A Genetic Implementation of Graph Coloring for Cellular Network

Vinod Mann¹ & Dhiraj Sangwan²

¹Assistant Professor, Computer Science Department
²Lecturer, ECE Department, Mody Institute of Technology and Science, Lakshmangarh

Abstract: The aim of this paper is to introduce a new evolutionary formulation of the graph coloring[9] problem based on compatible matrix for cellular network. In cellular network set of channels (colors) must be assigned to cells (vertices) while avoiding interference. We have used the coloring task of assigning sets of colors to vertices (cells) of cellular networks so that the sets of colors assigned to adjacent vertices are disjoint. In this paper we propose a methodology for channel assignment so that by using minimum number of colors with constraint to be followed that no adjacent cell are having same color.

Keywords: Channel Assignment, Graph Coloring, Compatible Matrix and Cellular Network.

INTRODUCTION

The objective of channel assignment[8] is to assign a required number of channels to each cell such that efficient frequency spectrum utilization is provided and interference effects are minimized. A fixed channel assignment[14] problem models the task of assigning channels to a set of cells on a permanent basis. The major problem in designing cellular network is interferences like co channel, adjacent channel and co site channel interferences [10]. In this approach the same cell pattern is to be repeated again and again to cover entire network area so that the co channel cells are always far apart from themselves in the same network. Co cells make a significant amount of interference that they are far apart. Adjacent cells although having wide frequency separation make interference because of shortest distance between cell sites. In fixed channel assignments the nodes are portioned in two independent sets, and each such set is assigned a separate set of channels [10]. Co channel interference constraints are modeled in terms of reused distance; it is assumed that the same channel can be assigned to two different nodes in the graph if and only if their graph distance is at least r.

In this paper, we model cellular network as graph with each node representing a base station in a cell in the network and edges representing geographical adjacency of cells. A graph equivalent matrix $G$ is drawn where $G_{ij}$ indicates the minimum path length between vertices $i$ and $j$. The G matrix is converted to an interference matrix $I$ where each element $I_{ij}$ indicates degree of interferences between cell $i$ and $j$[17].

GENETIC ALGORITHM

Genetic Algorithms[4] are adaptive methods, which may be used to solve search and optimization problems. They are based on the genetic process of biological organisms. Before a genetic algorithm can be run, an encoding (or representation) for the problem must be devised. A fitness function, which assigns a figure of merit to each encoded solution, is also required. An initial population of size N of randomly selected potential solutions (binary strings called chromosomes) is constructed and each solution is evaluated. In a direct representation, a chromosome represents a solution of the original problem, and is usually called genotype, each potential solution in the population is considered as an individual and the objective function value for an individual is referred to its fitness. The fitness of an individual is a measure of its relative functional performance. A single entity in a chromosome is called a gene and a value that can be assigned to gene is called an allele. Then, in the selection step, two of the above chromosomes at a time are selected from the population. The individuals having higher fitness values are more likely to be selected. The solution space is traced using three basic genetic operators, namely, selection, crossover and mutation. The selection operator allows individual strings to be copied into a mating pool for a possible contribution in the next generation. The chance that a string will be copied in the mating pool is proportional to its fitness value. The crossover takes two randomly chosen individuals from the mating pool and recombines them to generate two new individuals, which carry a mixture of their parental genetic information. There are different crossover operators namely simple (single point), order, partially mapped, and cycle. The simple crossover, for instance, works by choosing a random cut point in both parent chromosomes (the cut point should be the same in both parents) and generating the offspring by combining the segment of one parent to the left of the cut point with the segment of the other parent to the right of the cut. In this implementation, the simple crossover is used. Mutation is applied to each child
individually after the crossover. It randomly alters each gene with a very small probability. The mutation operator is used to introduce new random information in the population. Mutation provides a small amount of random search, and helps to ensure that no point in the search space has a zero probability of being examined. Mutation is usually applied to some individuals, to guarantee population diversity. If the GA has been correctly implemented, the population will evolve over successive generations so that the fitness of the best and average individuals in each generation increases towards the global optimum. Convergence is the progression towards increasing uniformity. Practically, a population is said to have converged when the best individual found so far has the desired fitness value.

