Heuristic Based Dynamic Spectrum Assignment in Cognitive Radio Network

Zhiyong Liu
School of Computing,
Queen's University, Kingston,
Ontario, K7L 3N6 Canada
7zml@queensu.ca

Nidal Nasser
Electrical and Software Engineering
Department, Alfaisal University
P.O Box 50927, Riyadh 11533
Kingdom of Saudi Arabia (KSA)
nnasser@alfaisal.edu

Hossam S. Hassanein
School of Computing,
Queen's University, Kingston,
Ontario, K7L 3N6 Canada
hossam@cs.queensu.ca

***Abstract*—Cognitive radio networks (CRNs) significantly improve spectrum utilization efficiency by allowing secondary users (SUs) to opportunistically share unused spectrum bands with primary users (PUs). In this paper, we present a spectrum assignment model and propose a genetic algorithm (GA) and a heuristic algorithm to determine the proper spectrum assignment, which optimizes the SUs' reward and the network operator's revenue while satisfying capacity constraints, interference constraints and rate requirement constraints. We show that both algorithms greatly outperform the random assignment approach.**

***Index Terms*—Cognitive radio network (CRN), genetic algorithm, spectrum assignment, wireless communication**

I. Introduction

Wireless communication has enjoyed tremendous growth in the past two decades and according to the International Telecommunications Union (ITU), the number of cell phone subscribers has surpassed 5 billion at the end of 2010. Such high demand puts a huge burden on existing limited radio resources. Surprisingly, studies have showed that the bottleneck is not because of lack of radio resources (4 GHz at the moment), but because of inefficient spectrum usage. Current wireless networks employ fixed spectrum assignment and have a 15%~85% utilization rate with high variance in time, leading to waste of resources [5]. On the other hand, congestion pricing, as a way to alleviate traffic jam, is often used individually on each access network and hence, prematurely leading to loss of revenue as the aggregated capacity of all access networks could still be under-utilized. These deficiencies can be exploited by a dynamic spectrum assignment policy in a Cognitive Radio Network (CRN).

A CRN, first introduced in [10, 11], is a new paradigm for wireless communication in which either a network or a wireless node can changes its transmission or reception parameters based on interaction with the environment, in hope to exploit spectrum holes in an opportunistic manner (See Fig. 1). A spectrum hole (or white space) is the portion of the spectrum that is not used by a licensed network. Due to the difficulty of frequency assignment, spectrum holes could occupy up to 85% of the licensed band. Such inefficiency forms the inspiration for cognitive radio technology. The two key ingredients in CRN are cognitive capability and re-configurability [8, 14].

- *Cognitive capability*: A CRN is not only capable of sensing available spectrum opportunity in frequency, temporal, spatial and geographical domains, but also able to extract the characteristics of such opportunity and determine the corresponding data rate, bandwidth and transmission mode.
- *Reconfigurability*: A mobile station (MS) in a CRN is capable of dynamically adjusting its operating frequency, changing its modulation scheme, altering its transmission power or even adapting to different communication technology in accordance with the radio environment.

Besides conventional radio management services, a CRN must support the following new functionalities in order to take advantage of the available spectrum opportunistically.

- *Spectrum sensing*: Detecting spectrum holes and sharing the spectrum without disturbance to existing users.
- *Spectrum management*: Selecting the best available spectrum to meet user communication requirements.
- *Spectrum mobility*: Maintaining seamless communication for a secondary user (SU) during the transition to better spectrum. It is triggered by the appearance of new primary user (PU), SU's movement or traffic variation in the network. It needs to monitor user behaviors. This is a largely unexplored area.
- *Spectrum sharing*: Providing appropriate spectrum assignment among coexisting users.

