On the Effectiveness of using Genetic Algorithm for Spectrum Allocation in Cognitive Radio Networks

Abdelbaset S. Hamza
Dept of Electronics & Communication Eng.
Cairo University
Giza, Egypt
Email: bhamza@ieee.org

Mona M. Elghoneimy
Dept. of Electronics & Communication Eng.
Cairo University
Giza, Egypt
Email: melghoneimy@eng.cu.edu.eg

***Abstract*—In this paper, an extensive study is performed to understand the effect of the various parameters used in Genetic Algorithms (GA) to solve the spectrum allocation problem in cognitive radio networks. Two utilization functions, namely, Mean-Reward, and Max-Proportional-Fair, are used to evaluate the performance of the GA under different parameters values. Extensive simulation shows that, the parameters of the GA can be fine-tuned to achieve up to 90% and 62% improvements in speed and error, respectively, as compared to the GA used in the literature.**

***Index Terms*—Wireless networks, spectrum allocation, open spectrum, cognitive radio, genetic algorithm.**

I. Introduction

Wide range of services such as, broadcasting services, mobile communications, satellite, and military rely on the use of the radio spectrum. However, spectrum ranges are allocated to these services depending on the country and frequency bands in a what so called *administrative spectrum management approach*. This approach is attractive to the regulators since it facilitates the prevention of excessive interference among systems using the same frequency bands but in different geographical areas. It also guarantees adequate quality of service (QoS) for licensed users.

According to a recent study by the Federal Communications Commission (FCC) it is found that spectrum utilization ranges from $15-85\%$ in the bands below 3 GHz [1]. This represents a major underutilization for the available spectrum under the administrative approach. Therefore the new concept of the dynamic spectrum access is presented. In particular, for the opportunistic spectrum access, two different types of users exist in the network, either primary (licensed) user, or secondary user (unlicensed). In such networks, secondary user seeks a highly reliable communications whenever and wherever needed and to utilize the radio spectrum efficiently. In order to perform this, secondary user needs cognition capabilities, such as being aware for the surrounding, learning and understanding the variations and activities, and accordingly, adjusts its parameters to operate efficiently.

Efficient utilization of open spectrum in cognitive radio networks requires appropriate allocation of idle spectrum frequency bands (not used by licensed users) among coexisting cognitive radios (secondary users) while minimizing interference [2]. This problem is known as *resource allocation* in cognitive radio, and is shown to be NP-hard in the literature [3].

Several heuristics were proposed to solve the resource allocation problem based on game theory [4], pricing and auction mechanisms [5] [6], local bargaining [7], and vertex labeling [3] [8]. Recently, evolutionary algorithms are used to address the resource allocation problem. In particular, in [2], three evolutionary algorithms was performed including genetic algorithm (GA), quantum genetic algorithm (QGA), and particle swarm optimization (PSO) techniques.

In genetic algorithms, several parameters (e.g., population size, crossover scheme, and selection methods) need to be tuned in order to obtain best solutions and optimize the performance of the algorithm. Various combinations of these parameters may lead to different solutions. Accordingly, the objective of this paper is to investigate the impact of the various parameters on the genetic algorithm performance and its produced solution. In particular, we present a comprehensive study for the impact of the population size, crossover methods and probability, probability of mutation, and selection methods on the performance of the GA and the quality of the solution it produces. The study shows that a fine-tuned GA can achieve up to 90% and 62% improvements in speed and error, respectively, as compared to the GA proposed in [2].

The rest of this paper is organized as follows. Section II gives the system model and outlines the problem statement. The evaluation approach is explained in Section III. In Section IV, simulation results and analysis are presented. Conclusions are presented in Section V.

II. System Model and Problem Statement

Figure 1 shows a simple structure of a cognitive radio network. A typical cognitive network consists of a set of primary users each is assigned a channel selected from a pool of M orthogonal, non-overlapping spectrum bands that differ in bandwidth and transmission range. There are N coexisting secondary users that are planned to utilize these idle channels occupied by primary users in order to provide their services. It is assumed that each secondary user can utilize multiple channels at one time, but limited to the radio interface constraint.

978-1-4244-9924-3/10/$26.00 ©2010 IEEE

Each secondary user keeps a list of available channels that it can use without interfering with neighboring primary users. The spectrum access problem becomes a channel allocation problem. A secondary user's transmission can not overlap with the transmission of the primary user who uses the same channel. Therefore, each secondary user can adjust its transmission range by tuning its transmit power on channels to avoid interference with primary users.

