$  if there is an edge between vertices  $i$  and  $j$  and  $e_{ij} = 0$  implies that  $i$  and  $j$  may use the same channel; note that since  $G$  is an undirected graph,  $E$  is symmetric. In a similar notation, we represent the availability of channels at vertices of  $G$  by an  $N \times K$  matrix  $L = \{l_{ik}\}$ , referred as the coloring matrix. In particular,  $l_{ik} = 1$  means that channel  $k$  is available at vertex  $i$ , and  $l_{ik} = 0$  otherwise. For instance, Fig. 1 shows four PUs, represented by numbers I to IV, using channel bands A, B, and C. These channels cannot be utilized by secondary users in the vicinity and the nodes within a certain range of each primary user cannot reuse the same channel. The same Fig. 1 shows five different secondary users, represented by numbers 1 to 5. If a secondary user is within the dashed circle of a specific primary user, it cannot access the channel band used by this primary user. For instance, node 3 is within the interference range of primary user I, who uses channel B, and is within the interference range of primary user II, who uses channel A. Therefore, it cannot reuse channels A and B. As a consequence, each node has access to a different set of bands. In Fig. 1, the available channels at vertex 1 are {A, B, C} and {A, C} at vertex 5 [19]. Figure 1 is then represented by the following matrices:



$$E = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} L = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$

Let us denote a channel assignment policy by an  $N \times K$  matrix  $S = \{s_{ik}\}$ , where  $s_{ik} = 0$  or  $1$  and  $s_{ik} = 1$  if channel  $k$  is assigned to the nodes  $i$  and  $0$  otherwise. We call  $S$  a feasible assignment if the assignments satisfy the interference graph constraint and the channel availability constraint. More specifically, for any node  $i$ , we have  $s_{ik} = 0$  if  $l_{ik} = 0$  (i.e., a channel can be assigned only if it is available at that node). Furthermore,

$$s_{ik}s_{jk}e_{ij} = 0, \forall i, j = 1, \dots, N, k = 1, \dots, K. \tag{3.1}$$

In other words, two connected nodes cannot be assigned the same channels. The objective of the resource allocation is to maximize the spectrum utilization. This problem can be formally represented as the following nonlinear integer programming problem.

$$\begin{aligned} \max_s \quad & \sum_{i=1}^N \sum_{k=1}^K s_{ik} \\ \text{s.t.} \quad & s_{ik} \leq l_{ik}, \quad s_{ik}s_{jk}e_{ij} = 0, \quad s_{ik} = 0, 1, \end{aligned} \tag{3.2}$$

For each node  $i$ , and for  $k = 1, \dots, K$ , if  $s_{ik} = 1$ , the  $l_{ik} = 1$  (i.e., a channel is assigned to  $i$ ), or else if  $s_{ik} = 0$ , the  $l_{ik} = 1$ , (i.e., a channel is available but not assigned at that node).

For all  $i, j = 1, \dots, N, k = 1, \dots, K$ . The above problem is sometimes referred to as a list multicoloring problem. When time is taken into account, a time index can be introduced into the equation where the objective is to maximize the utilization averaged over time and the three constraints are satisfied at each time instant. The corresponding decision list coloring problem is formulated below.

**Problem 3.1 [19] (*DListColor* Problem)** Given a graph  $G = (V, E, L)$  and a positive integer  $B$ , is there a solution with the same set of constraints as in (3.2)? Such that:

$$\sum_{i=1}^N \sum_{k=1}^K s_{ik} > B, \tag{3.3}$$

**3.1 Proposition 3.1**

The *DListColor* problem is NP-complete.

**PROOF:** This problem is clearly in NP since once a valid coloring assignment  $S$  is obtained, condition (3.3) may be verified in  $O(|V|K)$  time [19].

Note that a clique is a fully connected subgraph (a subnet), i.e., a clique consists of a set of nodes where any pair of its nodes has an edge in between.

We now show that the maximum clique problem can be reduced to the *DListColor* problem in polynomial time and that the maximum clique problem has a solution if and only if *DListColor* has a solution.

Let  $G = (V, E, L)$  be the undirected graph of the maximum clique problem. We construct the graph  $G' = (V', E', L')$  for our *DListColor* problem, such that  $V' = V$ , and  $E'$  is the complementary set of  $E$ . Furthermore, the color matrix  $L$  is of dimension  $|V| \times 1$ , where  $L = [1, 1, \dots, 1]^T$ . Since any pair of nodes connected in  $G$  are not connected in  $G'$  and vice versa, we cannot simultaneously assign to nodes in  $G'$  the same color if these nodes form a clique in  $G$ . Therefore, there exists a clique in  $G$  of size at least  $m$  if and only if there is no solution for *DListColor* for  $B = |V| - m$ . This reduction is obviously polynomial time as shown in [19].

