A genetic approach to ARMA filter synthesis for EEG signal simulation

Cesar Janeczko
Department of Electronics,
CEFET-PR
Av. 7 de setembro, 3165,
Curitiba (PR), Brazil
janeczko@daeln.cefetpr.br

Heitor S. Lopes
Bioinformatics Laboratory
CPGEI / CEFET-PR
Av. 7 de setembro, 3165,
Curitiba (PR), Brazil
hslopes@cpgei.cefetpr.br

Abstract- This paper describes the computational simulation of an electroencephalographic (EEG) signal (background activity, alpha waves) by filtering a white noise with an ARMA (Autoregressive Moving Average) filter. The filter coefficients were obtained interactively using genetic algorithms, comparing the spectrum of a real and a simulated signal. Results demonstrate the feasibility of the technique.

1 Introduction

The human electroencephalographic (EEG) signal is very complex, stochastic by nature and dependent of a number of variables, such as the localization in the scalp, the level of awareness and the underlying mental state of the individual, among others. Frequently, a typical EEG has a random component overlapped with some known rhythm, specially alpha, beta or theta waves. In particular, alpha waves appear mainly in the occipital region of the head, when the individual is relaxed (but not sleepy) and with eyes closed.

The computational simulation of the EEG is a challenging task due to the variability and complexity of the signal. As consequence, it is rarely found in the literature aproaches to this task. In the past, a mathematical method was proposed for simulating some simple patterns of the electric activity of the brain (Zetterberg, 1973; Zetterberg and Ahlin, 1975). This method is based on the filtering of a white noise by an Autoregressive-Moving Average (ARMA) filter. More details about this method will be presented in the next section.

The Zetterberg's method requires the computation of the ARMA filter coefficients, but this is not trivial at all, since this requires the minimization of nonlinear equations based on some assumptions. Anyway, it is possible to circumvent this difficulty by using quasi-optimal estimators. Typical approximated methods are: maximum likelihood estimator (MLE), Akaike´s method, modified Yule-Walker method, singular value decomposition (SVD) method, Durbin method, etc. Some of these are described in details in Zetterberg (1977), Makhoul (1975) and Eykhoff (1974), but even so, the computational implementation of these methods are very hard.

The difficulty in estimating the filter coefficients using traditional methods, which usually employ statistical parameters and matrix operations, has motivated the development of an iterative methodology. This methodology is based on the idea that the adjustment of the filter coefficients is like a search in a n-dimensional space, where the dimension depends on the order of the filter.

2 Background

2.1 Zetterberg's method for simulating EEG signals

The computational simulation of an EEG signal can be satisfactory accomplished by means of a method developed by Zetterberg (1973, 1975, 1978). In this method, the simulated EEG is the result of filtering a white noise source of specific characteristics (normal distribution, zero mean, variance σ^2 and flat spectrum in the frequency range of interest) with an ARMA filter. The linear differential equation [1] represents the filter, order p (where $p \geq q$).

$$x_v + a_1 x_{v-1} + \cdots + a_p x_{v-p} = e_v + b_1 e_{v-1} + \cdots b_q e_{v-q} \quad [1]$$

where e_v is the input (white noise), x_v is the output of the filter, a_p and b_q are the filter coefficients.

2.2 Autoregressive-Moving Average filter

The ARMA filter is a composition of the AR (Autoregressive) and the MA (Moving Average) filters, and can be expressed by the following transfer function:

$$H_{ARMA} = \frac{B(z-1,q)}{A(z-1,p)} \quad [2]$$

where the numerator is related to the MA and the denominator, to the AR models. Usually, the AR filter is more suitable for the simulation of signals which spectrum displays sharp peaks. In the other hand, the MA filter is more suitable to signals with deep valleys in the spectrum.

