A Performance Study of Genetic Algorithm-Assisted Beamforming in Distributed Cognitive Radio Networks

Andrew Minturn
University of Nebraska-Lincoln, aminturn@unomaha.edu

Follow this and additional works at: http://digitalcommons.unl.edu/ceendiss
Part of the Computer Engineering Commons

http://digitalcommons.unl.edu/ceendiss/34

This Article is brought to you for free and open access by the Electrical & Computer Engineering, Department of at DigitalCommons@University of Nebraska - Lincoln. It has been accepted for inclusion in Theses, Dissertations, & Student Research in Computer Electronics & Engineering by an authorized administrator of DigitalCommons@University of Nebraska - Lincoln.
A PERFORMANCE STUDY OF GENETIC ALGORITHM-ASSISTED BEAMFORMING IN DISTRIBUTED COGNITIVE RADIO NETWORKS

by

Andrew Minturn

A THESIS

Presented to the Faculty of
The Graduate College at the University of Nebraska
In Partial Fulfillment of Requirements
For the Degree of Master of Science

Major: Telecommunications Engineering
Under the Supervision of Professor Yaoqing Yang
Lincoln, Nebraska
December 2014
A PERFORMANCE STUDY OF GENETIC ALGORITHM-ASSISTED BEAMFORMING IN DISTRIBUTED COGNITIVE RADIO NETWORKS

Andrew Minturn, M.S.

UNIVERSITY OF NEBRASKA 2014

Adviser: Yaoqing (Lamar) Yang

Cognitive Radio (CR) is a technology that has gained much interest recently due to the increasing scarcity of the radio frequency spectrum. Large portions of the radio frequency spectrum are licensed to users who then have exclusive access to the bandwidth, and unlicensed bands can be a challenge to use due to interference from unlicensed users. Despite the seeming scarcity, tests of bands allocated by the Federal Communications Committee (FCC) to licensed and unlicensed user have shown that many are underutilized and often unoccupied by the user to whom they are licensed. CR aims to exploit this unused spectrum and thus use it more efficiently. This must be accomplished in a way that does not obstruct licensed communications. To achieve this, CR must sense the availability of the spectrum in one or more dimensions such as space, time, frequency, etc. and adjust its parameters to communicate in the unoccupied spectrum. In this thesis, a CR network uses beamforming to communicate in the presence of a primary network such that a beam is steered towards cognitive receivers while a null is steered toward primary receivers to prevent interference. As the number of users in the primary and secondary networks increase and the constraints become
more varied, a closed-form solution becomes more difficult to find. Due to the potentially dynamic nature of the network parameters and constraints, it would be of interest to study the application of a general algorithm that can be used to solve the problem. The Genetic Algorithm (GA) is a broadly applicable algorithm inspired by evolutionary biology in which solutions are encoded onto “chromosomes” and go through a process of natural selection to optimize some function. For this work, the GA was used to optimize the CR beamforming problem under various networks configurations and the effect of the GA parameters on its performance were studied.
ACKNOWLEDGEMENT

I would like to sincerely thank my advisor Dr. Yaoqing Yang for all of his help and guidance during my graduate studies, in research, and in writing my thesis. He is a patient and caring teacher and I am grateful for all of the knowledge and experience that he has given to me.

I would also like to thank Dr. Lim Nguyen and Dr. Yi Qian for giving me their support in my coursework and in writing my thesis. I have gained indispensable knowledge and experience from studying under such accomplished engineers.

Finally, I would like to thank my mother, Mary Ann Wendland, and my sister, Megan Minturn, for giving me unconditional love and support throughout my life and always believing in my abilities.

Andrew Minturn
Table of Contents

Chapter 1. Introduction ........................................................................................................... 1
  1.1 History and Motivation for Cognitive Radio .................................................................... 1
  1.2 Introduction to Cognitive Radio Networks ................................................................. 3
  1.3 Introduction to Beamforming in Wireless Communications ....................................... 6
  1.4 Introduction to the Genetic Algorithm .......................................................................... 7

Chapter 2. Beamforming Techniques in Wireless Communications ................................. 8
  2.1 Background ..................................................................................................................... 8
  2.2 Array Factor .................................................................................................................. 8
  2.3 Diversity Gain ............................................................................................................... 8
  2.4 Array Beamforming ...................................................................................................... 9
  2.5 Distributed Beamforming ............................................................................................. 11
  2.6 Effects of Imperfect Carrier Synchronization ............................................................. 14

Chapter 3. Genetic Algorithm ............................................................................................. 15
  3.1 Background .................................................................................................................. 15
  3.2 The Canonical Genetic Algorithm .............................................................................. 16
    3.2.1 Evaluation and Fitness Function ........................................................................... 18
    3.2.2 Crossover Selection .............................................................................................. 19
    3.2.3 Crossover ............................................................................................................. 21
    3.2.4 Mutation .............................................................................................................. 21

