A Fuzzy Gravitational Search Algorithm to Design Optimal IIR Filters

Authors:

Danilo Pelusi, Raffaele Mascella, Luca Tallini

Date Submitted: 2020-06-23

Keywords: fuzzy systems, gravitational search algorithm, IIR filters, optimization algorithms

Abstract:

The goodness of Infinite Impulse Response (IIR) digital filters design depends on pass band ripple, stop band ripple and transition band values. The main problem is defining a suitable error fitness function that depends on these parameters. This fitness function can be optimized by search algorithms such as evolutionary algorithms. This paper proposes an intelligent algorithm for the design of optimal 8th order IIR filters. The main contribution is the design of Fuzzy Inference Systems able to tune key parameters of a revisited version of the Gravitational Search Algorithm (GSA). In this way, a Fuzzy Gravitational Search Algorithm (FGSA) is designed. The optimization performances of FGSA are compared with those of Differential Evolution (DE) and GSA. The results show that FGSA is the algorithm that gives the best compromise between goodness, robustness and convergence rate for the design of 8th order IIR filters. Moreover, FGSA assures a good stability of the designed filters.

Record Type: Published Article

Submitted To: LAPSE (Living Archive for Process Systems Engineering)

Citation (overall record, always the latest version):	LAPSE:2020.0695
Citation (this specific file, latest version):	LAPSE:2020.0695-1
Citation (this specific file, this version):	LAPSE:2020.0695-1v1

DOI of Published Version: https://doi.org/10.3390/en11040736

License: Creative Commons Attribution 4.0 International (CC BY 4.0)

energies

Article

A Fuzzy Gravitational Search Algorithm to Design Optimal IIR Filters

Danilo Pelusi *, Raffaele Mascella and Luca Tallini

Faculty of Communication Sciences, University of Teramo, 64100 Teramo, Italy; rmascella@unite.it (R.M.); ltallini@unite.it (L.T.)

* Correspondence: dpelusi@unite.it; Tel.: +39-086-126-6036

Received: 24 February 2018; Accepted: 18 March 2018; Published: 23 March 2018

Abstract: The goodness of Infinite Impulse Response (IIR) digital filters design depends on pass band ripple, stop band ripple and transition band values. The main problem is defining a suitable error fitness function that depends on these parameters. This fitness function can be optimized by search algorithms such as evolutionary algorithms. This paper proposes an intelligent algorithm for the design of optimal 8th order IIR filters. The main contribution is the design of Fuzzy Inference Systems able to tune key parameters of a revisited version of the Gravitational Search Algorithm (GSA). In this way, a Fuzzy Gravitational Search Algorithm (FGSA) is designed. The optimization performances of FGSA are compared with those of Differential Evolution (DE) and GSA. The results show that FGSA is the algorithm that gives the best compromise between goodness, robustness and convergence rate for the design of 8th order IIR filters. Moreover, FGSA assures a good stability of the designed filters.

Keywords: optimization algorithms; IIR filters; gravitational search algorithm; fuzzy systems

1. Introduction

The design of optimal Infinite Impulse Response (IIR) digital filters is a very interesting challenge. The main techniques to design IIR filters are traditional design technique and optimization techniques. The first method is commonly known as bilinear transformation approach [1]. The second approach regards the applications of optimization techniques to design optimal filters. Among linear optimization algorithms, the steepest-descent and quasi-Newton (QN) algorithms are used for IIR filters design [1,2]. QN algorithms offer the advantages of robustness and fast convergence. Moreover, because QN optimization approach is very flexible, it can be used to design filters with arbitrary amplitude and/or phase responses. QN algorithms have also been used to design linear-phase IIR filters [3]. Chen et al. [4] proposed a technique for IIR filters design based on the minimization the error between the order-reduced filter's response and the desired one in the Hankel-norm sense. In the optimization algorithm proposed by Lu and Hinamoto [5] for the design of optimal IIR filters, the coefficients of all sub-filters are jointly optimized through a sequence of linear updates with each update carried out using second-order cone programming.

IIR filters are used in a wide range of applications where a high-selectivity processing of discrete signals is needed [6]. Lai and Lin [7] imposed two elliptic constraints on the frequency response of an IIR filter: the first one to minimize the maximum phase error, whereas the second one to constrain the maximum magnitude error. Another constrained optimization was introduced by Nongpiur et al. [8] for the design of IIR Digital Differentiators. The method in [8] minimizes the group-delay deviation under the constraint that the maximum amplitude-response error must be below a fixed level. Constraints on the magnitude and phase responses for the design of nearly linear-phase IIR filters was the main contribution of [9]. Lang [10] presented a method with the possibility to specify a maximum radius for the poles of the designed rational transfer function. A computationally low intensive method for designing IIR multi-Notch filters was proposed by Duarte et al. [11]. The design of IIR filters may be oriented on magnitude and delay together: by combining the root-mean-square error function of variable frequency response and a suitable stability constrained function, the stability problem is overcome [12].

Generally, the problem of designing IIR filters is formulated as a nonlinear optimization problem. Moreover, the traditional methods based on gradient search can easily be stuck at local minima of error surface. In order to solve this problem, some methods based on metaheuristic approaches have been proposed. Due to their fast convergence property, Differential Evolution (DE) algorithms [13] have been applied to design robustly stable IIR filters [14–16]. Karaboga [17] proposed a technique to design IIR filters through DE. A seeker-optimization-algorithm based on evolutionary methods has been proposed for digital IIR filter design [18]. Other evolutionary algorithms such as Particle Swarm Optimization (PSO) [19] have been used for the design of IIR filters to reconstruct missing segments of multidimensional data [20]. A multi-swarm PSO with particle reallocation strategy is applied to design IIR filters with null constraint and specified error in the stop band [21]. Wang and Chen [22] proposed the use of multi-objective optimization evolutionary algorithms with the aim of minimizing magnitude response error, phase response error and order of IIR filters. An improved Immune Algorithm (IA) was proposed by Tsai and Chou to solve the problem of designing optimal IIR filters [23].