The computational implementation of an ARMA filter requires that the linear differential equation [1] must be changed to its recursive form, as shown in equation [3].

$$x(n) = e(n) + b_1 e(n-1) + \cdots + b_q e(n-q) - a_1 x(n-1) - a_2 x(n-2) - \cdots - a_p x(n-p) \quad [3]$$

To estimate the order of an ARMA filter, we used the same approach proposed by Vaz, Oliveira and Príncipe (1987) for the estimation of autoregressive (AR) filters. Also, Lopes da Silva (1987) suggests that both AR and ARMA filters can be used for both EEG simulation and signal analysis. The analysis of the EEG using autoregressive modeling is done by applying the EEG signal to the inverse of the estimated ARMA filter. As result, it is expected a noise with normal distribution, zero mean and variance σ^2, since that it is considered a stationary process (for the sake of the background activity). If the resulting noise does not follow the normal distribution, an artifact or relevant event in the EEG is detected, such as spikes or sharp waves.

3 Methodology

3.1 White noise generator

To meet the requirements of the Zetterberg method, the white noise source has to have normal distribution, with zero mean and a given variance. With these settings, the result has a Gaussian distribution and the power spectrum is flat in the frequency range of interest (0-100 Hz). A routine in C programming language was developed to implement the white noise generator.

3.2 The EEG signal

A 5 seconds segment (corresponding to 1000 samples) taken in the C3 lead of a real patient was used to compare with the simulated signal. This segment had no significant graphoelements and was chosen by an expert physician who has classified it as being a background activity with predominance of alpha waves. This real signal was used for comparison with the simulated signal.

3.3 Time to frequency transformation

The fitness function, described below, compares the spectrum of the real EEG with the spectrum of the simulated EEG. A routine in C was implemented to transform the signal from the time to the frequency domain. Since the selected segment for comparison has 1000 samples and not 1024, it was not possible to use the Fast Fourier Transform (FFT) instead, the Discrete Fourier Transform (DFT) algorithm was used.

3.4 Chromosome structure

In the chromosome, the variables of the problem are represented as genes. These variables are the filter coefficients, and the number of genes in the chromosome depends on the filter order.

As first approach, a 5th-order filter was used to simulate a 5-sec. EEG segment. For this filter, nine coefficients have to be found: a_1, a_2, a_3, a_4, a_5, b_1, b_2, b_3 and b_4. The filter coefficients are in the range [-1,+1], and 12 bits per variable were used, giving a chromosome 108 bits long, thus a search space of approximately 3,25 x 10^{32}. For higher order filters, the size of the chromosome (and the search space) increases proportionally. Therefore, for the 5th-order filter, we have a chromosome 168 bits long and a search space 3,74 x 10^{50}.

3.5 Fitness function

In order to compute how close is a simulated signal from the original, the fitness function shown in equation 4 was used. It is as a function of the error between the spectrum of the simulated and the real signal, computed point-to-point in the frequency domain.

$$fitness = 100.\left(100 + \sqrt{\sum_{i=0}^{100} error_i}\right)^{-1} \quad [4]$$

Equation 4 makes the fitness function less sensible to large errors, inducing a faster convergence for the genetic algorithm when the error decreases, as shown in figure 1.

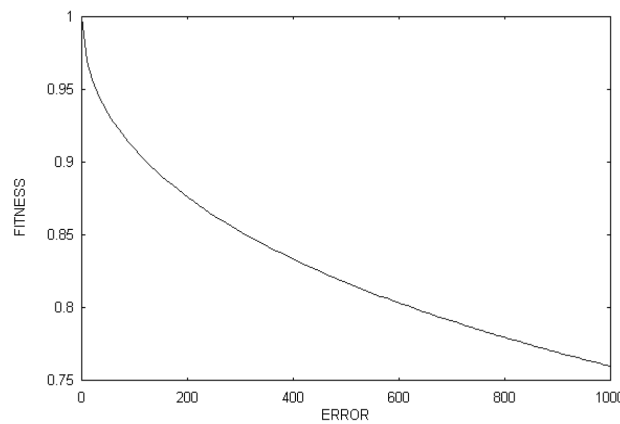


Figure 1: Shape of the fitness function.

The error used in equation 4 is computed as a weighting function according to the frequency range and the absolute value of the DFT. This is done in order to induce the convergence of the AG in such a way to favour individuals with the spectral shape similar to the real signal. The weights used were empirically obtained by observing the DFT of the real signal, shown in figure 2(a). Using this weighting system, it is possible to penalty individuals that generate a simulated signal somewhat similar (to the real signal) in the time domain but quite different in the frequency domain. The weights were set according to table 1, where W_i is the weight value, F_i is the frequency range (or limit) in Hz and A_i is the amplitude of the spectral component of the simulated signal.

