HIGH RESOLUTION SPECTRAL ANALYSIS USING A COMBINATION OF AN ORTHOGONAL APPROACH AND A GENETIC ALGORITHM

Jean-Marc Vesin
Signal Processing Laboratory
Swiss Federal Institute of Technology
CH-1015 Lausanne, Switzerland
Tel: +41 21 693 3996; fax: +41 21 693 7600
e-mail: vesin@ltssg4.epfl.ch

ABSTRACT

We describe in this paper how a method for parsimonious sinusoidal representation of signals based upon an orthogonalization technique can be suitably modified by embedding it into a genetic algorithm. We first describe the orthogonalization formalism, then we present the genetic algorithms in general and the specific form, based on a floating–point parameter representation, that we have employed in this work. Experiments are presented and possible extensions are discussed.

1 INTRODUCTION

In a series of papers, whose good representatives are [1, 2], Korenberg *et al.* introduced an orthogonal method for nonlinear signal model estimation and parsimonious sinusoidal representation of data. This method consists in finding the optimal projection in a least-squares sense of the input time series upon a non orthogonal basis of signal vectors by using a Gram-Schmidt orthogonalization process upon this basis. In the case of the parsimonious sinusoidal series representation, this basis is formed by sine and cosine signal pairs whose frequencies have been preselected. Examination of the error reduction brought by the corresponding components of the orthogonalized basis allows to retain in the original basis only the sine/cosine pair yielding the highest error reduction. The coordinates of the input time series in the original basis, i.e. the coefficients of the sine/cosine signals in the representation of this series, are readily obtained from its coordinates in the orthogonalized basis. From them the amplitudes and phases of the sinusoidal components of the data series are of course easily computed. This scheme allows to find significant frequencies in the time series which, in contrast to those obtained with a conventional Fourier approach, need not be commensurate nor integral multiples of the fundamental frequency corresponding to the length of the time series. Moreover, the length of this time series may take any value, and this scheme is easily extended to irregularly sampled time data. Of course, this parsimonious representation can be used for signal compression purposes.

However, this technique presents the disadvantage of requiring a selection of the frequencies that will be used to constitute the original signal basis. Accuracy of the selection method presented in [1] in the context of spectral analysis (selection based upon error reduction on a set of regularly spaced candidate frequencies) depends upon the frequency step. Additionally, it is not appropriate in the context of wide-band signal compression, since the fact that the original signal basis is not orthogonal imposes a *simultaneous* selection of all its components.

This is why we propose in this paper to use a genetic algorithm (GA) to select the appropriate frequencies for the estimation process. GA's were introduced by J. Holland [3] as a class of search procedures, and they consist in an imitation of Darwinian evolution on a population of initially randomly selected fasible solutions. GA's have been applied with success to various optimization problems [4]. However, in a vast majority of applications, the GA used is based upon a bit string representation of the "chromosomes" (solutions) in the population, which implies a quantification of the parameters. We use here an alternative class of GA's using a numerical representation which alleviates the problems associated with quantization.

The organization of the paper is as follows: First, Korenberg's method for parsimonious sinusoidal is briefly reviewed in section 2. Then the GA using a numerical representation is introduced in section 3 and its adaptation to the problem of frequency selection is described in section 4. Section 5 presents some experimental results and section 6 concludes the paper.

2 ORTHOGONAL SEARCH OF A PARSIMONIOUS SINUSOIDAL REPRESENTATION

Let $x(n)$, $n = 1, ..., N$, represent the time series under analysis, that we will suppose for now to be regularly sampled. The basic principle of the method consists in finding the best representation in a least-squares sense

$$x(n) = \sum_{i=0}^{K} a_i g_i(n) + e(n) \qquad (1)$$

where the $g_i(n)$ constitute a set of functions of interest. The coefficients a_i should thus minimize $\frac{1}{N}\sum_{n=1}^{N} e^2(n)$.

