Regadío, A., Sánchez-Prieto, S., Tabero, J. & González-Castaño, D. M. 2015, "Synthesis of optimal digital shapers with arbitrary noise using a genetic algorithm", Nuclear Instruments and Methods in Physics Research Section A, vol. 795, pp. 115-121

Synthesis of optimal digital shapers with arbitrary noise using a genetic algorithm

Alberto Regadío[a,b,*], Sebastián Sánchez-Prieto[a], Jesús Tabero[b], Diego M. González-Castaño[c]

[a] *Department of Computer Engineering, Space Research Group, Universidad de Alcalá, 28805 Alcalá de Henares, Spain*
[b] *Electronic Technology Area, Instituto Nacional de Técnica Aeroespacial, 28850 Torrejón de Ardoz, Spain*
[c] *Radiation Physics Laboratory, Universidad de Santiago, 15782 Santiago de Compostela, Spain*

Abstract

This paper presents the structure, design and implementation of a novel technique for determining the optimal shaping, in time-domain, for spectrometers by means of a Genetic Algorithm (GA) specifically designed for this purpose. The proposed algorithm is able to adjust automatically the coefficients for shaping an input signal. Results of this experiment have been compared to a previous simulated annealing algorithm. Lastly, its performance and capabilities were tested using simulation data and a real particle detector, as a scintillator.

Keywords: Spectroscopy, Noise, Shaping, Digital Signal Processing, Genetic Algorithm

1. Introduction

In spectroscopy, the value of energy of incident particles can be extracted from the peak amplitude of the input pulses coming from particle detectors. This method is called Pulse Height Analysis (PHA) and provides a value of energy proportional to the incident particle energy. Thus, identical particles with the same energy must generate identical peak values. The ability of a given measurement to resolve fine detail in the incident energy of the radiation is improved as the width of the response function becomes smaller. This feature is called resolution. Nowadays, this property remains determining for all spectroscopy systems [1–4].

The resolution of these measurements is affected by noise. This noise has a spectral density that depends on the type of detector and the features of the spectroscopy system. To mitigate this type of noise, spectroscopy systems have filters at the output of particle detectors called shapers.

The shaper's effectiveness in a spectroscopy system depends on the spectral density of noise. However, finding the optimal shaper is a problem with multiple degrees of freedom. This fact implies that optimal shapers should be selected using numerical and/or iterative procedures (e.g. [3, 5–8]).

*Corresponding Author
Email addresses: regadioca@inta.es (Alberto Regadío), sebastian.sanchez@uah.es (Sebastián Sánchez-Prieto), taberogj@inta.es (Jesús Tabero), diego.gonzalez@usc.es (Diego M. González-Castaño)

This article describes the development of an algorithm based on a GA for providing the optimal shaping for spectroscopy systems. The paper is structured as follows. Section 2 presents the fundamentals of the GA. Section 3 provides details of the GA used and the cost functions. Section 4 presents the theoretical and experimental results of this algorithm. Finally, Section 5 covers the conclusions and the future work.

2. Genetic algorithms

In the computer science field of artificial intelligence, a GA is a heuristic search that tries to imitate the process of natural selection and mutations. This heuristic is used to generate useful solutions to optimization and searching problems [9, 10]. GAs belong to the larger class of evolutionary algorithms, which generate solutions to optimization problems using techniques inspired by the natural evolution, such as inheritance, mutation, selection, and crossover.

In a genetic algorithm, a population of candidate solutions (called individuals or phenotypes) to an optimization problem is evolved toward better solutions. Each candidate solution has a set of properties (its chromosomes or genotype) which can be mutated and altered. Traditionally, solutions are represented as strings of information, usually in binary format [11].

The evolution process usually starts from a population of randomly generated individuals. The population in each iteration is called *generation*. In each generation, the fitness of every individual in the population is evaluated; the fitness is usually the value of the objective function in the optimization problem being solved. The individuals best suited are stochastically selected from the current population, and selected individual's genome is modified (recombined and possibly randomly mutated) to form a new generation. The new generation of candidate solutions is then used in the next iteration of the algorithm. Finally, the searching process terminates when either a maximum number of generations has been produced, or a satisfactory fitness level has been reached for the population.

Interest in such algorithms is intense because some important combinational optimization problems can be solved exactly in a reasonable time.

3. Proposed genetic algorithm

A typical genetic algorithm requires: (a) a cost function to evaluate the candidate solutions, (b) chromosomic representation of the solution domain.

A combinational optimization problem is aimed at finding among many configurations the one which minimizes a given function which is usually referred to as the *cost function*. This function is a measurement of goodness of a particular configuration of parameters. The selection of an appropriate cost function is crucial for achieving good results using this algorithm.