The process of IIR filters' design optimization is difficult because some constraints should be satisfied: (i) the determination of the lowest filter order; (ii) the filter stability; and (iii) the minimum value of passband and stopband ripple magnitudes. Because the Genetic Algorithms (GA) [24] are able to optimize complex and discontinuous functions that are difficult to analyze mathematically, some research [25–29] proposed different methods based on GA to solve the digital IIR filter design problems. A multi-crossover approach to design optimal GA-aided IIR filters was proposed by Chang [27]. Robust D-Stable IIR filters was designed by using GA where the stability criterion is embedded in the evolution of each generation [29]. Yu and Xinjie [28] proposed a coevolutionary GA that evolves coordinately as two different species: the control species and the coefficient species. A multi-parameter and multi-criterion optimization method based on a quantum genetic algorithm was proposed by Zhang et al. [25]. Stable IIR filters have been designed with the application of GA [26].

IIR filters' designing problems can be formulated as a multi-modal optimization problem with multiple decision variables. The Gravitational Search Algorithm (GSA) is a search method based on a law of gravity [30] able to optimize multi-modal functions. Saha et al. [31,32] proposed a simple GSA and a GSA with Wavelet Mutation for the optimization of 8-th order IIR filter design. On the other hand, GSA has been combined with fuzzy logic for various applications [33–36]. A fuzzy logic-based adaptive gravitational search algorithm dedicated to the optimal tuning of fuzzy controllers for servo systems was proposed by Precup et al. [33]. GSA and fuzzy logic have been combined to design optimal Proportional Integral (PI) controllers for a class of servo systems characterized by saturation and dead zone static nonlinearities [34]. The idea of enhancing GSA using fuzzy logic is inspired from the exploration and exploitation principle in meta-heuristics. The fuzzy regulation of GSA parameters assures this principle. Fuzzy Gravitational Search Algorithms (FGSA) with dynamic alpha parameter value adaptation for the optimization of modular neural networks in echocardiogram and pattern recognition have been proposed [35,36]. Moreover, other versions of GSA with a fuzzy dynamic parameters adaptation have been proposed [37–43]. The improvements of GSA are based on the dynamic regulation of suitable parameters during the search procedure.

This paper aims to design optimal IIR filters with the help of a revised GSA and the design of suitable Fuzzy Inference Systems (FIS). The first contribution of the work is the re-definition of a parameter of GSA able to improve the search performances. The second one is the design of two FIS's for GSA parameters adjustment. Both the approaches give rise to a Fuzzy Gravitational Search

Algorithm (FGSA) with dynamic parameter adaptation. This algorithm is applied to design 8th order IIR filters.

The paper is organized as follows. Section 2 contains the description of IIR filters design. The designed algorithm is presented in Section 3. Section 4 illustrates the achieved results. Section 5 contains the paper conclusions.

2. IIR Filter Design

The relation between inputs and outputs of IIR filters is given by Equation (1) [44]:

$$y(p)+\sum_{k=1}^{n}a_k y(p-k)=\sum_{k=0}^{m}b_k x(p-k), \tag{1}$$

where $x(p)$ is the filter input, whereas $y(p)$ is the output. The order of filters is defined by n with $n \geq m$. By assuming that $a_0 = 0$, the transfer function of IIR filter can be expressed as in Equation (2):

$$H(z)=\frac{\sum_{k=0}^{m}b_k z^{-k}}{1+\sum_{k=1}^{n}a_k z^{-k}}. \tag{2}$$

Assuming that $z = e^{j\omega}$, it follows that IIR filter frequency response becomes as in Equation (3)

$$H(\omega)=\frac{\sum_{k=0}^{m}b_k e^{-jk\omega}}{1+\sum_{k=1}^{n}a_k e^{-jk\omega}}, \tag{3}$$

where $\omega \in [0, \pi]$ is the digital frequency.

Generally, the used approach to design IIR filters is to consider a Mean Square Error optimization problem [45–47]. MSE fitness function can be expressed as in Equation (4)

$$J_1(\omega)=\frac{1}{N_s}[(d(p)-y(p))^2], \tag{4}$$

where N_s is the number of frequency points used for the computation of the error fitness function; $d(p)$ and $y(p)$ are the filter's desired and actual responses, respectively. The actual response $y(p)$ is calculated through Equation (1), whereas the values of $d(s)$ are set to be very close to ideal filters' values. The difference between $d(p)$ and $y(p)$ is the error between the desired and the actual filter responses. The design goal is to minimize the MSE $J1(\omega)$ with proper adjustment of filter coefficients $b_0, ..., b_m, a_0, ..., a_n$.

IIR filters' optimization problem depends on the choice of transfer function coefficients $b_0, b_1, ..., b_m$ and $a_0, a_1, ..., a_n$. Because the quality of a IIR filter depends on pass band ripple, stop band ripple and transition band, we propose a new fitness function that takes into account these three parameters. In particular, a good IIR filter has small pass and stop band ripple, and narrow transition band. In order to assure such constraints, the fitness function $J_2(\omega)$ in Equation (5) is defined. In Equation (5), n_s is the number of samples, δ_p is the pass band ripple, δ_s is the stop band ripple and $|H(\omega)|$ is the absolute value of $H(\omega)$ in Equation (3):

$$J_2(\omega)=\sum_{i=1}^{n_s}(abs(|H(\omega_i)|-1)-\delta_p)^2+\sum_{l=1}^{n_s}(|H(\omega_l)|-\delta_s)^2. \tag{5}$$

In particular, the absolute value of $H(\omega)$ is calculated for each $i = 1, ..., n_s$ with (6)

$$|H(\omega_i)|=\left|\frac{\sum_{k=0}^{m}b_k e^{-jk\omega_i}}{1+\sum_{k=1}^{n}a_k e^{-jk\omega_i}}\right|, \tag{6}$$

where the values of ω_i are spaced frequency points between 0 and the pass band normalized edge frequency ω_p.