In the case of parsimonious sinusoidal representation, the set of functions is defined by $g_0(n) = 1$ and, with K even, $g_{2i-1}(n) = \cos(2\pi f_i n)$, $g_{2i}(n) = \sin(2\pi f_i n)$, $i = 1, ..., \frac{K}{2}$. Eq. 1 can be rewritten as:

$$x(n) = \sum_{i=0}^{K} \alpha_i \gamma_i(n) + e'(n) \qquad (2)$$

where the functions $\gamma_i(n)$ are now orthogonal, i.e. $\sum_{n=1}^{N} \gamma_i(n)\gamma_j(n) = \delta_{ij}$, with δ_{ij} standing for the Kroneker symbol. The $\gamma_i(n)$ are obtained from the $g_i(n)$ by the well-known Gram-Schmidt orthogonalization procedure. The first function vector chosen is $g_0(n)$, and the corresponding coefficient is of course $a_0 = \frac{1}{N}\sum_{n=1}^{N} x(n)$. Once this procedure has been completed, it is easy to compute the coefficients α_i and, from them, the a_i. An immense advantage of this approach is that, at each step j of the orthogonalization procedure, it is possible to get the error reduction brought by the term $\gamma_j(n)$ and, hence by $g_j(n)$. So, in fact, in order to select the best $\frac{K}{2}$ frequencies, it is preferable to consider a much larger set of candidate frequencies. At each step, all as yet unselected frequencies are tested, and the one giving the highest error reduction is selected.

Usually, these candidate frequencies are uniformly spaced on the interval of admissible frequencies, which restricts the accuracy to the frequency step. Moreover, while this strategy seems appropriate for signals presenting mainly spectral peaks, it is less so for signals having a continuous power spectrum: since the original basis is not orthogonal, a *sequential* selection of the frequencies, and hence of the terms of the terms of the sinusoidal representation, is not guaranteed to be optimal. An alternative selection scheme based on a genetic algorithm will be presented now.

3 GENETIC ALGORITHMS

3.1 Classical genetic algorithms

GA's constitute a class of robust optimization techniques, which do not require differentiability or even continuity of the search space. One may also say that GA's are easy to implement and that their tuning relies on a proper selection of only a few parameter values.

A GA typically consists in:

1. A population of suitably encoded solutions to the problem at hand.

2. An evaluation function allowing to rank the solutions (often called the fitness measure)

3. Operators inspired by biology, permitting to create new solutions starting from existing ones.

4. An evolution strategy for the creation of a new population starting from the current one, i.e. a way to create successive generations of solutions.

5. A termination criterion, usually based on a maximum number of generations or a threshold on the fitness measure.

The operators mimick the biological ones of crossover and mutation, and the choice of the solutions upon which they are used is dictated by evolutionary principles such as *survival and reproduction of the fittest*, the fittest solutions being of course the best scoring ones with respect to the evaluation function. In an overwhelming majority of published works encoding of the solutions is performed as follows: The parameter values forming a solution are quantized, then binarized, and all these binary representations are chained to form a unique bit string called for obvious reasons a *chromosome*. Three operators are generally considered for building a new generation starting from the current one: reproduction, crossover and mutation. Reproduction corresponds to copying an existing chromosome into the new population. Crossover generally consists in building a new chromosome from two other ones referred to as the parents by concatenating randomly chosen parts of these two chromosomes. Mutation creates new chromosomes starting from existing ones by random alteration of some bits, which enables the GA to explore new regions of the solution space. Usually this operation is performed by considering in turn each bit of a chromosome, and replacing it by a randomly selected one with a predefined probability called the mutation rate.

Extensive use of this scheme and of the corresponding operators is mainly due to its simplicity and its generality. However, two major disadvantages of this approach are the following: First, the operators do not take into account any possible link between different solution parameters, and crossover mixes these parameters in an arbitrary way. Convergence may therefore be slowed. Second, the solution obtained is a quantized one. Improvement in the accuracy implies additional bits in the chromosomes and hence a decrease in the convergence speed.

3.2 Genetic algorithms in the continuous space

In order to overcome these limitations, we have decided to opt for a different kind of GA, subsequently called genetic algorithm in the continuous space (GACS), in which no binarization is performed, the chromosomes being the usual parameter vectors. Suitable mutation and crossover operators have to be defined. The idea introduced in [5] consists in creating operators linked to constraints on the solution parameters, in order both to obtain admissible solutions and to possibly enhance the convergence process. More precisely the focus is on situations where the solution parameter vector is constrained to belong to a convex domain, called the acceptance domain from now on. This type of constraint is very common, and it encompasses constraints of boundedness or positivity. Nevertheless, these operators should be consistent with the principles they stand for: Mutation must remain a mean for a population to explore the parameter space, and crossover must constitute a way to merge the good characteristics of two chromosomes.