Table 1: Weights used in the fitness function.

W_i	value	F_i	A_i
W_1	1.5 – 2.5	< 8	> 20
W_2	1.0 – 2.0	8 – 10	> 60
W_3	3.5 – 4.5	> 10	> 1
W_4	0.1 – 0.7	none of the above	

Using the weights shown in table 1, the error was computed for each spectral component, using equation 5:

$$error_i = W_x.(S_i - R_i)^2 \qquad [5]$$

where S_i is the amplitude of the spectral component of the simulated signal and R_i the same of the real signal.

3.6 Genetic algorithm parameters

In all experiments, we used the tournament selection using 10% of the population. This selection method imposes less selective pressure on the GA than the traditional roulette wheel and is suitable for a slower convergence. The genetic operators were: double point crossover with probability of 95% and single bit mutation with probability of 10% per bit. The GA was run for 100 generations and used a population of 200 individuals.

3.7 Simulation and analysis

The GA used in this work was based in a public domain software named GALOPPS, version 3.2 (Goodman, 1996). For each filter order, the GA was run 10 times with the same parameters, but different initial random seed. The best individual of all runs was elected as result for that filter. After running the GA, the better individual found was used in a simulation program, written in C, with a graphic output. This program simulates the ARMA filter, whose coefficients were found by GA, using the Zetterberg approach, previously mentioned. In addition, it was necessary to use a digital filter following the ARMA filter in order to remove high frequency noise. This filter was a low-pass second-order Butterworth filter, with unity gain and –3 dB point at 20 Hz.

4 Results

The real signal used in this work had some useful characteristics for comparison with a simulated signal. Its largest spectral component had an absolute value of 60 and was in 9 Hz (characteristic of alpha waves). By integrating the whole spectrum, from 0 to 100 Hz, it is possible to have an additional quantitative parameter for comparison, resembling the spectral power of the signal. For the original signal, the integral was 268.25.

Table 2 summarizes the better results found for the simulations of ARMA filters from order 5 up to 8. In this table, symbols S_{error} , A_{lsc} , F_{lsc} stands respectively for: the weighted sum of the errors (according to equation 5), the amplitude of the largest spectral component and the frequency of the largest spectral component. The table also shows the best filter coefficients a_i and b_i obtained by the GA.

The real signal as well as the simulated signals using the filter coefficients of table 2 are shown in figure 3. In these figures 200 samples are equivalent to 1 second of the signal. In the same way, the real signal in frequency domain (calculated using the DFT), as well as the simulated ones are shown in figure 2.

Table 2: Results for four different filter orders.

Filter order	**5**	**6**	**7**	**8**
S_{error}	128.4	93	85	41
A_{lsc}	65	64	65	69
F_{lsc}	7	8	9	9
Integral	732.5	537	526	307
a_1	0.27099	0.40479	-0.42432	0.56396
a_2	-0.74268	-0.67139	0.30518	-0.39941
a_3	0.66162	-0.62061	-0.54443	-0.90088
a_4	0.10059	-0.48780	-0.38135	-0.19628
a_5	-0.61035	0.41309	0.13281	-0.06396
a_6		0.64453	-0.03174	0.24014
a_7			0.55127	0.46973
a_8				0.48633
b_1	0.04932	-0.73340	-0.63867	-0.61621
b_2	0.27002	0.93506	0.78662	0.73828
b_3	0.72559	-0.81787	-1.0	-0.41162
b_4	0.41748	0.27099	-0.750	-0.07178
b_5		0.27246	-0.26123	0.52246
b_6			-1.0	-0.87939

5 Discussion

According to Vaz, Oliveira and Príncipe (1987), the simulation of signals of small duration (1 to 2 seconds) requires smaller orders for the ARMA filter. We started the GA searching for the coefficients of an ARMA filter using an 5-sec EEG segment with predominance of alpha rhythm. Results obtained were very bad and were not reported here, although successive decreasing of the signal duration yielded better results. Then we decided to keep the original duration and increase the filter order, thus leading to a harder problem for the GA.