Chapter 4. Performance Study of GA-Assisted Beamforming in a Distributed Cognitive Radio Network ............................................................... 23
  4.1 System Model ............................................................................................................... 24
  4.2 Analysis ......................................................................................................................... 25
  4.3 Results ......................................................................................................................... 26

Chapter 5. Conclusion ......................................................................................................... 35

Appendices ........................................................................................................................... 37

References .............................................................................................................................. 42
### Table of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Utilization of a 7 MHZ band below 1 GHz</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>Cognition Cycle [1]</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>Array Beamformer with M elements with spacing d</td>
<td>10</td>
</tr>
<tr>
<td>5</td>
<td>Distributed Beamforming Network</td>
<td>12</td>
</tr>
<tr>
<td>6</td>
<td>Average beampattern of N randomly placed nodes in circle of radius R</td>
<td>14</td>
</tr>
<tr>
<td>7</td>
<td>Genetic Algorithm Flowchart</td>
<td>17</td>
</tr>
<tr>
<td>8</td>
<td>Crossover in the GA</td>
<td>19</td>
</tr>
<tr>
<td>9</td>
<td>Distributed CRN System Model</td>
<td>24</td>
</tr>
<tr>
<td>10</td>
<td>GA Parameters</td>
<td>27</td>
</tr>
<tr>
<td>11</td>
<td>Mean and Best Fitness</td>
<td>28</td>
</tr>
<tr>
<td>12</td>
<td>BP with two PUR's, N = 16, R = 4</td>
<td>29</td>
</tr>
<tr>
<td>13</td>
<td>BP with varying N, R = 4, PUR1 = [20 25], PUR2 = [95 100], SU = -20</td>
<td>30</td>
</tr>
<tr>
<td>14</td>
<td>Primary Protection Region PUR = [20 25]</td>
<td>31</td>
</tr>
<tr>
<td>15</td>
<td>BP with varying R, N = 16, PUR1 = [20 25], PUR2 = [95 100], SU = -20</td>
<td>32</td>
</tr>
<tr>
<td>16</td>
<td>SU Power vs. power constraint for different N</td>
<td>33</td>
</tr>
<tr>
<td>17</td>
<td>SU receiver power vs. population size with various PU power constraints, N=8</td>
<td>34</td>
</tr>
</tbody>
</table>
### Nomenclature

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CR</td>
<td>Cognitive Radio</td>
</tr>
<tr>
<td>PU</td>
<td>Primary User</td>
</tr>
<tr>
<td>SU</td>
<td>Secondary User</td>
</tr>
<tr>
<td>CRN</td>
<td>Cognitive Radio Network</td>
</tr>
<tr>
<td>BP</td>
<td>Beampattern</td>
</tr>
<tr>
<td>PUR</td>
<td>Primary User Region</td>
</tr>
<tr>
<td>FCC</td>
<td>Federal Communications</td>
</tr>
<tr>
<td>AOA</td>
<td>Angle of Arrival</td>
</tr>
<tr>
<td>SNR</td>
<td>Signal to Noise Ratio</td>
</tr>
<tr>
<td>GA</td>
<td>Genetic Algorithm</td>
</tr>
<tr>
<td>PLL</td>
<td>Phase Locked Loop</td>
</tr>
<tr>
<td>PDF</td>
<td>Probability Distribution</td>
</tr>
<tr>
<td>CDF</td>
<td>Cumulative Distribution</td>
</tr>
</tbody>
</table>
Chapter 1. Introduction

1.1 History and Motivation for Cognitive Radio

Cognitive radio promises a more efficient use of scarce radio spectrum and much research has been done into applying existing techniques in wireless communications to achieve the aims of this new technology. Interest in cognitive radio has risen due to the increasing scarcity of available frequency spectrum. Although licensed spectrum is scarce, surveys of licensed bands show that they are often unoccupied by the licensed user [2]. Therefore, it would be desirable if a technology existed that could exploit this unused spectrum, which is what cognitive radio aims to do. In the literature, the licensed user is commonly referred to as the primary user and the nodes in the cognitive radio network are referred to as the secondary users. The goal of the secondary users is to communicate in the licensed spectrum, while limiting interference to the primary user.

