Applying CHC Algorithms on Radio Network Design for Wireless Communication

D. Molina[1], D. Pandolfi[1], A. Villagra[1], G. Leguizamón[2]

[1] Universidad Nacional de la Patagonia Austral, Argentina
{dmolina, dpandolfi, avillgra}@uaco.unpa.edu.ar
[2] Universidad Nacional de San Luis, Argentina
legui@unsl.edu.ar

Abstract. The diffusion of wireless communication services (telephone, internet, etc.) is continuously growing these days. Unfortunately, the cost of the equipment to provide the service with the appropriate quality is high. Thus, selecting a set of geographical points allowing optimum coverage of a radio frequency signal by minimizing the use of resources is essential. The above task is called the Radio Network Design (RND) and is a NP-hard problem, i.e., this can be approached by using metaheuristics techniques. Metaheuristics are methods comprising local improvement procedures and high-level strategies for a robust search in the problem space. In this work, different versions of the CHC algorithm with a fitness function based on the efficiency of resource use are proposed. The achieved results are encouraging in terms of efficiency and quality in all the analysed scenarios.

1 Introduction

Nowadays wireless media are perhaps the largest generators of the growth in communications. A message is encoding using electromagnetic waves between the transmitter and the receiver in order to let the messages exchange. Waves are subject to noise through the free space (attenuation, reflection, refraction, and diffraction) caused by atmospheric phenomena and obstacles [13]. Consequently there are losses in the signal level and the receiver may not be able to decode the message. The focus of this work is based on radio frequency, allowing to extend the application of metaheuristics to any service (radio, television, Internet, cell phone, etc.) [8]. Hence is a paramount challenge to achieve positioning the antennas where the most of the receivers can have an unobstructed line of sight and thus, ensuring as much as possible a high quality signal level during the whole transmission time. The task for solving this problem in the traditional method is called RND [7]. In this work we propose to analyze and evaluate population-based metaheuristics, more precisely a variant of Genetic Algorithms (GAs) [5] to find an acceptable solution for the RND problem. Former versions of GAs are implemented with one cut point, two cut point and uniform recombination operators. The CHC Algorithm (Crossover elitism population, Half uniform crossover combination, Cataclysm mutation) is a variant of a GA proposed in [4]. Our proposed CHC Algorithm consists of the implementation of three alternative methods for the "Cataclysm mutation" which used in CHC for breaking a local optimum convergence: (1) the classical mutation method, (2) retaining a proportion of bits of the best individual method, and (3) a local search iterated hybridization [6]. Besides we introduce a variation mechanism of the Hamming distance for incest prevention and also a fitness function based on the efficiency of surface coverage with respect to the radiated signal. In this way, each installed resources (e.g., antennas) will be used to its full potential, and consequently minimize the amount of them necessary to fulfill the objective. This paper is organized as follows: Section 2 contains the problem definition and explains the proposed fitness function. Section 3 describes the proposed algorithms and develop alternative mechanism to detect the population convergence. Section 4 presents the experimental study and the achieved results, and finally in Section 5, the conclusions of this work are given.

2 Problem Definition

According to [3] the problem of positioning antennas can be described as: given a set of candidate sites, a discretized geographic area, and a set of points that need to intercommunicate with each other; the objective is to select a subset of sites from the set of candidate sites that maximize the coverage by using a minimum number of resources and satisfying the traffic estimation and threshold signal reception between points. The signal level between the transmitter and receiver [15] determines the available bandwidth, being the indicator to satisfy the estimated traffic and threshold reception. This is regardless of the equipment capacity, technology or service to provide (e.g., radio, television, mobile phone, Internet, etc.). In order to discretize the ground surface, a matrix M of $f \times c$ is used, relating the latitude and longitude with the subscripts i, j of the cells in the matrix. Then, the height of each point at sea level is stored in each cell in M. This form of discretization responds to the raster model used in geographic information systems (GIS) and has been used in [9] and [10] to solve the RND problem. The candidates for the installation of antennas sites are represented as a subset of cells belonging to the matrix M. For this problem, omnidirectional antennas (radiating in all directions) are used and a ground plane is considered. Figure 1 shows the achieved coverage by a possible radio network. The value 1 in a cell indicates that the site is covered by a single antenna brand, the value -1 indicates that the site is covered by more than one cell; therefore, an interference is produced. Finally, a value 0 means that the site is not covered by any antenna. Each possible radio network is modeled by the activation or not of each of the candidate sites. The problem search space is determined by 2^n where n is the number of candidate sites.