Similarly, for each $l = 1, ..., n_s$

$$|H(\omega_l)| = \left| \frac{\sum_{k=0}^{m} b_k e^{-jk\omega_l}}{1 + \sum_{k=1}^{n} a_k e^{-jk\omega_l}} \right|, \quad (7)$$

where the values of ω_l are spaced frequency points between 0 and the stop band normalized edge frequency ω_s. Note that, when $i = n_s$, it follows that $\omega_i = \omega_{n_s} = \omega_p$. In the same way, the stop band normalized edge frequency ω_s is achieved when $l = n_s$; thus, $\omega_l = \omega_{n_s} = \omega_s$.

In order to design optimal IIR filters, a constrained minimization of the error fitness function defined in Equation (5) is needed. On the other hand, the stability is an important issue for IIR digital filters design [10,48,49]. Jiang and Kwan [50] proposed a stability constraint with a prescribed pole radius derived from the argument principle of complex analysis. The optimization of the proposed fitness function defined in Equation (5) follows the stability constraints in [50].

3. The Fuzzy Gravitational Search Algorithm

The proposed error fitness function in Equation (5) has to be minimized through an optimization algorithm. The design of our algorithm starts with a suitable definition of GSA parameter, which supplies the number of agents that apply the force to other individuals [30]. Such GSA parameter is referred as *Kbest* and it decreases linearly to 1 over the increment of iterations. The idea is to increase the convergence speed by defining *Kbest* as in Equation (8)

$$Kbest(i) = \left\lfloor n_a \exp\left(-\beta \frac{i}{n_a}\right) \right\rfloor, \quad (8)$$

where i is the i-th iteration, n_a is the number of agents and β a parameter.

A key parameter in GSA is the gravitational constant $G(t)$ [30], which depends on the initial value G_0, the number of iterations N and the value of parameter α (see Equation (9)):

$$G(i) = G_0 \exp(-\alpha \frac{i}{N}). \quad (9)$$

The next step is to design two Fuzzy Inference Systems (FIS) able to adjust β and α parameter in Equation (8) and Equation (9), respectively. The tuning of these parameters must assure a good trade-off between exploration and exploitation of the search process. For this aim, we define a quantity $P_p \in [0, 1]$, which gives a measure of the population progress (see Equation (10)),

$$P_p(i) = \left| \frac{\overline{J_2(\omega)}_{(i-1)} - \overline{J_2(\omega)}_{(i)}}{max(\overline{J_2(\omega)}_{(i-1)}, \overline{J_2(\omega)}_{(i)})} \right|, \quad (10)$$

where $\overline{J_2(\omega)}_{(i)}$ is the error fitness mean value calculated on the n_a agents. This computation is accomplished for each iteration $i = 1, ..., N$, where N represents the iterations number.

Generally, the accuracy of an FIS depends on the number of membership functions (MF): a higher membership functions number tends to cause an increase in FIS. Moreover, the number of MFs has a huge impact on the system complexity; therefore, an optimal trade-off between accuracy and complexity is needed. Thus, the fuzzy inputs definition for the first FIS depends on this issue. We define the iterations number N and the population progress P_p as fuzzy inputs with membership functions Low (L), Medium (M) and High (H) (see Figures 1 and 2). In order to achieve more refined values of α, nine membership functions for the fuzzy output are defined (see Figure 3). The choice of triangular/trapezoidal MFs depends on the performance of FIS for a generic system [51].

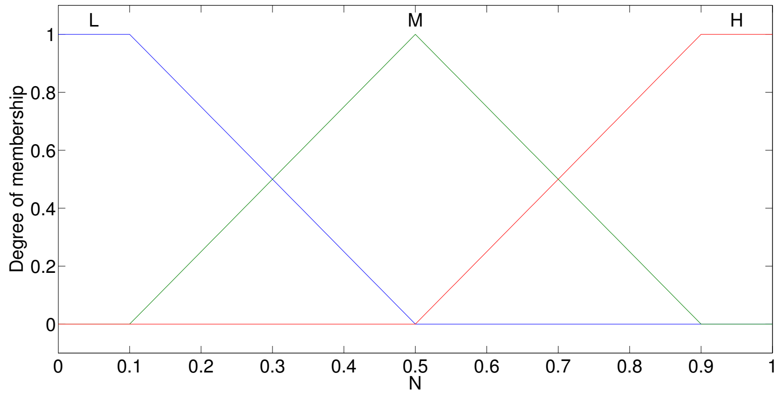

Figure 1. Membership functions of the fuzzy input N.

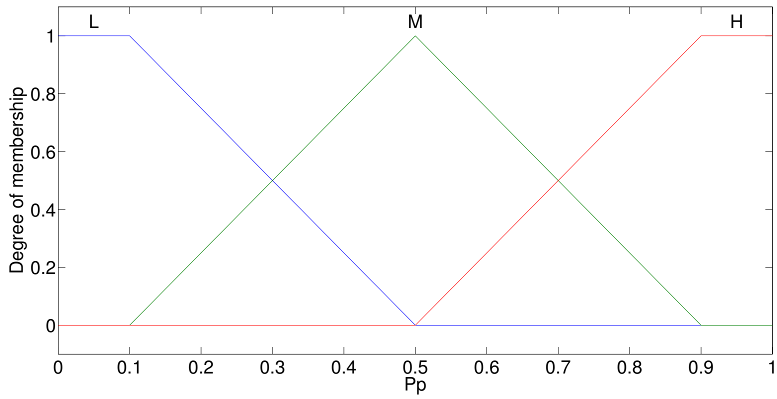

Figure 2. Membership functions of the fuzzy input P_p.