The analysis of table 2 shows that the results obtained for the 8th order filter, what is concerned to A_{lsc}, F_{lsc} and the integral, are quite close to those for the original signal. The frequency of the largest spectral component is the same and relation between A_{lsc} for the simulated signal and the real is 60/69=0.8695. In the same way, the relation between F_{lsc} for the simulated signal and the real is 268.25/307=0.8738.

Therefore, the simulated signal has a larger average amplitude (in time domain) than the real signal, what is not much significant.

6 Conclusions

The main goal of this work was achieved since the coefficients of the ARMA filter were successfully computed for the simulation of a very complex signal.

When examining the results, it should be kept in mind the motivation of this work, since they have to be analysed per se, due to the difficult of comparison of this technique with others for ARMA filters design.

ARMA filters are useful not only in biological signal processing, but also in control systems. The method proposed in this work may be of some interest in replacing existing methods.

Further work shall be done including the simulation of other types of EEG signals (with predominance of other rhythms) and the comparison of the Zetterberg's method to others for EEG simulation as in (Barlow, 1993) and (Janeczko and Lopes, 2000).

Also, we intend to use a more sophisticated fitness function that would take into account not only the error between the simulated and the real signal as used here, but also other objectives.

Acknowledgements

Authors would like to thank Dr. P. Gewehr and Dr. A. Stelle for the many insightful comments and reviews in previous drafts of this paper. Also thanks to CAPES for the grant to C. Janeczko.

Bibliography

Barlow, J.S. (1993) *The Electroencephalogram, its Patterns and Origins,* Cambridge: The MIT Press.

Eykhoff, P. (1974) *System Identification, Parameter and State Estimation*, London: John Wiley & Sons.

Goodman, E.D. (1996) "An introduction to GALOPPS, the Genetic Algorithm Optimized for Portability and Parallelism System", *Technical Report #96-07-01*, East Lansing : Michigam State University.

Janeczko, C. And Lopes, H.S. "Background EEG activity simulation using autoregressive models and double modulation". In: *Proceedings of World Congress on Medical Physics and Biomedical Engineering*, Chicago, IL, july 2000.

Lopes da Silva, F. (1987) "EEG analysis: theory and practice". In: Niedermeyer, E. and Lopes da Silva, F. (eds.), *Eletroencephalography: basic principles, clinical applications, and related fields*, 2st ed, Baltimore: Urban & Schwarzenberg, pp. 885-888.

Makhoul, J. (1975) "Linear prediction, a tutorial review," *Proceedings of the IEEE*, vol. 63, pp. 561-580.