In this work, and in order to reduce the searching space and the processing time, we assume that the chromosomic representation is a monotonically increasing function until it reaches the maximum level, and then it follows a monotonically decreasing function. Thus, for each individual,

$$\mathbf{I} = \left\{x_1, x_2, \cdots, x_{N/2} \ : \ 0 \leq x_1 \leq x_2 \leq \ldots \leq x_{N/2} = 1\right\} \tag{1}$$

where N is the shaper order. From these individuals, a symmetrical shaper can be obtained

$$\mathbf{S} = \left\{\mathbf{I}, \mathbf{I}^R\right\} = \left\{x_1, x_2, \ldots, x_{N/2} = 1, \cdots, x_2, x_1\right\} \tag{2}$$

where $\mathbf{I}^R$ is $\mathbf{I}$ reversed.

For all the considered shapers, the flat-top duration is equal to T_s. As in [8], when flat-tops with a duration of τ_t clock cycles, an additional constraint must be included with a number of ones equal to $L = \tau_t/\tau_s$ added in the middle of $\mathbf{S}$. In this case, the new equation is

$$\mathbf{S} = \left\{\mathbf{I}, 1 \cdots 1, \mathbf{I}^R\right\} = \left\{x_1, x_2, \ldots, x_{N/2-L/2} = 1, \cdots, x_{N/2+L/2} = 1, \cdots, x_2, x_1\right\} \tag{3}$$

The shaper $\mathbf{S}$ works as a digital Finite Impulse Response (FIR) filter. Thus x_n are the coefficients of the FIR filter.

Once both genotype and phenotype are defined, a GA proceeds to initialize a population of shapers, and then to improve it through repetitive application of the mutation, crossover and selection operators according to a cost function. Thus, in order to get an optimal shaper, the following steps are to be taken:

1. Establish the sampling period T_s of the input signal, the maximum shaping time $\tau_{\max}$ and the maximum shaper order $N_{\max}$. The relationship among these parameters is

$$N_{\max} = \frac{\tau_{\max}}{T_s} \tag{4}$$

2. Establish the number of generations G (i.e. iterations), the population P for each generation and the cost function. If mutations are desired, set p_m (probability of mutation) and S_n mutation maximum value.
3. Create a population of P shapers. Each shaper shall have a random integer N where $N \in [1, N_{\max}]$ to try different values of shaping time.
4. For each generation:
   (a) Generate a new population based on the crossover between the set that had got the best score (based on the cost function) in the present population. For this algorithm, the crossover is given by the following equation

$$\mathbf{I}_{\text{new}} = \frac{\varphi\mathbf{I}_1 + (1-\varphi)\mathbf{I}_2}{\max(\varphi\mathbf{I}_1 + (1-\varphi)\mathbf{I}_2)} \tag{5}$$

where $\mathbf{I}_1$, $\mathbf{I}_2$ are two individuals $\mathbf{I}_{new}$ the resulting individual from the crossover and $\varphi \in [0, 1]$ is a real number to set the weight of $\mathbf{I}_1$ and $\mathbf{I}_2$ proportional to the score of both individuals according to the following equation

$$\varphi = \frac{\text{score}(\mathbf{I}_1)}{\text{score}(\mathbf{I}_1) + \text{score}(\mathbf{I}_2)} \tag{6}$$

(b) Include within the population the individual of the past generation that get the best score.

(c) For each value of $\mathbf{I}_{\text{new}}$, add mutations ramdomly with a probability p_m. If a mutation occurs, the new value of $x_n \in \mathbf{I}_{\text{new}}$ is now equal to $\tilde{x}_n$ in this way

$$\tilde{x}_n = x_n + \chi S_n \tag{7}$$

where $\chi \in [-1, 1]$ is a real random number.

(d) Generate a shaper $\mathbf{S}$ for each individual $\mathbf{I}$ (see Eq.(2)) and test it.

(e) Evaluate $\mathbf{S}$ according to a cost function previously selected (see Section 3.1). Assign a score to each shaper based on the evaluation.

5. At the end of the process, the optimal shaper will be the final best shaper.

In specific environments, it can be interesting the execution of this algorithm at a certain intervals. For instance, in space systems, the GA could be executed at regular intervals to counter the effects of radiation damages as was proposed in [12].

3.1. Cost functions

In this work, the cost function used for simulated tests is the Equivalent Noise Charge (ENC), calculated using the noise indices [13], whereas for real test, the cost function is the Signal/Noise Ratio (SNR). In the experimental ones, the Full Width at Half Maximum (FWHM), as a percentage, was used to measure the quality of the final shaper, but it has not been used as cost function due to the enormous burden of calculation and time taken to generate a histogram for each individual in the population.

3.1.1. ENC

To evaluate the results of simulation tests, noise indexes have been used as a cost function. Noise indexes in analog domain were introduced by Goulding in [13]. The noise indexes, calculated in time-domain, are inversely proportional to the SNR, and they can be used to calculate the ENC [14]. This noise analysis is valid for any detector/preamplifier/analog filtering/ADC/PHA combination.