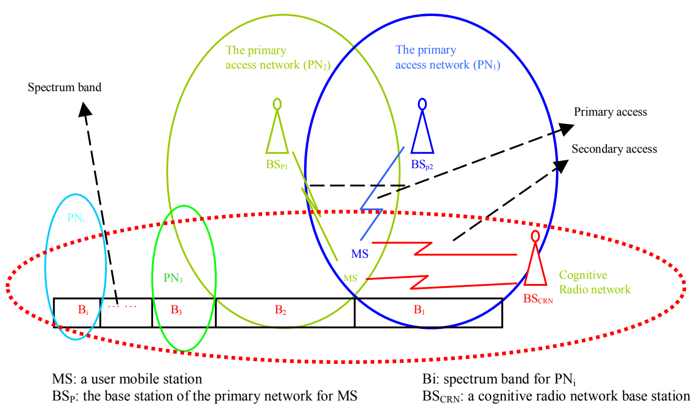


Fig. 1 A cognitive radio network

There are two types of users in a CRN. A PU is the subscriber of a licensed band network and a SU is the one that leases the unused licensed band of the network. In essence, a PU and a SU share the same spectrum band, typically in a preemptive manner where priority is given to the PU. In other words, if a PU requests a spectrum band that is currently used by a SU, this SU must vacate the band immediately and either migrates to another spectrum hole or has to hang up. Such preemptive sharing scheme is widely accepted for the purpose of avoiding harmful interference to existing PUs.

Many attempts have been made to tackle the spectrum sharing problem. One common approach is the color sensitive graph coloring approach [9, 17], in which spectrum channels are represented by different colors and each SU is denoted by a vertex. A vertex can use a number of colors depending on the channels in which the corresponding SU can operate, as well the conflicts it might have with adjacent vertices (i.e. adjacent SUs). A set of approximation algorithms are used to find the proper labeling and coloring scheme that maximizes some utility function. This approach is improved in [6] by taking proportional fairness into account. In [12], game theory is used to find the proper spectrum assignment. SUs and the network operator are modeled as players in a game, each with its own utility function that is defined as a function of price. Depending on the degree of cooperation between the SUs and the network operator, a pricing strategy is computed that will put the game in an equilibrium and the channel assignment is determined. Game theory is also used in [1] but formulated differently. In [1], Each SU expresses his channel preference in terms of throughput while each channel specifies the type of SU it prefers in terms of transmission power. The idea is to find a matching channel for each SU based on their preferences. In [4], an auction-based channel allocation scheme is proposed, where each SU senses the channel whose quality fits his need and announces a bid for that channel. The CRN base station then determines how to assign the SUs based on their bids, channel availabilities and channel capacities.

Instead of letting the CRN base station (or the network operator) deciding how to allocate the channels, a practice that often leads to performance bottleneck, some researchers propose to allow each SU to make his own decision about his spectrum access strategy. This is called distributed spectrum sharing [2, 3, 15, 18, 19]. In [14], a price-based iterative water-filling algorithm is implemented that allows SUs to repeatedly negotiate their spectrum and transmission powers. In this approach, a SU selfishly adjusts his transmission power and his pricing factor based on the interference he observes locally, in hope to maximize his own utility function. This process is repeated for each SU. They prove that the proposed algorithm will drive the SUs to reach a Nash equilibrium. Similar works include a heuristic algorithm based on fairness bargaining that considers both throughput and fairness [16], a distributed load balancing strategy where channels are randomly assigned and slowly adjusted according to traffic [13], and an allocation strategy based on reinforce learning where each network cell employs a learning strategy to adapt to the time-varying environment and offers opportunistic access to SUs when not congested [2]. Note that because spectrum allocation in distributed spectrum sharing is done based on local observation, the solution is often less than optimal. The above approaches, either centralized-based or distributed-based, are often compared based on their spectrum utilization, fairness, throughput, traffic overhead and system complexity.

A genetic algorithm (GA) is a search heuristic that mimics the process of natural evolution. A GA attempts to find a solution out of a pool of candidate solution that maximizes or minimizes certain objective(s). It shapes a population of candidate solution through the survival of its fittest member. The power of natural selection across a population of varying individuals has been demonstrated in the emergence of species in nature, as well as through the social process underlying culture change [7]. The algorithm usually starts from a population of randomly generated candidate solutions, called chromosomes or genome, and happens in generations. In each generation, the fitness of every candidate solution in the population is evaluated, multiple candidate solutions are stochastically selected from the current population based on their fitness, and modified through crossover and mutation operations to form a new population. The new population is then used in the next iteration of the algorithm. This is motivated by a hope, that the new population will be better than the old one. Commonly, the algorithm terminates when either a maximum number of generations has been produced, convergence has reached (that is, every individual in the population is the same), or a satisfactory fitness level has been reached for the population. GA strikes to find a “reasonable” solution within “reasonable” time.