The Abstract form of a Simple Genetic Algorithm is given below:

\[\text{Procedure Genetic Algorithm}\]
1. Generate Initial population: Inipop;
2. Evaluate population: fitnessmat;
3. Initialize generation counter: \( g \);
4. While stopping criteria not satisfied repeat;
5. Select some elements from Inipop to form mating pool;
6. Crossover some elements of mating pool and form bitmatrix;
7. Mutate some elements of bitmatrix to form new Population;
8. Evaluate new population;
9. Increment generation counter: \( g \) \( g+1 \);
10. end while;
end Genetic Algorithm

1. PROBLEM FORMULATION

In case of cellular networks the basic problem is to assign a set of radio channels to transmitters so as to avoid unacceptable interference which results because of using the same frequency. Adjacent cells although having wide frequency separation make interference because of shortest distance between cell sites. A minimum separation is assumed between channels assigned to two transmitters depending on the proximity of the transmitters.

In this paper a generalized channel allocation scheme based on genetic algorithm has been presented.

A cellular network of 17 cells is taken as the test network. The genetic algorithm has been used to verify the minimum number of colors for all the co-cells in this network.

Figures 1-3

For the given cellular network an equivalent graph has been constructed to show the cells as nodes connected by edges. A 17x17 matrix is to be described based on graph theory.

\[
G = \begin{bmatrix}
0 & 1 & 1 & 1 & 1 & 1 & 3 & 2 & 2 & 2 & 2 & 3 & 3 & 2 & 2 & 2 & 2 \\
1 & 0 & 1 & 2 & 2 & 1 & 4 & 3 & 3 & 2 & 3 & 3 & 4 & 3 & 3 & 3 & 3 \\
1 & 1 & 0 & 1 & 2 & 2 & 2 & 3 & 3 & 3 & 3 & 3 & 4 & 4 & 3 & 3 & 3 \\
1 & 2 & 1 & 0 & 1 & 2 & 2 & 2 & 1 & 1 & 2 & 2 & 3 & 2 & 2 & 3 & 2 \\
1 & 2 & 2 & 1 & 0 & 1 & 2 & 2 & 1 & 2 & 2 & 3 & 3 & 2 & 1 & 1 & 1 \\
1 & 2 & 2 & 1 & 0 & 1 & 3 & 2 & 3 & 3 & 4 & 4 & 2 & 1 & 2 & 1 & 2 \\
1 & 1 & 2 & 2 & 1 & 0 & 4 & 3 & 3 & 3 & 4 & 4 & 3 & 2 & 3 & 3 & 3 \\
3 & 4 & 3 & 2 & 2 & 3 & 4 & 0 & 1 & 1 & 2 & 2 & 3 & 3 & 3 & 3 & 3 \\
3 & 4 & 3 & 2 & 2 & 3 & 4 & 1 & 0 & 1 & 2 & 3 & 3 & 4 & 2 & 2 & 1 \\
2 & 3 & 2 & 1 & 1 & 2 & 3 & 1 & 0 & 1 & 2 & 2 & 3 & 2 & 2 & 2 & 1 \\
2 & 3 & 3 & 1 & 0 & 1 & 1 & 2 & 3 & 3 & 2 & 1 & 1 & 1 & 2 & 3 & 3 \\
2 & 2 & 1 & 1 & 2 & 3 & 3 & 2 & 2 & 3 & 2 & 1 & 1 & 1 & 4 & 3 & 3 \\
3 & 3 & 2 & 2 & 3 & 4 & 2 & 3 & 2 & 1 & 1 & 1 & 4 & 4 & 3 & 3 & 3 \\
3 & 3 & 4 & 2 & 3 & 4 & 4 & 3 & 4 & 3 & 3 & 1 & 1 & 0 & 5 & 4 & 4 \\
3 & 4 & 4 & 3 & 2 & 2 & 3 & 3 & 2 & 2 & 4 & 4 & 5 & 1 & 1 & 1 & 0 \\
2 & 3 & 3 & 2 & 1 & 1 & 2 & 3 & 2 & 2 & 2 & 3 & 4 & 4 & 1 & 1 & 0 \\
2 & 3 & 3 & 2 & 1 & 2 & 3 & 1 & 1 & 1 & 3 & 3 & 4 & 4 & 1 & 1 & 0
\end{bmatrix}
\]
For the given network we are using a seven cell pattern where each cell is in neighborhood of six other cells. After obtaining \( G \) matrix it is transformed into an interference matrix that would indicate the degree of interferences. The elements of interference matrix are obtained as \( I_{ij} = 7 - G_{ij} \). Since the interference matrix is symmetric so we can use only either upper or lower triangular matrix for the analysis. After this a compatible matrix is formed with a constraint such that the minimum path length between co-cells is 3. Hence \( I_{ij} = 7 - 3 = 4 \) denotes co-cell that is frequency could be reused.