The noise indexes for serial (white) noise N_Δ^2, parallel (red or brownian) noise N_S^2 and $1/f$ series (pink) noise N_F^2 were adapted to the digital domain in [8]:

$$N_S^2 = \frac{1}{S^2} \sum_{n=0}^{\tau_s/T_s} w^2[n]\, T_s \tag{8}$$

$$N_{\Delta}^2 = \frac{1}{S^2} \sum_{n=1}^{\tau_s/T_s} \left(w[n] - w[n-1]\right)^2 \frac{1}{T_s} \tag{9}$$

$$N_F^2 = \frac{1}{S^2} \sum_{n=1}^{\infty} \left(\frac{1}{\sqrt{\pi n T_s}} * \left(w[n] - w[n-1]\right) \right)^2 T_s \tag{10}$$

where τ_s is the shaping time, S is the maximum amplitude of the shaper and $w[t]$ is the weighting function of the shaper. For time-invariant shapers, $w[t]$ is equal to the step response of the system [13] given by the x_n coefficients of Eq. (1).

Furthermore, although the $1/f$ *parallel* noise is nowadays negligible compared to other types of noise above, the index of this noise has also been adapted in this work with the aim of evaluating the GA as discussed in Section 4.1. The $1/f$ parallel noise is proportional to τ_s^2 and equal to

$$N_{FS}^2 = \frac{1}{S^2} \sum_{n=1}^{\infty} \left(\frac{1}{\sqrt{\pi n T_s}} * w[n] \right)^2 T_s^3. \tag{11}$$

These formulae (8, 9, 10, 11) are used to get the four noise indexes and calculate the ENC. Thus,

$$\mathrm{ENC}^2 = i_n^2 N_S^2 + v_n^2 C_i^2 N_{\Delta}^2 + {v_{fn}}^2 C_i^2 N_F^2 + i_{fn}^2 N_{FS}^2 \tag{12}$$

where v_n, i_n, v_{fn} and i_{fn} are the spectral density of white series, white parallel, $1/f$ series noise and $1/f$ parallel noise, respectively. C_i is the equivalent capacitance at the input of the amplifier.

3.1.2. SNR

In a real benchmark experiment, the spectral densities of each noise type are not available unless they are calculated. However, a pulse sample $S[n]$ and a noise sample $N[n]$, that is, the value at the output of the shaper when no events are produced, can be easily captured. Using this pair of samples, SNR can be estimated by the following expression

$$\mathrm{SNR}^2 = \frac{\sum S^2[n]}{\sum N^2[n]} \tag{13}$$

In Section 4.2 the values for $S[n]$ and $N[n]$ are defined.

4. Experiments

To validate the robustness of the proposed algorithm, two different types of experiment have been carried out. The first attempts to reach a known target shaper for applying this algorithm. The second one validates the entire design using real data.

Results of the first group of tests shows that the GA works properly. Results of the second group of tests check that the algorithm also works properly with real data obtained from a scintillator.

4.1. Simulation experiments

The aim of the first experiment is to get the optimal shaper for different noise types, also obtained in [15] to check that the algorithm works properly.

