

# FREQUENCY ESTIMATION OF UNDAMPED EXPONENTIAL SIGNALS USING GENETIC ALGORITHMS

Amit Mitra<sup>1\*</sup>, Debasis Kundu<sup>1</sup> and Gunjan Agrawal<sup>2</sup>

<sup>1</sup>Department of Mathematics & Statistics, Indian Institute of Technology Kanpur,  
Kanpur– 208016, India.

<sup>2</sup>Department of Operations Research and Statistics, University of North Carolina at  
Chapel Hill, USA.

**Abstract:** In this paper, we consider the problem of frequency estimation of undamped superimposed exponential signals model. We propose two iterative techniques of frequency estimation using genetic algorithms. The proposed methods use an elitism based generational genetic algorithm for obtaining the least squares and the approximate least squares estimates. In the simulation studies, it is observed that the proposed methods give nearly efficient estimates, having mean square error almost attaining the corresponding Cramér-Rao lower bound. The proposed methods significantly do not depend on the initial guess values otherwise required for other iterative methods of frequency estimation. It is also observed that the proposed methods have fairly high breakdown point with respect to different types of outliers present in the data. Outlier robustness and accuracy of the proposed methods are compared with the classical approaches for this problem.

**Keywords:** *Approximate least squares estimator, Cramér-Rao bound, frequency estimation, genetic algorithm, least squares estimator, undamped exponential signals model.*

---

\* Corresponding author: Phone: 0091-0512-2596064; Email: [amitra@iitk.ac.in](mailto:amitra@iitk.ac.in)

## 1. Introduction

Estimating the frequencies of superimposed exponential signals embedded in additive noise is a fundamental problem in signal processing. In several applications, particularly in communications, radar, sonar, geophysical seismology, the signals dealt with can be described by the following superimposed undamped exponential signals model:

$$y(t) = \mu(t) + \varepsilon(t); \quad \mu(t) = \sum_{k=1}^M \alpha_k e^{i2\pi \omega_k t}; \quad t = 1, 2, \dots, n, \quad (1)$$

where  $\mu(t)$  denotes the noise free complex valued sinusoidal signal;  $\{\alpha_k\}_{k=1}^M$ ,  $\{\omega_k\}_{k=1}^M$  are its amplitudes and frequencies, respectively; and  $\{\varepsilon(t)\}$  is an additive observational noise.

The complex valued form (1), of course, is not encountered in practice, as practical signals are always real valued. However, in many applications both the in-phase and quadrature components of the studied signals are available. In case of a real-valued sinusoidal signal, this means that both the sine and the corresponding cosine components are available. These two components may be processed by arranging them in a two dimensional signal vector or a complex valued signal of the form of (1) (see Stoica and Moses [2003], page 144, for more details). Since the complex valued description of the in-phase and quadrature components of a sinusoidal signal is the most convenient one from a mathematical viewpoint, most of the literature in signal processing deals with the complex valued superimposed undamped exponential signals.

The noise  $\{\varepsilon(t)\}$  in the model is usually assumed to be complex valued circular white noise defined as;

$$\begin{aligned} E[\varepsilon(t) \varepsilon^*(s)] &= \sigma^2 \delta_{t,s}, \\ E[\varepsilon(t) \varepsilon(s)] &= 0 \quad \text{for all } t \text{ and } s, \end{aligned}$$

where  $\delta_{t,s}$  is the delta function. The variance or power of the noise is  $E|\varepsilon(t)|^2 = \sigma^2$ .

Conditions on  $\{\varepsilon(t)\}$  can be rewritten as

$$\left. \begin{aligned} E\{\operatorname{Re}[\varepsilon(t)] \operatorname{Re}[\varepsilon(t)]\} &= \frac{\sigma^2}{2} \delta_{t,s} \\ E\{\operatorname{Im}[\varepsilon(t)] \operatorname{Im}[\varepsilon(t)]\} &= \frac{\sigma^2}{2} \delta_{t,s} \\ E\{\operatorname{Re}[\varepsilon(t)] \operatorname{Im}[\varepsilon(t)]\} &= 0 \end{aligned} \right\} \quad (2)$$

Thus, the real and imaginary parts of a complex circular white noise are real-valued white noise sequences of identical power equal to  $\sigma^2/2$ , and uncorrelated with one another. The frequencies  $\{\omega_k\}_{k=1}^M$  are such that  $\omega_k \in [0,1]$  and are distinct. Given a sample of size  $n$ ,  $\{y(1), y(2), \dots, y(n)\}$ , the problem is to estimate the unknown amplitudes, the unknown frequencies and also order  $M$ , if the order is unknown. In the present paper, we assume  $M$  to be known and focus on the problem of frequency estimation, i.e. determination of  $\{\omega_k\}_{k=1}^M$  from the set of observations  $\{y(1), y(2), \dots, y(n)\}$ . Note that if  $M$  is unknown, first we can estimate  $M$ , using the method proposed by Kundu and Mitra [2000a]. Once the frequencies are estimated, estimation of the amplitudes becomes a simple linear regression problem. More precisely, as we will see later, for given  $\{\omega_k\}_{k=1}^M$ , the observations  $y(t)$  can be written as a linear regression function whose unknown regression coefficients are equal to the unknowns amplitudes.

The estimation of the frequencies of the sinusoidal components, given by model (1), embedded in additive white noise is a fundamental problem in signal processing. Due to the practical importance of this model, a battery of methods attempting to obtain efficient estimators have been proposed in the recent past. These methods can broadly be classified into two groups, one that attempts to find efficient computational algorithms for the solution of the least squares estimators (LSEs) through iterative techniques and the other group providing non-iterative parameter estimates of the unknown parameters. Notable among the iterative methods aimed at finding the LSEs are the approaches of Bressler and MaCovski [1986], Hwang and Chen [1993], Kundu [1993], Kannan and Kundu [1994], Li and Stoica [1996]. Non-iterative methods include the methods of Pisarenko [1973], Ulrych and Clayton [1976], Schmidt [1979], Tufts and Kumaresan [1982], Roy and Kailath [1989], Quinn [1994], Kundu and Mitra [1995] and Bai, Rao, Chow and Kundu [2003].

It is observed in all the iterative procedures that the performances of the different estimators depend heavily on the initial value chosen. Most of the iterative procedures may not even converge if the initial guesses are not 'sufficiently close' to the corresponding true values. The estimation problem becomes even more complex when outliers are present in the data. Most of the commonly used methods for frequency

estimation fail even with moderate levels of contamination. Recently, researchers have started exploring various robust frequency estimation techniques that are robust to presence of outliers in the data. The aim of this paper is to find an algorithm whose performance does not depend significantly on initial guesses and also having a reasonably high breakdown point with respect to presence of outliers as compared to the non-robust methods. Breakdown point, with respect to presence of outliers, indicates the outlier contamination proportion at which the method fails to resolve the frequencies correctly. In this paper, we propose a genetic algorithm (Quagliarella et. al. [1997]) based iterative algorithm to obtain the least squares estimates for the model (1). It is observed that the proposed method produces efficient frequency estimates attaining the Cramér-Rao lower bound (CRLB) and performs quite well in the presence of outliers. The rest of the paper is organized as follows. In section 2, we give the least square formulation of the problem. The approximate least squares formulation is described in section 3. Section 4, presents the proposed genetic search based iterative algorithms for frequency estimation. The empirical studies, implementing the proposed algorithms and comparing them with the classical approaches, will be presented in section 5. Finally, the conclusions will be discussed in section 6.

## 2. Least Squares formulation

The least squares estimates of the signal parameters for the model (1) are the minimizers of the following criterion function,

$$\psi(\omega, \alpha) = \sum_{t=1}^n \left| y(t) - \sum_{k=1}^M \alpha_k e^{i 2\pi \omega_k t} \right|^2, \quad (3)$$

where  $\omega$  is the vector of frequencies and  $\alpha$  is the vector of amplitudes. The sinusoidal model determined through minimization of (3) has the smallest sum of squares distance to the observed data. Since  $\psi(\omega, \alpha)$  is a nonlinear function of its arguments  $\omega$  and  $\alpha$ , the parameter estimates minimizing (3) are called the nonlinear least squares (NLS) estimators. When the white noise  $\varepsilon(t)$  is Gaussian, the minimization of (3) can also be interpreted as the method of maximum likelihood (Stoica and Moses [2003], page 377).

Although, the criterion in (3) depends on both  $\varpi$  and  $\alpha$ , it can be conveniently concentrated with respect to the ‘nuisance parameters’  $\alpha$ . Introducing the notations,

$$\underline{Y} = [y(1), y(2), \dots, y(n)]^T, \quad (4)$$

$$A(\varpi) = \begin{bmatrix} e^{i2\pi\varpi_1} & \dots & e^{i2\pi\varpi_M} \\ \vdots & \ddots & \vdots \\ e^{in2\pi\varpi_1} & \dots & e^{in2\pi\varpi_M} \end{bmatrix}, \quad (5)$$

$$\text{and } \underline{\alpha} = [\alpha_1, \dots, \alpha_M]^T, \quad (6)$$

we can write  $\psi(\varpi, \alpha)$  in (3) as

$$\psi(\varpi, \alpha) = (\underline{Y} - A(\varpi)\underline{\alpha})^* (\underline{Y} - A(\varpi)\underline{\alpha}), \quad (7)$$

where, ‘\*’ denotes the complex conjugate of a vector or a matrix.

Under the condition of distinct frequencies, the Vandermonde matrix  $A(\varpi)$  is of rank  $M$  (under the weak condition that  $n \geq M$ ) and  $(A(\varpi)^* A(\varpi))^{-1}$  exists. We observe that for given  $\varpi$ , minimizing  $\alpha$  is given by  $\hat{\alpha} = (A(\varpi)^* A(\varpi))^{-1} A(\varpi)^* \underline{Y}$ . Plugging in this  $\hat{\alpha}$  in (7), we get the concentrated likelihood function  $\phi(\varpi) = \underline{Y}^* A(\varpi) (A(\varpi)^* A(\varpi))^{-1} A(\varpi)^* \underline{Y}$  and the LSE of  $\varpi$  is obtained by minimizing  $\phi(\varpi)$ . Thus the non-linear least squares (NLS) estimators of  $\varpi$  and  $\alpha$  which minimize  $\psi(\varpi, \alpha)$  are given by

$$\hat{\varpi} = \arg \max_{\varpi} \left[ \underline{Y}^* A(\varpi) (A(\varpi)^* A(\varpi))^{-1} A(\varpi)^* \underline{Y} \right], \quad (8)$$

$$\hat{\alpha} = (A(\hat{\varpi})^* A(\hat{\varpi}))^{-1} A(\hat{\varpi})^* \underline{Y} \Big|_{\varpi=\hat{\varpi}}. \quad (9)$$

The NLS estimators for this problem are optimal under various considerations. The estimators are strongly consistent (Kundu and Mitra [2000b]), asymptotic normal with a covariance matrix that coincides with the Cramér-Rao bound under the normality assumption on the error random variables (Rao and Zhao [1993]). Hence, under the Gaussian hypothesis, the NLS method provides frequency estimates that are the most accurate, in the sense of minimum variance in a fairly general class of estimators. It is further observed that, the performance of the NLS method does not critically depend on the assumption that the noise process is white. If the noise process is non white, the NLS still gives consistent frequency estimates (Stoica and Nehorai [1989]). The fact that the

NLS approach gives optimal estimates, spurred a renewed interest in the NLS approach and in reliable algorithms for performing the minimization required in (3) (see, e.g., Hwang and Chen [1993]; Kannan and Kundu [1994]; Li and Stoica [1996]).