**3.1.1 Color decoupling**

The list coloring problem may be reduced to a set of maximum-size clique problems when fairness is not in consideration. In other words, in the process of finding the maximum in (3.2), nodes are allowed to be assigned zero channels.

According to the logic diagram of the proposed resolution model, the channel assignment problem in a cognitive radio system is based on opportunist channel exploitation where each channel will be used by the maximum number of nodes. Therefore, the initialization of such “possible channels allocation” for a maximum clique problem as a list of channels is constructed to produce a solution for a subnetwork, which can be extended to the entire network. For each channel, a greedy assignment is determined to maximize the number of nodes assigned to this channel, and then a subnet of connected nodes is generated. Finally, nodes are connected in the original graph  $G$ .

The problem of assigning each node a set of colors (channels) may be solved by coloring the graph in sequence with individual colors:

$$\max_s \sum_{i=1}^N \sum_{k=1}^K s_{ik} \Leftrightarrow \sum_{k=1}^K \max_s \sum_{i=1}^N s_{ik} \tag{3.4}$$

The feasible assignment matrix is  $S = \{s_{ik}\}$  where  $s_{ik} = 1$  if channel  $k$  is assigned to the nodes  $i$  and  $0$  otherwise.  $S_k$  denotes the channel allocation with respect to channel (color)  $k$ . More specifically,  $S_k$  is the  $k^{th}$  column in the assignment matrix  $S$ . Note that the equality in (3.4) does not hold in general situations, e.g., a graph coloring problem that requires each node to be colored with nonempty colors. Note that when fairness is taken into account, e.g., each node has to be assigned at least one color, and then the decoupling property does not apply [19].

With fewer assigned channel to the connected nodes, our proposed algorithm can result in very unfair allocation. Nodes with greatest degrees of saturation will be first served, and their subnet obtains the maximum number of used channels. The algorithm will allow the addition of new assignments in its structure during the search. This is one of the great primary advantages of the proposed genetic algorithm.

#### 4 Spectrum allocation using genetic algorithm

A genetic algorithm (GA) is an optimization heuristic for searching very large spaces that mimic the process of natural selection. An initial population of individual solutions can be combined to make fitter solutions (Fig. 2). At each step, the GA uses three main types of rules to create the next generation from the current population: selection rules, crossover rules, and mutation rules.

In this section, we first discuss the background of GA and then the implementation of the proposed solution model on the channel allocation in CR.

##### 4.1 Main genetic algorithms

Genetic algorithms were originally developed by John Holland. They are used in order to find a solution, usually numerical solving of a given problem, without having a prior knowledge on the search space [20].

##### 4.2 Genetic operators

The main steps of the proposed genetic algorithm are individual coding, and then for each generation, child population division, niche crowding operation, and by

evaluating the fitness function the GA can exclude individuals that do not accord with constraint conditions, and then apply selection, crossover, and mutation operations to ameliorate quality of solution. The adjustment of these parameters is an important issue for GA.

- *Individual coding*

Let there be  $N$  secondary users and  $K$  channels in the system. A cell can use any available channel if such a choice satisfies the interference graph and the channel availability constraints.

In the proposed models, each gene represents a channel state and the channels of a node form a chromosome. The set of all nodes composes an individual representing an assignment policy. Hence, each individual is denoted by a vector of dimension  $V = N \times K$ , as shown in Fig. 3. Furthermore, a population can be represented by an array of strings (individuals).

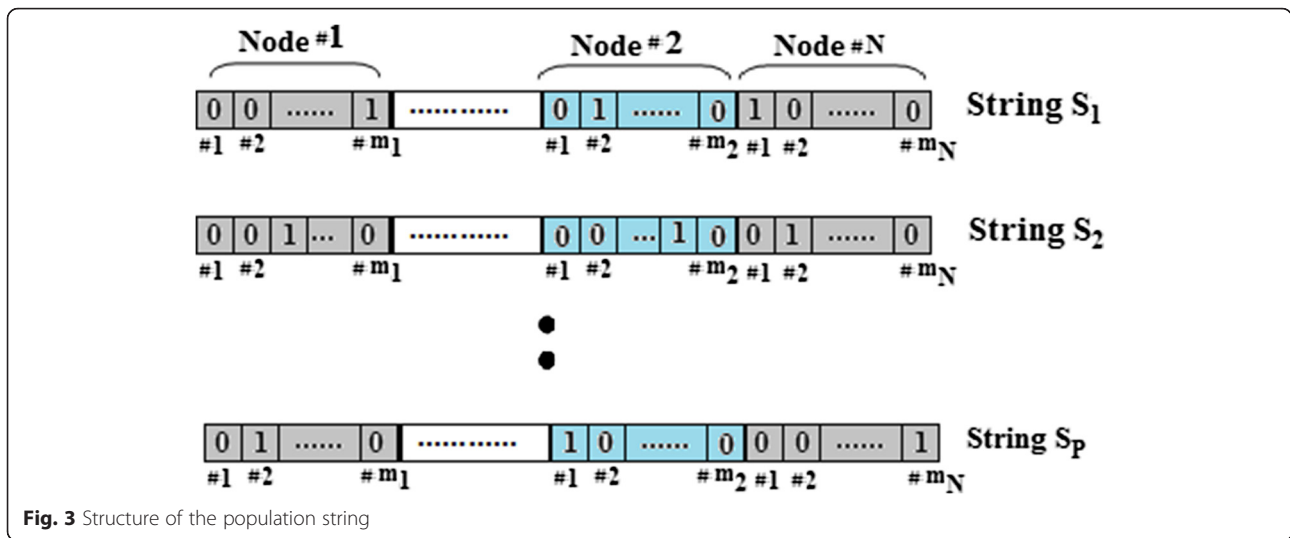
The rows of the array represent strings (of bits) in a population, and the columns represent the (the status bit of the condition of availability) available channel numbers which will be assigned. There are  $P$  strings for a population and each string has  $Q$  (possible states) channels which are the total number of available channels in the system  $Q < K$ .