The following are the key components in the used model [3]:

- *Channel availability*: $L = \{l_{n,m} | l_{n,m} \in 0, 1\}_{N \times M}$ is a matrix representing per user available spectrum: $l_{n,m} = 1$ if and only if channel m is available to user n.
- *Channel reward*: $B = \{b_{n,m}\}_{N \times M}$, a matrix representing the channel reward: $b_{n,m}$ represents the maximum reward that can be acquired by user n using channel m.
- *Interference constraint*: Let $C = \{c_{n,k,m} | c_{n,k,m} \in \{0, 1\}\}_{N \times N \times M}$, a matrix represents the interference constraints among secondary users. $c_{n,k,m} = 1$, if users n and k would interfere if they use channel m simultaneously. $c_{n,m} = 1 - l_{n,m}$.
- *Conflict free channel assignment*: Let $A = \{a_{n,m} | a_{n,m} \in \{0, 1\}, a_{n,m} \leq l_{n,m}\}_{N \times M}$, a matrix that represents the assignment: $a_{n,m} = 1$ if channel m is assigned to user n.
- *User Reward*: $\Re = \{\beta_n = \sum_{m=0}^{M-1} a_{n,m} \cdot b_{n,m}\}_{N \times 1}$ represents the reward vector that each user gets for a given channel assignment. In our context, the reward is the coverage area of the secondary user.

The objective of the spectrum allocation problem in open spectrum systems is to find a channel allocation that maximizes a utilization function $U(\Re)$.

To define $U(\Re)$, two spectrum design factors can be considered, namely, spectrum *utilization* and *fairness*. Different combinations of these two factors result in different definitions for the $U(\Re)$ function. We consider two utilization functions, namely, *Mean-Reward (MR)*, and *Max-Proportional-Fair (MPF)*. The following equations reproduced from [3] to describe these three utilization functions:

$$U_{MR} = \frac{1}{N} \sum_{n=0}^{N-1} \beta_n = \frac{1}{N} \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} a_{n,m} \cdot b_{n,m} \quad (1)$$

$$U_{MPF} = \left[\prod_{n=0}^{N-1} \left(\sum_{m=0}^{M-1} a_{n,m} \cdot b_{n,m} + 1E - 4 \right) \right]^{\frac{1}{N}} \quad (2)$$

A baseline reward of $1E-4$ is used in order to prevent the case of having a user with no channels (user starvation).

III. Evaluation Approach

For the two considered utilization functions, conventional genetic algorithm is used. However, our study considers the effect of GA's parameter variations. In particular, population size (chromosomes/generation), crossover probability, mutation probability, crossover scheme, and selection method are the parameters investigated in our study. In the following a brief description of GA parameters, their ranges, and options:

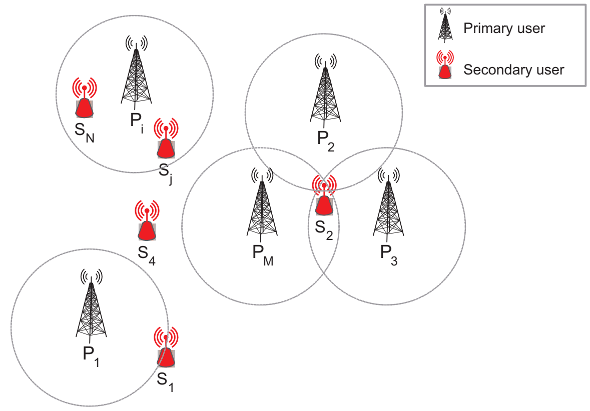

Fig. 1. Sample structure of a cognitive radio network.

- *Population Size.* The population size is one of the most important factors that affects the performance and the efficiency of the GA. Population size needs to be selected based on the size of the search space. If population size is too small, it may fail to cover the entire solution space, however, too large population may causes a divergence. In our study, we examine both small and large population size and analyze their impact on the GA performance and its produced solutions.
- *Selection Schemes.* Selection is used to generate the new population without the elite members. A variety of selection methods exists, and discussed in literature, such as: random selection, roulette-wheel, and tournament selection [9]. In this paper, we study the effect of the these three selection schemes.
- *Crossover Techniques and Parameters.* Crossover is the mating form used by the GA in order to generate a new population. There are three techniques that are used to perform this mating:
  1) Single point crossover,
  2) Multipoint crossover, and
  3) Uniform crossover.