Unfortunately, the good statistical performance associated with the NLS method of frequency estimation is difficult to achieve, for a number of reasons. The optimizing function (3) has a complicated multimodal shape with a very sharp global maximum corresponding to  $\hat{\omega}$  (Stoica et al. [1989]). Hence, finding  $\hat{\omega}$  by a local search algorithm requires very accurate initialization. Initialization procedures that provide fairly accurate approximations of the maximizer of (3) have been proposed in (Kumaresan, Scharf, and Shaw [1986], Bressler and MaCovski [1986], Kundu and Mitra [1995]). However, there is no available method that is guaranteed to provide frequency estimates within the attraction domain of the global maximum  $\hat{\omega}$  of (3), especially in situations when the observation set is contaminated with outliers. As a consequence, general-purpose search algorithms, like the Gauss-Newton algorithm, the Newton-Raphson algorithm or the Levenberg-Marquardt algorithm, may well fail to converge to  $\hat{\omega}$ , or may even diverge. The least squares frequency estimation for the exponential signals model given in (1) becomes even more difficult when outliers are present in the dataset.

### 3. Approximate Least Squares

In the previous Section, we have discussed the least squares estimators and it is observed that finding the least squares estimators involves an  $M$ -dimensional optimization procedure. To reduce the computational complexity, approximate least squares estimators (ALSEs) were proposed by Kundu and Mitra [2000b] and their properties were studied. It is observed that the least squares estimators and the approximate least squares estimators enjoy the same asymptotic properties. The approximate least squares estimators can be briefly described as follows;

In the formulation of the ALSE, we assume, without loss of generality that  $|\alpha_1^2| \geq |\alpha_2^2| \geq \dots \geq |\alpha_M^2|$ . Consider the following function

$$I(\omega) = \frac{1}{n} \left| \sum_{t=1}^n y(t) e^{-i 2\pi \omega t} \right|^2. \quad (10)$$

Suppose  $\tilde{\omega}_1$  maximizes  $I(\omega)$ , then  $\tilde{\omega}_1$  is the ALSE of  $\omega_1$ . The amplitude  $\alpha_1$  can be estimated as

$$\tilde{\alpha}_1 = \frac{1}{n} \sum_{t=1}^n y(t) e^{i2\pi\tilde{\omega}_1 t}. \quad (11)$$

Now to obtain the ALSE of  $\tilde{\omega}_2$ , we first obtain an adjusted data vector  $\tilde{y}$  from the original data, where

$$\tilde{y}(t) = y(t) - \tilde{\alpha}_1 e^{i2\pi\tilde{\omega}_1 t}. \quad (12)$$

Using the adjusted data  $\tilde{y}$ , we find the  $\omega$  that maximizes (10). Maximizing  $\omega$  provides the estimate for  $\tilde{\omega}_2$ . We repeat this process of finding sequentially the frequency estimates and an adjusted data vector until all the unknown frequencies are estimated. For details of the ALSE algorithm and their asymptotic statistical properties, the readers are referred to Kundu and Mitra [2000b].

#### 4. Frequency Estimation using Genetic Algorithms

In this section, we present the genetic search based iterative algorithms for the frequency estimation problem. First, we give a brief introduction about the basic concepts of a genetic search procedure and then present the proposed algorithms.

##### 4.1 Basics of Genetic Search Procedures

In this subsection, we discuss in brief the technique of solving optimization problems using genetic algorithms.

Genetic Algorithm (GA) is a branch of artificial intelligence that borrows the ideas from Darwinian evolution. It is a stochastic global optimization search method that mimics the natural biological evolution, which is based on the ideas of ‘natural selection’ and ‘genetics’. The process of natural selection is nature’s way of searching for better and better organisms and argues that individuals with certain characteristics are more suited to survive and pass those characteristics to their next generation. Genetic algorithms operate on a population of potential solutions and applying the principle of ‘survival of the fittest’ produce better and better approximations to a solution.

Detailed discussions on Genetic algorithms and their varied applications may be found in Chambers [1995a and 1995b] and Quagliarella et al. [1997].

A standard generational GA is a method of stochastic optimization that can be formulated as;

$$\begin{aligned} & \text{Maximize } F(x) \\ & \text{Subject to } x \in \Theta = \{0,1\}^n \subset \mathfrak{R}^n \end{aligned} \quad (13)$$

The function to be maximized,  $F: \Theta \rightarrow \mathfrak{R}$ , is called the fitness function in a GA framework. To start the GA stochastic optimization, a population consisting of possible solutions is initialized. Each individual of this population, also called a chromosome, is an  $n$ -dimensional binary string.

The members of the initial population are first evaluated for their fitness. We use a rank based fitness function (Baker [1985]). In this procedure, the individuals (chromosomes) are assigned fitness according to their rank in the population rather than their raw performance. Simpler versions of genetic algorithms use a proportional fitness function.

Based on the fitness values of the chromosomes, a variety of *selection* rules can be applied for selection of the fit chromosomes (also called the fit parents). The commonly used selection schemes are stochastic random sampling with replacement, stochastic sampling with partial replacement, remainder stochastic sampling with or without replacement, stochastic universal sampling and stochastic tournament. For a detailed discussion on various selection procedures, see for example Goldberg [1989]. We have adopted a stochastic sampling with replacement approach in this paper. Members selected from the current population using the selection operator, are next combined to produce new chromosomes by passing their genetic string material.

The operator for producing new chromosomes is the *crossover* operator. The crossover operator is applied, according to a pre-assigned crossover probability, on two selected parents. Crossover produces new individuals that have some parts of both the parent's genetic material. The simplest form of crossover is the one-point crossover. More frequently used crossover operators are multipoint crossover, uniform crossover and reduced surrogate crossover. The simple one-point crossover is performed in the following way. Consider the following 2 parent binary strings

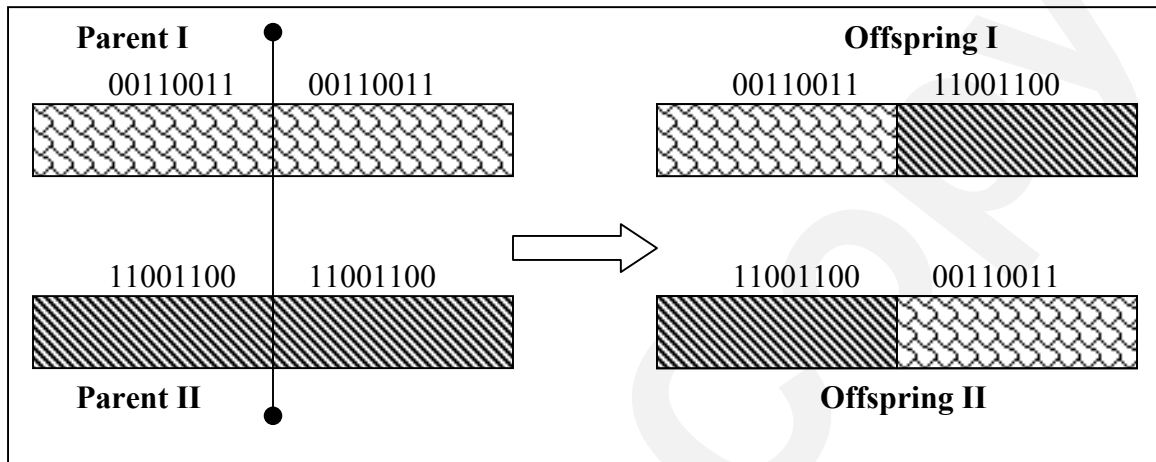
**Parent I:** 0011001100110011

and

**Parent II:** 1100110011001100.

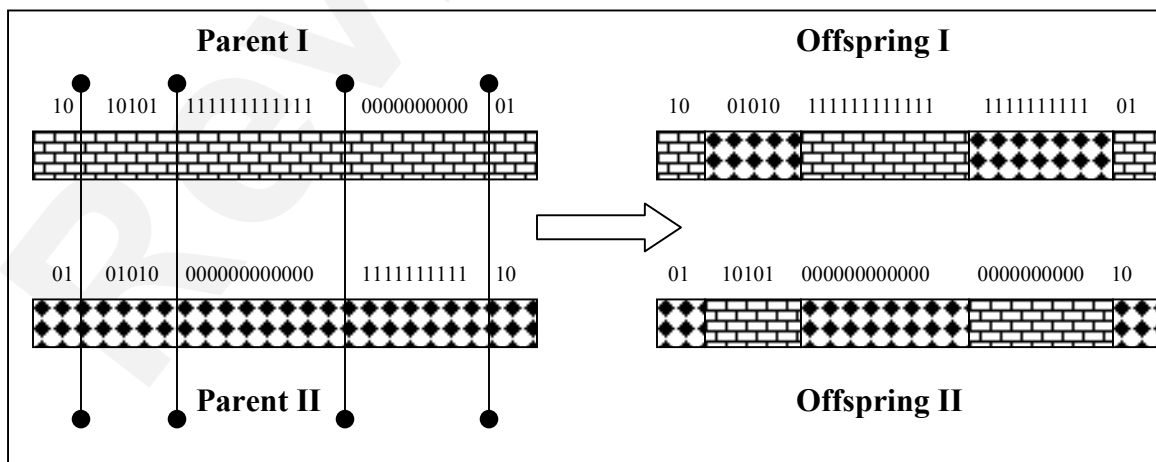


An integer position  $i$ , is selected uniformly at random between 1 and the chromosome length minus 1 and the genetic information is exchanged between the parents about this point producing the two new offspring chromosomes. Figure 1a illustrates a one-point crossover at the crossover point  $i = 9$ .



**Figure 1a:** A one-point crossover

For a multi-point crossover, say an  $m$ -point crossover,  $m$  crossover positions,  $k_1, k_2, \dots, k_m$ ;  $k_i \in \{1, 2, \dots, \text{chromosome length}-1\}$  are chosen at random with no repetitions and arranged in ascending order. The bits between the successive crossover points are exchanged between the two parents to produce the two new offspring chromosomes. A 4-point crossover is illustrated in Figure 1b, vertical lines indicating the 4 crossover points.



**Figure 1b:** A 4-point crossover

A genetic operator called *mutation*, is further applied to the new chromosomes produced by the crossover process. Mutation is randomly applied with a pre-assigned low mutation probability and is considered to be a genetic operator that ensures that the probability of searching any given string will never be zero and thus has the effect of tending to inhibit the possibility of convergence of the genetic algorithms to a local optimum. Mutation causes the chromosomes genetic representation to be changed according to a probabilistic rule. In the binary string representation, mutation will cause a single bit to change its state, i.e.  $0 \Rightarrow 1$  or  $1 \Rightarrow 0$ .

While populating the next generation after selection, crossover and mutation have been performed, we adopt an elitist strategy (De Jong [1975], Thierens [1997]). Elitism encourages the inclusion of highly fit genetic material, from earlier generations, in the subsequent generations. Under this approach, a predetermined fraction of the most-fit individuals is deterministically allowed to propagate through successive generations. The fractional difference between the number of chromosomes in the old population and the number of chromosomes produced by selection and recombination is termed as the generation gap and is filled using the elitist approach.

Thus, starting from the initial population of chromosomes and applying the GA operators of selection, crossover and mutation, GA moves on to the next generation of chromosomes. Chromosome strings of the next generation are decoded (if necessary), objective function is evaluated, a fitness value assigned and new set of chromosomes are selected, using the selection operator, for creation of the subsequent generation and so the process continues through subsequent generations.

Since GA is a stochastic search procedure, it is difficult to formally specify its convergence criteria. It is quite possible under the GA stochastic optimization that the fitness of a population may remain static for a number of generations before a superior individual is found and hence application of conventional termination criteria becomes inappropriate. Commonly used approaches are; termination of GA after a prespecified number of generations or termination of GA if no better solution is found after a predetermined number of generations, keeping track of the best solution found over generations under both the approaches.