The push for cognitive radio technology is driven by the saturation of the licensed radio frequency spectrum as well as studies which show that it is being highly underutilized. The Federal Communication Commission (FCC) is the body responsible for allocating frequency spectrum and dictating how it is used in the United States. As seen in Figure 1, the radio frequency has been divided into frequency bands, each having a specific use [2].
Testing of some of these licensed bands by the FCC has shown that many are underutilized. Underutilized can mean that for large portions of time the band is unused or that in some areas signals from the licensed users are not present. These are referred to as the time dimension and the spatial dimension of the frequency spectrum. Figure 2 shows the utilization of a 7 megahertz band in three different major US cities.
Figure 2: Utilization of a 7 MHZ band below 1 GHz

The curves represent the percentage of a thirty second window that was occupied in a 7 MHz band below 1 GHz. As can be seen in the plot, even in a major city such as Chicago, the band was unoccupied for almost half of the time. Tests have also shown that many channels in some of the licensed bands are more underutilized than others. This would be considered the frequency dimension of the spectrum. According to a 2002 FCC Spectrum Policy Task Force Report, “the existence of such low average occupancy bands in urbanized areas with increasing spectrum demand raises interest in the possibility of matching supply and demand better with alternative technology”.

Therefore, it would be advantageous if a technology existed that was capable of detecting the state of the time, space, and frequency dimensions of the RF spectrum and adapting to this environment to improve spectral efficiency. This is what Cognitive Radio aims to do.

1.2 Introduction to Cognitive Radio Networks

An ideal Cognitive Radio Network (CRN) has the ability to detect the state of the RF environment, identify available spectrum, and exploit the available spectrum without interference to licensed users’ communication. The users in the licensed network are
referred to as primary users (PUs) and the users in the cognitive network are referred to as the secondary users (SUs). The SUs must exploit the spectrum of the PU network autonomously, because it is assumed that the PU network is a separate, independent network that does not provide feedback to the SU network. Therefore, it is critical that the SU network accurately assess the channel-state of the PU network and vacate a channel before the PU reoccupies it. This can be accomplished with spectrum sensing and predictive modeling [3]. According to [1], there are two primary objectives for cognitive radio networks:

- Highly reliable communications whenever and wherever needed
- Efficient utilization of the radio spectrum

In order to accomplish these objectives, a CRN must repeatedly go through a basic cognition cycle as shown in Figure 3.
The figure shows the tasks that must be accomplished by the CRN and the parameters that must be considered to achieve those tasks. The steps in the Cognition cycle can be summarized as follows [1]:

- **Radio scene analysis**: Radio scene analysis consists of sensing the interference temperature of the radio environment and detecting available spectrum.
• **Channel Identification**: Channel identification consists of estimation of the channel state information and estimation of the channel capacity.

• **Transmit-power control**: Transmit power control consists of controlling the transmission parameters of the network to manage the spectrum.

This cognition cycle consists of two key components: radio scene analysis and adaptation of transmission parameters. The adaptation of transmission parameters is the focus of this thesis.

### 1.3 Introduction to Beamforming in Wireless Communications

Beamforming (BF) is used in wireless communications in order to produce a high gain signal or construct a directional beam. Beamforming is accomplished through adjusting the amplitudes and phases of signals such that they combine constructively or destructively at a receiver. This can be used to direct a beam to a desired receiver or to direct a null in the beampattern to a receiver to avoid interference.

Beamforming can be performed by a linear array of transmitters or by a distributed network of transmitters. Due to the ability to manage the spatial dimension of spectrum, beamforming lends itself well to Cognitive Radio. Beamforming can be accomplished by either using the positions of the intended and unintended receivers or by using channel-state information (CSI) [10]. A more detailed survey of beamforming techniques is presented in Chapter 2.
1.4 Introduction to the Genetic Algorithm

Many problems that arise in CR BF do not have a clear analytical solution or require an analytical solution that is complex due to optimization problems formed with the constraints of a CRN. The Genetic Algorithm provides a way to arrive at reasonable solutions to these problems with a low complexity. The GA is an algorithm inspired by the biological process of natural selection. The basic idea of the GA is that for a given population, the more fit individuals will produce more offspring and each generation will be better adapted. The genetic diversity of the population changes due to random mutations and desirable mutations will be passed on to future generations. The GA is a simple algorithm that has been proven to be very powerful and widely applicable. [14] Another benefit of the GA is the potential for parallelization, which is attractive for a distributed application. A detailed description of the GA and its parameters will be discussed in Chapter 3.
Chapter 2. Beamforming Techniques in Wireless Communications

2.1 Background

One technique that lends itself well to Cognitive Radio is beamforming. Cognitive radio operates under the condition that the interference to a primary user is kept below a certain threshold or probability while communication between the secondary cognitive users is maintained. Studies into beamforming show that N nodes can cooperate together to achieve a gain approaching $N^2$ [4]. Additionally, beamforming can be used to steer a beam towards a desired receiver while steering a null to avoid interference to a different receiver. This capability can be applied to cognitive radio to limit interference to primary users by steering a null towards them [15].

2.2 Array Factor

The array factor represents the gain of the array in a particular direction. The array factor, $F(\phi)$, is a function of the azimuth angle, $\phi$. The array factor of an antenna array is determined by the weighting applied to each antenna and the geometry of the array.