  The crossover operation is controlled by a *crossover probability*, which directly affects the performance of the GA.

  In this study, we study the impact of the three mating techniques above. In addition, we examine the effect of varying the crossover probability on the GA performance and its solutions.
- *Mutation Parameter.* Mutation represents another technique by which the GA explores the solution space. A single point mutation inverts a randomly selected bit, in an arbitrary chromosome. The process of mutation is performed according to a *mutation probability*. Generally, mutation probability less than 0.1 is usually used [10]. In this paper, we study the impact of varying the mutation probability in the range from 0.02 to 0.1.

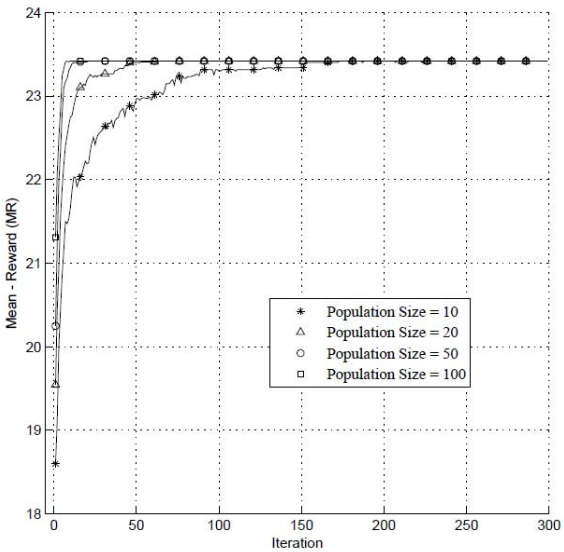

Fig. 2. GA results under MR utility function and different values of population size.

During this study, parameters except the one under investigation are kept constant. Initially, simulation setup used in [2] is used, and then an iterative study is performed. In particular, the population size is set to 20 chromosome, with an elitism, and survival rate of 15%, the crossover and mutation probabilities used are 0.8 and 0.01, respectively. For the selection method, roulette wheel selection scheme is carried out, and two-point crossover scheme is performed. The given results in this study are the average of 50 experiments, each of 300 iterations. However, some curves are displayed with less number of iterations (100 iterations), for better performance illustration.

IV. Simulation and Analysis

This section summarizes the simulation results of the genetic algorithm parameter study under the two utilization functions Mean-Reward (MR), and Max-Proportional-Fair (MPF). For each function, the results for the GA parameters discussed in Section III are reported.

Experimental simulation reported in this paper are performed using MATLAB (running on a 1.83 GHz processor PC with 1 GB RAM) is used to perform simulations.

A. Results for Mean-Reward Utility Function

Figure 2 depicts the performance of the GA under different values of population size. In particular, four different values are used: 10, 20, 50, and 100. As it is shown, the performance under the four values of the population size saturates at the same utility value, but at different number of iterations.

A summary of number of iterations required by the algorithm using these different values is included in Table I. In order to determine the best value to be used, a unified factor which is the total time required by the algorithm to achieve that saturation is used as a performance measure.

TABLE I
Summary of GA Performance under MR Utility Function and Different Values of Population Size.

Population Size	Mean No. of Iterations	Mean Time/ Iteration (sec)	Total time required (sec)
10	180	0.0907	16.326
20	64	0.1985	12.704
50	21	0.8158	17.131
100	10	2.7920	27.920

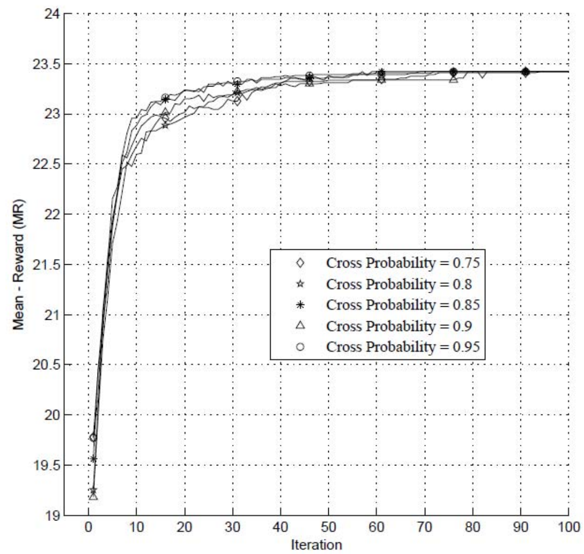

Fig. 3. GA results under MR utility function and different values of crossover probability.