The algorithmic steps of the discussed generational GA using elitism are presented in Figure 2.

To apply the discussed algorithm to any arbitrary optimization problem given by;

$$\left. \begin{array}{l} \text{Maximize } G(\underline{y}) \\ \text{Subject to } \underline{y} \in \Omega \subset \mathfrak{R}^n \end{array} \right\}, \quad (14)$$

a correspondence between the search space  $\Omega$  and appropriate space of binary strings  $\Theta$  is established. The correspondence is usually established using a coding technique. The use of Gray coded binary strings is the most commonly used approach. For calculating the objective function of the optimization problem, the Gray coded binary strings can be decoded using a linear or a logarithmic scaling.

**Step 1:** Randomly initialize the initial population generation of chromosomes of coded binary strings representing possible solutions.

**Step 2:** Evaluate the objective function for each of the chromosomes (after decoding, if necessary) and obtain their fitness values.

**Step 3:** Using selection operator, populate the fit parents pool, size of the pool being dependent on the generation gap.

**Step 4:** Apply crossover (with a pre-assigned crossover probability), exchanging genetic material of parents to obtain offspring.

**Step 5:** Apply mutation on the mated chromosome strings with pre-assigned mutation probability.

**Step 6:** Use elitist strategy to fill the generation gap.

**Step 7:** Repeat the steps 2 to 6 till termination criteria is met.

**Figure 2:** Algorithmic steps of a generational GA using elitism

#### 4.2 Proposed Method of Least Square Frequency Estimation

In this subsection, we present the proposed algorithm of least squares frequency estimation using a genetic search procedure. We have observed that the least square frequency estimation for the undamped exponential signals model (1), reduces to finding the solution of the optimization problem

$$\hat{\omega} = \arg \max_{\omega} \left[ \underline{Y}^* A(\omega) (A(\omega)^* A(\omega))^{-1} A(\omega)^* \underline{Y} \right]. \quad (15)$$

In the genetic search formulation of the frequency estimation problem, we take the function

$$\xi(\omega) = \tilde{Y}^* A(\omega) (A(\omega)^* A(\omega))^{-1} A(\omega)^* Y, \quad (16)$$

as the objective function and aim to find the optimum solution through repeated application of the three genetic operators of selection, crossover and mutation, over successive generations.

The parameter space,  $\Omega$  for the present frequency estimation problem is given by

$$\Omega = [0, 1] \times [0, 1] \times \dots \times [0, 1] \subset \mathfrak{R}^M. \quad (17)$$

Since genetic algorithms work with coded information rather than directly with the optimization variables, it is necessary here to use a coding technique. There are various approaches for coding and a particular coding method has influence on the computation speed and the accuracy of the obtained results. The simplest approach is to use a binary representation with only two bits information data 0 and 1 and most of the coding method for GA searching is usually done using binary coding and decoding techniques.

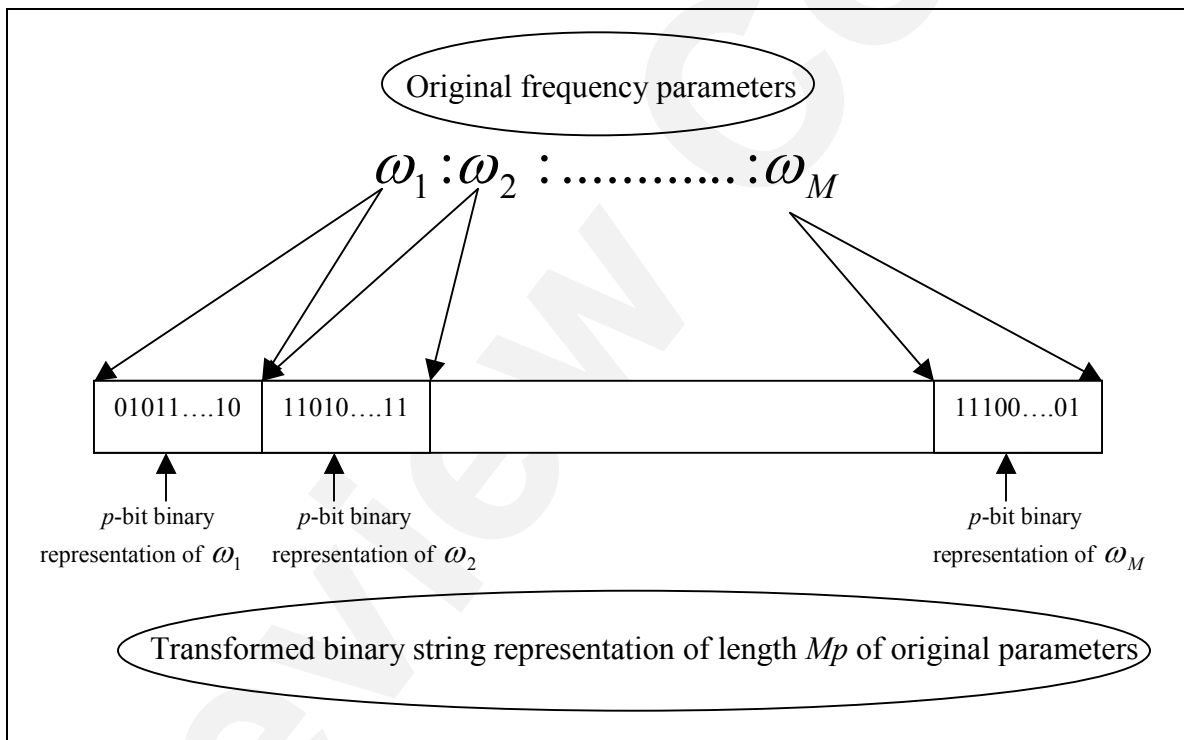
We first look for a simple binary chromosomal representation of a typical parameter vector in the parameter space  $\Omega$ . We can encode any  $\omega \in \Omega$  to a binary string of length  $Mp$ , where,  $p$  denotes the length of the binary bit representation of any component of the parameter vector  $\omega$ , i.e. for each of the  $M$  frequencies, we obtain a  $p$ -bit encoded binary representation. Thus encoding each parameter in the parameter vector as a binary string of length  $p$  and then concatenating these, we can form a single chromosome. The number of bits,  $p$ , depends on the level of precision desired. The binary coding of the parameter space is schematically presented in Figure 3.

It is however well-known that, ordinary binary coding can result in a search process being deceived or unable to efficiently locate the global minima due to large Hamming distances in the representational mapping between adjacent values (Hollstien [1971]). Hamming distance between two binary strings being defined as the minimum number of bits that must be changed in order to convert one bit string into another.

Use of Gray coding, derived from the binary coding, is advocated in the GA literature, as a method to overcome the hidden representational bias. The literature of GA and its applications report that Gray coding exhibits accelerated convergence rate of the objective function, and provides better accuracy than the binary coded GA (Caruana and

Schaffer [1988], Yokose et. al. [2000]). Superior performance of a Gray coded GA is mainly attributed to the fact that Gray codes do not bias the searching direction, as is the case of ordinary binary coding having a large Hamming distance between adjacent values.

A Gray code represents each number in the sequence of integers  $\{0,1,\dots,2^K-1\}$  as a binary string of length  $K$  in an order such that adjacent integers have Gray code representations that differ in only one bit position. Use of Gray code thus allows going through the integer sequence requiring flipping just one bit at a time. This is called the adjacency property of Gray codes. Gray code takes a binary sequence and shuffles it to form a new sequence with the adjacency property.



**Figure 3:** Binary coding of the original frequency parameters

We use here a Gray coding derived from the initial binary coding presented in Figure 3. To initialize the genetic search, we populate an initial population of a predetermined number of chromosomes. Each member of this initial population is a Gray coded binary string of length  $Mp$ , each of which corresponds to a possible parameter vector solution,  $\omega^0 \in \Omega$ . The value of the objective function (16), for each of the members of

this initial population, is evaluated by decoding the Gray coded binary strings and computing their corresponding fitness values using a ranking method. We use a linear scaling for decoding of the encoded binary strings.

**Step 1:** Randomly initialize initial population generation (of a predetermined size) of chromosomes of Gray coded binary strings of length  $Mp$ .

**Step 2:** Decode the Gray coded binary strings using a linear scaling.

**Step 3:** Evaluate the objective function (16) for each of the decoded strings and obtain their fitness values using a ranking method.

**Step 4:** Preserve the information about the string with highest fitness value.

**Step 5:** Using a stochastic sampling with replacement method, populate fit parents pool, size of the pool depending on the generation gap.

**Step 6:** From the selected parents pool, we select pairs in order and apply a two-point crossover (with a pre-assigned crossover probability), exchanging genetic material of parents to obtain offspring chromosome.

**Step 7:** Apply mutation on the mated chromosome strings with small pre-assigned mutation probability.

**Step 8:** Use elitist strategy to fill the generation gap.

**Step 9:** Repeat the steps 2 to 6 till maximum number of generations is reached or no better solution is found after 25 generations.

**Step 10:**  $\hat{\omega}_{GA-LSE}$  is the most fit decoded string found among all the generations.

**Figure 4:** GA-LSE algorithm

In the empirical section, we will apply the proposed procedure of GA based least square frequency estimation for parameter estimation of simulated models. We will also present extensive simulation studies to investigate the possible effect of different type of outliers present in the data.

### 4.3 Genetic Algorithm based Approximate Least Square Frequency Estimation

In this subsection, we outline the proposed algorithm of approximate least squares frequency estimation using the genetic algorithms. We have observed in the formulation of the approximate least squares estimators for the undamped exponential signals model

(1), that we can conveniently reduce the  $M$ -dimensional optimization search required for finding the exact least squares estimators to  $M$  1-dimensional optimization searches for finding the approximate least squares estimators. This leads to a significant reduction in the computational time required for genetic algorithms based formulation.

Under the genetic algorithms based approximate least squares estimation formulation, we obtain the frequencies sequentially, sequencing depending upon the norm of the corresponding amplitudes. Accordingly, we first estimate the frequency with highest norm amplitude, then using the adjusted data obtained using equation (12); we obtain the estimate for the frequency having the next highest norm amplitude. This process goes on until we estimate all the frequencies. At each step, genetic algorithms based ALSE for that step frequency is computed through a 1-dimensional generational GA with elitism strategy.

For computing the genetic algorithms based ALSE of the frequency corresponding to the highest norm amplitude, i.e.  $\omega_1 \in [0,1]$ , we code it to a binary string of length  $p$  using Gray coding and use a linear scaling for decoding.  $p$  denoting the length of the binary string representation of the parameter  $\omega_1$ . To start the GA based ALSE procedure, we populate an initial population, of a predetermined size, of binary strings of length  $p$ -bits. The optimization criterion function for the chromosomes, after decoding, is given by (10). We walk through the GA steps, as presented in Figure 2, repeatedly, till the termination criterion is reached. The GA based approximate least square (GA-ALSE) solution of  $\omega_1$ , say  $\hat{\omega}_{(1)GA-ALSE}$ , is the most fit individual evolving among all the generations, at the point when termination criterion is reached. Once we obtain  $\hat{\omega}_{(1)GA-ALSE}$ , the estimate of corresponding amplitude is obtained using (11). The adjusted data  $\{\tilde{y}(1), \tilde{y}(2), \dots, \tilde{y}(n)\}$  is obtained using (12). From the adjusted data, proceeding in the same way, as that used for obtaining  $\hat{\omega}_{(1)GA-ALSE}$ , we obtain  $\hat{\omega}_{(2)GA-ALSE}$ . The process is continued until all the frequencies are estimated. The algorithm for obtaining the ALSE of the parameters is presented in Figure 5. The main advantage of using the GA-ALSE approach over the GA-LSE approach is in terms of the computational complexity and time, which we will investigate in the simulation studies.