Compared to other optimization techniques, GA offers flexible trade-off between the quality of the solution and the time to compute the solution. This feature makes GA an excellent choice for the spectrum assignment problem. When the network is highly congested and many SUs are waiting to be admitted, we can lower the number of generations and the population size to speed up the evolution process. On the other hand, when traffic is light, we can increase the number of generations and the population size to improve the quality of the solution.

A GA and a quantum genetic algorithm (QGA) are proposed in [20]. They attempt to find a spectrum allocation, based on channel availability and interference constraints, that optimizes SUs’ rewards. However, these approaches do not consider the spectrum capacity and the SUs’ transmission requirements, nor do they consider the revenue for the network operator.

In this paper, we propose a customized GA-based solution that determines the spectrum assignment for a group of SUs. It optimizes SU's reward as well as network operator's revenue. The algorithm offers flexible trade-off between the accuracy and speed. Our system model includes operator's incentive function as part of the objective and considers spectrum capacity as part of the constraint. We also introduce a heuristic algorithm that quickly computes the spectrum assignment.

The rest of the paper is organized as follows. Section II describes the system model and formulates the spectrum assignment problem. The customized GA and the heuristic algorithm to solve the problem are introduced in Section III and IV, respectively. Simulation results are shown and discussed in Section V. Finally, we give our conclusions and future research extensions in Section VI.

II. PROBLEM FORMULATION

Consider a CRN currently serving an arbitrary number of PUs and SUs. The network is not congested and there are some unused spectrum bands. Each unused band has different characteristics and, therefore, might have different transmission capacity and support different transmission rates. There are several SUs waiting to be admitted into the network. The objective is to find a spectrum assignment to the unassigned SUs so that the network throughput as well as the revenue of the network operator will be maximized. Because existing users' performance will not be affected, maximizing the network throughput is the same as maximizing the new SUs' throughput.

Our study is conducted based on the following assumptions:

- We focus on the overlay spectrum assignment problem, in which unassigned SUs can only be admitted to unused band and admitting SUs in unused bands will not affect the performance of existing PUs and SUs.
- A SU can only be admitted to one unused band but an unused band can be shared by multiple SUs.
- There is a central CRN base station (BS) responsible for sensing what spectrum bands are available, their transmission capacity and transmission rate, and deciding which SU should be assigned to which band.

Under these settings, we formulate the spectrum assignment as follows:

- Spectrum band vector: $\mathbf{b} = \{b_1, b_2, ..., b_M\}$ is a vector of M spectrum bands, representing all the bands owned by the CRN.
- Spectrum availability vector: $\mathbf{a} = \{a_1, a_2, ..., a_M\}$ is a binary vector of M bits, where a 1 indicates the corresponding band is unused and 0 indicates otherwise.
- Spectrum capacity vector: $\mathbf{c} = \{c_1, c_2, ..., c_M\}$ is a vector where each element represents the maximum transmission rate supported by the corresponding spectrum band.
- $\mathbf{u} = \{u_1, u_2, ..., u_N\}$ is a vector of N secondary users waiting to be admitted into the network.
- Rate request vector: $\mathbf{r} = \{r_1, ..., r_M\}$ is a vector where each element represents the requested transmission rate of the corresponding SU.
- Spectrum reward matrix: $\boldsymbol{G} = \{g_{n,m} \geq 0\}_{N*M}$ is a $N \times M$ matrix where $1 \leq n \leq N$, $1 \leq m \leq M$, and element $g_{n,m}$ represents the maximum reward (the maximum throughput in this case) that secondary user u_n acquires if being assigned to spectrum band b_m. Note that if the entry is 0, it means that u_n cannot operate in b_m.
- Utility matrix: $\boldsymbol{K} = \{k_{n,m} \geq 0\}_{N*M}$ is an $N \times M$ matrix where $1 \leq n \leq N$, $1 \leq m \leq M$, element $k_{n,m}$ represents the willingness to pay of secondary user u_n if he is assigned to spectrum band b_m. In other words, it is the revenue that the operator will receive if u_n is allocated to b_m. Note that the following relationships hold between $\boldsymbol{G}$ and $\boldsymbol{K}$:

  $$If\, g_{n,m} = 0,\ then\ k_{n,m} = 0, \qquad (1)$$

  $$If\, g_{n,m} > g_{n,l}, then\ k_{n,m} > k_{n,l} \qquad (2)$$

  where n is the index of the SU, m and l are the indexes of the band it could be assigned to. (1) and (2) simply state that the higher the reward a SU gets, the more willing he is to pay for the service. However, (2) might not hold between two SUs. That is, it is possible that $g_{z,m} > g_{y,m}$, but $k_{z,m} < k_{y,m}$, where $1 \leq z, y \leq N$ and $1 \leq m \leq M$.
- Interference constraint matrix: $\boldsymbol{I} = \{i_{n,m} \in (0,1)\}_{N*N}$ is a N by N binary matrix where $1 \leq n \leq N$, $1 \leq m \leq M$, and if $i_{n,m} = 1$, it means that secondary user u_n and u_m interfere with each other and must not be assigned to the same spectrum band.
- Spectrum assignment vector: $\mathbf{x} = \{x_1, x_2, ..., x_N\}$ is a vector where element x_n means that secondary user u_n is assigned to spectrum band b_{x_n}. Note that if $x_n = 0$, it means that u_n is not admitted into the network. Vector $\mathbf{x^*}$ is a valid spectrum assignment if for $1 \leq n, m \leq N$, x_n satisfies:

  $$M \geq x_n \geq 0\ and \qquad (3)$$

  $$If\, x_n > 0, then\ a_{x_n} = 1\ and \qquad (4)$$

  $$If\, x_n > 0, then\ g_{n,x_n} \geq r_n\ and \qquad (5)$$

  $$For\ all\ x_n\ that\ are\ equal, \sum g_{n,x_n} \leq c_{x_n} \qquad (6)$$

  $$For\ all\ x_n\ that\ are\ equal, i_{n,m} = 0. \qquad (7)$$

  Constraint (3) states that the assigned band must be owned by the CRN. Constraint (4) means that the assigned band must be unused. Constraint (5) means that if a SU is indeed admitted, the assigned band must offer a rate no less than the requested rate. Constraint (6) and (7) means that if multiple SUs are assigned to the same band, their aggregated rate cannot exceed the band's capacity and they must not interfere with each other.

Given the above model, the objective of the spectrum assignment problem is to find a valid spectrum assignment $\mathbf{x^*}$ that maximizes the aggregated throughput and the revenue:

$$\arg max_{x*} \sum_{i=1}^{N} (\alpha g_{i,x_i} + \beta k_{i,x_i}), where\ \alpha + \beta = 1. \qquad (8)$$

III. GENETIC ALGORITHM

In a GA, each candidate solution is encoded as a chromosome. In the proposed GA, we use the spectrum assignment vector as chromosome, where each individual assignment x_n is regarded as a gene. Note that unlike [19], we do not distinguish whether a chromosome is a valid assignment or not. Instead, such information is implicitly reflected in the fitness function, in which invalid assignment receives a fitness value of 0. Such treatment reduces the processing time during each evolution. The evolution of a GA is driven by a set of genetic operations including fitness evaluation, selection, crossover, mutation and replacement. Our fitness function is based on the objective function (8):

$$f(\mathrm{x}) = \begin{cases} \sum_{i=1}^{N} (\alpha g_{i,x_i} + \beta k_{i,x_i}), \alpha + \beta = 1, if\ \mathrm{x}\ is\ valid \\ 0, otherwise \end{cases}. \qquad (9)$$

As for the rest of the genetic operations, we use roulette wheel selection strategy, one-point crossover scheme, and two-point swapping mutation scheme. The replacement strategy determines that after one round of evolution, out of the ***p*** (population size) old candidate solutions (called parents) and the ***p*** newly generated candidate solutions (called offsprings), which ***p*** individuals will survive and be selected to move onto the next round of evolution?

There are three widely used replacement strategies [5]:

RS1. The old and new populations are combined and the top ***p*** individuals are chosen to form the new generation.

RS2. The top 1% of candidate solutions from the old population is combined with the top 99% of the new population to form the new generation.

RS3. The best candidate solution from the old population is combined with the top ***p-1*** of the new population to form the new generation.

We will adapt RS3 as our replacement strategy in our first implementation. The proposed genetic algorithm is outlined below.