Comparing the total time required by the GA under different population sizes, it is found that the population of 20 chromosomes represents an adequate results among other sizes.

TABLE II
Summary of GA Performance under MR Utility Function and Different Values of Crossover Probability.

Crossover probability	Mean No. of Iterations	Mean Time/ Iteration (sec)	Total time required (sec)
0.75	94	0.1993	18.7342
0.8	69	0.2018	13.9242
0.85	59	0.1992	11.7528
0.9	83	0.2024	16.7992
0.95	72	0.2019	14.5368

Figure 3 plots the performance of the GA under different values of crossover probability. For this study, five different values are used starting from 0.75 to 0.95 with an increment of 0.05. Table 3 summarizes the results obtained. Results show that the GA under the MR utilization function performs the best using a crossover probability of 85%.

For the mutation probability, five different values are used for this factor; 0.02, 0.04, 0.06, 0.08, and 0.1 as it is shown in Figure 4. According to results tabulated in Table III, a mutation probability of 0.1 gives the best performance among the selected values.

Using a population size of 20 chromosome, crossover probability of 0.85, and a mutation probability of 0.1, the

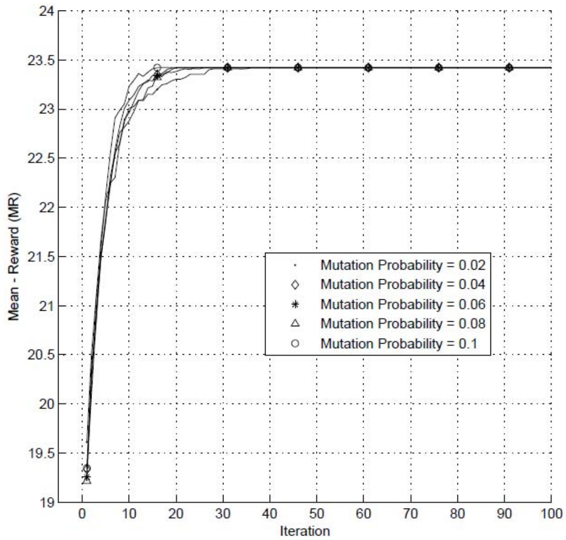

Fig. 4. GA results under MR utility function and different values of mutation probability.

TABLE III
SUMMARY OF GA PERFORMANCE UNDER MR UTILITY FUNCTION AND DIFFERENT VALUES OF MUTATION PROBABILITY.

Mutation probability	Mean No. of Iterations	Mean Time/ Iteration (sec)	Total time required (sec)
0.02	36	0.2040	7.344
0.04	24	0.2071	4.970
0.06	26	0.2090	5.434
0.08	20	0.2100	4.200
0.1	16	0.2336	3.737

performance of the GA using four different crossover methods is investigated.

Results are shown in Figure 5, and tabulated in Table IV. It can be seen that the single point crossover presents the worst performance, as it requires a large number of iterations to reach the best solution. On the other hand, the two points and multipoints both have a nearly similar performance, either with respect to the number of iterations, or the average time required per iteration. For the uniform crossover method, it is found that it outperforms other methods. It requires less number of iterations to get to the best solution, with an overall processing time of 3.661 seconds.

TABLE IV
SUMMARY OF GA PERFORMANCE UNDER MR UTILITY FUNCTION AND DIFFERENT CROSSOVER METHODS.

Crossover method	Mean No. of Iterations	Mean Time/ Iteration (sec)	Total time required (sec)
Simple point	27	0.2150	5.805
Two points	22	0.2177	4.789
Multipoints	20	0.2199	4.398
Uniform	17	0.2154	3.661

In this study, three selection methods mentioned in Section III are used for the evaluation process. For tournament selection scheme, tournament sizes of 2, 3, and 5 are considered. Different performance curves are plotted in Figure 6.