[13]

2.3 Diversity Gain

Diversity gain refers to the gain achieved through the use of multiple transmit antennas. Diversity can be desirable in a wireless communication system, because the directivity
increases as $N^2$. For example, an isotropic antenna with a transmit power of $N\cdot P$ achieves a uniform beampattern with power $N\cdot P$. An array of $N$ isotropic antennas each with $P$ transmit power can be phased appropriately in order to achieve a beampattern with a maximum array factor of $N^2$ although the same power was dissipated.

2.4 Array Beamforming

Array beamforming refers to an array of local antennas that are equally spaced and can be weighted to achieve the desired radiation pattern. Antenna arrays can be used for directional of arrival (DOA) estimation and for interference avoidance by steering nulls toward unintended receivers and steering beams towards intended receivers. Figure 4 shows an array beamformer with $M$ elements spaced a distance, $d$, apart.
The array factor of the beamformer is found by determining the relative phase shift of the wavefront at each antenna element. The relative phase shift can be found using simple trigonometry as:

\[ \Delta \theta = -\omega d(M - n) \cos(\phi) / c. \] (1)
Where \( n \) is the element number, \( \omega \) is the frequency, and \( c \) is the speed of light. The array factor can then be found as

\[
a(\phi) = [1, e^{-\frac{j\omega d \cos(\phi)}{c}}, \ldots, e^{-\frac{j\omega (M-1) d \cos(\phi)}{c}}]. \tag{2}
\]

This is also known as the steering vector and the opposite signed array factor can be applied as element weights to steer a beam towards angle \( \phi \).

### 2.5 Distributed Beamforming

Distributed beamforming is performed by a distributed network of transmitters that are not fixed locally in some predictable geometry such as in mobile network. Despite the unpredictable and complicated geometry of distributed beamforming networks, the distributed nodes can still cooperate to achieve a desired beampattern. In [6] it was shown that an array of \( N \) distributed nodes can achieve the same diversity gain as a linear array of \( N \) nodes as the number of nodes approaches infinity. A model of a distributed beamforming network can be seen in Figure 5.
The array factor of this network can be found similarly to that of an antenna array. The relative phase difference is determined by finding the distance between the receiver and each node. In the case of the receiver being in the same plane as the network, the distance between each node and the receiver can be found trigonometrically to be:

\[ d_k = \sqrt{A^2 + r_k^2 - 2r_k A \cos(\Phi_0 - \gamma_k)}. \] (3)

If the far-field condition of \( A >> r_k \) is met, the distance can be approximated as:

\[ d_k \approx A - r_k \cos(\Phi_0 - \Phi_k). \] (4)

If the initial phase of each node is set to:
\[ \psi = \frac{-2\pi}{\lambda} d_k, \]  
\[ (5) \]

Then the array factor is
\[ F(\phi) = \frac{1}{N} \sum_{k=1}^{N} e^{j \frac{2\pi}{\lambda} r_k [\cos(\phi_0 - \gamma_k) - \cos(\phi - \gamma_k)]} \]  
\[ (6) \]

The far-field beampattern is then
\[ P(\phi) = |F(\phi)|^2 \]  
\[ (7) \]

The authors in [6] then use the uniform distribution of the nodes to find the average beampattern as:
\[ P_{av}(\phi) = \frac{1}{N} + \left( \frac{1}{N} \right) |2 \frac{J_1(\alpha(\phi))}{\alpha(\phi)}|^2 \]  
\[ (8) \]

Where \( J_1 \) is the first-order Bessel function of the first kind and
\[ \alpha(\phi) = \frac{4\pi R}{\lambda} \sin \left( \frac{\phi}{2} \right). \]  
\[ (9) \]

The plot of the average beampattern can be seen in Figure 6 for different network radiuses and transmitting nodes.
The beampattern produced in a distributed beamforming system uses knowledge of the relative locations of the nodes and assume a synchronized carrier frequency. If the carrier frequency is offset due to noise in the phase-locked loop of the node, the beampattern produced may not be the desired beampattern. In CR, this effect can cause protection to the primary user to be compromised [7],[8].