Algorithm *GA(m, c, size, g, Select(P), FF(P), Replace(O, N,))* spectrum assignment algorithm

---

Input: *m is the mutation rate, c is the crossover rate, size is the population size, g is the number of generation the algorithm will execute, Select(P) is the selection strategy, FF(P) is the fitness function and Replace(O, N) is the replacement strategy.*
Output: *a spectrum assignment*

```
Let oldPop = null   /* Store the old population */
Let newPop = null  /* Store the newly generated population */
/* Randomly generate the initial population */
for (i := 0; i < size; i++) do
    randomly generate an assignment a
    add a to oldPop
end for

for (i := 0; i < g; i++) do
    /* use the fitness function to evaluate members of oldPop */
    FF(oldPop)

    for (j := 0; j < g/2; j++) do
        /* Select two parents from the old population to mate */
        p1 = Select(oldPop)
        p2 = Select(oldPop)
        /* Apply crossover according to the crossover rate */
        with probability c
            c1 = crossover(p1, p2)
        with probability c
            c2 = crossover(p1, p2)
        /* Apply mutation according to the mutation rate*/
        with probability m
            c1 = mutation(c1)
        with probability m
            c2 = mutation(c2)
        add c1 and c2 to newPop;
    end for
    /* Use the replacement strategy to form a new population */
    oldPop = Replace(oldPop, newPop)
    /* Reduce the mutation rate by 10m% per 10 generations */
    if (g mod 10) == 0
        m = m * 90%
end for
return the chromosome with the best fitness value from oldPop
```

In most applications, the crossover rate is close to 100%, that is, crossover always occurs. However, finding the values for population size and mutation rate is often a balance between speed and accuracy. The smaller the population size and the lower the mutation rate, the faster to find a solution. But the quality of the solution often suffers. A larger population size and larger mutation rate reduces the probability of being trapped in a local optimum. Besides slowing down the algorithm, a larger mutation rate makes the search random. As a compromise, we recommend the following approach which is an idea borrowed from simulated annealing: the mutation rate is high at start (say 10%), then gradually lower to 0% (say reduced by 1% for every 10 generations).

IV. HEURISTIC ALGORITHM

In this section, we introduce a heuristic algorithm that quickly computes the assignment priorities for SUs and then assigns the SUs to unused spectrum bands one by one according to their priorities. The higher the priority, the earlier a SU will be assigned. The main feature of this algorithm is its simplicity while still attempting to maximize reward and revenue. This heuristic algorithm uses the same system model as the one described in Section II. The assignment priority score s_i for secondary user u_i is computed according to the following equation:

$$s_i = \omega * avg_i + \theta * avk_i + \gamma * cf_i - \delta * nb_i \qquad (10)$$

where $1 \le i \le N$; avg_i is the average reward for u_i, which can be easily computed based on $\boldsymbol{G}$; avk_i is the average revenue generated by admitting u_i, which can be computed based on $\boldsymbol{K}$; cf_i is the number of interference conflicts of u_i, which can be deduced from $\boldsymbol{I}$; nb_i is the number of bands that u_i can operate in, which can be deduced from $\boldsymbol{G}$; ω, θ, γ and δ are weights assigned to each component and $\omega + \theta + \gamma + \delta = 1$. Note that the higher the reward and revenue a SU brings in, the higher the assignment priority it has. Similarly, the more conflicts a SU has, the sooner it should be considered. However, the more spectrum bands a SU can operate in, the more accommodating it is and hence the lower the priority it gets. The following describes how the algorithm works:

Step 1: use (10) to compute the assignment priority score for each SU in **u**.

Step 2: based on the priority scores, sort the SUs in **u** in decreasing order.

Step 3: assign u_1 to a spectrum band b_n that satisfies constraints (3)-(7) and produces the highest total of reward and revenue. If no such band exists, set x_{u_1} to 0, indicating that u_1 is not admitted into the network.

Step 4: update the capacity vector **c** and the availability vector **a**.

Step 5: remove u_1 from **u**.

Step 6: repeat Step 3 until **u** is empty.