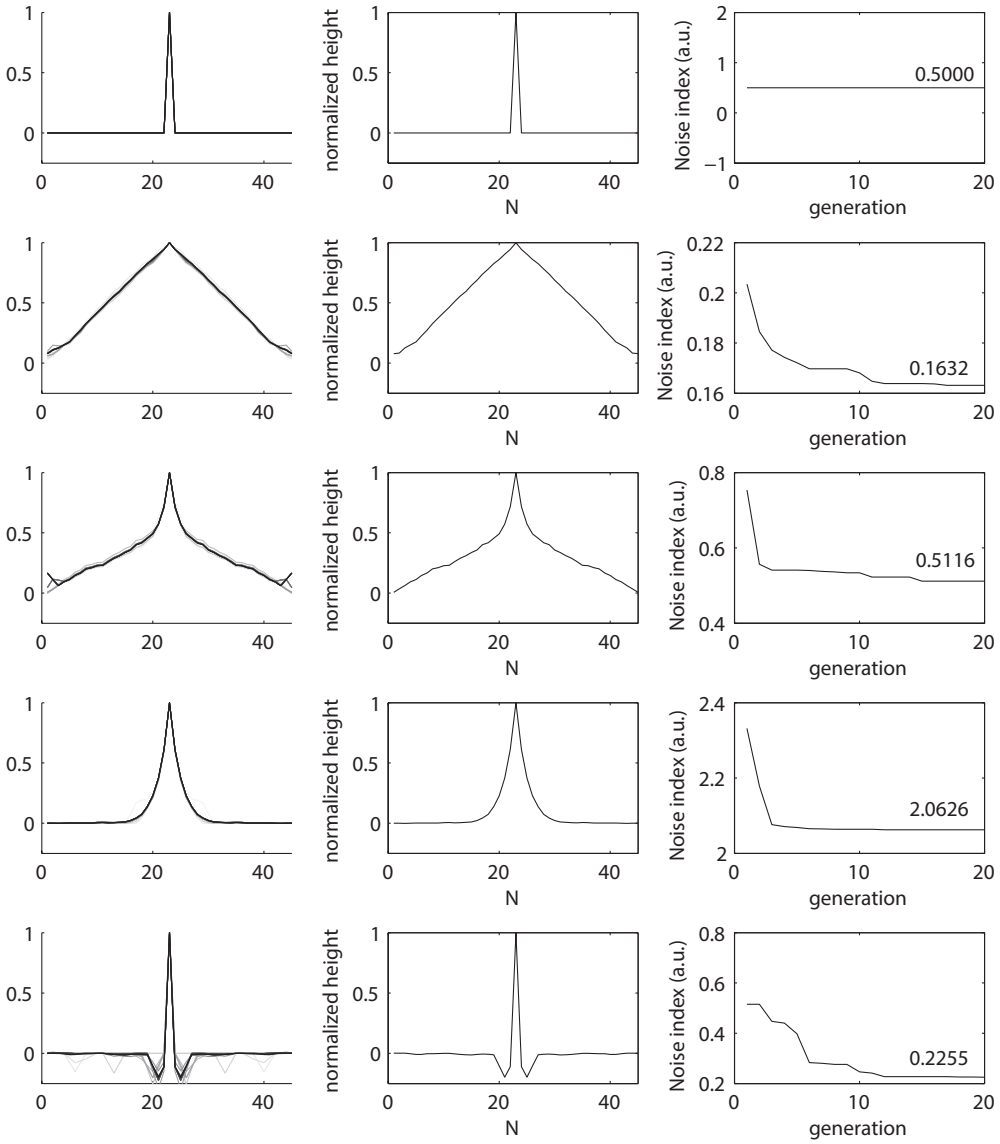

Figure 1: Algorithm results for (a) $v_n > 0$, others= 0. (b) $i_n > 0$, others= 0. (c) $v_{fn} > 0$, others= 0. (d) $v_n = i_n > 0$, others= 0. (e) $i_{fn} > 0$, others= 0. In all of them $T_s = 0.5$ s.

Fig. 1 shows the result of the application of the algorithm. The left column shows the resulting shaper for each generation (lighter lines imply low G). The central column shows the final shaper. The right column represents the evolution of the function cost (in this case, Eq. (12)). As a result of this test, it can be observed the optimal shapers for each type of noise: (a) series noise, (b) parallel noise, (c) equal influence of series and parallel noise (cusp-like shaping) [16], (d) shaper for $1/f$ series noise [15], (e) shaper for $1/f$ parallel noise [17].

The test was carried out with $G = 20, P = 300, p_m = 0.2$ and $S_m = 0.2$. In all cases, it can be observed that noise indexes are decreased as G increases. A special case is the shaper for $1/f$ parallel noise. According to Eq. (1), x_n cannot be higher than x_{n+1}. Moreover, x_n cannot be lower than 0. However, the effect of mutations forces the individuals to ignore Eq. (1) in order to find the optimal shaper. If there were no mutation (i.e. $p_m = 0$), the optimal shaper for (e) would not have been found.

The execution time of the algorithm is directly proportional to $N \cdot G \cdot P$. The addition of mutations implies an increase of 12% in the total time. The execution time in a Intel Core i7 at 2.2 GHz has been 0.37 seconds for each shaper. Thus, the execution time is negligible compared to the time needed to capture pulses and to generate a histogram, even with a much slower processor.

In the second test, the result of the GA for several values of G and P have been compared. In Fig. 2 can be seen that the algorithm needs a different value of P and G to get noise indexes close to those of optimal noise shapers depending on the type of noise present. Moreover, above a certain value of P and G the algorithm provides acceptable results in all cases. In the left column of Fig. 2, it can be observed the effect of including mutations, that in all cases provides better results. However, an increase in the value of S_m above 0.2 makes the solutions of the shapers oscillate too much making them unfeasible.

4.2. Experimental results using a scintillator

Lastly, a group of tests to check the proposed GA in a real environment was performed. The main objective of these tests is to check that the GA works and try to improve the results obtained with a fixed shaper.

This test was performed in the Radiation Physics Laboratory located in Santiago de Compostela University, Spain using a scintillator. A diagram of the detection chain used in the experimental test is shown in Fig. 3. The scintillator model of NaI is Bricon 1M1/1.5 working at +475 V, with an integrated preamplifier Bricon PA-12. The amplifier N968 (with a shaping of 2 μs and gain $\times$14 was connected to a Digital Phosphor Oscilloscope Tektronix TDS 3014B. An amount of 500 points were taken for each pulse (each pulse duration was 0.4 μs). The resolution of the vertical scale was 128 bits for 5 V. This oscilloscope performs the function of DAQ, receiving the raw data from the amplifier and storing it in a PC. The scintillator receives radiation sources of ^{137}Cs, ^{22}Na or ^{60}Co whose features are listed in Table 1.