2.6 Effects of Imperfect Carrier Synchronization

![Figure 6: Average beampattern of N randomly placed nodes in circle of radius R](image)
Chapter 3. Genetic Algorithm

3.1 Background

The genetic algorithm (GA) is a population search algorithm inspired by the evolutionary process of natural selection. The algorithm arrives at a solution by performing operations on a population of solutions referred to as chromosomes over several iterations, which are referred to as generations. The operations used are borrowed from genetic concepts such as mutation, reproduction, fitness, generations etc. The algorithm uses fitness evaluation to rank solutions and randomness in creating new generations of solutions in order to promote diversity. The GA is most commonly used as an optimization algorithm and has been used to solve optimization problems across many disciplines. There are many different implementations of the genetic algorithm with many different parameters. The conventional genetic algorithm, which will be detailed in this section, can be referred to as the “canonical” genetic algorithm [17]. The GA can also be combined with other optimization methods making it different from the canonical GA. These forms of the GA are referred to as hybrid genetic algorithms and in many cases hybrid genetic algorithms have been shown to outperform the canonical genetic algorithm for solving certain types of problems[18]. This section will detail each step of the canonical genetic algorithm and will demonstrate its operation by using it to solve a distributed beamforming problem.
3.2 The Canonical Genetic Algorithm

The diagram for the canonical GA is shown in Figure 7. Operations used on the population for each iteration of the algorithm are shown. The GA is initialized by generating an initial population (i.e. the first set of chromosomes used in the algorithm). The number of individual chromosomes and the domain from which they are selected may vary. There is a trade-off between the size of the population and the number of generations required to arrive at a solution. Also, the domain may be limited to prevent known poor solutions from being generated [17]. Once the initial population is generated, the fitness of each chromosome is evaluated. Chromosomes will then be selected for crossover based on their fitness, which involves combing two chromosomes to produce a new individual. Mutations are then applied randomly to the population and the fitness is re-evaluated. This process is repeated until the solution converges or until the maximum number of generations is exceeded. Each of these operations has parameters that can vary greatly and are the subject of much research in the study of genetic algorithms. The main considerations for each component of the canonical genetic algorithms are discussed below.
Figure 7: Genetic Algorithm Flowchart
3.2.1 Evaluation and Fitness Function

The GA uses an evaluation function in order to assess a solution according to the particular parameters of the optimization problem and uses a fitness function in order to calculate the solutions fitness relative to the rest of the population. The fitness function therefore can be represented as,

$$ F_i = \frac{f_i}{\bar{f}} $$

where $f_i$ is the evaluation of chromosome $i$ and $\bar{f}$ is the average evaluation of the population. If a solution does not satisfy the constraints of the optimization problem, it will be assigned an evaluation of zero and a fitness of zero.

If the goal is to maximize some function, $F(X)$, where $X$ is an array of arguments, then the evaluation of a solution would be the value of the function given that solution. The argument $X$ is often represented as bit string, but in practice it is an array of one or more variables from a domain appropriate for the particular problem. Once the evaluation of the function is found for each solution, the average evaluation can be found and the fitness of each solution can be found using the equation above. These fitness values are then used to determine which solutions are selected for crossover. [17]
3.2.2 Crossover Selection

Crossover selection is performed in order to determine the “parents” used to form the next generation. Chromosomes from the current generation are selected based on their fitness to be parents for the next generation. Crossover of some form is then used to combine the chromosomes of the parents to produce offspring. This process can be seen in Figure 8.

Roulette wheel selection is used in the canonical GA using either the fitness value of each solution or the ranked value of each solution [17]. Solutions with higher fitness values are more likely to be selected. A roulette wheel analogy is often used to illustrate the selection process as
follows: each solution is assigned a slice of a roulette wheel proportional to its fitness as a fraction of the total fitness. Spinning the roulette wheel N times will select N solutions to be used for recombination and the likelihood of the solutions being selected will be proportional to their fitness. The following algorithm can be used to simulate this in practice [11]:

- Sum the fitness of all chromosomes to find $T$
- Generate a random number, $r$, from a uniform distribution between 0 and $T$
- Add the fitness of each chromosome one-by-one to $s = 0$ until $s \geq r$
- The chromosome whose fitness caused the sum to exceed $r$ is chosen for crossover
- Repeat the last three steps $N$ times to select $N$ parents
3.2.3 Crossover

Once the parents are selected, they are combined in order to produce the next generation. This is accomplished with crossover which merges different solutions at some crossover point. For instance, the following shows two chromosomes each with four elements and a vertical bar designating an arbitrary crossover point.

\[
\begin{array}{c|cccc}
A_1 & A_2 & A_3 & A_4 \\
\hline
B_1 & B_2 & B_3 & B_4 \\
\end{array}
\]

The result of crossover would produce the two chromosomes as follows:

\[
\begin{array}{c}
A_1 & B_2 & B_3 & B_4 \\
B_1 & A_2 & A_3 & A_4 \\
\end{array}
\]

This is an example of one-point crossover, but multi-point crossover is a possibility as well and crossover with real-coded chromosomes is more complicated. Crossover is conducted with some probably, \( p_c \), between two randomly selected chromosomes at a random point in the chromosome.