V. PERFORMANCE EVALUATION

We first investigate the impact of GA parameters like the number generations and the population size on the performance. We fix the number of available bands M to 10 and the number of new SUs N to 8. In one experiment, we set the population size to 200 and study how the performance changes as we manipulate the number of generations the algorithm will execute. In another experiment, we set the number of generations to 100 and study how the performance changes as the population size changes. The results are shown in Fig. 2. As we can see, the performance is pretty stable (only fluctuates between 1240 and 1250) once the number of the generations is over 100 and if we set the number of generations to 100, the performance peeks when the population size reaches 200. As such, we use these settings for the proposed GA in the rest of the experiments.

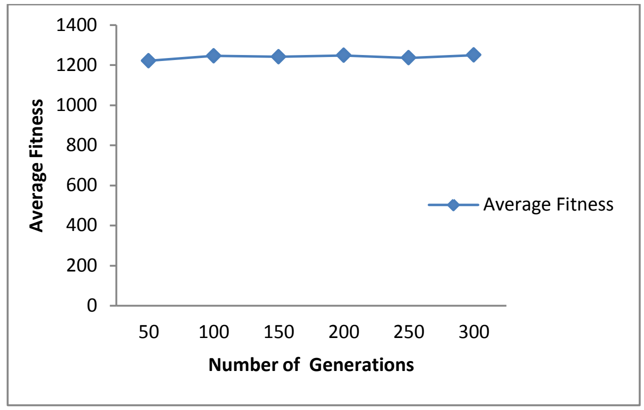
We use the random assignment approach as the base of comparison and solutions produced by the three algorithms are compared using evaluation function (9). For GA, the population size and the number of generations are set to 200 and 100, respectively. The crossover rate is set to 100% while the mutation rate starts at 10% and is reduced by 1% for every 10 generations till it reaches 0. The coefficients in (9) and (10) are set to $\alpha = \beta = 0.5$ and $\omega = \theta = \gamma = \delta = 0.25$. The probability of interference between two randomly chosen SUs is 50%.

We test the algorithms under two different conditions: when the network is not congested, which is simulated by having more available spectrum bands than SUs; when the network is congested, which is simulated by having more SUs than available spectrum bands. The results of over 300 experiments are plotted in Fig.3. For simplicity, we keep the number of available spectrum band a constant that is set to 10, while dynamically adjusting the number of SUs N in different experiments. While the proposed GA performs better than the heuristics algorithm when the network is not congested or not heavily congested, both the proposed GA and the heuristic algorithm outperform the random approach in a large margin in most experiments, which validates the effectiveness of the two proposed algorithms. It is interesting to see that when the network is heavily congested where there are twice as many SUs as the number of available spectrum bands ($M = 10$ and N 21), GA fails to produce any valid assignment. This is largely because the number of conflicts increases exponentially as the number of SUs increases. As a result, any crossover operation between two randomly chosen assignments would almost certainly produce new assignments with conflicts. To prove this conjuncture, we reduce the interference probability from 50% to 10% and repeat the experiments. The results are shown in Fig. 4. As suspected, as the probability of interference decreases, GA's performance is greatly improved even when the network is heavily congested, while the performances of the other two algorithms remain relatively unchanged.

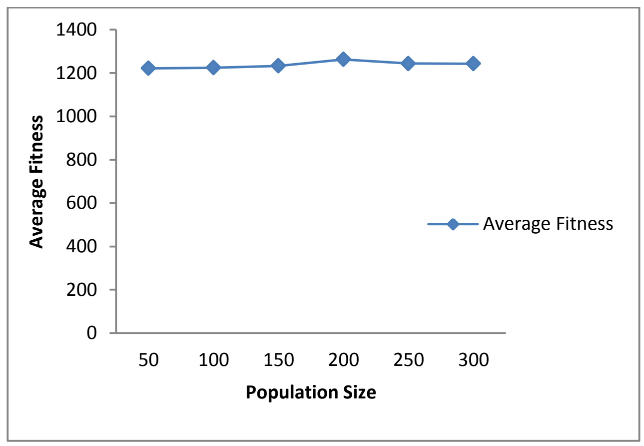
In conclusion, GA performs the best when the network has light to medium traffic or when the interference is low, while the heuristic algorithm offers consistent performance and works well regardless of the network condition.