Raw data reuse, stored in the PC, allows using the same data multiple times without recapturing new

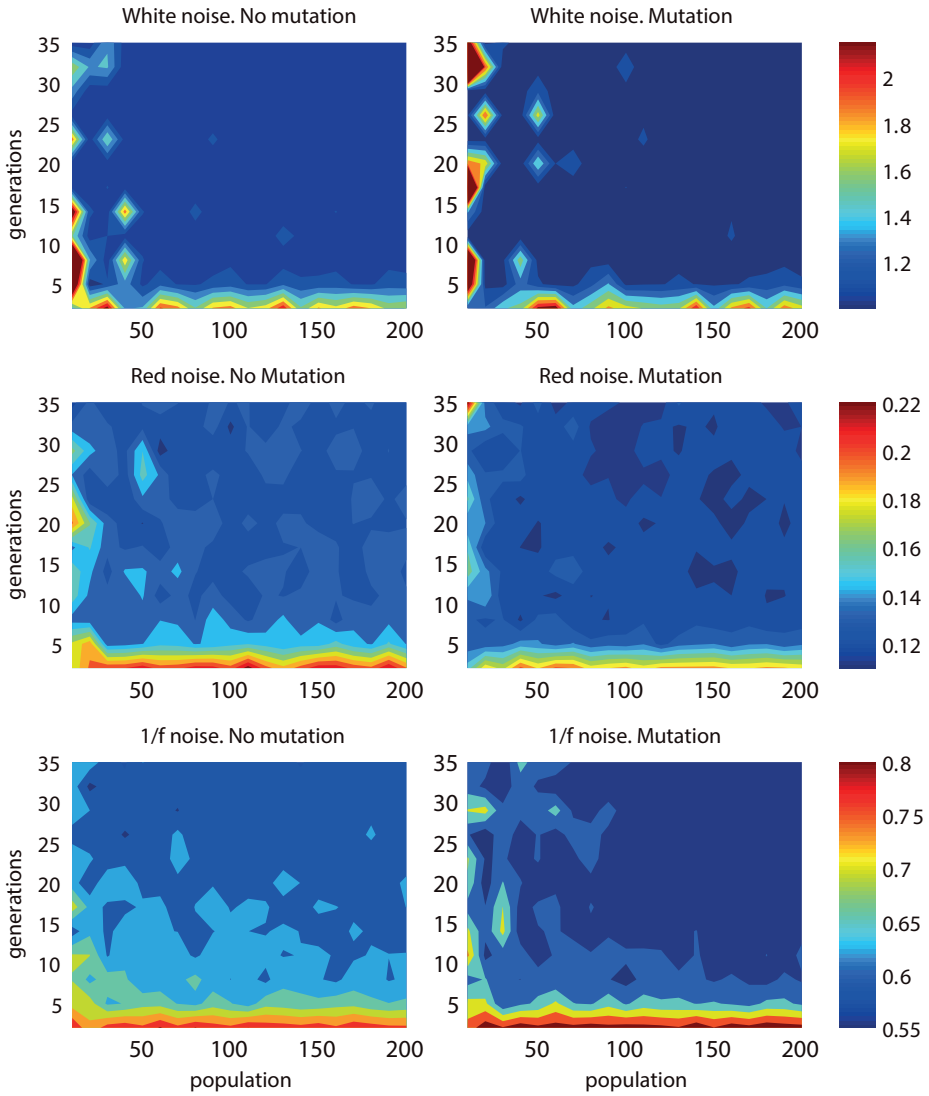

Figure 2: Effect of P, G and mutations for several noise types.

Table 1: Radiation sources used for the experimental tests.

Isotope	Activity (kBq)	Main energies (keV)
^{22}Na	105	511; 1274
^{137}Cs	8.71	32; 661.6
^{60}Co	28.5	1173.2; 1332.5

data and it ensures that changes in the results obtained during the test are exclusively due to the digital signal processing.

Using Matlab code, the raw data are filtered using a digital shaper generated by the GA. Finally, the

160 height of each pulse of the filtered raw data are extracted to generate a histogram to compare results.

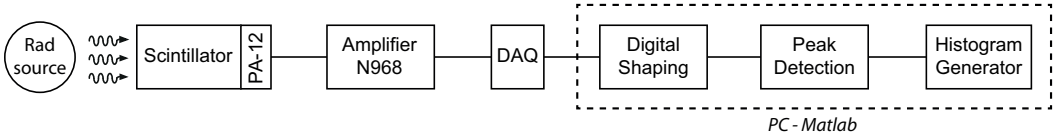

Figure 3: Diagram of the detection chain used for the experimental test.