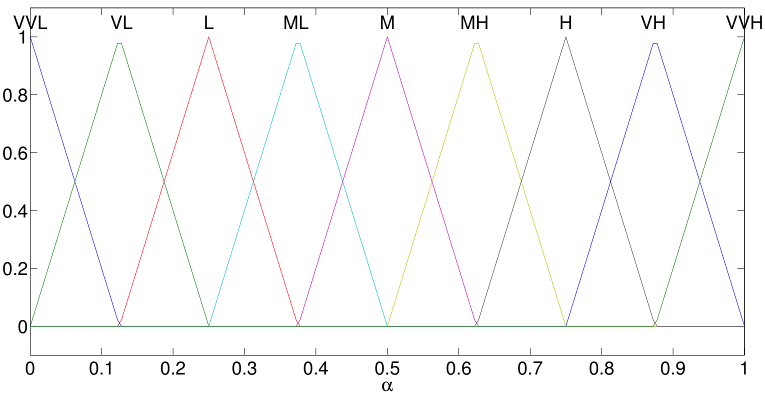

Figure 3. Membership functions of the fuzzy output α.

The definition of fuzzy rules for FIS-α depends on GSA behavior. To assure exploration at the beginning of iterations, the agents must have a huge acceleration: this condition is achieved with low values of α (see Equation (9)). A lack of improvement means premature convergence: supplying a lower value of α, the agents escape from local optima. Low values of α are required when GSA lies in the middle of the procedure and there are no improvements. Very very high values of α are necessary with improvements at the end of iterations. Table 1 shows FIS-α rules. The architecture of FIS-α is shown in Figure 4.

The main step in the design of an FIS is the definition of fuzzy rules. Table 1 shows the fuzzy rules for FIS-α. The rules are based on the behavior of GSA. At the beginning of iterations, i.e., when $N = L$, to assure exploration, the agents must have a big acceleration; therefore, a low value of α is necessary. In fact, if α is low, then G increases (see Equation (9)), and then the acceleration tends to increase [30]. On the other hand, an early lack of improvement, that is $P_p = H$ and $N = L$, is a sign of premature convergence: with a very very low value of α, the individuals can escape from local optima. When GSA is at the middle of the procedure ($N = M$) and there is lack of improvement ($P_p = H$), the values of α must be basically low. At the end of the iterations ($N = H$), if there is an improvement in the optima research ($P_p = L$), then α must be very very high. High values of α tend to decrease the value of the gravitational constant and therefore the acceleration. Figure 4 shows the architecture of FIS-α to adjust the parameter α.

Table 1. Fuzzy rules for FIS(Fuzzy Inference System)-α.

N	P_p	α
L	L	L
L	M	VL
L	H	VVL
M	L	MH
M	M	M
M	H	ML
H	L	VVH
H	M	VH
H	H	H

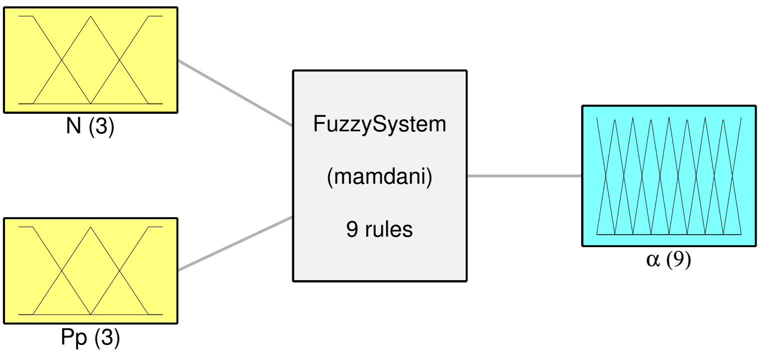

Figure 4. Architecture of Fuzzy Inference System (FIS) to adjust α.

The second FIS (referred as FIS-β) has the task of adjusting the parameter β of quantity *Kbest* defined in Equation (8). This FIS has one input and one output as shown in Figure 5. The fuzzy input is the number of iterations N that has three triangular/trapezoidal membership functions: Low (L), Medium (M) and High (H) as in FIS-α (see Figure 1). In this way, FIS-β has three rules and thus we define three possible values of the fuzzy output β. Figure 6 shows the three triangular/trapezoidal membership functions Low (L), Medium (M) and High (H) of β. The definition of the fuzzy rules is based on the fact that there is a tendency for *Kbest* to decrease with higher iteration number. The base rule is shown in Table 2 and Figure 5 illustrates the architecture of FIS-β.

The target of the search procedure is to find the best coefficients $b_0, b_1, ..., b_m$ and $a_0, a_1, ..., a_n$ of H with $m = n$, where n is the order of IIR filter. In order to do this, Algorithm 1 is proposed. Both of the designed FISs are used in the optimization process.

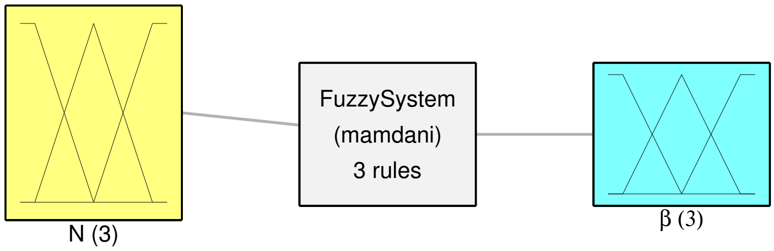

Figure 5. Architecture of Fuzzy Inference System to adjust β.

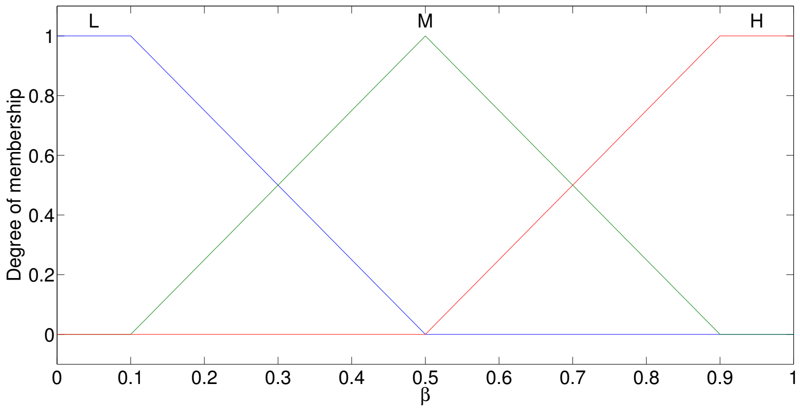

Figure 6. Membership functions of the fuzzy output β.