VI. CONCLUSION

In this paper, we propose a customized GA and a heuristic algorithm to solve the spectrum assignment problem. They maximize the network throughput as well as network operator's revenue, while taking into account of the spectrum capacity constraint, interference constraint and user requirement constraints. Both algorithms output the random assignment approach and offer trade-off between the speed and performance. One of the major extensions of this work is to identify the set of traffic indicators that can signal the GA to automatically adjust its parameters (population size, number of generation, crossover and mutation rates) according to the network conditions. Another direction we are working on is to fine-tune the fitness function so that it can give differential treatment to different classes of traffics.



a) The impact of increasing the number of generations the GA will execute while the population size is fixed at 200.



b) The impact of increasing the population size while the number of generations is fixed at 100.

Fig. 2 The impact of GA parameters.

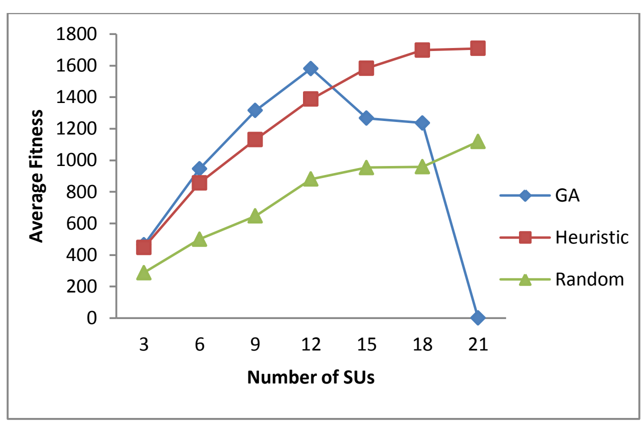
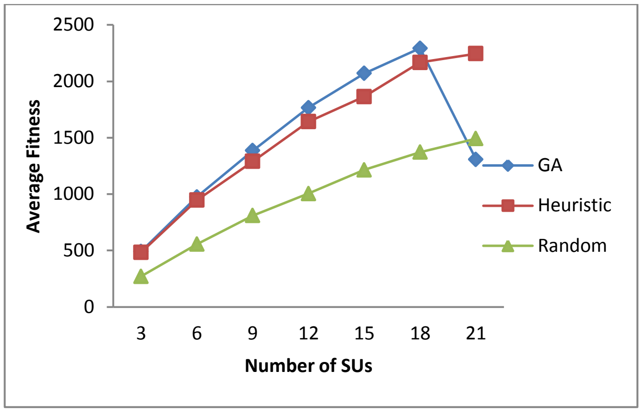
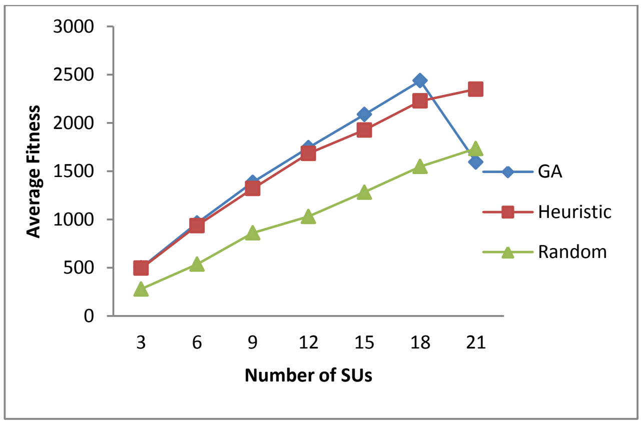


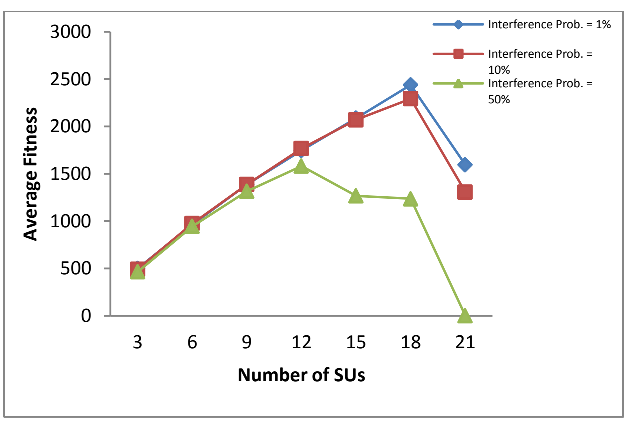
Fig. 3 Average finesses for GA, Heuristic and Random approach



a) The impact of reducing the interference probability to 10%.



b) The impact of reducing the interference probability to 1%.



c) Comparing GA's performance when the interference probability is 50%, 10% and 1%, respectively.