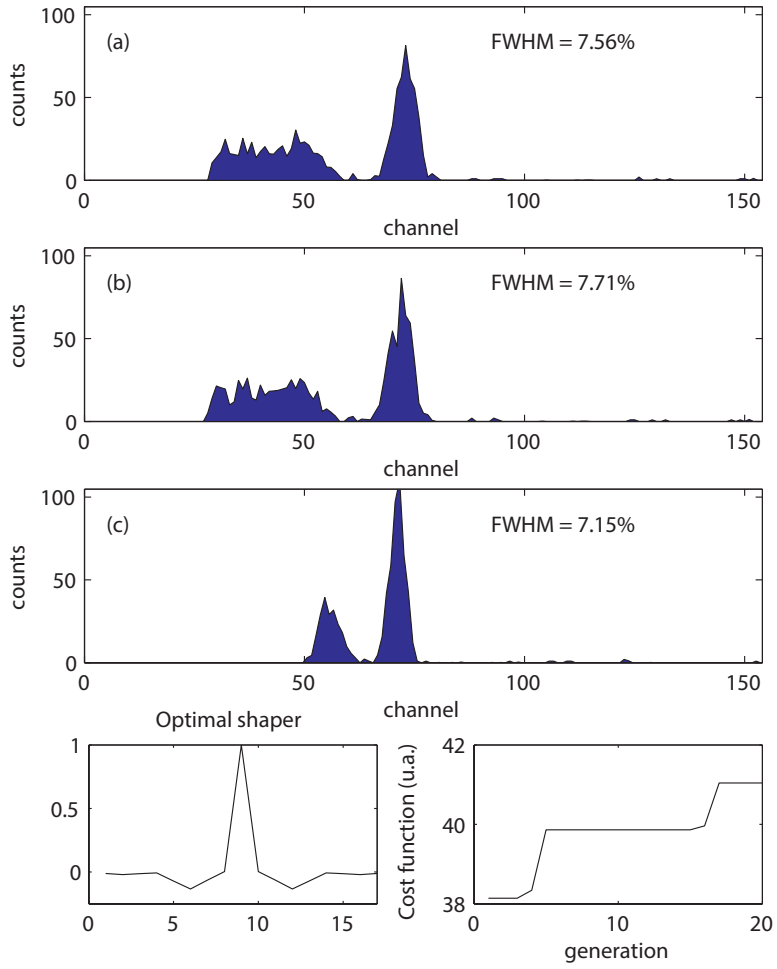

Figure 4: Histograms and result of GA for ^{137}Cs.

161 For these tests, $P = 30$, $G = 20$ and $N = 17$. The raw data length were 1121 kSamples for ^{137}Cs,
162 2973 kSamples for ^{22}Na and 1812 kSamples for ^{60}Co. Besides, a signal with a length of 603 kSamples was
163 captured when no radiation sample was in front of the scintillator to measure the environmental noise.

164 From these signals, the height of each pulse was extracted using the Matlab software. The sum of the
165 heights of the pulses for each radiation source was $S[n]$ whereas the sum of the signals height when no
166 radiation source was present was $N[n]$. Both $S[n]$ and $N[n]$ allows to calculate the cost function presented

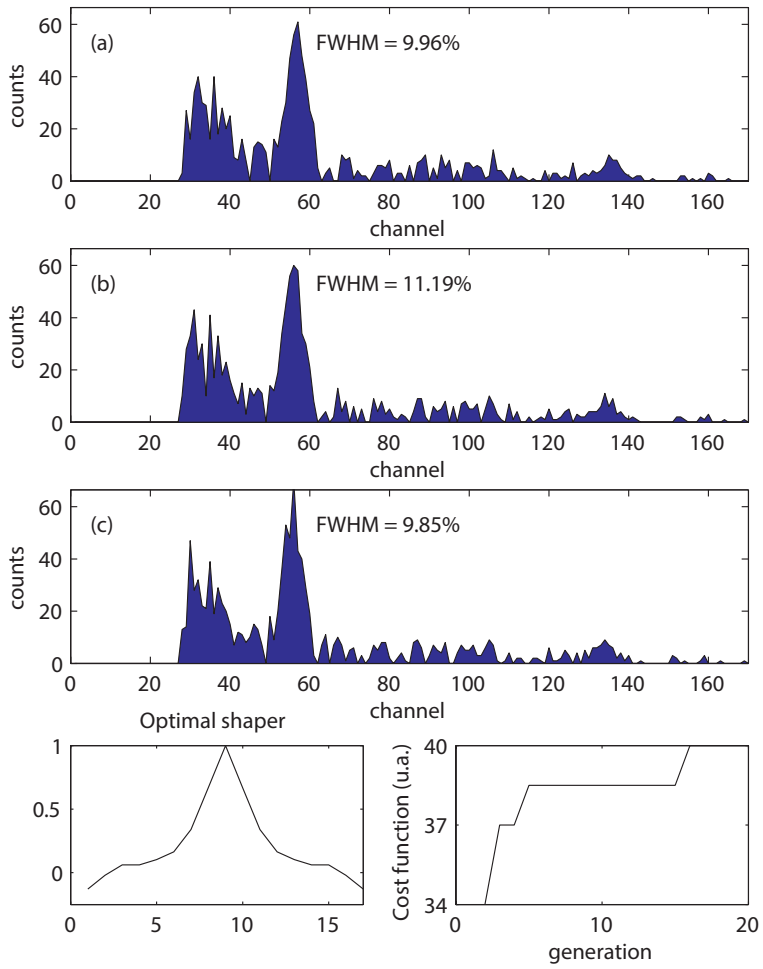

Figure 5: Histograms and result of GA for ^{22}Na.

in Section 3.1.2.