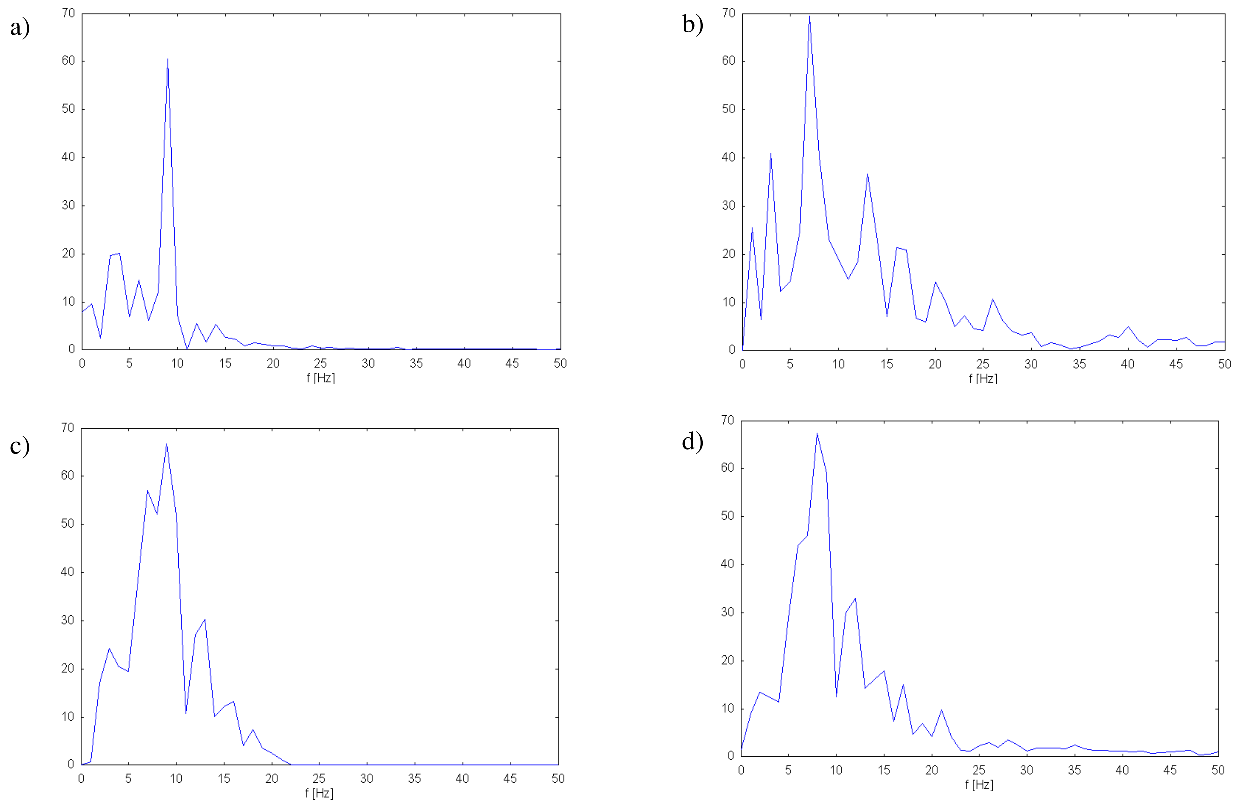
Vaz, F., Oliveira, P.G. and Príncipe, J.C. (1987) "A study on the best order for autoregressive EEG modeling," *International Journal Bio-Medical Computing*, vol. 20, pp. 41-50.

Zetterberg, L.H. (1973) "Experience with analysis and simulation of EEG signals with parametric description of spectra". In: Kellaway, P. and Petersén, I. (eds.). *Automation of Clinical Eletroencephalography*, New York: Ravem Press, pp. 161-201.

Zetterberg, L.H. (1977) "Means and methods for processing of physiological signals with emphasis on EEG analysis". In: Lawrence, J.H. et al. (eds.), *Advances in Biology and Medical Physics, New York: Academic Press, vol. 16, pp. 41-91.*

Zetterberg, L.H., (1978) "Recent advances in EEG data processing", *Contemporary Clinical Neurophysiology*, vol. 34, pp. 19-36.

Zetterberg, L.H. and Ahlin, K. (1975) "Analogue simulator of EEG signals based on spectral components," *Medical and Biological Engineering and Computing*, march, pp. 272-278.



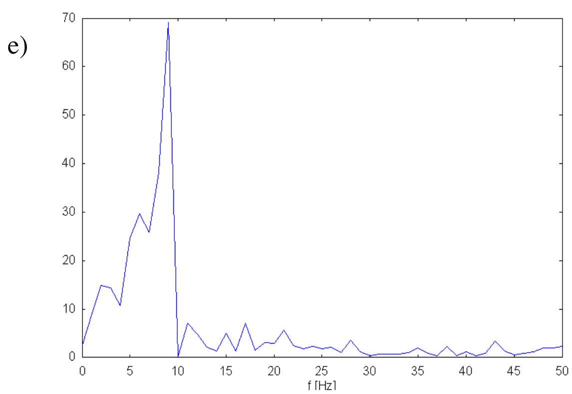


Figure 2: Real and simulated signals in the frequency domain. Horizontal axis represents the spectral component number and vertical axis the amplitude. (a) Spectrum of the real signal. (b) Spectrum of the simulated signal with a 5th-order ARMA filter. (c) Same, with a 6th-order ARMA filter. (d) Same, with a 7th-order ARMA filter. (e) Same, with a 8th-order ARMA filter.