A string  $S_r$  is composed of  $N$  substrings which is the number of nodes in the network. Each substring  $S_{ri}$  (for node  $i$ ) is composed of  $m_i$  genes (channel: bits of the condition of availability). Each gene is represented as a binary bit.

##### Basic Structure of a GA :

1. **Given:**
2.  $|\text{chr}|$ : number of chromosomes. /\*the size of population\*/
3.  $\text{MaxGeneration}$ : terminal criteria for while loop.
4. **begin**
5.  $k := 1$ ;
6. Generate initial population  $X^k = \{x_1^k, x_2^k, \dots, x_{|\text{chr}|}^k\}$ ;
7. Calculate  $\text{Fitness}(x_i^k)$ , for  $i = 1, \dots, |\text{chr}|$ ;
8. **do**
9.  $k = k + 1$ ;
10. **for**( $\text{iter} = 1$ ;  $\text{iter} \leq |\text{chr}|/2$ ;  $\text{iter} := \text{iter} + 1$ )  
/\*produce new generation  $X^{\text{new}}$ \*/
11. Randomly pick two chromosomes  $x_p$  and  $x_q$  from  $X^{k-1}$ ;
12. Produce two new chromosomes  $x_p^{\text{new}}$  and  $x_q^{\text{new}}$  by crossover  $x_p$  and  $x_q$  with crossover probability  $p_c$ ;
13. Perform mutation operation on  $x_p^{\text{new}}$  and  $x_q^{\text{new}}$  with mutation probability  $p_m$ ;
14. Calculate  $\text{Fitness}(x_p^{\text{new}})$  and  $\text{Fitness}(x_q^{\text{new}})$ ;
15. Insert  $x_p^{\text{new}}$  and  $x_q^{\text{new}}$  into  $X^{\text{new}}$ ;
16. **end for**
17. Pick  $|\text{chr}|$  best chromosomes from  $X^{k-1}$  and  $X^{\text{new}}$  to form  $X^k$ ;
18.  $x^* :=$  the best chromosome in  $X^k$ ;
19. **while**( $k \leq \text{MaxGeneration}$ ) && ( $\text{Fitness}(x^*) > 0$ )
20. **return**( $x^*$ );
21. **end**

**Fig. 2** Pseudo code for the GA algorithm. [4]



**Fig. 3** Structure of the population string

An  $A \times Q$  two-dimensional array is constructed to implement a number of strings (a population) as shown in Fig. 2 step 6.

- *Initial population*

According to [14], the initial population for each string is as follows:

- (1) For each node  $i^*$  with the largest number of constraints, the channel for the  $i^{th}$  node is the minimal channel interval in the available channel domain.
- (2) For all nodes with a maximum degree of saturation, choose the node  $i^*$  with the greatest degree of saturation, assign  $i^*$  with the minimum channel, and update the degrees of saturation of the neighbors of  $i^*$ .

Notice that the degree of separation  $D(i^*)$  of node  $i^*$  is the sum of the incident constraints values for each node. They are compiled once and for all. The degree of saturation  $d_j(i^*)$  in step  $j$  is used as a criterion for selecting the node that we will assign in this step. This degree is determined from the non-allowed intervals for nodes that are not yet affected (this process provides to use the minimum number of channels and reusing, to its maximum, the same channels, while avoiding deadlock)

- (3) Repeat step 2 until all nodes have the assigned channels.

- *Reproduction*

According to the target of channel allocation, the interference constraint will be presented in reutilization matrix  $C = \{C_{i,j,k} | C_{i,j,k} \in \{0, 1\}\}_{N \times N \times K}$ , where  $C_{i,j,k} = 1$ , if users  $i$  and  $j$  would cause interference if they used the

spectrum band  $k$  simultaneously. Note that constraints are spectrum band specific. Note also that two users who are constrained by one spectrum band cannot use this band simultaneously.