In these experiments, the resolution using the Full Width at Half Maximum (FWHM) was calculated to compare the data captured (a) without shaping, (b) with a fixed triangular shaper and (c) with the shaper obtained using the GA proposed.

The results of the experiment are shown in Fig. 4, 5 and 6.

In all these figures, (a) histogram is generated without shaping, (b) is the one generated with $N = 20$ triangular shaping, and (c) is the histogram generated with the optimal shaper obtained using the proposed GA. Finally, at the bottom of each figure, the optimal shaper and the evolution of the function cost are depicted.

The FWHM expressed as a percentage is defined as the width of the distribution at a level that is just half the maximum ordinate of the peak divided by the location of the peak maximum. For all the histograms, the width at a half the maximum ordinate of the peak is depicted in grey numbers whereas the peak maximum

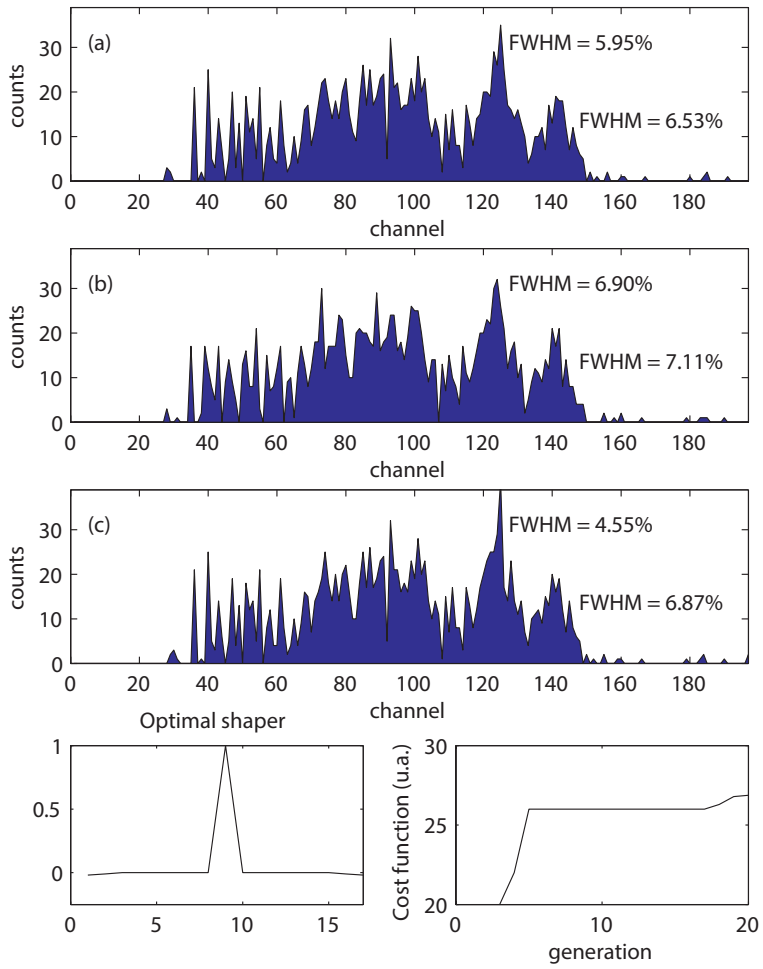

Figure 6: Histograms and result of GA for ^{60}Co.

is depicted in bold grey numbers. In all the histograms is also depicted the FWHM.

As it can be observed, the FWHM improves when shapers are used. In addition, the improvement is even greater when the GA is used. In the case of ^{137}Cs, the peak at 32 keV could not be captured because the present noise at that spectrum area. In the first generations, the SNR decreases but then is increased. This behavior is produced when the optimal result is difficult to get as a consequence of the solution space. However, it is normal because the latter generation may contain worse chromosomes than the previous one.

Once the optimal shaper is found, this shaper is linear and time-invariant because, as stated in Section 3, it works as a FIR filter. Thus, the maximum event rate of this shaper depends on the shaping time and on the pile-up management selected in the same way than other non-adaptive, linear, time-invariant shapers.

4.3. Comparison with simulated annealing

The purpose of this section is to compare the GA proposed in this paper with the simulated annealing algorithm proposed in [8].

In both algorithms the number of software instructions performed are directly proportional to the population P and the number of iterations (generations in the case of GA and temperature in the case of annealing algorithm). The number of operations performed during each phase of the algorithm is each phase of the algorithm is similar, and therefore, the computing time is also similar. However, both algorithms have advantages and disadvantages when compared.