The fitness function uses the values in the ground matrix M and assigns a unique numerical value to each solution. This value is a measure of the quality of the radio network evaluated and leads to different search algorithms on the radio network to maximize coverage, and minimize interference and use of resources.

1	1	1	0	0	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0	0	0
1	1	1	0	0	0	1	1	1	0	0	0
0	0	0	0	0	0	1	1	1	0	0	0
0	0	0	0	0	0	1	1	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	-1	1	1	0	1	1	1
0	0	0	1	1	-1	1	1	0	1	1	1
0	0	0	1	1	-1	1	1	0	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0

Fig. 1. Discretized ground by a matrix with possible covering values for a candidate network radio

The amount of $1s$, $0s$, and $-1s$ indicate respectively the total number of covered sites, uncovered sites, and illuminated sites by more than one antenna. Equation 1 displays the fitness function using the values obtained, the first factor determines the degree of covering ground and the second factor determines the quality of the covering ground. When the factors tend to one, the maximization of the covered area and the minimization of the use of resource is satisfied. Finally, the closer to 1 is the fitness value, the better is the solution (i.e., the maximum coverage with minimum use of resources).

$$f(x) = (1 - \frac{\#UncoveredPoints}{\#GroundPoints}) \times (1 - \frac{\#InterferedPoints}{\#CoveredPoints}) \quad (1)$$

3 CHC Algorithm

The CHC Algorithm combines a conservative selection strategy that preserves the best individuals found [4],[14]. The recombination operator generates offspring that maximizes their genetic differences from their parents. Reproduction occurs only if the Hamming distance (genetic difference between the parents) is greater than the threshold of incest (1/4 the size of the chromosome). The half uniform crossover scheme (HUX) is used to maximize the genetic distance between individuals. The new population is generated with a selection of the best individuals between parents and children. For each not improving generation, the genetic distance is decreased if the number of bits different from the selected parents is less than a threshold, i.e., the individuals are too similar. This determines that the population has converged, so that the population is restarted by triggering a *cataclysm* method[2]. Algorithm 1 shows the pseudo-code of CHC.

3.1 Proposed CHC Algorithms

For the RND problem with large chains bits (e.g., 200 or more bits) incest threshold for restarting the population demands a large computational effort

Algorithm 1 CHC pseudo-code

```
1:  t = 0 // time generation
2:  Initialize (Pa, incestThreshold)
3:  while not (endCondition(t, Pa)) do
4:     parents=parentsSelection(Pa,incestThreshold)
5:     offspring=HUX(P(t))
6:     evaluate(offspring, parents)
7:     Pn = elitismSelection(offspring, parents)
8:     if not (improvement(Pa,Pn))
9:        incestThreshold=incestThreshold-1
10:         if incestThreshold <= k
11:           Pn = cataclysm(Pa)
12:        end if
13:    endif
14:    t = t+1
15:    Pa = Pn
16: end while
17: return best solution
```

and number of iterations. The proposed method uses the genetic variability of the population to measure the level of diversity of the individuals. It is based on the probability of acceptance of the Simulated Annealing (SA)[1], but in reverse way as the temperature increases with the number of iterations and therefore increasing the chance of a cataclysm. For each unimproved iteration, the variability of the generated population is compared against a variability pattern. The difference between these two variability patterns must always be lower than the environment centered on the pattern variability and the radius must always be equal to the probability of acceptance. As iterations progress, the environment is reduced and if the variability of the generated population is outside of the environment, then the cataclysm is triggered. The variability v is defined by:

$$v = \frac{\sum_{i=1}^{n} \left(\sum_{j=1}^{c} x_{ij} \right)}{n \times c}$$

where $x \in (0,1)$, v is the population variability, c the chromosome size and n the number of individuals in the population.sThe acceptance environment ea is defined by:

$$ea = 1 - e^{\left(\frac{(f(x')-1)}{T} \right)}$$

where $f(x')$ is the local optimum value and T the number of iterations without improvements. The proposed based CHC algorithm is obtained by replacing the lines 9 to 13 of Algorithm 1 by the pseudo-code displayed in Algorithm 2. The parameter "method" in line 5 varies from values 1 to 3 (see explanation below) giving rise to the three CHC versions proposed here.