**Step 1:** Randomly initialize initial population generation (of a predetermined size) of chromosomes of Gray coded binary strings of length  $p$ .

**Step 2:** Decode the Gray coded binary strings using a linear scaling.

**Step 3:** Evaluate the objective function (10) for each of the decoded strings and obtain their fitness values using a ranking based method.

**Step 4–Step 9:** Same as that in GA-LSE algorithm.

**Step 10:** Obtain  $\hat{\omega}_{(1)GA-ALSE}$ , as the most fit decoded string found among all the generations.

**Step 11:** Compute the amplitude estimate  $\hat{\alpha}_{(1)GA-ALSE}$  and the adjusted data  $\{\check{y}(1), \check{y}(2), \dots, \check{y}(n)\}$  is obtained using (12).

**Step 12:** Using the adjusted data, obtained in Step 11, repeat the steps Step 1–Step 9 to get  $\hat{\omega}_{(2)GA-ALSE}$  and  $\hat{\alpha}_{(2)GA-ALSE}$ .

**Step 13:** Continue the above steps till all the frequencies are estimated.

**Figure 5:** GA-ALSE algorithm

## 5. Simulation Studies and Discussions

In this section, we present the simulation studies to access the performance of the proposed procedures, GA-LSE and GA-ALSE under various conditions. We also perform simulations to check the robustness and performance with respect to outliers being present in the data. All the simulations have been done using MATLAB random number generators.

### 5.1 Effect of bits on the performance the estimators

Binary string representation of the frequencies impels a discretization of the frequency range, discretization depending upon the number of bit length  $p$  used for each parameter. It is thus natural that the performance of the estimators will depend upon the bit length. To ascertain the effect of the number of bits in the binary string representation of the frequency parameters, we perform a simulation on the following model:

$$y(t) = 1.0 \exp(i(2\pi)0.5t) + \varepsilon(t); t = 1, 2, \dots, 25. \quad (18)$$



$\varepsilon(t)$  is taken as independently and identically distributed complex normal noise sequence, with mean zero and variance  $\sigma^2/2$  for both the real and imaginary parts of  $\varepsilon(t)$ . For each data set, we estimated the GA-LSE and the GA-ALSE for the frequency parameter. Table 2 presents the average estimates and the mean square errors (MSEs) of the GA-LSE over 100 simulation runs with  $\sigma^2 = 1$ , obtained using the GA parameters given in Table 1. We also present the corresponding CRLB for comparison in Table 2.

**Table 1:** Choice of genetic parameters for the simulations of model (18)

<i>Genetic Parameter</i>	<i>Values</i>
Number of chromosomes in one population	150
Coding	Gray coding
Scaling	Linear
Range of parameter	$\omega \in [0,1]$
Crossover probability	0.70
Crossover method	2-point
Mutation probability	0.01
Elitism	Top 10%
Maximum number of generation	200

We observe from the results that initially the performance of the GA-LSE improves drastically with increase in string length and stabilizes around 15 to 20 bits where the MSE attains the CRLB. We observe similar results for the GA-ALSE. In all subsequent computations we use a 20-bit representation of a single frequency parameter.

**Table 2:** Performance of the GA-LSE for different bit lengths

Length of binary string	Average Estimate	MSE	CRLB
4	0.4553	0.0641	<b>9.726 E-6</b>
6	0.5006	6.298 E-5	
8	0.5001	1.430 E-5	
10	0.5000	1.214 E-5	
15	0.4993	9.766 E-6	
20	0.5001	9.367 E-6	
30	0.4998	9.370 E-6	

## 5.2 Performance of GA-LSE and GA-ALSE at different SNR levels and sample sizes

We next investigate the performance of the proposed method at various SNR levels and sample sizes. We consider the following 2–component simulation model:

$$y(t) = \alpha_1 \exp(i2\pi \omega_1 t) + \alpha_2 \exp(i2\pi \omega_2 t) + \varepsilon(t); t = 1, 2, \dots, n. \quad (19)$$

The true values of the parameters of the simulation model are  $\alpha_1 = 2.0$ ,  $\alpha_2 = 3.0$ ,  $\omega_1 = 0.5$  and  $\omega_2 = 0.7$ .  $\varepsilon(t)$  is taken as independently and identically distributed complex normal noise sequence, with mean zero and variance  $\sigma^2/2$  for both the real and imaginary parts of  $\varepsilon(t)$ . Furthermore, the real and the imaginary parts are independently distributed. Simulation is performed over three different sample sizes ( $n$ ), 25, 50 and 75. For each of these sample sizes, we generated different scenarios, varying the noise power  $\sigma^2$  from 0.5 ( $SNR \cong 14.2$ ) to 2 ( $SNR \cong 8.1$ ).  $SNR$  denotes the ‘Signal-to-Noise’ ratio, given by  $SNR = 10 \log_{10}(\text{signal energy}/\text{noise energy})$ . For each data set, we estimated the GA-LSE and the GA-ALSE for the frequency parameter vector (0.5,0.7). The choice of the genetic parameters for the genetic algorithm setup is same as that presented in Table 1. The bit length corresponding to each parameter is taken as 20. The average estimates, the MSEs and average number of GA generations required over 100 simulations for all the components of the frequencies are computed. We have also reported the corresponding CRLB for comparison. The results for sample size 25 are reported in Table 3, sample size 50 in Table 4 and results for sample size 75 are reported in Table 5.

From the result tables, we observe that the GA-LSE and GA-ALSE frequency estimation techniques perform quite well. The MSE of the proposed methods are almost equal to the corresponding CRLB even for reasonably low SNR and small sample sizes. In most of the cases, we observe that the estimators are nearly efficient, having their MSEs equal or nearly equal to the CRLB. Thus they are able to attain the maximum possible accuracy corresponding to unbiased estimators of the respective model parameters. During the simulations, we observe that GA-ALSE is computationally much more efficient than the GA-LSE method. For completing a single run of simulation with population size of 50 chromosomes and 200 generation, GA based ALSE procedure takes 14 seconds on a 1000 Mhz, Pentium 3 processor, as against 90 seconds for GA based LSE. It is also observed that the computationally efficient GA-ALSE takes on an average less than half the number of generations as that required for GA-LSE, giving estimates, on an average, as accurate as GA-LSE. From the results tables, it is further observed, that for a fixed

level of  $\sigma^2$ , MSE decreases as  $n$  increases and for a fixed value of  $n$ , MSE increases as  $\sigma^2$  increases. This is true for both the GA-LSE and GA-ALSE.

**Table 3:** GA-LSE and GA-ALSE frequency estimation results for sample size 25

$\sigma^2$ (SNR)		Frequency 1 $\omega_1 = 0.5$		Frequency 2 $\omega_2 = 0.7$	
		GA-LSE	GA-ALSE	GA-LSE	GA-ALSE
<b>0.5 (14.15)</b>	Av. Est.	0.4999	0.4998	0.7002	0.7015
	MSE	1.552E-6	1.591E-6	5.291E-7	2.985E-6
	Av. Gen.	75	32	75	32
	CRLB	1.215E-6		5.395E-7	
<b>1(11.14)</b>	Av. Est.	0.4997	0.4997	0.7001	0.7014
	MSE	1.990E-6	2.025E-6	1.247 E-6	3.297E-6
	Av. Gen.	73	30	73	32
	CRLB	2.432E-06		1.082E-06	
<b>1.5 (9.38)</b>	Av. Est.	0.5000	0.4999	0.7003	0.7016
	MSE	3.647E-6	3.720E-6	1.634 E-6	4.118E-6
	Av. Gen.	74	34	74	32
	CRLB	3.648E-06		1.621E-06	
<b>2 (8.13)</b>	Av. Est.	0.4998	0.4997	0.7002	0.7016
	MSE	5.275 E-6	5.352 E-6	2.486 E-6	4.929 E-6
	Av. Gen.	72	30	72	29
	CRLB	4.863E-06		2.161E-06	

**Table 4:** GA-LSE and GA-ALSE frequency estimation results for sample size 50

$\sigma^2$ (SNR)		Frequency 1 $\omega_1 = 0.5$		Frequency 2 $\omega_2 = 0.7$	
		GA-LSE	GA-ALSE	GA-LSE	GA-ALSE
<b>0.5 (14.15)</b>	Av. Est.	0.5001	0.5000	0.7000	0.7004
	MSE	1.692 E-7	1.579 E-7	6.488 E-8	1.935 E-7
	Av. Gen.	73	31	73	34
	CRLB	1.520E-07		6.763E-08	
<b>1(11.14)</b>	Av. Est.	0.5000	0.5000	0.7000	0.7004
	MSE	3.393 E-7	3.375 E-7	1.651 E-7	4.127 E-7
	Av. Gen.	73	29	73	36
	CRLB	3.040E-07		1.350E-07	
<b>1.5 (9.38)</b>	Av. Est.	0.4999	0.4999	0.7000	0.7003
	MSE	5.173 E-7	5.089 E-7	2.093 E-7	3.203 E-7
	Av. Gen.	73	31	73	33
	CRLB	4.559E-07		2.026E-07	
<b>2 (8.13)</b>	Av. Est.	0.4999	0.4999	0.7000	0.7004
	MSE	5.417 E-7	5.959 E-7	2.911 E-7	4.275 E-7
	Av. Gen.	71	33	71	30
	CRLB	6.079E-07		2.710E-07	

**Table 5:** GA-LSE and GA-ALSE frequency estimation results for sample size 75

$\sigma^2$ (SNR)		Frequency 1 $\omega_1 = 0.5$		Frequency 2 $\omega_2 = 0.7$	
		GA-LSE	GA-ALSE	GA-LSE	GA-ALSE
<b>0.5 (14.15)</b>	Av. Est.	0.5000	0.5000	0.7000	0.7002
	MSE	4.961 E-8	4.454 E-8	2.212 E-8	4.996 E-8
	Av.Gen.	74	37	74	29
	CRLB	4.509E-08		2.001E-08	
<b>1(11.14)</b>	Av. Est.	0.5000	0.5000	0.7000	0.7002
	MSE	9.331 E-8	8.496 E-8	4.469 E-8	2.165 E-7
	Av.Gen.	71	31	71	29
	CRLB	9.018E-08		4.002E-08	
<b>1.5 (9.38)</b>	Av. Est.	0.5000	0.5001	0.7000	0.7002
	MSE	1.413 E-7	1.367 E-7	6.080 E-8	8.689 E-8
	Av.Gen.	72	28	72	33
	CRLB	1.350E-07		6.003E-08	
<b>2 (8.13)</b>	Av. Est.	0.5000	0.5000	0.7000	0.7001
	MSE	1.775 E-7	1.713 E-7	9.185 E-8	1.004 E-7
	Av.Gen.	72	34	72	33
	CRLB	1.801E-07		8.004E-08	

### 5.3 Performance of GA-LSE and GA-ALSE for closely spaced frequencies

In this subsection, we study the performance of the proposed methods when the frequencies are closely spaced. We provide a detailed performance analysis corresponding to different degree of separation of the frequencies. We consider the same 2-component superimposed exponential model (19). The true values of the amplitude parameters of the simulation model are  $\alpha_1 = 2.0$ ,  $\alpha_2 = 3.0$ . The noise sequence  $\varepsilon(t)$  is same as that considered earlier. In order to ascertain the effect of the degree of separation of the frequencies on the performance, we vary the degree of separation from 0.20 to 0.01 at two different  $\sigma^2$  levels ( $\sigma^2=0.5$  and  $\sigma^2=2.0$ ). The sample size is fixed at 25. The results are tabulated in Table 6. The CRLB of the frequency parameter estimate corresponding to  $\alpha_1$ , at  $\sigma^2=0.5$  being 1.215E-6 and 4.863E-06 at  $\sigma^2=2.0$  and the CRLB of the frequency parameter estimate corresponding to  $\alpha_2$ , at  $\sigma^2=0.5$  being 5.395E-7 and 2.161E-06 at  $\sigma^2=2.0$ .