Considering the spectrum allocation matrix  $S = \{S_{ik}\}_{N \times K}$  which denotes the effectiveness of spectrum allocation, where  $S_{ik} = 1$  denotes that spectrum band  $k$  is assigned to user  $n$ .  $S$  satisfies all the constraints defined by  $C$ , i.e.,  $S_{i,k}S_{j,k} = 0$ , if  $C_{i,j,k} = 1$ ,  $\forall i, j < N, k < K$ .

If we consider  $B = \{b_{i,k}\}_{N \times K}$  describe the reward that a user  $i^*$  gets by successfully acquiring available spectrum band  $k^*$ , i.e.,  $b_{i,k}$  represents the maximum bandwidth/throughput that can be acquired (assuming no interference from other neighbors), and from the matrix  $R = \{l_{i,k} \cdot b_{i,k}\}_{N \times K}$  representing the throughput or the bandwidth of each channel which is available for each user to use. The resource allocation performance can be expressed as the following fitness value and the total bandwidth of the system:

$$\sum_{i=0}^{N-1} \sum_{k=0}^{M-1} S_{ik} b_{ik} \tag{3.5}$$

The fitness value of each individual is calculated according to the fitness function (objective function). After evaluation of the fitness function, a certain pair of individuals (strings) should be selected for the parents. We can use a simple biased roulette wheel to select individuals.

Each string in the population has a roulette wheel slot size in proportion to the ratio of its fitness over the total sum of fitness in the population. A random number between 0 and 1 is generated for each selection. A string is selected for reproduction if the random number is within the range of its roulette wheel slot. The copy of the selected string is gathered into a mating pool, in

which they are mated for further genetic operations. Strings with higher fitness have higher probabilities of selection so that those with higher fitness produce more offspring's than those with lower fitness in the next generation [14].

- *Crossover*

The reproduced strings in the mating pool are mated under crossover operation at random. Crossover operation is performed with a pair of substrings in the mated strings for each model [14].

The crossover technique adopted in our approach is the conflict-based crossover; this method distinguishes between genes that do not create a conflict from genes creating conflicts. Two children are generated based on the parent's genes: for each created child, values not creating conflict are duplicated on the children. If conflicts are generated by a value, it must be modified. Then, a deterministic change could be used: If a value is in conflict for a parent, and is not for the other parent, the non-conflict value is duplicated for the two children. However, if a value is in conflict with both parents, then each child will copy the value of the gene of its parent, even if it causes conflict. Figure 4 shows an example of possible crossover.

- *Mutation*

The mutation operator used in our approach is named Conflict-Based Mutation (CBM); it applies individually

for the conflict chosen channel. The CBM consists of three main steps:

- Select randomly one user from the conflict users.
- Choose the best value of available channel bands that provide more choices for the neighborhoods users.

From a purely intuitive point of view, this operator should give better results than random mutation, which consists in randomly selecting a channel of the solution and modify it on taking a new value of the channel at random [22].

### 4.3 Operation parameters

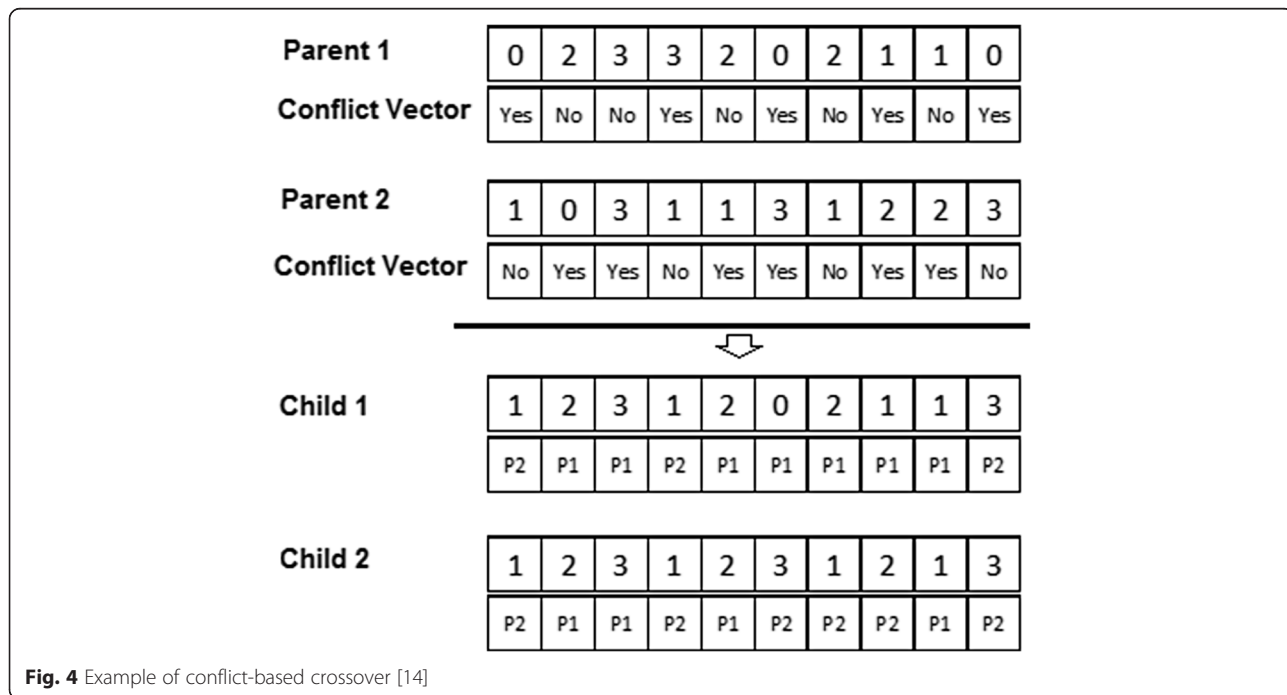
The implementation of our GA requires the adjustment of some parameters like the population size, the number of generations, and the mutation rate. The stopping criterion can be defined as the maximum number of iterations or detection of an optimum solution.