Algorithm 2 Pseudo-code for detecting population convergence

```
varPattern =0.5 // variability pattern
  standingPopulationVariability(Pa)
  acceptationRadius(T, fitnessUnimprovedValue)
   if variability is out of range
     Cataclysm(Pa, method)
   else
     T = T+1 // T increases and decreases acceptance range
  end if
```

The restart operators proposed for this work are: (1) Random Method (RM), the new population keeps a copy of the best individual and the rest individuals are obtained by mutating the best individual [11]; (2) Bits Conservation Method (BCM), the new population is generated retaining the best individual and varying only a percentage of bits of the original string [12]; (3) hybridization with Iterated Local Search (ILS). The proposed CHC versions differ from the canonical CHC in the mechanism to escape from suboptimal regions, so they are called QCHC (Quasi CHC). To identify the method used in each version we call QCHC-RM for method (1), QCHC-BCM for method (2), and QCHC-ILS for method (3).

4 Experiments and Analysis of Results

To analyze the performance of the proposed CHC Algorithms (QCHC-RM, QCHC-BCM, and QCHC-ILS) we additionally considered a set of GAs with different crossover operators: one-point crossover (GA-OPX), two-point crossover (GA-TPX), and uniform crossover (GA-UX). For each problem instance (i.e., matrix) the number of candidate sites was varied. The configuration of the seven instances used are shown in Table 1. Column *Instance* displays the name of each instance, column *Matrix* shows the matrix size used to discretize the respective ground scenery. Column *size* indicates the number of cells of each ground matrix as a result of product of the #rows by #columns. Column *Sites* indicates the number of candidates for each instance. The behavior of the studied algorithms is analyzed using various degrees of complexity related to the matrix dimensions and number of sites conforming the optimal solution. The antennas used for each site are omnisdirectional and they cover nine cells per antenna on the ground matrix, i.e., if placed in c_{ij} the covering radius is one cell on all directions.

The GAs use a population of 100 individuals and 5,000 iterations with a probability of crossover and mutation of 0.9 and 0.001, respectively. For the QCHC-RM Algorithm the mutation probability is set to 0.05; for QCHC-BCM was used a template pattern of 95%, varying only 5% of the bits of the best solution. All the QCHCs algorithms used a population of 500 individuals and $2,500$ iterations. In all cases, the best individual was always kept in the new population. For each of the instances of the experiment, 30 independent runs

Table 1. Description of the set of instances considered

Instance	Matrix	Size	Sites
Inst_01	12x12	144	32
Inst_02	15x15	225	50
Inst_03	18x18	324	72
Inst_04	21x21	441	98
Inst_05	24x24	576	128
Inst_06	27x27	729	162
Inst_07	30x30	900	200

were performed to ensure statistical significance. Windows 8 is used as a platform running on an Intel Core @ processor i7-3630QM 12 GB RAM and 1 TB hard disk. To analyze the performance of the proposed algorithms, only are compared those which obtain the 100% of success in reaching the ideal coverage. Also, the appropriate statistical tests are applied to the following performance variables: number of evaluations and execution time. Table 2 shows the rate success obtained by each of the algorithms in the seven instances. Column *Instance* contains the reference of the various configurations described in Table 1. The remaining columns have the name of each used algorithm and the success rate obtained on each instance. The displayed values indicate the percentage of success obtained by each algorithm. Row *average* shows a first approximation on the efficiency of each algorithm according to the complexity of each instance. We can observe that the proposed QCHC-ILS Algorithm achieves 100% success over the all instances. QCHC-RM achieves 100% success in five of the seven instances. On the side of the classical GAs, we can see that GA-TPX obtains the highest percentage (81.90%) of success (Inst_01, Inst_02, and Inst _03) with respect to GA-UX and GA-OPX.

Table 2. Success rate obtained by GA-UX, GA-OPX, GA-TPX, QCHC-RM, QCHC-BCM and QCHC-ILS Algorithms

Instance	GA-UX	GA-OPX	GA-TPX	QCHC-RM	QCHC-BCM	QCHC-ILS
Inst_01	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**
Inst_02	93.33%	70%	**100%**	**100%**	**100%**	**100%**
Inst_03	66.7%	66.7%	**100%**	**100%**	90%	**100%**
Inst_04	**100%**	**100%**	86.67%	**100%**	90%	**100%**
Inst_05	90.00%	80.00%	70.00%	**100%**	93.33%	**100%**
Inst_06	63.33%	66.67%	86.67%	90%	93.33%	**100%**
Inst_07	26.67%	33.33%	30%	43.33%	86.67%	**100%**
Average	77.41%	73.81%	81.90%	90.48%	93.33%	**100%**

A significance level of $\alpha = 0.05$ was used for all the applied statical tests. Due to the nature of the experiments, the values obtained by the algorithms satisfy the condition of independence, thus Kolgomorov-Smirnov test is applied to determine whether the data fit a normal distribution and the Levene test is applied to verify whether the data have homoscedasticity. As the results obtained by the algorithms do not meet the conditions of normality and homoscedasticity, non-parametric test (Kruskal-Wallis) was used for the algorithms on instances Inst_01 to Inst_04. The Wilcoxon test is applied to the algorithms on the Inst_05 since only two algorithms are compared: QCHC-RM and QCHC-ILS.