When things are considered from a numerical viewpoint , it seems natural to opt for a mutation operator consisting in the addition of a Gaussian distributed random vector to the chromosome vector subject to mutation. Mutation of a chromosome $\vec{v}_d$ into a chromosome $\vec{v}_a$ can thus be defined by the following operation:

$$\vec{v}_d \longmapsto \vec{v}_a = \vec{v}_d + \nu, \qquad (3)$$

where ν is a random vector with independent Gaussian components, whose variances must be appropriately chosen. This mutation is accepted only if the resulting chromosome $\vec{v}_a$ lies inside the acceptance domain.

In what concerns crossover, it is also desirable that acceptable parent chromosomes produce an acceptable offspring. In order to fulfill this requirement, and since the resulting chromosome should be "somewhere in between" its parents, a natural choice is the following one. Crossover of two chromosomes $\vec{v}_f$ and $\vec{v}_m$ yielding a chromosome $\vec{v}_s$ can be defined in the following way:

$$\vec{v}_f, \vec{v}_m \longmapsto \vec{v}_s = (1-\lambda)\vec{v}_f + \lambda\vec{v}_m, \qquad (4)$$

where λ is a random variable uniformly distributed in $[0,1]$. Convexity of the acceptance domain ensures that the resulting chromosome $\vec{v}_s$ will be a valid solution if $\vec{v}_f$ and $\vec{v}_m$ are valid too.

4 ADAPTATION OF A GACS TO THE PROBLEM AT HAND

If a GACS is to be considered in the present context, each chromosome is of course composed of $\frac{K}{2}$ frequencies. If they are normalized, they all belong to the interval $[0, \frac{1}{2}]$. In consequence, the acceptance domain is an hypercube of dimension $\frac{K}{2}$, and is obviously convex. We remarked that, in order to obtain faster convergence, it was advisable to sort the frequencies of the chromosomes in the population, so that crossover "mixes" frequencies in the same ranges.

In what concerns the evaluation function, the orthogonalization procedure is applied to the set of sine/cosine pairs defined by the frequencies of a chromosome, and the best one is the one minimizing the least-squares error.

We will describe briefly below the specific evolution strategy used in the experiments presented in this paper. This scheme, which is well adapted to small populations, was first introduced in [6] . It is summarized as follows:

1. Generate an initial population of M chromosomes inside the acceptance domain, where M is a small odd number.

2. Rank these chromosomes with the evaluation function. The best ranking one is then kept unchanged for the next generation (reproduction).

3. Remaining chromosomes compete in pairs. Crossover is then applied to the winners of these local competitions until $\frac{(M-1)}{2}$ new chromosomes are formed. These latter ones are incorporated into the next generation. Considering local instead of global competitions allows to conserve genetic diversity in the next generation, due to the fact that global winners would have a stronger tendency to be close to each other.

4. Complete the next generation by applying mutation to the best ranking chromosome $\frac{(M-1)}{2}$ times.

5. Repeat steps 2 to 4 until convergence occurs, i.e. no change takes place during a given number of generations (typically 10 to 20).

6. Place the best chromosome in a new initial generation where the $(M-1)$ other chromosomes are randomly created and start from step 2 again. Iterate the process until a satisfactory solution is obtained. This procedure is introduced to avoid premature convergence.

Since the best chromosome is always kept for the next generation, with $\frac{(M-1)}{2}$ chromosomes created by mutation, and mutation for this GACS consists in the addition of a Gaussian random vector, convergence is indeed guaranteed. This is due to the fact that the GACS performs at least like a random search algorithm in a bounded search space with infinite-support distribution perturbations . We observed however in the experiments presented below that crossover greatly enhances the convergence speed. A last improvement consists in defining the initial population so that the interval $[0, \frac{1}{2}]$ is regularly covered.

5 EXPERIMENTS

5.1 General presentation

In all the experiments that follow, a population of $M = 11$ chromosomes was used, with a pre-defined chromosome size (i.e. number of frequencies). The variance of the components of the vectors added for mutation was taken to be 0.0025. The results presented correspond to Monte Carlo simulations over 50 runs.