In our simulation, the population size is set to be 100. Hence, the initial population procedures are repeated 100 times.

### 5 Performance evaluation and comparison

For verification of operation, the proposed genetic algorithm was modeled using some aspects or parameters/genes and several simulations were run and results verified.

Computer simulations are used to analyze the percentage of removed interference by the algorithm when the



number of nodes and the available channels increase. The approach was designed to assign one channel for each opportunistic user under the primary users by using a uniform cost function to select the best available channel (and for its neighbors alike).

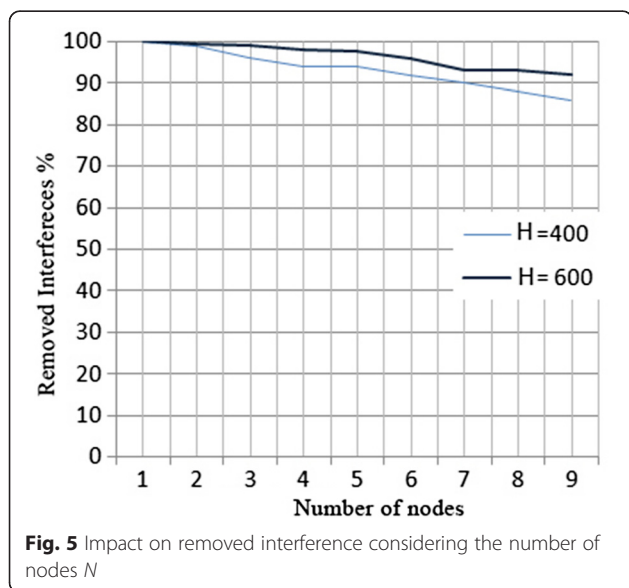
For each scenario, we consider a total number of PUs equal to 10 and then different interference topology graph instances are randomly generated, with the number of nodes varying from 10 to 50, with the interference ranges defined in a reutilization matrix  $C$ , where each channel is occupied by a primary user in a certain range, and each node within a PU's range cannot use the channel occupied by this PU. Each node in the network has its own available channel set, according to the positions of the PUs. Two nodes can use the same channel if they are not in range.

The impact of the dynamic occupation of channel by the PUs was modeled considering an average inactive time at least ten times larger than the time needed for the algorithm to reach the stopping criterion.

The channel assignment problem of this network is solved by the proposed algorithm, and analyzed, for three different scenarios:

- (1) varying the number of available channels  $K$ ;
- (2) varying the number of network nodes  $N$ ; and
- (3) varying the stopping criterion  $H$  (number of generations).

According to the observations made experimentally, we can remark that, as shown in Fig. 5, the performance in terms of removed interference (i.e., satisfied constraints during the search) decreases when the number of nodes increases with an acceptable degradation not



exceeding 6 %, and each node has less chance to find and use a forbidden channel.

In addition, the algorithm obtains good results and keeps good performance in terms of the percentage of expected goals calculated as the number of assigned nodes; the algorithm can reduce the network interference up to 90–98 %.

According to Table 1 and Fig. 6, one can deduce that the processing requirement has increased with channel increase, and then the optimal fitness decreases. An increase in the number of channels causes an increase in the Time per Generations ( $TpG$ ) and in the optimal generation. Note that all of the tests were performed on a core 2 duo CPU 2.00 GHz with 1 GB RAM memory.

Therefore, during all simulated cases, for both states before and after channel allocation, and when the stopping criterion is reached, channel availability corresponds to keeping stable the number of available channels varying from 1 to 8.