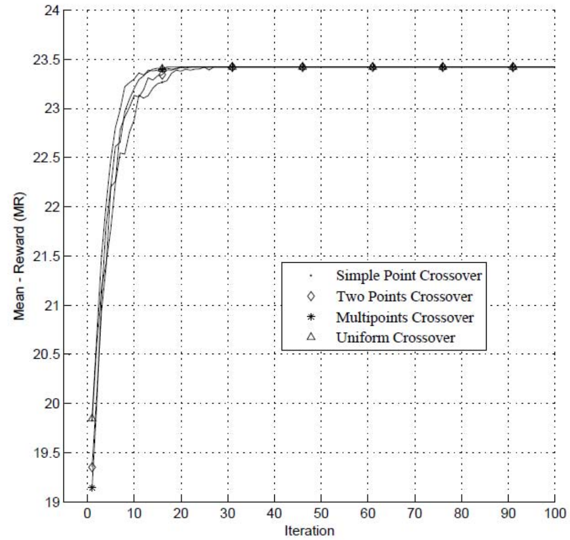

Fig. 5. GA results under MR utility function and different crossover methods.

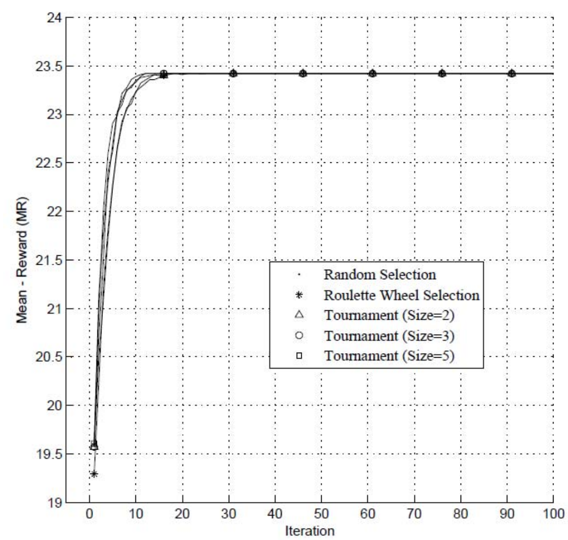

Fig. 6. GA results under MR utility function and different selection methods.

It worth to note that the roulette wheel presented the worst performance among the different methods tested. As it is shown in Table V, RW selection method required 17 iterations to find the best solution, each of 0.2070 seconds on average. This means that for the RW selection method, 4.347 seconds are required to find the best solution.

On the other hand, Tournament selection methods generally outperforms the random and the RW selection methods. In particular, tournament selection method with size of 5 required 12 iterations to find the best solution, and an average time of 0.1086 seconds, resulting in a total processing time of 1.288 seconds.

Comparing the performance of the GA with the parameters setup presented in [2], and the fine-tuned GA employing

TABLE V
SUMMARY OF GA PERFORMANCE UNDER MR UTILITY FUNCTION AND DIFFERENT SELECTION METHODS.

Selection method	Mean No. of Iterations	Mean Time/ Iteration (sec)	Total time required (sec)
Random	17	0.1054	1.791
Roulette Wheel	21	0.2070	4.347
Tournament (size=2)	17	0.1083	1.841
Tournament (size=3)	15	0.1083	1.407
Tournament (size=5)	12	0.1086	1.288

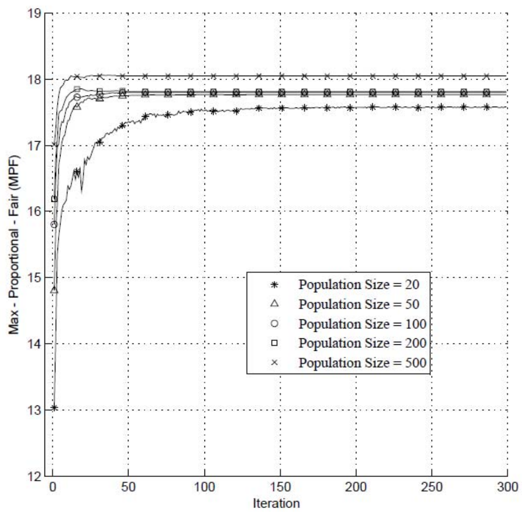

Fig. 7. GA results under MPF utility function and different values of population size.

the 20 chromosomes, crossover and mutation probabilities of 0.85, and 0.1, respectively. And using uniform crossover, with tournament selection method of tournament size equals to 5, we find that the GA in literature requires 64 iterations, each of 0.1985 seconds, resulting in an overhaul time required of 12.704 seconds to achieve the best value, however, for the tuned GA only 12 iterations are required, with an average time per iteration of 0.1074 seconds, this implies that a total time of 1.288 seconds is required to find the best value. This represents an improvement of around 90% with respect to the total time required.