Table 2. Fuzzy rules for FIS-β.

N	β
L	H
M	M
H	L

Algorithm 1 Fuzzy Gravitational Search Algorithm

1: Random initialization of the coefficients $(b_0, b_1, ..., b_n, a_0, a_1, ..., a_n)$ (see S1).
2: Settings of the fuzzy output extrema values and initialization of the first two values of α (see S2).
3: For each iteration from 1 to n_a (see S3):
4: Search space boundaries control (see S3.1)
5: Agents evaluation (see S3.2)
6: Agents masses calculation (see S3.3).
7: Gravitational constant calculation (see S3.4).
8: *Kbest* computation (see S3.5)
9: Agents acceleration computation (see S3.6).
10: Population progress calculation (see S3.7).
11: Computation of fuzzy-α (see S3.8).
12: Agents velocity and position computation (see S3.9).
13: Final computation of the best coefficients (see S4).

S1. Initialize randomly the coefficients $(b_0, b_1, ..., b_n, a_0, a_1, ..., a_n)$ for each agent in the search space, where n is the order of IIR filter. Note that the number of coefficients is equal to $2(n+1)$ and n_a is the number of agents. Let up and low be the extrema of search interval, the matrix of coefficients for all agents $X^{(n_a, 2(n+1))}$ is computed by

$$x(i,j) = rand(i,j)(up - low) + low \tag{11}$$

for each agent $i \in [1, n_a] \cap \mathbb{N}$ and coefficient $j \in [1, 2(n+1)] \cap \mathbb{N}$; $rand(i,j)$ is a function that generates random numbers between 0 and 1.

S2. Fix the extrema value α_{inf} and α_{sup} of fuzzy output α and the initial value for the first two iterations $\alpha(1)$ and $\alpha(2)$.

S3. For each iteration from 1 to n_a, execute the steps from **S3.1** to **S3.9**.

S3.1. Check the search space boundaries for agents according to the coefficients in X; agents that go out of the search space, are reinitialized randomly, i.e., for the i-th agent such that in the j-th coefficient $x(i,j) > up$ or $x(i,j) < low$, compute $x(i,j)$ by using Equation (11).

S3.2. Evaluate the agents; compute the fitness $J_2(\omega)$ of each agent by passing the coefficients of X to the test function and select the minimum fitness value among agents.

S3.3. Calculate the masses of each agent, i.e., Equations (12)–(16), where M_{aj} and M_{pi} are the active and passive gravitational masses related to agent j and i, respectively, and $M_i(t)$ is the inertial mass of i-th agent at iteration t [30]:

$$M_{ai} = M_{pi} = M_{ii} = M_i, i = 1, 2, ..., n_a, \tag{12}$$

$$m_i(t) = \frac{J_2(\omega)_{(t)} - worst(t)}{best(t) - worst(t)}, \tag{13}$$

$$M_i(t) = \frac{m_i(t)}{\sum_{j=1}^{n_a} m_j(t)}, \tag{14}$$

$$best(t) = \min_{j \in 1,...,n_a} J_2(\omega)_{(t)}, \tag{15}$$

$$worst(t) = \max_{j \in 1,...,n_a} J_2(\omega)_{(t)}. \tag{16}$$

S3.4. Compute the gravitational constant G (see Equation (9)) according to the value of $\alpha(i)$, with $i = 1, 2, ..., n_a$.

S3.5. Compute $Kbest$ as defined by (8); $Kbest$ is computed by changing β by means of FIS-β. The iteration number t is normalized in $[0, 1]$ through the formula $t_n = t/N \in [0, 1]$, where t_n is the normalized iteration number. This normalization is necessary because the fuzzy inputs only accept values in $[0, 1]$. The normalized iteration number is passed as an input to FIS-β, which gives as output the parameter β_{out} with values in $[0, 1]$. This value is normalized between β_{inf} and β_{sup} with the formula

$$\beta = \beta_{out} \cdot (\beta_{sup} - \beta_{inf}) + \beta_{inf}. \tag{17}$$

Therefore, the new value of $Kbest$ is computed by Equation (8).

S3.6. According to $Kbest$, calculate the acceleration of each agent in gravitational field, see Equations (9) and (18)–(22),

$$F_{ij}^d(t) = G(t)\frac{M_{pi}(t) \times M_{ai}(t)}{R_{ij}(t) + \epsilon}(x_j^d(t) - x_i^d(t)), \tag{18}$$

$$R_{ij}(t) = ||X_i(t), X_j(t)||_2, \tag{19}$$

$$F_i^d(t) = \sum_{j=1, j \neq i}^{n_a} rand_j F_{ij}^d(t), \tag{20}$$

$$a_i^d(t) = \frac{F_i^d(t)}{M_{ii}(t)}, \tag{21}$$

$$F_i^d(t) = \sum_{j \in K_{best}, j \neq i}^{n_a} rand_j F_{ij}^d(t), \tag{22}$$

where $d = 2(n+1)$, with n filter order.
S3.7. Compute the population progress $P_p \in [0,1]$ as defined in Equation (10).
S3.8. The values of normalized iteration number t_n and population progress P_p are passed as inputs to FIS-α, which gives a value of α between 0 and 1, denoted by α_{out}. This value is normalized between α_{inf} and α_{sup} with the formula

$$\alpha = \alpha_{out} \cdot (\alpha_{sup} - \alpha_{inf}) + \alpha_{inf}. \tag{23}$$

S3.9. Update the velocity v and coefficients in X of i-th agent, with the Equations (24) and (25), respectively,

$$v_i^d(t+1) = rand_i \times v_i^d(t) + a_i^d(t), \tag{24}$$

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1), \tag{25}$$

where
$x_i^1 = b_0,$
$x_i^2 = b_1,$
...,
$x_i^{n+1} = b_n,$
$x_i^{n+2} = a_0,$
$x_i^{n+3} = a_1,$
...,
$x_i^{2(n+1)} = a_n.$

S4. Give in output the values of best coefficients $(b_0, b_1, ..., b_n, a_0, a_1, ..., a_n)_{opt}$.