3.2.4 Mutation

Mutation is performed to promote diversity in the population and to help avoid getting stuck in local optima. Mutation is applied to each element
in the chromosome string with probability, $p_m$. The probability of
mutation is very small and if an element is selected for mutation it will be
changed to a random value within the bounds of the solution.
Chapter 4. Performance Study of GA-Assisted Beamforming in a Distributed Cognitive Radio Network

In this section, a system model consisting of a network of CR transmitters communicating with a SU receiver in the far-field is considered. The network operates in the presence of a primary network and it must avoid interfering with primary communications. It is assumed that the location of the SUs is known relative to an origin and that the azimuth angles of the primary users are known a priori. The CRN employs beamforming in order to steer a null towards the PUs, and the range of azimuth angles that must be protected are referred to as primary user regions (PUR). A primary user region can be considered a protection region for a single PU receiver or a network of primary user receivers.

The main objective of the CRN is to weight each node to maximize power towards the SU receiver while limiting power in the direction of the PUR to some threshold. A similar system model was employed in [12] in which a wireless sensor network used collaborative null-steering to limit the average beampattern towards certain azimuth angles. In [15], phase-only distributed beamforming (PODB) is used to accomplish the same objective. In these cases, the average beampattern is found and the distribution of the side-lobes and nulls are studied. In this thesis, the instantaneous beampattern is
considered and spread nulls are used in order to provide arbitrarily wide protection to the PU network. This creates a problem for which there is no known closed-form solution, so the GA is used to approximate the optimization problem. The authors of [10] and [16] also formulated optimization problems to solve different beamforming problems. The details of the system model are presented in the following section.

4.1 System Model

The distributed CRN is shown in Figure 9. It consists of N SU transmitters uniformly distributed within a circle of radius R. The polar coordinates of each node are denoted by \((d_n, \phi_n)\) and the azimuth angle is denoted by \(\phi\). The SU receiver is located within \(\phi \in \Phi_s\) and the PU protection region is \(\phi \in \Phi_p\).
The array factor at angle $\phi$ can be expressed as:

$$
F(\phi) = \left[ e^{\frac{j2\pi d_1 \cos(\phi - \phi_1)}{\lambda}}, e^{\frac{j2\pi d_2 \cos(\phi - \phi_2)}{\lambda}}, \ldots, e^{\frac{j2\pi d_N \cos(\phi - \phi_N)}{\lambda}} \right]
$$  \hspace{1cm} (11)

and the weighting vector can be expressed as:

$$
w = \left[ \frac{1}{N} e^{j\beta_1}, \frac{1}{N} e^{j\beta_2}, \ldots, \frac{1}{N} e^{j\beta_N} \right]
$$  \hspace{1cm} (12)

The resulting beampattern can then be found as:

$$
P(\phi) = |F \ast w'|^2
$$  \hspace{1cm} (13)

The objective of the CRN is to determine $\beta_1 \ldots \beta_N$ such that the beampattern is maximized in the direction of the SU receiver and limited in the PU protection region. Each node transmits the same power and only the phase is adjusted for weighting.

### 4.2 Analysis

An optimization problem can be formed using the beampattern as follows:

$$
\max_{\beta_1 \ldots \beta_N} |F \ast w'|^2, \quad \phi \in \Phi_S
$$  \hspace{1cm} (14)

subject to:

$$
|F \ast w'|^2 \leq \gamma_p, \quad \phi \in \Phi_P
$$  \hspace{1cm} (15)

where $\gamma_p$ is the limit on the interference in the PU protection region. The problem can then be rewritten as follows to provide the fitness and constraint function for the GA:

$$
\min_{\beta_1 \ldots \beta_N} \min_{\Phi_S} |F \ast w'|^2, \quad \phi \in \Phi_S
$$  \hspace{1cm} (16)

subject to:

$$
\max_{\Phi_S} |F \ast w'|^2 \leq \gamma_p, \quad \phi \in \Phi_P.
$$  \hspace{1cm} (17)
The first expression maximizes the minimum of the power directed at the SU receiver and the constraint limits the maximum power in the primary protection region to $\gamma_P$.

The minimization of a negative function is used in the first expression because the Matlab GA function minimizes the objective function.

### 4.3 Results

The optimization function was solved using the Augmented Lagrangian Genetic Algorithm in order to handle the nonlinear constraints of the beamforming problem [19]. The GA parameters used for the simulation can be seen in Figure 10. A scattered crossover function refers to elements of a chromosome being swapped according to a randomly generated bit string. The crossover fraction refers to the fraction of chromosomes used for crossover. The adaptive feasible mutation function was used so that mutation satisfied constraints and bounds. It was observed after experimentation that a population of more than fifteen provided no added benefit to the optimal solution and the algorithm always met stopping criteria within fifteen generations. The constraint tolerance refers to the maximum constraint violation that can be tolerated before the algorithm terminates. The function tolerance refers to the maximum average change in the best fitness value tolerated before the algorithm terminates.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Crossover Function</td>
<td>Scattered</td>
</tr>
<tr>
<td>Crossover Fraction</td>
<td>0.8</td>
</tr>
<tr>
<td>Mutation Function</td>
<td>Adaptive Feasible</td>
</tr>
<tr>
<td>Max Generations</td>
<td>15</td>
</tr>
<tr>
<td>Population</td>
<td>15</td>
</tr>
<tr>
<td>Constraint Tolerance</td>
<td>1e-5</td>
</tr>
<tr>
<td>Function Tolerance</td>
<td>1e-11</td>
</tr>
</tbody>
</table>