B. Results for Max-Proportional-Fair Utility Function

In this part of the study, similar experiments performed under the MR utilization function, are repeated employing the MPF utilization function. Starting with the size of the population parameter, single experiments achieve a maximum reward of 18.1, however, on the average of the 50 experiments performed, the GA does not find this best value under any of the investigated population sizes. In particular, five different values of the population size: 20, 50, 100, 200, and 500 are considered.

Figure 7 shows the simulation results of the GA employing the MPF utility function, under the five population sizes. Average time per iteration of the population sizes considered are tabulated in Table VI. Also, data from curves are extracted

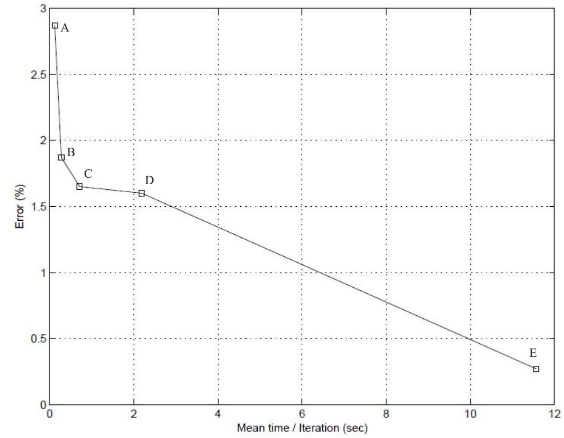

Fig. 8. Mean time per iteration versus minimum error achieved under MPF utility function and different values of population size.

and tabulated in Table VII in which six different error values are defined, and for each population size, number of iterations and time required to find a solution with the corresponding error are listed. None of the five population values find solution of 0%. For a solution with error of 2.87%, a population size of 50 chromosomes represents best result with respect to the total time required to find this solution among other population sizes.

TABLE VI
MEAN TIME PER ITERATION OF GA UNDER MPF UTILITY FUNCTION AND DIFFERENT VALUES OF POPULATION SIZE.

Population Size	Mean Time/ Iteration (sec)
20	0.188
50	0.276
100	0.714
200	2.188
500	11.569

From another point of view, a curve of average time per iteration versus the minimum error achieved is shown in Figure 8. It is clear that the optimum value to be selected is bounded between the two points B and C, hence, a population size of 50 chromosomes is selected.

It is suggested that, either a population of 50 chromosomes is selected, for adequate error and processing time, or a population of 500 for minimum error. A population in between does not represents an efficient operating population size.

For crossover probability parameter under the MPF utility function, different performance curves are shown in Figure VIII. Data of mean time per iteration, and minimum error achieved are presented in Table 9. Comparing the mean time per iteration of the four values considered, the difference is almost negligible. Therefore, a decision of the best value to be used can be performed based on the minimum error achieved. Performance using crossover probability of 0.85 represents the minimum error achieved among the investigated range of probabilities, whereas it finds a solution with error of 1.43%.

TABLE VII
GA PERFORMANCE AND MINIMUM ERROR UNDER MPF UTILITY FUNCTION AND DIFFERENT VALUES OF POPULATION SIZE.

Error (%)	Size=20		Size=50		Size=100		Size=200		Size=500	
	Iterations	Time	Iterations	Time	Iterations	Times	Iterations	Time	Iterations	Time
0	—	—	—	—	—	—	—	—	—	—
0.27	—	—	—	—	—	—	—	—	42	485.93
1.6	—	—	—	—	—	—	49	107.221	5	57.84
1.65	—	—	—	—	47	33.56	12	26.258	4	46.27
1.87	—	—	51	14.11	26	18.56	10	21.882	3	34.70
2.87	146	17,22	14	3.87	10	7.142	7	15.317	2	23.13

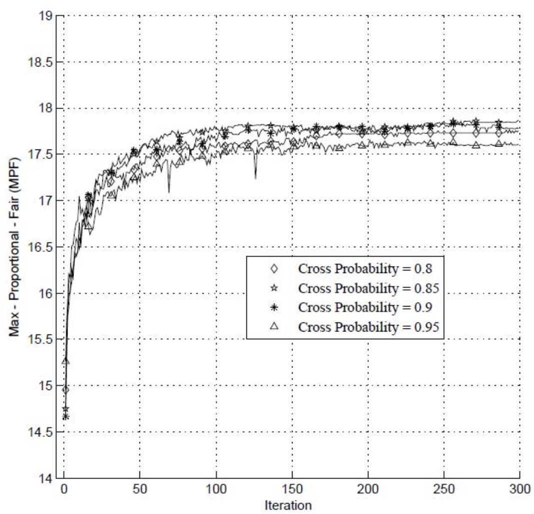

Fig. 9. GA results under MPF utility function and different values of crossover probability.