Fig. 4 The impact of reducing the interference probability to 10%

REFERENCES

[1] C. An, L. Zhang, "A spectrum allocation algorithm based on matching game", 978-1-4244-4496-4/09, IEEE 2009.

[2] R. Bernardo, R. Agusti, J. Perez-Romero, O. Sallent, "Distributed spectrum management based on reinforce learning", *Proceedings of the 4th Internal Conference on CROWNCOM*, 2009.

[3] L. Cao, H. Zheng, "Distributed spectrum allocation via local bargaining", *Proceeding IEEE Sensor and Ad Hoc Communications and Networks (SECON) 2005*, September 2005, pp.475–486.

[4] B. Chen, H. Wu, A. Hoang, Y. Liang, "Optimizing the second-price auction algorithm in a dynamic cognitive radio network", ICCS 2008, 978-1-4244-2424-5/08, IEEE 2008.

[5] FCC, *ET Docket No 03-222 Notice of proposed rule making and order*, December 2003.

[6] Y. Ge, J. Sun, S. Shao, L. Yang, H. Zhu, "An improved spectrum allocation algorithm based on proportional fairness in cognitive radio network", 978-1-4244-4496-3/10, IEEE 2010.

[7] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Chapter 01, Kluwer Academic Publishers, Boston, MA., 1989.

[8] J. Huang, R.A. Berry, M.L. Honig, "Spectrum sharing with distributed interference compensation", *Proceeding IEEE DySPAN 2005*, November 2005, pp. 88–93.

[9] Y. Liang, L. Lai, J. Halloran, "Distributed cognitive radio network management via algorithms in probabilistic graphical models", *IEEE Journal of Selected Areas in Communications*, Vol. 29, No. 2, pp. 338 – 347, February 2011.

[10] J. Mitola III, "Cognitive radio for flexible mobile multimedia communication", *Proceeding IEEE International Workshop on Mobile Multimedia Communications* (MoMuC) 1999, November 1999, pp. 3–10.

[11] J. Mitola III, "Cognitive radio: an integrated agent architecture for software defined radio", *Ph.D Thesis*, KTH Royal Institute of Technology, 2000.

[12] D. Niyato and E. Hossain, "Market-equilibrium, competitive, and cooperative pricing for spectrum sharing in cognitive radio networks: analysis and comparison", *IEEE Transactions on Wireless Communications*, VOL. 7, NO. 11, NOVEMBER 2008.

[13] N. Olano, M. Petrova, P. Mahonen, "SDR implementation and verification of distributed load balancing spectrum allocation algorithm", 978-1-4244-5213-4/09, IEEE 2009.

[14] R.W. Thomas, L.A. DaSilva, A.B. MacKenzie, "Cognitive networks", *Proceeding IEEE DySPAN 2005*, November 2005, pp. 352–360.

[15] F. Wang, M. Krunz, and S. Cui, "Price-based spectrum management in cognitive radio networks", *IEEE Journal of Selected Topics in Signal Processing*, VOL. 2, NO. 1, pp. 74 – 87, FEBRUARY 2008

[16] L. Yu, C. Liu, Z. Liu, W. Hu, "Heuristic spectrum assignment algorithm in distributed cognitive networks", 978-1-4244-3709-2/10, IEEE 2010.

[17] T. Zhang, B. Wang, Z. Wu, "Spectrum assignment in infrastructure based cognitive radio network", 978-1-4244-4496-3/09, IEEE 2009.

[18] Q. Zhao, L. Tong, A. Swami, "Decentralized cognitive MAC For dynamic spectrum access", *Proceeding IEEE DySPAN 2005*, pp. 224–232.

[19] J. Zhao, H. Zheng, G. H. Yang, "Distributed coordination in dynamic spectrum allocation networks", *Proceeding IEEE DySPAN 2005*, November 2005, pp. 259–268.

[20] Z. Zhao, Z. Peng, S. Zheng and J. Shang, "Cognitive radio spectrum allocation using evolutionary algorithms", *IEEE Transactions on Wireless Communications*, Vol. 8, No. 9, September, 2009.