5.2 Experiment 1

The algorithm was applied to a signal composed of a single sinusoid of amplitude $A = 1$ at normalized frequency $f = 0.387$ with various signal lengths N and a SNR of 20 dB, in order to check the accuracy of the method. The results are summarized in the following table:

par. est.	$N = 25$	$N = 13$
$\widehat{f}$	0.3870 ± 0.0003	0.387 ± 0.001
$\widehat{A}$	1.002 ± 0.001	1.00 ± 0.03

Note that the accuracy of the frequency estimates is superior to the Fourier resolution (resp. 0.04 and 0.08). Runs over 40 generations took place, which means that on the whole 440 (40 times 11 chromosomes) estimations were performed. In order to get the same accuracy, at least 500 estimations would have been necessary with Korenberg's approach.

5.3 Experiment 2

In this experiment we considered a test signal introduced in [2], that is, a 128–point time series composed of 5 sinusoids with additive white noise and an overall SNR of 20 dB. The SNRs for the individual frequencies are (frequency / SNR): 0.01 / -5.4 dB, 0.03 / 14.1 dB, 0.031 / 14.1 dB, 0.06 / 8.2 dB, 0.088 / 0.3 dB. Runs over 100 generations took place, and the results are summarized in the following table:

True		Estimated	
freq.	ampl.	freq.	ampl.
0.010	0.1	0.010 ± 0.0002	0.12 ± 0.03
0.030	1.0	0.029 ± 0.004	0.9 ± 0.3
0.031	1.0	0.031 ± 0.001	1.1 ± 0.3
0.060	0.5	0.060 ± 0.0002	0.50 ± 0.01
0.088	0.2	0.088 ± 0.0006	0.20 ± 0.01

These results are as good as the ones obtained with Korenberg's method, except for the sinusoids at 0.030 and 0.031, whose amplitudes display a large variance. However, it is to be remarked that in reference [2], the frequency interval on which the candidate frequencies were chosen was limited to $[0, 0.1]$, while in our case the entire interval $[0, \frac{1}{2}]$ was taken into account, i.e. no a priori information was used.

5.4 Experiment 3

In this last experiment we applied our sinusoidal representation search method to a 32-point passband signal generated with a second order linear autoregressive model. Runs over 40 generations were performed for increasing numbers of sinusoids. The following table presents the approximation error reduction in dB when compared to the representation built with the same number of largest–amplitude DFT terms.

number of terms	1	2	3	4
error reduction (dB)	0.75	1.0	1.1	1.4

In this experiment, the advantage of the GA approach is clear: since no a priori selection of the representative frequencies has to be done, the algorithm tries to find the best *global* representation.

6 DISCUSSION AND CONCLUSION

We have presented in this paper a genetic overhead to a method for parsimonious sinusoidal representation of signals, which alleviates the need to pre–select the frequencies of interest by performing a global search on the frequency axis. Experiments show that automatic frequency selection is indeed possible, with no limit upon the frequency resolution, except of course those imposed by the data themselves. The major drawback of the method is that it imposes to fix the number of frequencies. However, it can be applied with different values of this parameter, and the best solution in the sense of a criterion such as Rissanen's Minimum Description Length (MDL) [7] can be picked up. Another solution, which has only been preliminary tested but seems reliable, consists in modifying the chromosomes by allowing them to have variable lengths and adding to them the number of frequencies as an additional parameter. Mutation and crossover operators have to be adapted, and the evaluation function becomes the MDL.

References

[1] M.J. Korenberg. A robust orthogonal algorithm for system identification and time-series analysis. *Biol. Cybern.*, 60:267–276, 1989.

[2] M.J. Korenberg and L.D. Paarmann. Orthogonal approaches to time-series analysis and system identification. *IEEE Sig. Proc. Mag.*, pages 29–43, July 1991.

[3] J.H. Holland. *Adaptation in Natural and Artificial Systems.* Univ. of Michigan Press, Ann Arbor, MI, 1975.

[4] Davis L., editor. *Handbook of Genetic Algorithms.* Van Nostrand Reinhold, New York, 1991.

[5] J.M. Vesin. Genetic algorithms in the continuous space for recursive adaptive filter design. In *Proc. 7th European Sig. Proc. Conf. EUSIPCO-94*, pages 732–735, Edinburgh, U.K., September 1994.

[6] D.E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning.* Addison-Wesley, Reading, Mass., 1989.

[7] J. Rissanen. *Stochastic Complexity in Statistical Inquiry.* World Scientific, Singapore, 1989.