FGSA designed in [35,36] have a more simple fuzzy rule base than FIS-α: the just one fuzzy input is the number of iterations. A similar approach has been introduced by Sombra [42] with a fuzzy system characterized by three fuzzy rules to adjust the parameter α of GSA. However, Algorithm 1 guarantees a high accuracy in the computation of α, with a minor adding of complexity compared to the mentioned approaches. On the other hand, Khabisi and Rashedi [43] designed an FGSA with 36 fuzzy rules, without achieving relevant results. The dynamic Type-2 fuzzy logic α adaptation proposed in [39] improves the convergence performances with an increase of the system complexity. The parameter *Kbest* has been adjusted by Olivas [38] directly as output fuzzy, whereas the proposed algorithm tunes the parameter β with a new definition of *Kbest* (see Equation (8)). This fact assures a better regulation of *Kbest* during the steps of our FGSA. Moreover, GSA with Wavelet Mutation proposed in [32] has the drawbacks of the rigorous trials required for the tuning of control parameters for the wavelet mutation method. Finally, Algorithm 1 supplies a good trade-off between accuracy and complexity compared with the mentioned approaches.

4. Experimental Results

Algorithm 1 is compared with the application of GSA and DE for the design of IIR filters. All optimization algorithms are run in the MATLAB environment (R2016a, MathWorks, Natick, MA, USA) on 2.20 GHz-speed processor. In particular, the fuzzy toolbox of MATLAB is exploited. This tool gives the possibility of designing Fuzzy Inference Systems based on the Mamdani inference method [52]. Because the MATLAB environment allows for referring to objects created by tools with the MATLAB code, the settings of the designed FISs can be dynamically modified. Moreover, the parameters of the fuzzy part of Algorithm 1 are computed by using the Center of the Mass defuzzification method. The MATLAB environment has been also exploited for its workspace data storage capability.

Referring to FGSA, the number of agents n_a is set to 50 and the iterations number N is equal to 500. Moreover, $\alpha_{inf} = 1$, $\alpha_{sup} = 20$ and $\beta_{inf} = 2$, $\beta_{sup} = 4$ in Equations (23) and (17), respectively. The value of α in GSA is 20 (as in [30]), $n_a = 50$ and $N = 500$. In DE, the population size is 50, the crossover probability is 0.2 and the maximum number of iterations is 500.

Equation (11) computes the values of coefficients in H: we assume a range from $low = -2$ to $up = +2$ with a filter order $n = 8$, which is $b_0, ..., b_8, a_0, ..., a_8 \in [-2, 2]$. The design specifications of Low Pass (LP), High Pass (HP), Band Pass (BP) and Stop Band (SB) IIR filters are shown in Table 3, with w frequency width. Moreover, the frequency range from 0 to π is divided into $n_s = 256$ equally spaced sample points. DE, GSA and FGSA are run for 30 times to get the best solutions and the results in Tables 4–8 are the average (first sub-row) and standard deviation (second sub-row) on 30 experiments.

Table 3. Design specifications of Low Pass (LP), High Pass (HP), Band Pass (BP) and Stop Band (SB) IIR filters.

Filter	δ_p	δ_s	ω_p	ω_s	w
LP	0.01	0.001	0.45	0.50	-
HP	0.01	0.001	0.35	0.30	-
BP	0.01	0.001	0.35 and 0.65	0.3 and 0.7	0.3
SB	0.01	0.001	0.25 and 0.55	0.3 and 0.7	0.4

Table 4. Pass band ripple, stop band ripple and transition band of LP filters.

Algorithm	PB-Ripple	SB-Ripple	tb
Differential Evolution (DE)	0.209371 0.092159	0.174627 0.100366	0.430534 0.188224
Gravitational Search Algorithm (GSA)	0.067816 0.016867	0.066199 0.016857	0.120964 0.029634
Fuzzy Gravitational Search Algorithm (FGSA)	0.065990 0.022298	0.061591 0.017547	0.101172 0.027451

Table 5. Pass band ripple, stop band ripple and transition band of HP filters.

Algorithm	PB-Ripple	SB-Ripple	tb
Differential Evolution	0.301758 0.154653	0.136288 0.087169	0.401888 0.155471
Gravitational Search Algorithm	0.047248 0.016758	0.057344 0.010286	0.195117 0.098803
Fuzzy Gravitational Search Algorithm	0.067245 0.030699	0.058384 0.011865	0.217318 0.073569

Table 6. Pass band ripple, stop band ripple and transition band of BP filters.

Algorithm	PB-Ripple	SB-Ripple	tb
Differential Evolution	0.158257 0.096243	0.463677 0.489097	0.014583 0.413012
Gravitational Search Algorithm	0.067377 0.024833	0.080152 0.015516	0.127604 0.041768
Fuzzy Gravitational Search Algorithm	0.070821 0.017908	0.076286 0.017812	0.131966 0.061376

Table 7. Pass band ripple, stop band ripple and transition band of SB filters.

Algorithm	PB-Ripple	SB-Ripple	tb
Differential Evolution	0.254731 0.127932	0.077870 0.041679	0.248177 0.136460
Gravitational Search Algorithm	0.078905 0.022983	0.074704 0.014319	0.117773 0.028508
Fuzzy Gravitational Search Algorithm	0.066908 0.023753	0.083628 0.026077	0.111458 0.034889

Table 8. Convergence profile results.