TABLE VIII
SUMMARY OF GA PERFORMANCE UNDER MPF UTILITY FUNCTION AND DIFFERENT VALUES OF CROSSOVER PROBABILITY.

Crossover probability	Mean Time/ Iteration (sec)	Minimum Error (%)
0.8	0.3005	2.04
0.85	0.3068	1.43
0.9	0.3045	1.76
0.95	0.3017	2.76

Similarly, for the mutation probability, the average time per iteration is found to be almost the same over the range of mutation probabilities used. Table VIII lists the mean time per iteration, and minimum error achieved by each. As it is shown, performance under the probability of 0.1 outperforms other probabilities, finding a solution with error of 1.215%. Curves of the performance of the GA under different values of the mutation probabilities are shown in Figure 10.

It worth to note, that as we tune a parameter after another, the overhaul performance of the GA is improving, presenting a better solution quality with respect to the minimum error achieved.

Results of the GA performance under different crossover methods, show that an almost similar mean time per iteration is obtained, however, different minimum errors are found. Figure 11 depicts different performance curves. Table X lists

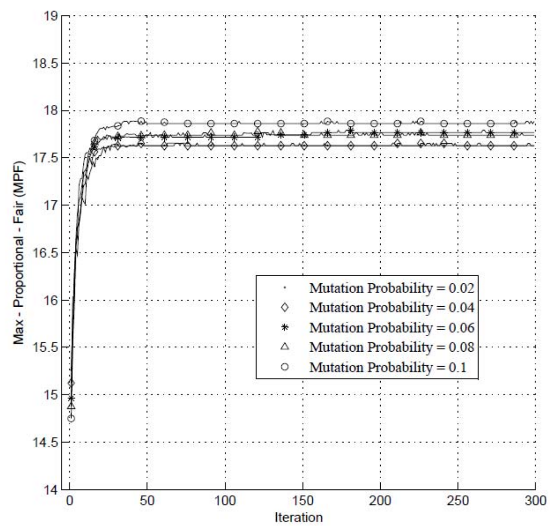

Fig. 10. GA Results under MPF utility function and different values of mutation probability.

the results of the simulations. Results imply that a single point crossover presents the best scheme to be used under the MPF utility function.

TABLE IX
SUMMARY OF GA PERFORMANCE UNDER MPF UTILITY FUNCTION AND DIFFERENT VALUES OF MUTATION PROBABILITY.

Mutation probability	Mean Time/ Iteration (sec)	Minimum Error (%)
0.02	0.2756	2.59
0.04	0.2807	2.48
0.06	0.2868	1.82
0.08	0.2905	1.98
0.1	0.2992	1.21

TABLE X
SUMMARY OF GA PERFORMANCE UNDER MPF UTILITY FUNCTION AND DIFFERENT CROSSOVER METHODS.

Crossover method	Mean Time/ Iteration (sec)	Minimum Error (%)
Simple point	0.2960	1.546
Two points	0.2972	1.878
Multipoints	0.3016	2.209
Uniform	0.2915	1.712

Figure 12 shows the performance results of the GA under different selection schemes. Mean time per iteration, and

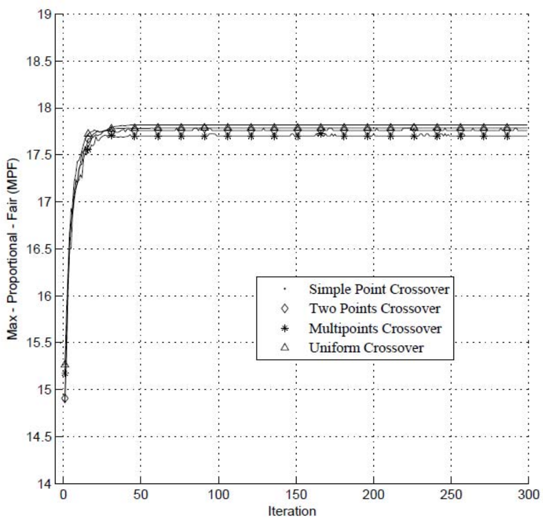

Fig. 11. GA results under MPF utility function and different crossover methods.