Figure 10: GA Parameters

Figure 11 shows how the best fitness and mean fitness value of the population converge as the number of generations increase. In the case of the beamforming problem, the maximum fitness value is -1 and it can be seen that the best and mean fitness values approach this as the number of generations increase.
Figure 11: Mean and Best Fitness

Figure 12 shows the beampattern found with $N = 16$ SU nodes and a radius of $4\lambda$ with a wavelength of 1.25 m. The SU receiver is located at $-20^\circ$ with a spread of $\Phi_S = [-21^\circ \ 20^\circ]$ and there are two primary user regions with spreads of $\Phi_{P1} = [95^\circ \ 100^\circ]$ and $\Phi_{P2} = [20^\circ \ 25^\circ]$. The primary power threshold was set to $\gamma_p = -20$ dB. It can be seen that a beam is steered towards the SU receiver and nulls are steered towards the primary user protection regions.
The simulation was then performed with varying number of SU transmitter nodes. The resulting beampatterns can be seen in Figure 13. It can be observed that increasing the number of nodes decreases the sidelobe levels.
Figure 13: BP with varying N, R = 4, PUR1 = [20 25], PUR2 = [95 100], SU = -20

Figure 14 shows a close-up of the primary user protection region at $\Phi_{P2} = [20^\circ 25^\circ]$. It can be seen that the beampattern power is kept below the primary power threshold of $\gamma_p = -20$ dB.
The simulation was also performed with varying CRN radius $R$. Figure 15 shows the effect of the network radius on the beampattern and it can be observed that a tighter cluster achieves a wider beampattern and also produces a beampattern with fewer sidelobes.
The effect of the value of the power constraint on the primary user protection region was also studied. Figure 16 shows the minimum power achieved in the direction of the SU receiver over a range of $\gamma_P$. It can be seen that loosening the constraint on the primary user region allows more power to be directed towards the SU receiver and that increasing the number of antennas increases the power that can be steered towards the
SU receiver for a given constraint.

Figure 16: SU Power vs. power constraint for different $N$

Due to the heuristic nature of the genetic algorithm, it was of interest to study the tradeoff between complexity and the optimality of the solution. Therefore, the power achieved in the direction of the SU receiver was observed for various population sizes. Figure 17 shows the minimum power achieved in the direction of the SU receiver versus the population used by the GA. It can be seen that the power does not improve for a
population above 15, so needless computation can be avoided by limiting the population size.

Figure 17: SU receiver power vs. population size with various PU power constraints, N=8
Chapter 5. Conclusion

In this thesis, the application of Cognitive Radio to the spectrum scarcity problem was studied by exploiting the spatial dimension of the radio spectrum. Cognitive Radio aims to use spectrum more efficiently than conventional wireless communications by operating within licensed spectrum when it is unoccupied by the licensed user. In order to achieve this, Cognitive Radio Networks must accurately sense their environment and detect available spectrum in one or more of the spectrum dimensions. Cognitive Radio must then adapt its parameters in a way that enables communication in the cognitive network while preventing interference to the primary network. The system model proposed in this thesis was comprised of a distributed network that used beamforming to steer a beam towards and secondary receiver and steer nulls in the direction of the primary network. A beamforming optimization problem with no known closed-form solution was formulated and the Genetic Algorithm was used to find a suitable solution.

Results show that the Genetic Algorithm can be used to solve the beamforming problem such that interference to the primary network is limited to a given threshold while a main beam is steered towards the secondary user receiver. The results also show that this can be accomplished with a relative low complexity in terms of the generations and population size required for the Genetic Algorithm. It was also shown that there is a tradeoff between the interference threshold in the direction of the primary user and the magnitude of the beam produced in the direction of the secondary receiver. The
interference threshold has less of an effect on the optimal beam in the direction of the secondary user as the number of secondary transmitters increases.