Algorithm	f_{LP}	f_{HP}	f_{BP}	f_{SB}
Differential Evolution	0.016140 0.004195	0.017622 0.005027	0.021384 0.003965	0.029527 0.004395
Gravitational Search Algorithm	0.006185 0.004130	0.004209 0.003159	0.011853 0.006739	0.010370 0.004559
Fuzzy Gravitational Search Algorithm	0.004364 0.004154	0.006941 0.005924	0.010368 0.006368	0.011207 0.008748

Table 4 shows that the proposed algorithm gives good results for LP filters because pass band ripple, stop band ripple and transition band are less than DE and GSA values. This fact is confirmed by comparing the magnitude response over normalized frequency of DE, GSA and FGSA (see Figure 7) for LP filters. By comparing FGSA and GSA, the pass band ripple (PB-ripple in the table) is improved by 3%, the stop band ripple (SB-ripple) is reduced by 7% and the transition band (tb) is improved by 16%. Moreover, the robustness of the proposed approach is analyzed by computing the standard deviation on 30 experiments. In fact, the robustness depends on standard deviation values: a result is more robust if the data have a smaller standard deviation. Observing Table 4, we can note that standard deviation values of GSA and FGSA are about the same, whereas DE results show less robustness than GSA and FGSA. Finally, the proposed approach shows a good robustness for the design of LP filters.

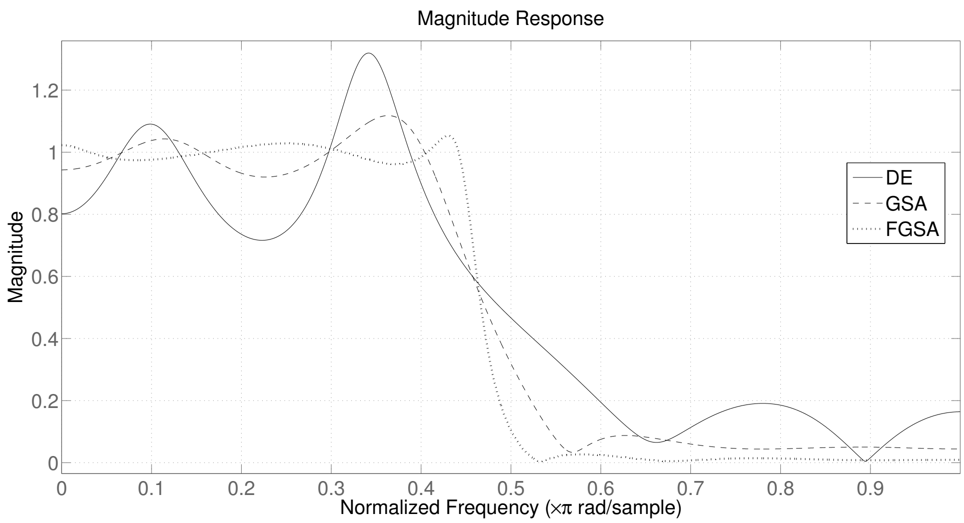

Figure 7. Magnitude response over normalized frequency for 8th order IIR Low Pass filter using Differential Evolution, Gravitational Search Algorithm and Fuzzy Gravitational Search Algorithm.

Table 5 shows the average pass and stop band ripples and transition band for HP filters. In this case, FGSA is better than DE. Moreover, FGSA has about the same value of stop band ripple of GSA, but pass band ripple and transition band are greater than GSA. However, FGSA gives a reasonable trend of magnitude response over frequency (see Figure 8). As in LP filters' results, the outcomes on 30 experiments show a good robustness of FGSA (see the standard deviation values in Table 5).

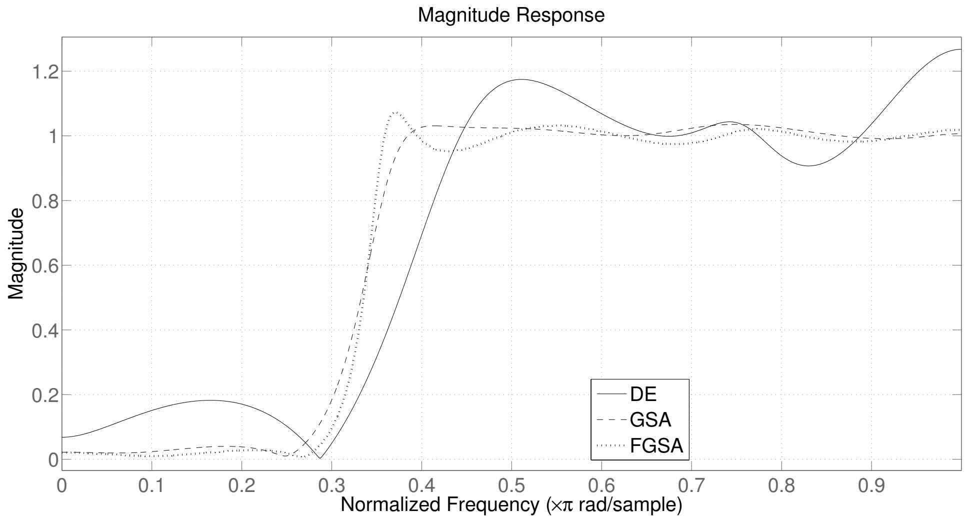

Figure 8. Magnitude response over normalized frequency for 8th order IIR High Pass filter using DE, GSA and FGSA.

For BP filters, FGSA and GSA are better than DE (see Table 6). The stop band ripple of FGSA is less than GSA, whereas pass band ripple and transition band are greater than GSA. Figure 9 shows a symmetric trend of GSA magnitude response. Moreover, good results of robustness are achieved (see Table 6).