Tables 3 and 4 show the results obtained by the statistical tests for performance variables of time and number of function evaluations to achieve the best solution. For both tables, column *Instance* indicates the scenario configuration, columns GA-UX, GA-OPX, GA-TPX, QCHC-RM, QCHC-BCM, and QCHC-ILS show respectively the average values obtained for each evaluated algorithm on the instances that achieved 100% of success. N/A (Not Available) indicates that the algorithm does not meet this condition. Column KW/W represents the initials of the Kruskal-Wallis and Wilcoxon test respectively. The symbols (+) and (−) stand respectively for there are and there are not statistical difference. The symbol $*$ indicates that the Wilcoxon test was applied. Tukey's test applies in all cases where statistically significant differences are detected to identify the best performing algorithm.

Table 3. Average values for the performance variable time and the statistically significant differences obtained in each test

Instance	GA-UX	GA-OPX	GA-TPX	QCHC-RM	QCHC-BCM	QCHC-ILS	KW/W
Inst_01	1.710	2.808	1.953	0.424	1.020	**0.405**	(+)
Inst_02	N/A	N/A	3.317	18.219	7.639	**0.754**	(+)
Inst_03	N/A	N/A	5.674	3.634	N/A	**1.127**	(+)
Inst_04	9.878	17.528	N/A	6.664	N/A	**1.806**	(+)
Inst_05	N/A	N/A	N/A	12.562	N/A	**2.527**	*(+)

Table 4. Average values for the performance variable number of evaluations and the statistically significant differences obtained in each test

Instance	GA-UX	GA-OPX	GA-TPX	QCHC-RM	QCHC-BCM	QCHC-ILS	KW/W
Inst_01	17023	28217	19663	6550	16167	**6417**	(+)
Inst_02	N/A	N/A	32307	298033	107733	**10550**	(+)
Inst_03	N/A	N/A	52793	40650	N/A	**13183**	(-)
Inst_04	86247	153673	N/A	67000	N/A	**18333**	(+)
Inst_05	N/A	N/A	N/A	112133	N/A	**22100**	(+)

We can see in Table 3 that the results obtained by QCHC-ILS in terms of performance time variable are lower than the remaining studied algorithms. The statistical test applied to the results obtained by the algorithms in terms of this performance variable indicate that there are statistically significant differences between the results. In all analyzed cases the differences arise between QCHC-ILS with respect to the other algorithms. In Table 4 we can observe that in the number of evaluations required to achieve the best solution, again the QCHC-ILS algorithm is one that requires the fewest number of evaluations in all instances tested. In this case, the algorithms present differences statistically significant in four of five instances. Here again statistically significant differences occur between QCHC-ILS with respect to the other algorithms. Figure 2 represents the number of instances with 100% of the hits obtained by each algorithm. We can observe that the proposed QCHC-ILS algorithm is the only one in all instances obtaining 100% success reaching the optimum value.