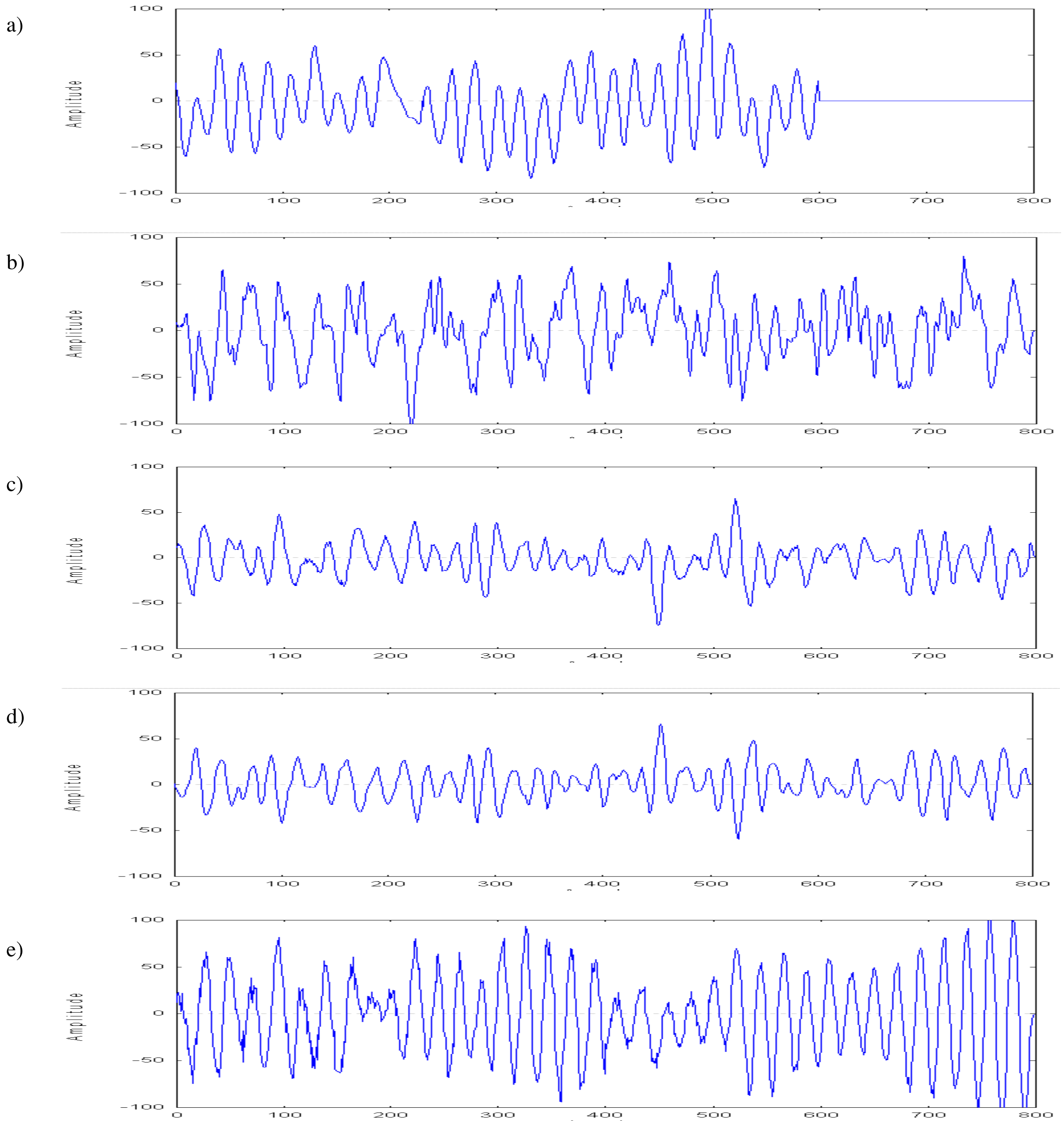


Figure 3: Real and simulated signals in the time domain. Horizontal axis represents the sample number (1 second = 200 samples) and vertical axis the normalized amplitude. (a) Real signal. (b) Simulated signal with a 5th-order ARMA filter. (c) Same, with a 6th-order ARMA filter. (d) Same, with a 7th-order ARMA filter. (e) Same, with a 8th-order ARMA filter.