**Table 6:** Performance of the proposed methods for different degree of separation

Degree of Separation of Signals	$\sigma^2$	Actual Frequency	Average Estimate		MSE	
			GA-LSE	GA-ALSE	GA-LSE	GA-ALSE
0.20	0.5	0.50	0.4999	0.4998	1.552E-6	1.591E-6
		0.70	0.7002	0.7015	5.291E-7	5.985E-7
	2.0	0.50	0.4998	0.4997	5.275 E-6	5.352 E-6
		0.70	0.7002	0.7016	2.486 E-6	4.929 E-6
0.10	0.5	0.50	0.5000	0.4999	1.642 E-6	1.781 E-6
		0.60	0.6000	0.6001	5.822 E-7	5.999 E-7
	2.0	0.50	0.5001	0.5001	5.248 E-6	5.575 E-6
		0.60	0.6000	0.6001	1.591 E-6	1.919 E-6
0.05	0.5	0.50	0.4997	0.4993	1.583 E-6	1.606 E-6
		0.55	0.5501	0.5517	5.952 E-7	3.199 E-6
	2.0	0.50	0.5003	0.4998	8.296 E-6	6.519 E-6
		0.55	0.5497	0.5516	3.308 E-6	3.913 E-6
0.02	0.5	0.50	0.5005	0.4863	7.697 E-6	1.880 E-4
		0.52	0.5195	0.5155	4.695 E-6	2.124 E-5
	2.0	0.50	0.4997	0.4884	3.138 E-5	2.060 E-4
		0.52	0.5203	0.5181	1.714 E-5	6.551 E-5
0.01	0.5	0.50	0.4976	0.4534	5.821 E-5	0.01404
		0.51	0.5182	0.5504	0.002468	0.01207
	2.0	0.50	0.4164	0.3883	0.02682	0.03678
		0.51	0.5779	0.6137	0.02322	0.03418

The results of Table 6 indicate that the proposed methods have reasonably high breakdown point while resolving closely spaced frequencies. Even at fairly low degree of separation, 0.05, both the methods provide estimates with MSEs very close to the CRLB, at both the SNR levels. In general, GA-LSE performs marginally better than the GA-ALSE and has a lower breakdown point than the GA-ALSE. The threshold of the GA-ALSE seems to be around 0.02 degree of separation. The threshold level of GA-LSE seems to be even lower. Thus, both the proposed methods are able to provide fairly accurate estimates when the degree of separation of the frequencies is greater than 0.02, even at low SNR levels.

#### 5.4 Performance of GA-LSE and GA-ALSE for higher order superimposed model

In this subsection, we perform simulations to ascertain the performance of the proposed methods in case of higher order superimposed model. We consider the following 10-component superimposed exponential signals model:

$$y(t) = \sum_{k=1}^{10} \alpha_k e^{i 2\pi \omega_k t} + \varepsilon(t); \quad t = 1, 2, \dots, n \quad (20)$$

The true amplitude parameter vector of model (20) is

$$\alpha = (1.5 \ 2.0 \ 2.25 \ 2.45 \ 2.2 \ 2.3 \ 1.2 \ 1.7 \ 2.4 \ 2.5)^T$$

and the true frequency vector is

$$\omega = (0.14 \ 0.21 \ 0.33 \ 0.39 \ 0.48 \ 0.56 \ 0.67 \ 0.73 \ 0.81 \ 0.86)^T.$$

The noise sequence  $\varepsilon(t)$  is same as that considered for models (18) and (19). Simulation is performed over two different sample sizes, 50 and 100. For each of these sample sizes, we generated different scenarios, varying the noise power  $\sigma^2$  from 0.5 to 2.0. Since the number of components is large, we only compute the computationally efficient GA-ALSE and report its performance. For comparison of the GA-ALSE results, we report also the corresponding CRLBs. The results are presented in Table 7.

We observe from the simulations study of model (20) that the GA-ALSE performs quite well for the 10-component superimposed exponential signals model and are able to resolve all the frequencies with reasonable degree of accuracy in most of the cases.

**Table 7:** Simulation results of GA-ALSE for model (20)

$\omega$	$\sigma^2$	Average Estimate		MSE		CRLB	
		$n = 50$	$n = 100$	$n = 50$	$n = 100$	$n = 50$	$n = 100$
<b>0.14</b>	<b>0.5</b>	0.1406	0.1400	7.452 E-7	5.774 E-8	2.701 E-7	3.376E-08
	<b>1.0</b>	0.1404	0.1400	8.395 E-7	1.051 E-7	5.403 E-7	6.754E-08
	<b>1.5</b>	0.1404	0.1399	1.146 E-6	1.132 E-7	8.105 E-7	1.013E-07
	<b>2.0</b>	0.1404	0.1404	1.719 E-6	1.719 E-6	1.080 E-6	1.350E-07
<b>0.21</b>	<b>0.5</b>	0.2100	0.2103	1.888 E-7	1.562 E-7	1.519 E-7	1.899E-08
	<b>1.0</b>	0.2098	0.2103	4.429 E-7	1.503 E-7	3.039 E-7	3.799E-08
	<b>1.5</b>	0.2097	0.2103	9.226 E-7	1.734 E-7	4.559 E-7	5.699E-08
	<b>2.0</b>	0.2099	0.2099	1.021 E-6	1.021 E-6	6.079 E-7	7.599E-08
<b>0.33</b>	<b>0.5</b>	0.3305	0.3301	5.215 E-7	3.762 E-8	1.200 E-7	1.500E-08
	<b>1.0</b>	0.3301	0.3300	7.467 E-7	7.611 E-8	2.401 E-7	3.001E-08
	<b>1.5</b>	0.3301	0.3300	1.194 E-6	7.724 E-8	3.602 E-7	4.503E-08
	<b>2.0</b>	0.3299	0.3299	1.186 E-6	1.186 E-6	4.803 E-7	6.004E-08
<b>0.39</b>	<b>0.5</b>	0.3919	0.3901	4.056 E-6	4.836 E-8	1.012 E-7	1.265E-08
	<b>1.0</b>	0.3918	0.3901	3.854 E-6	7.408 E-8	2.025 E-7	2.531E-08
	<b>1.5</b>	0.3917	0.3901	3.708 E-6	6.065 E-8	3.038 E-7	3.798E-08
	<b>2.0</b>	0.3916	0.3916	3.617 E-6	3.617 E-6	4.051 E-7	5.064E-08
<b>0.48</b>	<b>0.5</b>	0.4801	0.4800	4.876 E-7	4.549 E-8	1.256 E-7	1.570E-08
	<b>1.0</b>	0.4798	0.4799	8.678 E-7	6.809 E-8	2.512 E-7	3.140E-08
	<b>1.5</b>	0.4798	0.4799	1.087 E-6	1.041 E-7	3.768 E-7	4.710E-08
	<b>2.0</b>	0.4798	0.4798	1.225 E-6	1.225 E-6	5.024 E-7	6.280E-08
<b>0.56</b>	<b>0.5</b>	0.5618	0.5601	3.716 E-6	7.222 E-8	1.149 E-7	1.436E-08
	<b>1.0</b>	0.5615	0.5601	2.978 E-6	9.166 E-8	2.298 E-7	2.873E-08
	<b>1.5</b>	0.5614	0.5601	3.004 E-6	9.621 E-8	3.447 E-7	4.309E-08
	<b>2.0</b>	0.5615	0.5615	3.165 E-6	3.165 E-6	4.596 E-7	5.745E-08
<b>0.67</b>	<b>0.5</b>	0.6698	0.6700	4.018 E-7	5.088 E-8	4.221 E-7	5.276E-08
	<b>1.0</b>	0.6698	0.6699	1.169 E-6	1.055 E-7	8.443 E-7	1.055E-07
	<b>1.5</b>	0.6698	0.6699	1.943 E-6	1.627 E-7	1.266 E-6	1.583E-07
	<b>2.0</b>	0.6695	0.6695	3.040 E-6	3.040 E-6	1.688 E-6	2.110E-07
<b>0.73</b>	<b>0.5</b>	0.7315	0.7303	2.412 E-6	1.270 E-7	2.103 E-7	2.629E-08
	<b>1.0</b>	0.7313	0.7303	2.587 E-6	1.538 E-7	4.207 E-7	5.259E-08
	<b>1.5</b>	0.7314	0.7303	2.580 E-6	1.801 E-7	6.310 E-7	7.888E-08
	<b>2.0</b>	0.7313	0.7313	3.009 E-6	3.009 E-6	8.414 E-7	1.052E-07
<b>0.81</b>	<b>0.5</b>	0.8116	0.8103	2.814 E-6	1.809 E-7	1.055 E-7	1.319E-08
	<b>1.0</b>	0.8115	0.8104	2.698 E-6	2.513 E-7	2.110 E-7	2.638E-08
	<b>1.5</b>	0.8114	0.8103	2.577 E-6	2.418 E-7	3.166 E-7	3.958E-08
	<b>2.0</b>	0.8116	0.8116	3.203 E-6	3.203 E-6	4.221 E-7	5.276E-08
<b>0.86</b>	<b>0.5</b>	0.8604	0.8607	4.103 E-7	1.915 E-7	9.726 E-8	1.216E-08
	<b>1.0</b>	0.8604	0.8606	5.494 E-7	3.261 E-7	1.945 E-7	2.431E-08
	<b>1.5</b>	0.8603	0.8606	6.494 E-7	5.732 E-7	2.918 E-7	3.648E-08
	<b>2.0</b>	0.8602	0.8602	7.919 E-7	6.919 E-7	3.891 E-7	4.864E-08

### 5.5 Computational aspects of GA-LSE and GA-ALSE

There are two critical aspects of the GA based ALSE/LSE frequency estimation procedure. The first one is the number of generations required to arrive at the frequency estimate and the second one is the information about the generation-wise progress of the criterion function (LSE or ALSE), in terms of the criterion function value corresponding to the solution found up to that generation. We will report the average number of generations required over 100 simulations in different cases and will report graphically the generation-wise progress of the criterion function. Table 8 gives the average number of generations that were required over 100 simulations for GA-LSE and GA-ALSE. For the GA-ALSE, the first figure inside the parentheses indicates the number of generations required for estimating  $\omega_1$  and the second figure indicates the number of generations required for estimating  $\omega_2$ . The population size in all these computations is fixed at 150.

**Table 8:** Average number of GA generations for finding LSE and ALSE

$\sigma^2$ (SNR)	Average number of GA generations					
	Sample size 25		Sample size 50		Sample size 75	
	GA-LSE	GA-ALSE	GA-LSE	GA-ALSE	GA-LSE	GA-ALSE
<b>0.5</b> <b>(14.15)</b>	75	(32, 32)	73	(31, 34)	74	(37, 29)
<b>1</b> <b>(11.14)</b>	73	(30, 32)	73	(29, 36)	71	(31, 29)
<b>1.5</b> <b>(9.38)</b>	74	(34, 32)	73	(31, 33)	72	(28, 33)
<b>2</b> <b>(8.13)</b>	72	(30, 29)	71	(33, 30)	72	(34, 33)

We observe from the results that GA-LSE is computationally much faster, requiring less number of generations to arrive at the frequency estimates, having nearly the same accuracy as that of GA-ALSE. This is true for all the cases considered.