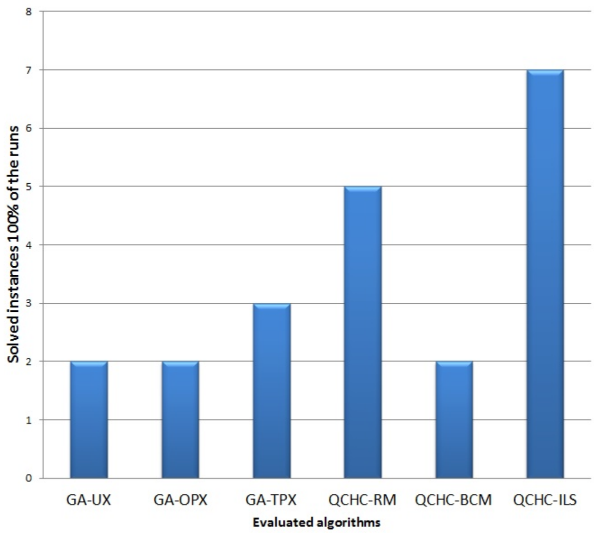

Fig. 2. Number of instances solved by the evaluated Algorithms

5 Conclusions

Three GAs called respectively GA-UX, GA-OPX, and GA-TPX analyzed, and three versions of the CHC algorithms called respectively QCHC-RM, QCHC-BCM, and QCHC-ILS are proposed and analyzed in this work. New incest prevention mechanism based on population variability and a fitness function considering coverage efficiency of signal was used in the scenarios. The proposed mechanism can detect incest prevention to avoid premature convergence at a low computational cost. All the CHC versions used the HUX operator and the elitist selection as usually applied in the former CHC algorithm. The QCHC-RM, QCHC-CBM and QCHC-ILS algorithms use the best individual to generate the new population. The QCHC-ILS algorithm, in addition, applies an iterated local search procedure that intensifies the search around each newest generated individual during the evolutionary process.

The first instance was solved by all competitors. However, as the complexity/size of the instances increase several algorithms were not able to reach the optimum in the 30 runs (i.e., a hit ratio less than 100%). From a statistical point of view, the hybrid algorithm (QCHC-ILS) is the best performing algorithm with respect to the complete set of studied algorithms. This is the only algorithm that achieves the optimal for all the instances considered. In addition to the performance variables of time and number of evaluations, the hybrid approach has the best performance with 95 % confidence level for the statistical test applied. This means that this version of the CHC Algorithm is an effective improvement over the competitors and the other QCHC versions. Future studies will address different size of problems in order to study scalability properties for large dimensional search spaces. Also, other aspects of the CHC improvements will be analyzed such as the selection mechanism and the use of parallel architectures.

Acknowledgments We acknowledge the constant support by the Universidad Nacional de la Patagonia Austral. The last author also acknowledges the constant support afforded by the Universidad Nacional de San Luis that finances his current researches.

References

1. V. Černỳ. Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. *Journal of optimization theory and applications*, 45(1):41–51, 1985.
2. P.K. Chawdhry, R. Roy, and R.K. Pant. *Soft Computing in Engineering Design and Manufacturing*. Springer London, 1998.
3. D. Corne, M.J. Oates, and G.D. Smith. *Telecommunications optimization: heuristics and adaptive techniques*. John Wiley, 2000.
4. L.J. Eshelman. The CHC Adaptive Search Algorithm: How to Have Safe Search When Engaging in Nontradi- tional Genetic Recombination. In *Foundations of Ge- netic Algorithms*, pages 265–283. Morgan Kaufmann, 1991.
5. D.E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Artificial Intelligence. Addison-Wesley, 1989.
6. O. Martin H. R. Lourenco and T. Stutzle. Iterated local search.in handbook of metaheuristics. In *Vol. 57 of Operations Research and Management Science*, pages 321–353. Kluwer Academic, 2002.
7. N. Gerlich K. Tutschku and P. Tran-Gia. An integrated Approach to Cellular Network Planning. In *Institute of Computer Science*. University of Wurzburg, 1995.

8. A.R.C. Lechtaler and R.J. Fusario. *Comunicaciones: una introducción a las redes digitales de transmisión de datos y señales isócronas*. Area: computación. Subárea: comunicaciones. Alfaomega, 2013.
9. F. Luna, J. J. Durillo, A. J. Nebro, and E. Alba. Evolutionary algorithms for solving the automatic cell planning problem: a survey. *Engineering Optimization*, 42(7):671–690, December 2010.
10. P. Kuonen D. Kobler P. Callegari, F. Guidec. Parallel island-based genetic algorithm for radio network design. In *Journal of Parallel and Distributed Computing (47)*, pages 86–90, 1997.
11. S.K. Pal and P.P. Wang. *Genetic Algorithms for Pattern Recognition*. Computer science mathematics. Taylor & Francis, 1996.
12. I.C. Parmee. *Evolutionary and Adaptive Computing in Engineering Design: With 98 Figures*. Springer London, 2001.
13. J.D. Parsons. *The Mobile Radio Propagation Channel*. Wiley, 2000.
14. S. Priem-Mendes, G. Molina, M. A. Vega-Rodríguez, J. A Gómez Pulido, Y. Sáez, G. Miranda, C. Segura, E. Alba, P. Isasi, C. León, and J. M. Sánchez-Pérez. Benchmarking a wide spectrum of metaheuristic techniques for the radio network design problem. *IEEE Trans. Evolutionary Computation*, 13(5):1133–1150, 2009.
15. R.T. S. *Wireless Communications: Principles And Practice, 2/E*. Pearson Education, 2010.