Thanks to the inclusion of mutations, an advantage that the GA presented is that equation of individuals (1) can stop being effective and thus the value of x_n of shapers can decrease before reaching its maximum value and even reach values below zero. This implies an increase of the search space that can be useful for better noise mitigation. In fact, the shaping obtained in Fig. 1(e) for $1/f$ parallel noise is impossible to get with annealing algorithm of [8].

However, an advantage of annealing algorithm is that only requires memory to store the last generated shaper and the current optimal shaper. In contrast, to allow the GA make the crossings, is necessary to store in memory two complete generations: the original and the new one. Thus, the amount of memory used is equal to $2 \cdot$ size of the number format used $\cdot\ N \cdot P$. Thus, for instance, using $P = 400$, and considering the size of the format number equal to 4 bytes, the amount of memory used is equal to 124800 bytes.

5. Conclusions and future work

In this study, an algorithm which uses GA for calculating optimal filters in presence of arbitrary noise type was designed and implemented. In order to test the efficiency of this algorithm, simulation examples were evaluated and one setup was measured in real radiation facilities. Additional constraints such as shaping time or even the peak time can be added modifying the parameters of this algorithm. It can be concluded that this algorithm is a promising method to be taking into account in successive digital spectroscopy systems due to its efficiency and simplicity.

Acknowledgments

This project was funded by the Spanish Administration as part of projects ref. AYA2011-29727-C02-02 and AYA2012-39810-C02-02.

Bibliography

[1] C Fiorini, et al., "Silicon Drift Detectors for Readout of Scintillators in Gamma-Ray Spectroscopy". *IEEE Transactions on Nuclear Science*, vol. 60, no. 4, (2013) 2923–2933.

[2] N. Casali, S. S. Nagorny, F. Orio, L. Pattavina *et al.* "Discovery of the ^{151}Eu α decay". Journal of Physics G: Nuclear and Particle Physics 41: 075101.

[3] X. Wen, H. Yang, *Nuclear Instruments and Methods in Physics Research A* (2014), http://dx.doi.org/10.1016/j.nima.2014.11.008

[4] U. Ackermann, et al., *Nuclear Instruments and Methods in Physics Research A* (2015), http://dx.doi.org/10.1016/j.nima.2015.03.016

[5] E. Gatti, A. Geraci, G. Ripamonti, "Automatic synthesis of optimum filters with arbitrary constraints and noises: a new method". *Nuclear Instruments and Methods in Physics Research A* 381 (1996) 117–127.

[6] E. Gatti, A. Geraci, S. Riboldi, G. Ripamonti, "Digital Penalized LMS method for filter synthesis with arbitrary constraints and noise". *Nuclear Instruments and Methods in Physics Research A* 523 (2004) 167–185.

[7] N. Menaa, P. D'Agostino, B. Zakrzewski, V. T. Jordanov. "Evaluation of real-time digital pulse shapers with various HPGe and silicon radiation detectors". *Nuclear Instruments and Methods in Physics Research A* vol. 652, (2011) 512–515.

[8] A. Regadío, S. Sánchez-Prieto, J. Tabero, "Synthesis of optimal digital shapers with arbitrary noise using simulated annealing", *Nuclear Instruments and Methods in Physics Research A* vol. 738, (2014) 74–81.

[9] M. Mitchell. *An Introduction to Genetic Algorithms*. Cambridge, MA: MIT Press. 1996. ISBN 0262133164.

[10] D. Simon. *Evolutionary Optimization Algorithms*. John Wiley & Sons, 2013.

[11] D. Whitley, "A genetic algorithm tutorial". Statistics and Computing vol. 4, issue 2 (1994) 6585.

[12] J. Lanchares, O. Garnica, J. L. Risco-Martín, J. I. Hidalgo, A. Regadío, "Real-Time Evolvable Pulse Shaper for Radiation Measurements". *Nuclear Instruments and Methods in Physics Research A* 735 (2014) 297–303.

[13] F. S. Goulding, "Pulse-Shaping in Low-Noise Nuclear Amplifiers: A Physical Approach to Noise Analysis", *Nuclear Instruments and Methods* 100, (1972) 493–504.

[14] K. A. Olive et al. (Particle Data Group), Chinese Physics C38, 090001 (2014). p.433.

[15] V. T. Jordanov, "Real time digital pulse shaper with variable weighting function", *Nuclear Instruments and Methods in Physics Research A* vol. 505, (2003) 347–351.

[16] P. W. Nicholson, *Nuclear Electronics*. John Wiley & Sons, Ltd., 1973.

[17] E. Gatti, A. Geraci, G. Ripamonti, "Optimum filter for $1/f$ current noise smoothed-to-white at low frequency (Letter to the Editor)", *Nuclear Instruments and Methods in Physics Research A* vol. 394, (1997) 268–270.