In Figure 6 and Figure 7, we graphically present the generation-wise progress of the (scaled) optimum criterion function values of two representative cases. Figure 6 presents the GA-ALSE results and Figure 7 reports the GA-LSE results. The results correspond to GA-ALSE and GA-LSE computations with sample size 25 and  $\sigma^2 = 0.5$ .



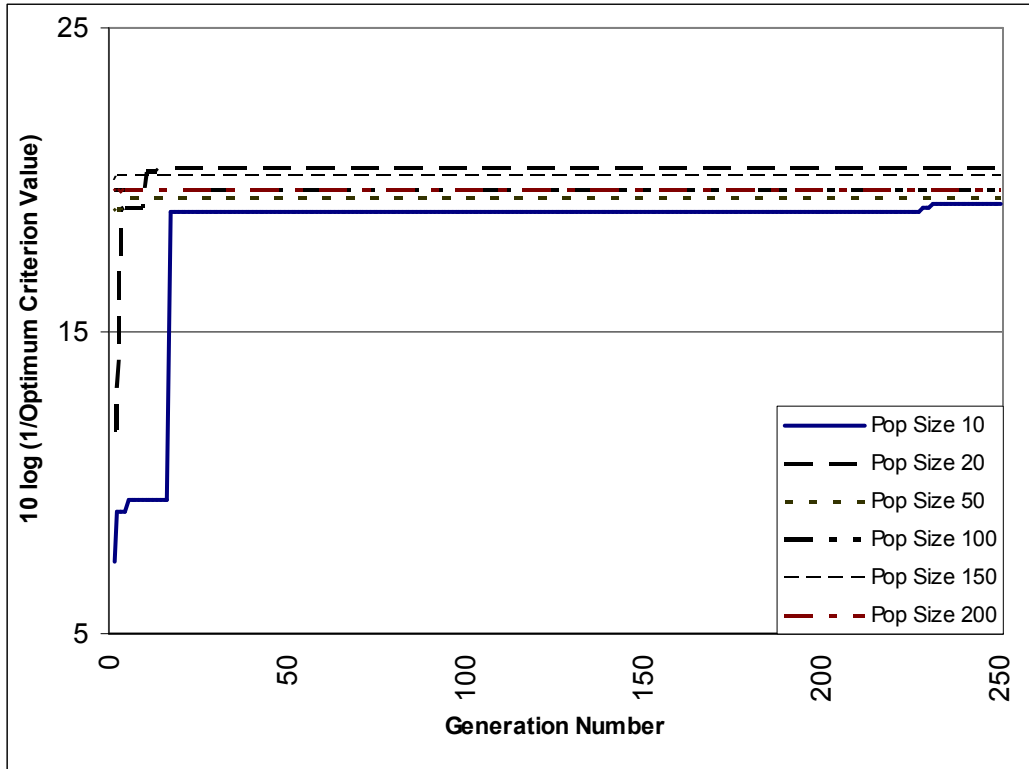


Figure 6: Generation-wise progress of the criterion function for GA-ALSE (frequency 1)

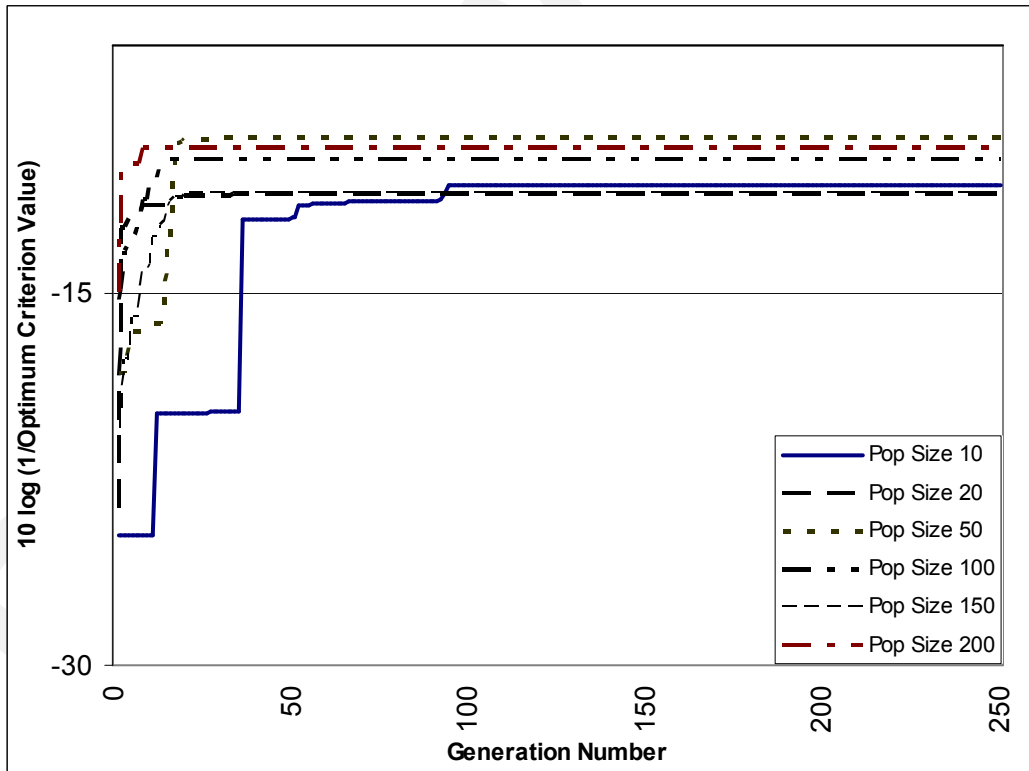


Figure 7: Generation-wise progress of the criterion function for GA-LSE

From the above figures, we observe that, for both GA-LSE and GA-ALSE, optimum criterion function value stabilizes fairly fast, irrespective of the population size. On an average, the GA-ALSE achieves this stabilization of the optimum criterion function value earlier than the GA-LSE. It is also observed that, for both the methods, the final fitness values for different population sizes are very close. This further indicates that the performance of the proposed genetic algorithms based frequency estimation procedures do not significantly depend on the population size, provided the number of GA generations is reasonably large.

### 5.6 Performance of GA-LSE and GA-ALSE in outlier contaminated datasets

We next investigate the effect of presence of various types of outliers on the GA-LSE and GA-ALSE frequency estimates, at different contamination levels. The data for the outlier simulation studies are generated in the following way. To contaminate a simulated dataset with  $k\%$  outliers, we randomly choose  $k\%$  of the total data points and generate the data corresponding to these points from a model different than the original simulation model (19). In the outlier simulation studies, we take the sample size as 25 and consider three different types of outliers. These are; (i) outlier observations are generated from a model with different amplitudes (we call these ‘amplitude outliers’), (ii) outlier observations are generated from a model with different frequencies (we call these ‘frequency outliers’) and (iii) outlier observations are generated from a model with different noise structure (we call these ‘noise outliers’). In case of the ‘amplitude outliers’, we consider the outlier data being generated from

$$y(t) = 1.85 \exp(i 2\pi (0.5)t) + 3.0 \exp(i 2\pi (0.7)t) + \varepsilon(t); \quad t=1, \dots, 25. \quad (21)$$

$\varepsilon(t)$  as in the formulation of model (16). For the ‘frequency outliers’, we consider the outlier data model as;

$$y(t) = 2.0 \exp(i 2\pi (0.35)t) + 3.0 \exp(i 2\pi (0.7)t) + \varepsilon(t); \quad t=1, \dots, 25. \quad (22)$$

Finally for the ‘noise outliers’, we consider the model (19) with  $\varepsilon(t)$  complex Gaussian having variance of its real and imaginary parts as  $10(\sigma^2/2)$ . We consider three levels of outlier contaminations, (i) 4 out of 25 observations are outliers ( $\cong 16\%$ ), (ii) 6 out of 25 observations are outliers ( $\cong 24\%$ ) and (iii) 8 out of 25 observations are outliers ( $\cong 32\%$ ).

$\sigma^2$  is varied from 0.5 ( $SNR \cong 14.2$ ) to 2.5 ( $SNR \cong 7.2$ ). We report the average estimates and the MSEs of the GA-LSE and GA-ALSE, over 100 simulation runs. For comparison of the performance of the proposed methods with usual method of least square frequency estimation, we report the results of Noise Space Decomposition (Kundu and Mitra [1995]) estimator initialized LSE using Quasi-Newton method with Levenberg-Marquardt adjustment (NSD-LM). The results for the 16% outliers are presented in Tables 9-11, 24% outliers in Tables 12-14 and 32% outliers in Tables 15-17.

From the results of the outlier study, we observe that the proposed GA-LSE and GA-ALSE methods perform quite well with the levels and types of outliers considered in the simulation study. The outlier simulation study also brings forth the breakdown points of the proposed methods and the NSD-LM method, with respect to contamination and SNR levels. For the frequency outlier case, it is observed that the GA based methods and the usual method perform equally well at high SNR levels for 16% contamination. It is also observed that, as compared to the proposed methods, the usual method breaks down much earlier, with respect to outlier contamination and SNR levels. The NSD-LM breaks down around 7 SNR at 16% level, around 8 SNR at 24% and around 14 SNR at 32%. The proposed methods clearly outperform the NSD-LM beyond these breakdown points. The breakdown points of the GA based methods are around 7 SNR at 24% and 32% contamination levels. At these points, estimation of the frequency with lower contribution to signal energy deteriorates drastically. For the amplitude outlier case, we observe that the proposed methods as well as the usual NSD-LM perform quite well in all the scenarios considered and this type of outlier perturbation does not seem to have a significant effect. For the noise outlier case, once again, we observe that the proposed methods clearly outperform the usual NSD-LM at higher contamination levels, especially at low SNR levels. Performance of the proposed methods and the NSD-LM method are similar at high SNR levels at all the contamination levels. For the NSD-LM method, we observe that the breakdown points are at the SNR level of 8 at contamination levels 16% and 24%; and at the SNR level of 11 at 32% contamination level. For the GA-LSE and GA-ALSE methods, the respective breakdown points arrives much later, at around SNR level of 7 at 32% contamination level. We thus observe from the outlier simulation study that the proposed methods of frequency estimation seem to be fairly robust in presence of

outliers in the data and they have a reasonably high breakdown point, even at low SNR levels of the signal.