The memory process in the algorithm improves the determination of the best solution. On one hand, the reproduction process ensures that the best string in the actual population (with the highest fitness) is maintained (and copied) and put in a mating pool for the next generation. On the other hand, the algorithm uses its feedback information to adapt its response.

The algorithm will be able to adapt its response to the cognitive radio's need to adapt to a changing environment and avoid the negative effect of other arrival nodes.

Therefore, we can conclude that the GA has good performance on the users' service and channel gain, achieves channel allocation robustness, and maintains minimum network interference.

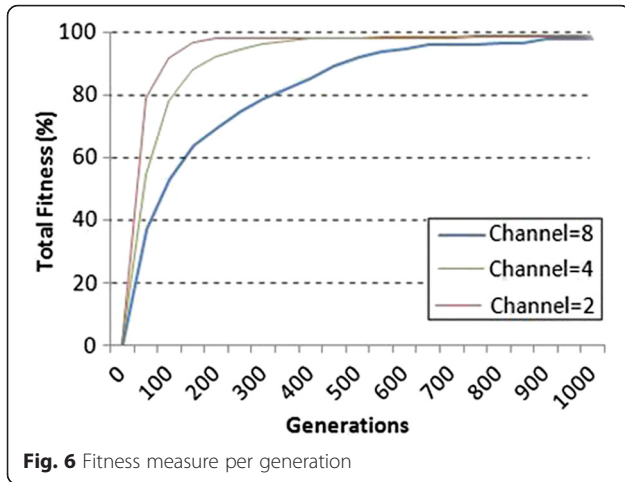
Furthermore, the GA can be easily extended to channels of different capacities by considering different versions of the dynamic channel assignment, as the dynamic assignment of coordination controls channels in cognitive radio networks (CRNs).

To examine the efficiency of the proposed algorithm in larger-scale problems, we show the evaluation results for the simulation service area considered in [23]. Nodes are randomly distributed in a  $200 \times 200$  unit square, and a certain number of PUs is generated, when each primary user occupies one channel. Each node in the

**Table 1** Resulting fitness measures and GA performance per number of channels

Number of channels	Optimal generation	Optimal fitness (%)	Time per generations (ms)
2	89	98,4	1,0
4	689	94,4	3,4
8	745	90,6	6,7





network has its own available channel set according to the positions of PUs.

The parameters used to compare the performance of various algorithms are (1) number of nodes in the network, (2) the total number of channels, and (3) interference range of primary users.

The settings of parameters are shown in Table 2 [23].

In order to evaluate such algorithm and compare its performance with existing ones, we assess the performance of the GA against three schemes in terms of the delivery rate:

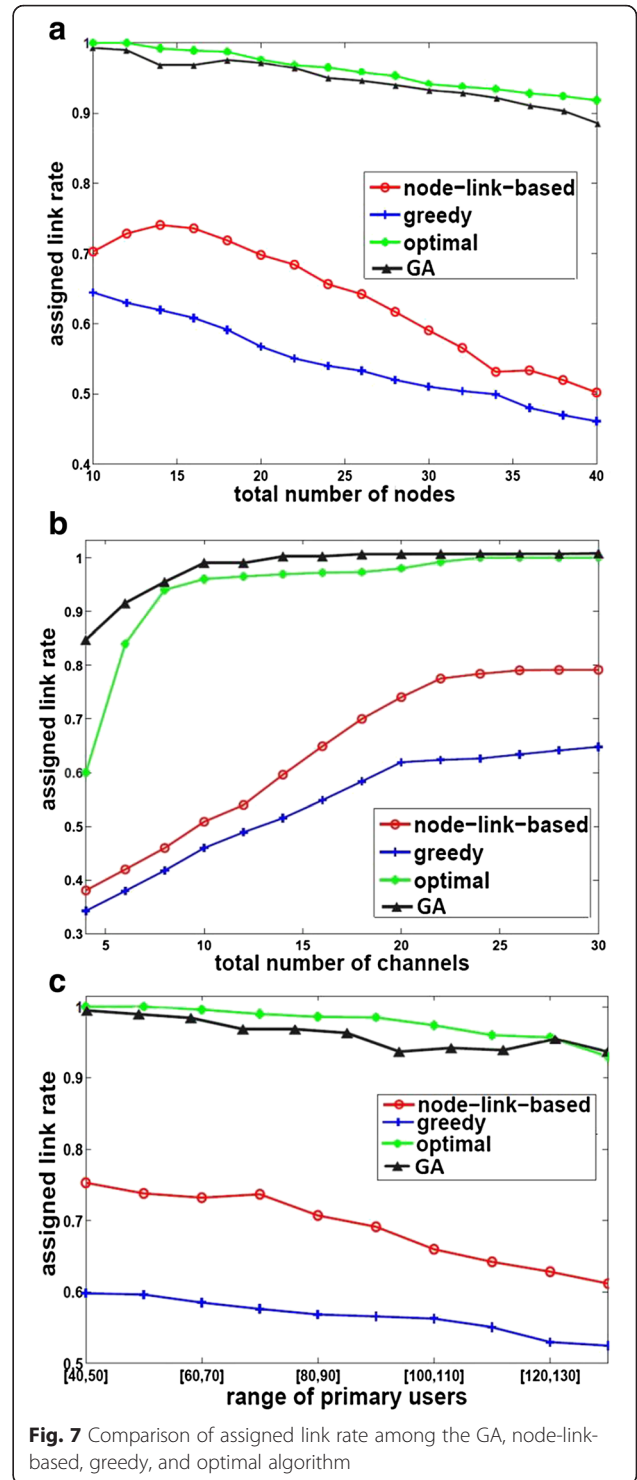
- a node-link-based [23] algorithm with a fast convergent localized protocol that assigns conflict-free channels to maximize connectivity in multi-chip Cognitive Radio Networks consisting of a cognitive source–destination node pair and multiple cooperating cognitive relays.
- an optimal algorithm which maximizes the assigned link rate, without consideration of the number of rounds [23]
- the distributed greedy algorithm in [24].