The interference matrix is transformed to an equivalent compatible matrix \( C \) based on

\[
C_{ij} = \begin{cases} 
I_{ij} - 4 : & \text{for } I_{ij} > 3 \\
0 : & \text{otherwise}
\end{cases}
\]

The co-cells are found using the matrix \( C \) by adopting the code mentioned below Pseudo code to find co-cells\[17\]

1. Use matrix \( C \) to find all co-cells of cell \( i \); 
2. Find the interference between cells of step 1 using elements of matrix \( C \); 
3. Exclude cells \( p \) and \( q \) if \( C_{pq} > 0 \); 
4. Set \( i = i + 1 \) and repeat 2 and 3.

The tabular representation of co-cells(fig.5) is presented below which is the implementation of the code mentioned above.

2. GENETIC IMPLEMENTATION

Genetic Algorithms generally have two steps; first, the initial population is needed. Second, for each generation GA are operated against an objective function which has been designed such that it optimizes GA toward global maxima. If a population element has better score it is more likely towards the maxima. The strength of GA lies in finding very quickly good optimal solutions in a complex search space. It is an iterative procedure that maintains a set of candidate solutions called the population \( P(t) \) for each iteration \( t \). At each iteration a new population \( P(t+1) \) is created from the previous population using a set of genetic operators.

A population can be represented by an array of strings (individuals). The row of the array represents strings in a population and the column represents the channel numbers which will be assigned. There are \( P \) strings in a population and each string has 51 bits to represent the 3 bit information of 17 cells used. Thus

\[
Q = \text{summation of total number of bits}
\]

A string \( S \) is composed of \( N \) substrings which is the number of cells in the network. Each substring \( S_i \) for cell \( i \) is composed of \( n \) bits.

For example, the \( r \)th string \( S_r \) in population \( P(t) \) may be \( S_r = (1, 2, 3, \ldots, j, \ldots, Q) \) and a substring \( S_{ri} \) for the cell \( I \) in the string \( S_r \) may be \( S_{ri} = (1, 2, \ldots, m_i) \), therefore \( S_r = S_{r1}S_{r2}\ldots S_{rN} \). A \( PxQ \) two dimensional array is constructed to implement a number of strings (a population) is shown in figure 6.
Once the co-cell table is obtained after this it is fed into the fitness function of the genetic algorithm where every population element is evaluated for its fitness corresponding to the bitpattern which it represents. The better individuals are chosen for crossover and mutation and new population elements are formed which are further evaluated in next genetic cycle.

For the network taken as example the following parameters are considered
1) Each chromosome is 51 bits long;
2) Each cell is represented using 3 bit binary pattern;
3) Total number of colors used can by 9 in maximum;
4) The maximum fitness obtained is 150;
5) Genetic Parameters taken are;
   a) Population size: 40
   b) Probability of Crossover: 0.8
   c) Probability of Mutation: 0.002.

3. RESULTS
The genetic algorithm is used to verify the minimum number of colors used for the cocells, it has been verified that by using 3 bit pattern for each co-cell in the network the minimum number of colors used is 7 out of 9 at maximum. The maximum fitness obtained for the population element in a genetic cycle is 150.

   01100101101001101
Columns 37 through 51
001001101010111

The following figure 7 shows the result of the different genetic cycles and the maximum fitness obtained.

4. CONCLUSION AND FUTURE WORK
In a cellular network the channel assignment to the active cells has been considered with the constraint that the neighboring cells do not have the same color used. By forming a compatible matrix it has been found that the cocells are at a uniform distance. A genetic framework has been presented to verify the minimum number of colors used for the cellular network. The work can be further extended to find explore the possibilities in bigger networks with more chances of interference from the nearby cells. It can also be found that the static allocation of colors would be minimum by exploring more networks.

REFERENCES
[1] Bruce Reed and Benny Sudakov, “List Coloring When the Chromatic Number is close to the Order of the Graph”, Combinatorica, 25(1), (2005), 117-123.