**Table 9:** GA-LSE, GA-ALSE and NSD-LM for 16% frequency outliers scenario

$\sigma^2$ (SNR)		Frequency 1: $\omega_1 = 0.5$			Frequency 2: $\omega_2 = 0.7$		
		GA-LSE	GA-ALSE	NSD-LM	GA-LSE	GA-ALSE	NSD-LM
<b>0.5</b> <b>(14.15)</b>	Av. Est.	0.5002	0.5002	0.5006	0.7003	0.7014	0.7001
	MSE	6.088E-6	6.200E-6	5.697 E-6	1.934E-6	3.709E-6	3.373 E-6
<b>1</b> <b>(11.14)</b>	Av. Est.	0.5006	0.5005	0.5005	0.7002	0.7013	0.7004
	MSE	6.969E-6	6.943E-6	7.784 E-6	2.354E-6	3.832E-6	2.568 E-6
<b>1.5</b> <b>(9.38)</b>	Av. Est.	0.5001	0.5001	0.4994	0.7001	0.7013	0.7005
	MSE	8.127E-6	8.392E-6	8.104 E-6	3.361E-6	4.911E-6	3.598 E-6
<b>2</b> <b>(8.13)</b>	Av. Est.	0.5002	0.5002	0.5004	0.7004	0.7015	0.7000
	MSE	1.163 E-5	1.186 E-5	1.131 E-5	3.821 E-6	5.731 E-6	2.916 E-6
<b>2.5</b> <b>(7.16)</b>	Av. Est.	0.5001	0.5000	0.4817	0.7000	0.7012	0.6900
	MSE	1.465 E-5	1.499 E-5	0.0326	3.026 E-6	4.124 E-6	0.0101

**Table 10:** GA-LSE, GA-ALSE and NSD-LM for 16% amplitude outliers scenario

$\sigma^2$ (SNR)		Frequency 1: $\omega_1 = 0.5$			Frequency 2: $\omega_2 = 0.7$		
		GA-LSE	GA-ALSE	NSD-LM	GA-LSE	GA-ALSE	NSD-LM
<b>0.5</b> <b>(14.15)</b>	Av. Est.	0.5000	0.5000	0.5001	0.6999	0.7013	0.7000
	MSE	1.591 E-6	1.623 E-6	1.374 E-6	5.466 E-7	2.996 E-6	6.309 E-7
<b>1</b> <b>(11.14)</b>	Av. Est.	0.5002	0.5002	0.4999	0.7001	0.7015	0.7000
	MSE	2.621 E-6	2.659 E-6	2.644 E-6	1.307 E-6	3.577 E-6	1.122 E-6
<b>1.5</b> <b>(9.38)</b>	Av. Est.	0.4998	0.4997	0.5000	0.7000	0.7013	0.6999
	MSE	4.288 E-6	4.406 E-6	4.127 E-6	1.827 E-6	4.130 E-6	1.634 E-6
<b>2</b> <b>(8.13)</b>	Av. Est.	0.4998	0.4998	0.4998	0.6999	0.7013	0.6999
	MSE	5.685 E-6	5.809 E-6	6.826 E-6	2.514 E-6	4.959 E-6	2.411 E-6
<b>2.5</b> <b>(7.16)</b>	Av. Est.	0.5002	0.5001	0.5000	0.6999	0.7013	0.6999
	MSE	6.050 E-6	6.252 E-6	6.213 E-6	2.588 E-6	4.181 E-6	2.909 E-6

**Table 11:** GA-LSE, GA-ALSE and NSD-LM for 16% noise outliers scenario

$\sigma^2$ (SNR)		Frequency 1: $\omega_1 = 0.5$			Frequency 2: $\omega_2 = 0.7$		
		GA-LSE	GA-ALSE	NSD-LM	GA-LSE	GA-ALSE	NSD-LM
<b>0.5</b> (14.15)	Av. Est.	0.5000	0.5000	0.4999	0.7001	0.7015	0.7001
	MSE	3.235E-6	3.214E-6	4.108 E-6	1.358E-6	3.474E-6	1.213 E-6
<b>1</b> (11.14)	Av. Est.	0.4996	0.4995	0.4997	0.6999	0.7013	0.7000
	MSE	8.769E-6	8.998E-6	6.356 E-6	2.767E-6	4.256E-6	2.684 E-6
<b>1.5</b> (9.38)	Av. Est.	0.4997	0.4996	0.4996	0.7002	0.7015	0.7001
	MSE	1.019E-5	1.031E-5	8.396 E-6	4.959E-6	6.631E-6	4.259 E-6
<b>2</b> (8.13)	Av. Est.	0.4982	0.4982	0.4968	0.6998	0.7011	0.7007
	MSE	5.662 E-5	6.292 E-5	5.364 E-4	6.594 E-6	7.532 E-6	7.179 E-6
<b>2.5</b> (7.16)	Av. Est.	0.5001	0.4999	0.5120	0.7001	0.7013	0.7167
	MSE	6.261 E-5	6.302 E-5	0.0104	8.169 E-6	9.534 E-6	0.0224

**Table 12:** GA-LSE, GA-ALSE and NSD-LM for 24% frequency outliers scenario

$\sigma^2$ (SNR)		Frequency 1: $\omega_1 = 0.5$			Frequency 2: $\omega_2 = 0.7$		
		GA-LSE	GA-ALSE	NSD-LM	GA-LSE	GA-ALSE	NSD-LM
<b>0.5</b> (14.15)	Av. Est.	0.5004	0.5004	0.5008	0.7002	0.7012	0.6999
	MSE	8.531 E-6	8.724 E-6	7.267 E-6	3.254 E-6	4.301 E-6	2.425 E-6
<b>1</b> (11.14)	Av. Est.	0.5010	0.5010	0.4956	0.7000	0.7010	0.7003
	MSE	1.313 E-5	1.338 E-5	7.639 E-4	2.685 E-6	3.679 E-6	2.984 E-6
<b>1.5</b> (9.38)	Av. Est.	0.5008	0.5008	0.4971	0.7002	0.7013	0.6999
	MSE	1.243 E-5	1.273 E-5	6.277 E-4	3.331 E-6	4.778 E-6	5.134 E-6
<b>2</b> (8.13)	Av. Est.	0.5004	0.5004	0.4924	0.7004	0.7014	0.7004
	MSE	1.684 E-5	1.729 E-5	2.066 E-3	3.261 E-6	4.952 E-6	5.141 E-6
<b>2.5</b> (7.16)	Av. Est.	0.4974	0.4974	0.4402	0.7013	0.7023	0.7100
	MSE	1.411 E-3	1.412 E-3	0.2851	1.494 E-4	1.517 E-4	0.0151

**Table 13:** GA-LSE, GA-ALSE and NSD-LM for 24% amplitude outliers scenario

$\sigma^2$ (SNR)		Frequency 1: $\omega_1 = 0.5$			Frequency 2: $\omega_2 = 0.7$		
		GA-LSE	GA-ALSE	NSD-LM	GA-LSE	GA-ALSE	NSD-LM
<b>0.5</b> (14.15)	Av. Est.	0.4998	0.4998	0.5002	0.7000	0.7013	0.7000
	MSE	1.608 E-6	1.673 E-6	1.467 E-6	5.490 E-7	3.011 E-6	4.039 E-7
<b>1</b> (11.14)	Av. Est.	0.5000	0.4999	0.4999	0.6999	0.7013	0.6999
	MSE	3.249 E-6	3.323 E-6	2.257 E-6	1.319 E-6	3.561 E-6	1.256 E-6
<b>1.5</b> (9.38)	Av. Est.	0.5001	0.5001	0.4999	0.7000	0.7014	0.6999
	MSE	4.766 E-6	4.848 E-6	4.007 E-6	1.838 E-6	4.131 E-6	2.185 E-6
<b>2</b> (8.13)	Av. Est.	0.5000	0.4999	0.4999	0.7002	0.7015	0.6999
	MSE	5.991 E-6	6.203 E-6	5.241 E-6	2.509 E-6	4.977 E-6	2.714 E-6
<b>2.5</b> (7.16)	Av. Est.	0.5006	0.5006	0.5000	0.6998	0.7012	0.7003
	MSE	6.883 E-6	7.016 E-6	6.084 E-6	3.043 E-6	4.393 E-6	2.630 E-6

**Table 14:** GA-LSE, GA-ALSE and NSD-LM for 24% noise outliers scenario

$\sigma^2$ (SNR)		Frequency 1: $\omega_1 = 0.5$			Frequency 2: $\omega_2 = 0.7$		
		GA-LSE	GA-ALSE	NSD-LM	GA-LSE	GA-ALSE	NSD-LM
<b>0.5</b> (14.15)	Av. Est.	0.5001	0.5000	0.5001	0.7000	0.7014	0.6997
	MSE	6.085E-6	6.127E-6	3.776 E-6	2.376E-6	4.162E-6	2.171 E-6
<b>1</b> (11.14)	Av. Est.	0.4992	0.4991	0.5003	0.7003	0.7016	0.7001
	MSE	9.923E-6	1.041E-5	7.333 E-6	3.697 E-6	6.338E-6	3.412 E-6
<b>1.5</b> (9.38)	Av. Est.	0.4999	0.4998	0.4996	0.7003	0.7016	0.6999
	MSE	1.803 E-5	1.877 E-5	1.139 E-5	7.181 E-6	9.186 E-6	6.214 E-6
<b>2</b> (8.13)	Av. Est.	0.5006	0.5004	0.4922	0.7007	0.7019	0.7024
	MSE	7.159 E-5	7.261 E-5	4.457 E-3	9.098 E-6	1.173 E-5	4.512 E-4
<b>2.5</b> (7.16)	Av. Est.	0.5016	0.5016	0.3662	0.7001	0.7013	0.7667
	MSE	1.462 E-4	1.451 E-4	0.6812	7.981 E-6	9.559 E-6	0.3672

**Table 15:** GA-LSE, GA-ALSE and NSD-LM for 32% frequency outliers scenario

$\sigma^2$ (SNR)		Frequency 1: $\omega_1 = 0.5$			Frequency 2: $\omega_2 = 0.7$		
		GA-LSE	GA-ALSE	NSD-LM	GA-LSE	GA-ALSE	NSD-LM
<b>0.5</b> (14.15)	Av. Est.	0.5011	0.5011	0.4824	0.7004	0.7014	0.7003
	MSE	8.745 E-6	8.825 E-6	2.951 E-3	2.517 E-6	4.363 E-6	3.272 E-6
<b>1</b> (11.14)	Av. Est.	0.5008	0.5007	0.4728	0.7006	0.7016	0.7003
	MSE	1.173 E-5	1.192 E-5	0.0138	3.021 E-6	5.269 E-6	4.564 E-6
<b>1.5</b> (9.38)	Av. Est.	0.5009	0.5008	0.4539	0.7003	0.7013	0.7010
	MSE	1.639 E-5	1.669 E-5	0.0878	3.627 E-6	5.149 E-6	1.208 E-3
<b>2</b> (8.13)	Av. Est.	0.5019	0.5019	0.3822	0.7001	0.7011	0.7079
	MSE	2.002 E-5	2.018 E-5	0.8688	4.296 E-6	5.509 E-6	0.0491
<b>2.5</b> (7.16)	Av. Est.	0.5036	0.5058	0.3003	0.7013	0.7042	0.7397
	MSE	1.321 E-3	1.744 E-3	0.9712	5.634 E-4	4.711 E-4	0.0967

**Table 16:** GA-LSE, GA-ALSE and NSD-LM for 32% amplitude outliers scenario

$\sigma^2$ (SNR)		Frequency 1: $\omega_1 = 0.5$			Frequency 2: $\omega_2 = 0.7$		
		GA-LSE	GA-ALSE	NSD-LM	GA-LSE	GA-ALSE	NSD-LM
<b>0.5</b> (14.15)	Av. Est.	0.5000	0.4999	0.4999	0.7000	0.7013	0.6999
	MSE	1.631 E-6	1.692 E-6	1.132 E-6	5.689 E-7	3.140 E-6	5.457 E-7
<b>1</b> (11.14)	Av. Est.	0.4995	0.4996	0.4998	0.7001	0.7014	0.6999
	MSE	2.503 E-6	2.608 E-6	2.552 E-6	1.391 E-6	3.668 E-6	1.021 E-6
<b>1.5</b> (9.38)	Av. Est.	0.4997	0.4997	0.4999	0.6999	0.7013	0.7000
	MSE	4.816 E-6	4.856 E-6	3.523 E-6	1.851 E-6	4.173 E-6	1.205 E-6
<b>2</b> (8.13)	Av. Est.	0.4999	0.4998	0.4998	0.7001	0.7015	0.7001
	MSE	6.231 E-6	6.382 E-6	5.236 E-6	2.529 E-6	5.016 E-6	2.395 E-6
<b>2.5</b> (7.16)	Av. Est.	0.5002	0.5001	0.4997	0.6999	0.7013	0.7001
	MSE	4.242E-6	4.310 E-6	6.534 E-6	3.288 E-6	5.017 E-6	3.680 E-6