The delivery rate evolution is analyzed by varying three parameters:

In Fig. 7a, the total number of nodes varies from 10 and 40 while keeping the total number of channels as 10 and the interference range of primary users is randomly selected in [60, 70]; one can observe that the GA algorithm can find better or equivalent results compared

**Table 2** Simulation settings

Total number of nodes	[10, 40]
Communication range of each node (in meters)	[50, 70]
Total number of channels	[4, 30]
Total number of PUs	10
Interference range of PUs (in meters)	[40, 140]



with existing optimization methods. In Fig. 7b, the number of channels varies from 4 to 30 while keeping the number of nodes at 15 and interference range of PUs in [60, 70].

The results show that the GA finds better solutions compared with the optimal algorithm and the node-link-

based algorithm and achieves self-configuring radio cognitive networks without the need of additional signaling loads and changes. It shows a rapid convergence to an optimal solution and guarantees that there are idle channels, for last affectation of a new cognitive node appears on the current channel assignment.

In Fig. 7c, the number of nodes is kept at 15 and number of channels at 10, and the interference range of primary users varies. The results show that our proposed approach has improved the forecasting accuracy, and it is almost 30 % better than the other schemes.

As shown in Fig. 7, we can conclude that for each scheme, and compared to the optimal algorithm, the proposed GA can reduce efficiently the network interference up to 84–98 % with an important increase of the number of available channels and keeping it relatively constant. Particularly, since the goal is to construct a robust topology with minimizing interference, our algorithm obtain the optimality in a reasonably short execution time for these instances; it can obtain high-quality solutions and much better than those of a node-link-based and optimal algorithm.

## 6 Conclusions

Channel allocation problem is the main issue in cognitive radio network, and the deployable real-time dynamic spectrum policies are one of its powerful means to improve the global spectral efficiency to develop new wireless communications services.

In this paper, we adopt a suitable mathematical formulation for opportunistic channel allocation. While taking into account spectrum dynamicity, we propose a genetic algorithm for channel assignment problem in cognitive radio and we give the steps of its implementation. The GA is based on the new enhanced crossover and mutation operators. It has a uniform distribution compared to the classical genetic algorithm, since it is oriented to explore promising solutions in the search space.

Experimental results show the validity of this approach. It cannot only find good solutions to the radio channel allocation problem but also obtain the optimal solution in a reasonably short execution time, with high-quality solutions and better than those of the best current approaches used in node-link-based and optimal algorithm. It promotes the real-time allocation of cognitive radio spectrum.

The algorithm ensures improved performance in terms of satisfying the demands of cognitive radio's need, and the adaptability to a changing environment; in the simulation, we assume that the channel availability is dynamic during the time. So, the algorithm can recompute allocations as the topology changes and adapt the environment change.

Furthermore, the simulation researches of this paper are principally executed on the theoretical benchmarks and oriented toward qualitative analysis; therefore, it is necessary to do some experiments on real benchmarks with quantitative analysis in the next studies to verify its superiority in the actual cognitive radio system.

Another possible future work will focus on channel allocation algorithm based on genetic algorithm to achieve a better performance and a better spectrum utilization and taking profit from the advantages of other heuristics through hybrid genetic algorithms.

Our future research direction will be the implementation of this algorithm for optimizing the 802.22 norm.

### Competing interests

The authors declare that they have no competing interests.