Future work on this problem should include research into methods for the network to perform the GA in a parallel and decentralized way. More work should also be done into the performance of the algorithm as the number of angles accounted for in the beamforming problem increases.
Appendices

Appendix A

%Matlab code for Chapter 2%

%beampattern of N nodes distributed randomly in radius R transmitting to receiver in the far-field with no interference restrictions

%number of nodes
N = 10;

%normalized radius
R_tilda = 4;

%azimuth angle
theta = -pi:0.01:pi;
alpha = 4*pi*R_tilda*sin(theta/2);

for N = 10:90:100
%average beampattern
Pav = (1/N) + ((1-(1/N))*(besselj(1,alpha)./(alpha/2)).^2);

%plot of beampattern
plot(theta*180/pi,10*log(Pav));
hold on;
end
Appendix B

% Matlab code for Chapter 4%
% beamforming with GA with a point beam and point nulls

%*****************************network parameters*****************************
lambda = 1.25;    %wavelength, 245 M
R = 4*lambda;    %radius of network

%number of cognitive transmitting nodes
N = 8;

%position of secondary user receiver (SU RX)
thetas1 = -21*pi/180;  %(-20 deg)
thetas2 = -20*pi/180;

%position of primary user receivers (PU RX)
thetap1 = 95*pi/180;  %(50 deg)
thetap2 = 100*pi/180;  %(100 deg)
thetap3 = 20*pi/180;  %(100 deg)
thetap4 = 25*pi/180;  %(100 deg)

%array of user distance from origin
dn = rand(1,N)*R;

%array of user azimuth angles(uniform from -pi to pi)
thon = rand(1,N)*2*pi-

%interference limit to the primary users
gamma_p = 10^(-10/10);  %(-10 dB)

%***************************** formulate array factors*****************************

%array factor in direction of PU1
PU_Rangle = thetap1:0.0001:thetap2;

PU_F1 = zeros(length(PE_Rangle),N);

for i = 1:N
PU_F1(:,i) = exp(1i*2*pi*dn(i)*cos(PU_Range - thetan(i))/lambda)/N;
end

%array factor in direction of PU2
PU_Range2 = thetap3:0.0001:thetap4;

PU_F2 = zeros(length(PU_Range2),N);
for i = 1:N
    PU_F2(:,i) = exp(1i*2*pi*dn(i)*cos(PU_Range2 - thetan(i))/lambda)/N;
end

%array factor in direction of SU
SU_Range = thetas1:0.0001:thetas2;

SU_F = zeros(length(SU_Range),N);
for i = 1:N
    SU_F(:,i) = exp(1i*2*pi*dn(i)*cos(SU_Range - thetan(i))/lambda)/N;
end

%total array factor
theta = -pi:0.0001:pi;

F = zeros(length(theta),N);
for i = 1:N
    F(:,i) = exp(1i*2*pi*dn(i)*cos(theta - thetan(i))/lambda)/N;
end

%***************invoke the genetic algorithm*****************
% 'MutationFcn',@mutationadaptfeasible,
% 'PlotFcns',{@gaplotmaxconstr}
options = gaoptimset('Display','iter','TolFun',1e-11);

%monte carlo trials to determine the average beampattern
pat_sum = zeros(length(theta),1);

%array of tolfun values
TF = [1e-1 1e-2 1e-3 1e-4 1e-5 1e-6 1e-7 1e-8 1e-9 1e-10 1e-11];

%array of generations
GENS = zeros(5,1);

%array of max fitness function values
Fit_Vals = zeros(5,1);

%array of function evaluations
Fun_Eval = zeros(5,1);

k=1;
for gam = 5:5:25
  options = gaoptimset('TolCon',1e-4,'TolFun',1e-11,'Display','iter','PopulationSize',gam);
  for mc = 1:50
    %determine optimal beamweight using the genetic algorithm
    [X,FVAL,EXITFLAG,OUTPUT] = ga(@(X)SU_Power_Fitness(X,SU_F),N,[],[],[],[],pi,pi,@(X)PU_constr(X,PU_F1,gamma_p),[],options);
    GENS(k) = GENS(k) + OUTPUT.generations;
    Fit_Vals(k) = Fit_Vals(k) + FVAL;
    Fun_Eval(k) = Fun_Eval(k) + OUTPUT.funccount;
    pat = (abs(F*transpose(exp(1i*X))).^2);
    pat_sum = pat_sum + pat;
  end
  GENS(k) = GENS(k)/mc;
  Fit_Vals(k) = Fit_Vals(k)/mc;
  Fun_Eval(k) = Fun_Eval(k)/mc;
  k=k+1;
end
% plot(theta*180/pi,10*log(pat_sum/mc));

function SU_Power = SU_Power_Fitness(X,SU_F)

SU_Power = -min((abs(SU_F*transpose(exp(1i*X))))).^2);

function [c,ceq] = PU_constr(X,PU_F1,gamma_p)

%limit power to PU RX to threshold
  c = max((abs(PU_F1*transpose(exp(1i*X))))).^2)-gamma_p;

%   c(2) = max((abs(PU_F2*transpose(exp(1i*X))))).^2)-gamma_p;

% No nonlinear equality constraints:
  ceq = [];

end
References