**Table 17:** GA-LSE, GA-ALSE and NSD-LM for 32% noise outliers scenario

$\sigma^2$ (SNR)		Frequency 1: $\omega_1 = 0.5$			Frequency 2: $\omega_2 = 0.7$		
		GA-LSE	GA-ALSE	NSD-LM	GA-LSE	GA-ALSE	NSD-LM
<b>0.5</b> <b>(14.15)</b>	Av. Est.	0.5000	0.5000	0.4998	0.7000	0.7013	0.7000
	MSE	4.214 E-6	4.195 E-6	5.475 E-6	2.295 E-6	3.950 E-6	2.058 E-6
<b>1</b> <b>(11.14)</b>	Av. Est.	0.4998	0.4997	0.4991	0.6999	0.7013	0.6999
	MSE	7.738 E-6	8.003 E-6	1.358 E-4	3.265 E-6	4.829 E-6	4.715 E-6
<b>1.5</b> <b>(9.38)</b>	Av. Est.	0.4980	0.4979	0.4994	0.6999	0.7011	0.7001
	MSE	4.382 E-4	4.384 E-4	1.395 E-4	4.898 E-6	6.003 E-6	6.488 E-6
<b>2</b> <b>(8.13)</b>	Av. Est.	0.5001	0.5000	0.4784	0.6999	0.7011	0.7091
	MSE	1.993 E-5	2.020 E-5	0.0421	9.308 E-6	1.059 E-5	0.0063
<b>2.5</b> <b>(7.16)</b>	Av. Est.	0.4999	0.4997	0.2988	0.7056	0.7067	1.0276
	MSE	1.543 E-3	1.545 E-3	3.5712	1.618 E-3	1.627 E-3	9.0657

## 5. Conclusions

In this paper, we propose two methods for frequency estimation of undamped superimposed exponential signals model. The proposed methods use generational GA, with elitism, to obtain least squares and approximate least squares estimates. The proposed GA-LSE method uses the elitist GA stochastic search procedure for locating the optima, with respect to the  $M$  unknown frequencies, of the concentrated likelihood function. The computationally efficient GA-ALSE solves for the unknown frequencies through a sequential  $M$  1-dimensional GA for finding the approximate least squares estimators.

The proposed frequency estimation techniques differ from standard approaches of least squares frequency estimation in a number of ways. Firstly, the GA based least square frequency estimation technique search a population of possible optimal solutions in parallel. Secondly, the proposed methods do not require derivative information or other auxiliary information, only the levels of fitness influence the direction of search. Thirdly, the proposed methods being based on genetic algorithms, use probabilistic transition rules rather than deterministic ones. Because of these, the proposed procedure does not suffer from the drawbacks of the standard least square frequency estimation for this problem.

Simulation results show that both GA-LSE and GA-ALSE gives frequency estimates that are nearly efficient in the sense that the mean square errors of the resultant estimates are

equal or almost equal to the corresponding CRLB. It is further observed that, GA-ALSE is computationally much more efficient than the GA-LSE, with significant reduction in the computational time, attaining almost the same accuracy as that of GA-LSE, even for small sample sizes. The proposed methods appear to be fairly robust with respect to outliers present in the data and are able to resolve the frequencies with high level of accuracy, performing significantly better than the classical methods at high outlier contamination levels.

The proposed methods of frequency estimation can easily be extended to other important signal processing models, known to have computational complexity and heavy dependence on initial value chosen for frequency estimation problem. Furthermore, we can extend the idea used in this paper to obtain M-estimates and other robust estimates, say, robust estimates based on least trimmed squares or least median squares type of approaches (Rousseeuw [1998]).

## REFERENCES

- [1] Bai, Z. D., Rao, C. R., Chow, M. and Kundu, D., 2003. An efficient algorithm for estimating the parameters of superimposed exponential signals. *Journal of Statistical Planning and Inference*, 110, 23-34.
- [2] Baker, J. E., 1987. Reducing bias and inefficiency in the selection algorithm, *Proc. International Conference on Genetic Algorithms 2*, 14-21.
- [3] Bressler, Y. and MaCovski, A., 1986. Exact maximum likelihood parameters estimation of superimposed exponential signals in noise. *IEEE Transactions on Acoustics Speech and Signal Processing*, ASSP 34, 1081-1089.
- [4] Caruana, R.A. and Schaffer, J.D., 1988. Representation and hidden bias: Gray vs. binary coding. *Proc. 6<sup>th</sup> International Conference Machine Learning*, 153-161.
- [5] Chambers, L., 1995a. *Practical Handbook of Genetic Algorithms: Applications*, Vol.I, CRC Press, Florida.
- [6] Chambers, L., 1995b. *Practical Handbook of Genetic Algorithms: New Frontiers*, Vol.II, CRC Press, Florida.
- [7] De Jong, K.A., 1975. An analysis of the behavior of a class of genetic adaptive systems. PhD thesis, University of Michigan.



- [8] Goldberg, D. E., 1989. Genetic algorithms in Search, Optimization, and Machine Learning, *Addison-Wesley*.
- [9] Hollstien, R.B., 1971. Artificial genetic adaptation in computer control systems, Doctoral Dissertation, University of Michigan, Dissertation Abstracts International, 32(3), 1510B, University Microfilms No. 71-23,773.
- [10] Hwang, J.K. and Chen, Y.C., 1993. A combined detection-estimation algorithm for the harmonic-retrieval problem. *Signal Processing*, 30, 177-197.
- [11] Kannan, N. and Kundu, D., 1994. On modified EVLP and ML methods for estimating superimposed exponential signals. *Signal Processing*, 39, 223-233.
- [12] Kumaresan, R., Scharf, L. L. and Shaw, A. K., 1986. An algorithm for pole-zero modeling and spectral analysis. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP 34, 637-640.
- [13] Kundu, D., 1993. Estimating the Parameters of Undamped Exponential Signals. *Technometrics*, 35, 215 – 218.
- [14] Kundu, D. and Mitra, A., 1995. Consistent method of estimating the superimposed exponential signals. *Scandinavian Journal of Statistics*, 22, 73-82.
- [15] Kundu, D. and Mitra, A., 2000a. Detecting the number of signals of undamped exponential model using cross validation. *Signal Processing*, 80, 525-534.
- [16] Kundu, D. and Mitra, A., 2000b. Asymptotic theory of the least squares estimator of superimposed exponential signals. *American Journal of Mathematical and Management Sciences*, 20, 345-365.
- [17] Li, J. and Stoica, P., 1996. Efficient mixed-spectrum estimation with applications to target feature extraction. *IEEE Transactions on Signal Processing*, 44, 281-295.
- [18] Pisarenko, V. F., 1973. The retrieval of harmonics from a covariance function. *Geophysical Journal of the Royal Astronomical Society*, 33, 347-366.
- [19] Quagliarella, D., Periaux, J., Poloni C. and Winter, G., 1997, *Genetic Algorithms in Engineering and Computer Science*, John Wiley and Sons.
- [20] Quinn, B. G., 1994. Estimating frequency by interpolation using Fourier coefficients, *IEEE Transactions on Signal Processing*, 42, 1264-1268.
- [21] Rao, C. R. and Zhao, L. C., 1993. Asymptotic behavior of the maximum likelihood estimators of superimposed exponential signals. *IEEE Transactions on Acoustics Speech and Signal Processing*, 41, 1461-1463.

- [22] Rousseeuw, P.J., 1998. Robust estimation and identifying outliers, H.M. Wadsworth (Ed.), Handbook of Statistical Methods for Engineers and Scientists, McGraw-Hill, New York, Chapter 17.
- [23] Roy, R. and Kailath, T., 1989. ESPRIT: Estimation of signal parameters via rotational invariance techniques. IEEE Transactions on Acoustics, Speech, and Signal Processing, ASSP-37, 984-995.
- [24] Schmidt, R. O., 1979. Multiple emitter location and signal parameter estimation. Proc. RADC, Spectral Estimation Workshop, Rome, NY, pp. 243-258.
- [25] Stoica, P., and Moses, R., 2003. Spectral Analysis of Signals, Prentice Hall, Upper Saddle River, New Jersey.
- [26] Stoica, P., Moses, R. L., Friedlander, B. and Söderström, T., 1989. Maximum likelihood estimation of the parameters of multiple sinusoids from noisy measurements. IEEE Transactions on Acoustics, Speech, and Signal Processing ASSP 37, 378-392.
- [27] Stoica, P. and Nehorai, A., 1989. Statistical analysis of two nonlinear least-squares estimators of sine-wave parameters in the colored-noise case. Circuits, Systems, and Signal Processing, 8, 3-15.
- [28] Thierens, D., 1997. Selection schemes, elitist recombination, and selection intensity. In T. Back (Ed.), Proceedings of the Seventh International Conference on Genetic Algorithms, 152-159, San Francisco, U.S.A.
- [29] Tufts, D. W. and Kumaresan, R., 1982. Estimation of frequencies of multiple sinusoids: Making linear prediction perform like maximum likelihood. Proceedings of the IEEE 70, 975-989.
- [30] Ulrych, T.J. and Clayton, R.W., 1976. Time series modeling and maximum entropy. Phys. Earth and Planetary Interiors, 12, 188-200.
- [31] Yokose, Y., Cingoski, V., Kaneda, K. and Yamashita, H., 2000. Performance Comparison Between Gray Coded and Binary Coded Genetic Algorithms for Inverse Shape Optimization of Magnetic Devices, Applied Electromagnetics, 115- 120.

## Table Captions

**Table 1:** Choice of genetic parameters for the simulations of model (18)

**Table 2:** Performance of the GA-LSE for different bit lengths

**Table 3:** GA-LSE and GA-ALSE frequency estimation results for sample size 25

**Table 4:** GA-LSE and GA-ALSE frequency estimation results for sample size 50

**Table 5:** GA-LSE and GA-ALSE frequency estimation results for sample size 75

**Table 6:** Performance of the proposed methods for different degree of separation

**Table 7:** Simulation results of GA-ALSE for model (20)

**Table 8:** Average number of GA generations for finding LSE and ALSE

**Table 9:** GA-LSE, GA-ALSE and NSD-LM for 16% frequency outliers scenario

**Table 10:** GA-LSE, GA-ALSE and NSD-LM for 16% amplitude outliers scenario

**Table 11:** GA-LSE, GA-ALSE and NSD-LM for 16% noise outliers scenario

**Table 12:** GA-LSE, GA-ALSE and NSD-LM for 24% frequency outliers scenario

**Table 13:** GA-LSE, GA-ALSE and NSD-LM for 24% amplitude outliers scenario

**Table 14:** GA-LSE, GA-ALSE and NSD-LM for 24% noise outliers scenario

**Table 15:** GA-LSE, GA-ALSE and NSD-LM for 32% frequency outliers scenario

**Table 16:** GA-LSE, GA-ALSE and NSD-LM for 32% amplitude outliers scenario

**Table 17:** GA-LSE, GA-ALSE and NSD-LM for 32% noise outliers scenario

## Figure Captions

**Figure 1a:** A one-point crossover

**Figure 1b:** A 4-point crossover

**Figure 2:** Algorithmic steps of a generational GA using elitism

**Figure 3:** Binary coding of the original frequency parameters

**Figure 4:** GA-LSE algorithm

**Figure 5:** GA-ALSE algorithm

**Figure 6:** Generation-wise progress of the criterion function for GA-ALSE (frequency 1)

**Figure 7:** Generation-wise progress of the criterion function for GA-LSE