Received: 8 July 2015 Accepted: 24 April 2016

Published online: 10 May 2016

### References

1. J Mitola, G Maguire, Cognitive radio: making software radios more personal. *IEEE Pers. Commun.* **6**(4), 13–18 (1999)
2. C.R. Stevenson, C. Cordeiro, E. Sofer, G. Chouinard, Functional requirements for the 802.22 WRAN standard. *IEEE 802.22-05/0007r47*, (2006)
3. S Haykin, Cognitive radio: brain-empowered wireless communications. *IEEE J. Sel. Areas Commun.* **23**(2), 201–220 (2005)
4. P-K Tseng, W-H Chung, P-C Hsiu, Minimum interference topology construction for robust multi-hop cognitive radio networks. *IEEE Wireless Commun. Netw. Conf. WCNC 2013*, 101–105 (2013)
5. J ElHachmi, Z Guennoun, An artificial intelligence approach for cognitive spectrum management. *Int. J. Comput. Technol. Appl.* **5**(3), 1219–1225 (2014)
6. C Clancy et al., Applications of machine, learning to cognitive radio networks. *IEEE Wireless Commun.* **14**, 4 (2007)
7. A Amraoui, B Benmammar, F Krief, FT Bendimerad, Intelligent wireless communication system using cognitive radio. *IJDPS Int. J. Distrib. Parallel Syst.* **3**(2), 91–104 (2012)
8. E. Hossain, D. Niyam, Z. Han, *Dynamic spectrum access and management in cognitive radio networks*. (Cambridge University Press, Cambridge, England, 2009)
9. C.R. Aguayo Gonzales, et al, *Design and implementation of an open-source software-defined cognitive radio testbed*, submitted to *IEEE Journal on Selected Areas of Communication (J-SAC)*.
10. T Rondeau, B Le, C Rieser, C Bostian, Cognitive radios with genetic algorithms: intelligent control of software defined radios, in *Software defined radio forum technical conference*, 2004, pp. C3–C8
11. Y Zhao, J Gaeddert, KK Bae, JH Reed, *Radio environment map-enabled situation-aware cognitive radio learning algorithms* (Proc. of the SDR '06 Technical Conference and Product Exposition, Orlando, 2006), pp. 13–17
12. R Deka, S Chakraborty, J Sekhar Roy, Optimization of spectrum sensing in cognitive radio using genetic algorithm. *Ser. Elec. Energ.* **25**(3), 235–243 (2012)
13. M Kaur, M Uddin, Optimization of QoS parameters in cognitive radio using adaptive genetic algorithm. *Int. J. Next-Generat. Netw. (IJNGN)* **4**(2), 1–15 (2012)
14. O Jegede, K Ferens, W Kinsner, A chaotic genetic algorithm for radio spectrum allocation, in *Proceedings of the International Conference on Genetic and Evolutionary Methods, Las Vegas, NV USA*, 2013, pp. 118–125
15. T Siddique, A Azam, *Spectrum optimization in cognitive radio networks using genetic algorithms* (Blenkinge Institute of Technology, Sweden, 2010)
16. F Ye, R Yang, Y Li, Genetic spectrum assignment model with constraints in cognitive radio networks. *IJ. Comput. Netw. Inf. Secur.* **4**, 39–45 (2011)
17. V Subramanian, S Rimal, Dynamic resource allocation algorithms for cognitive radio Systems, in *BTech thesis*. Engineering and Technology, Department of Electronics and Communication Engineering, code 2244, 2011
18. J. Mingjun, T. Huanwen, Application of chaos in simulated annealing. **21**(4), 931–944 (2004)
19. W Wang, X Liu, H Xiao, Exploring opportunistic spectrum availability in wireless communication networks, in *Proc. IEEE Vehicular Technology Conf. (VTC)*, 2005

20. L Mé, *Audit de sécurité par algorithmes génétiques*. PhD Thesis (University of Rennes 1, France, 1994)
21. E Adamopoulou, K Demestichas, M Theologou, Enhanced estimation of configuration capabilities in cognitive radio. *IEEE Commun. Mag.* **46**(4), 56–63 (2008)
22. JK Hao, R Dorne, Study of genetic search for the frequency assignment problem, in *Artificial Evolution AE'95* (Lecture Note in Computer Science, Best France, 1996), pp. 333–344
23. J Wu, Y Dai, Y Zhao, Local channel assignments in cognitive radio networks, in *Proceedings of 20th International Conference on Computer Communications and Networks (ICCCN)*, 2011
24. W Wang, X Liu, List-coloring based channel allocation for open spectrum wireless networks, in *Proc. of IEEE VTC*, 2005

**Submit your manuscript to a SpringerOpen<sup>®</sup> journal and benefit from:**

- ▶ Convenient online submission
- ▶ Rigorous peer review
- ▶ Immediate publication on acceptance
- ▶ Open access: articles freely available online
- ▶ High visibility within the field
- ▶ Retaining the copyright to your article

---

Submit your next manuscript at ▶ [springeropen.com](http://springeropen.com)

---