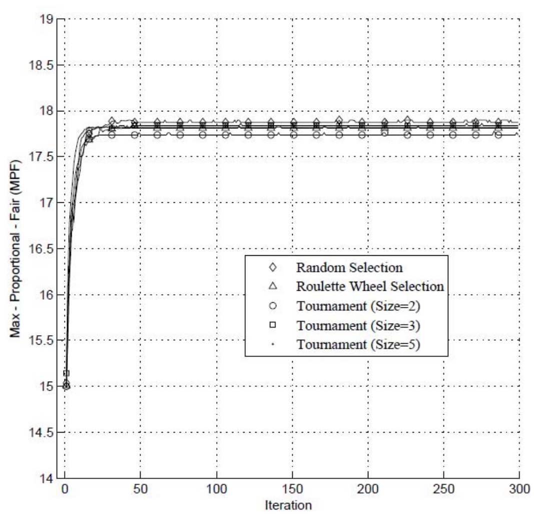

Fig. 12. GA results under MPF utility function and different selection methods.

minimum error are tabulated in Table XI. As it is shown, all selection methods has an almost same average time per iteration except the RW selection scheme. From the data available, it is clear that random selection scheme is the most efficient under MPF utility function, since it presents the minimum achievable error among other selection schemes.

TABLE XI
SUMMARY OF GA PERFORMANCE UNDER MPF UTILITY FUNCTION DIFFERENT CROSSOVER METHODS.

Selection method	Mean Time/ Iteration (sec)	Minimum Error (%)
Random	0.1920	1.10
Roulette Wheel	0.2867	1.60
Tournament (size=2)	0.1948	1.98
Tournament (size=3)	0.1964	1.43
Tournament (size=5)	0.1947	1.54

V. CONCLUSION

We performed an extensive parameter fine-tuning study for the GA used to address the problem of spectrum allocation in cognitive radio networks. Two utilization functions, namely, Mean-Reward (MR), and Max-Proportional-Fair (MPF), are used to evaluate the performance of the GA under varying parameters. Simulation results confirm that a judicious selection for GA parameters directly affects the solution quality found by the GA. In particular, for the MR function, careful parameter selection results in an improvement of 90% over the GA presented in literature. Moreover, for the MPF function, different parameters than those used in the MR study are found to be more efficient, presenting an improvement of 62%.

REFERENCES

[1] FCC Spectrum Policy Task Force, "Report of the spectrum efficiency working group,“ Nov. 2002. [Online]. Available: $http : //www.fcc.gov/sptf/files/SEWGFinalReport_1.pdf$

[2] Z. Zhao, Z. Peng, S. Zheng, and J. Shang, "Cognitive Radio Spectrum Allocation using Evolutionary Algorithms," IEEE Trans. on Wireless Communications, vol. 8, no. 9, pp. 4421 - 4425, 2009.

[3] C. Peng, H. Zheng, and B. Y. Zhao, "Utilization and Fairness in Spectrum Assignment for Opportunistic Spectrum Access," *ACM Mobile Networks and Applications (MONET),* vol. 11, no. 4, pp. 555-576, 2006.

[4] N. Nie and C. Comaniciu, "Adaptive channel allocation spectrum etiquette for cognitive radio networks,“ *Proc. IEEE DySPAN 2005,* pp. 269278, 2005.

[5] J. Huang, R. Berry, and M. L. Honig, "Auction-based spectrum sharing,“ *ACM Mobile Networks and Applications (MONET),* vol. 11, no. 3, pp. 405418, 2006.

[6] C. Kloeck, H. Jaekel, and F. K. Jondral, "Dynamic and local combined pricing, allocation and billing system with cognitive radios,“ *Proc. IEEE DySPAN 2005,* pp. 7381, 2005.

[7] L. Cao and H. Zheng, "Distributed spectrum allocation via local bargaining,“ *Proc. IEEE DySPAN 2005,* pp. 475486, 2005.

[8] H. Zheng and C. Peng, "Collaboration and Fairness in Opportunistic Spectrum Access," *Proc. 40th IEEE Int. Conf. on Communications (ICC),* pp. 3132-3136, 2005.

[9] Randy L. Haupt and Sue Ellen Haupt. Practical Genetic Algorithms. John Wiley & Sons. 2004.

[10] Goldberg DE., "Genetic algorithms in search, optimization and machine learning,“ Reading, MA: Addison-Wesley Publishing, Co 1989.