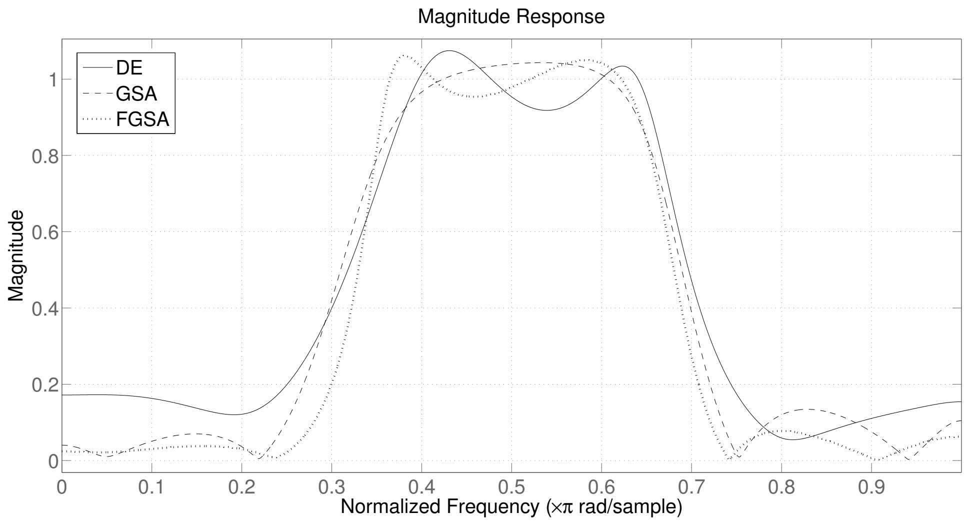

Figure 9. Magnitude response over normalized frequency for 8th order IIR Band Pass filter using DE, GSA and FGSA

Algorithm 1 gives good results for SB filters, where pass band ripple and transition band are better than DE and GSA (see Table 7). In particular, FGSA reduces the pass band ripple by 15% with respect to GSA. Moreover, Figure 10 shows that FGSA has a trend very close to an ideal SB filter. Table 7 shows low standard deviation values for FGSA: this fact assures a good robustness of the developed method.

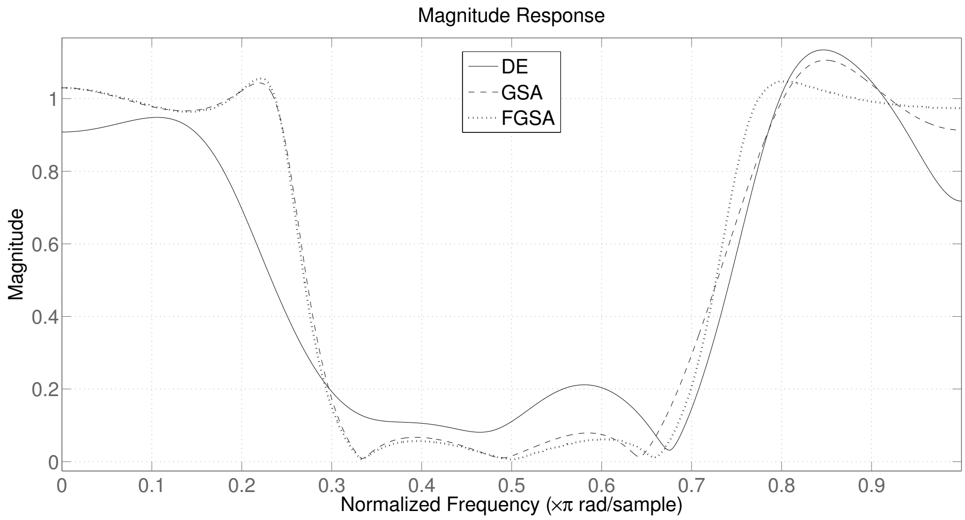

Figure 10. Magnitude response over normalized frequency for 8th order IIR Stop Band filter using DE, GSA and FGSA.

Usually, the search algorithm computation time is a measure of the procedure convergence speed. On the other hand, the computation time is not good to evaluate the convergence speed because it depends on hardware performances, programming language and designer skills. A good way to evaluate the convergence speed is to consider the objective function evaluations number up to the minimum value of the function (see [23,31–34,53]). Finally, we consider the ratio between the fitness function calculations number n_f and the evaluations number N: such ratio defines the convergence rate denoted by c_r (see Equation (26)):

$$c_r = \frac{n_f}{N}. \tag{26}$$

Table 8 contains the convergence profile results of DE, GSA and FGSA for 8th order LP, HP, BP and SB IIR filters on 30 experiments. Each row contains the best value of fitness error function and standard deviation obtained by using the specified algorithm. In the table, f_{LP} denotes the optimal value of fitness function for the filter LP, f_{HP} denotes the optimal value of fitness function for the filter HP, f_{BP} denotes the optimal value of fitness function for the filter BP and f_{SB} denotes the optimal value of fitness function for the filter SB. Note that FGSA shows a better design performance for LP and SB filters, whereas there is a certain equilibrium between FGSA and GSA for HP and BP filters. Moreover, DE is the worst IIR filter design algorithm in terms of fitness function minimization. Referring to robustness, DE, GSA and FGSA have about the same low values of standard deviation of order 10^{-3} (see Table 8). This fact assures a very good robustness of the proposed approach.

Table 9 shows the convergence rates of DE, GSA and FGSA for 8th order LP, HP, BP and SB IIR filters. They are referred to as $c_{r_{LP}}$, $c_{r_{HP}}$, $c_{r_{BP}}$ and $c_{r_{SB}}$ for LP, HP, BP and SB filters, respectively. The analysis on the convergence rate results shows that FGSA has a convergence rate better than GSA for LP and HP filters, whereas GSA is better than FGSA for BP and SB filters. However, FGSA gives the best results because the improvements on convergence rate are 6% and 10% for LP and HP, respectively, versus 7% and 1% of GSA for BP and SB. Moreover, DE gives the worst values of convergence rate. Figure 11 shows the trend of the error fitness function over the number of iteration N for LP filters by using DE, GSA and FGSA. Note that FGSA has a better convergence rate than GSA and DE.

Finally, FGSA achieves the best compromise between IIR filter design performances and convergence rate with a good robustness. These facts make FGSA better than DE and GSA for the optimal design of 8th order IIR filters.

Stability analysis of the designed IIR filters is shown in Figures 11–15, where the circle markers represent the zeros, whereas the cross markers are the poles. The pole-zero plots demonstrate the existence of poles within the unit circle, which assures the Bounded Input Bounded Output (BIBO) stability condition. However, adding constraints to optimization algorithms may cause an increase of computational complexity. Recently, Pelusi et al. [54] have proposed a Neural and Fuzzy Gravitational Search Algorithm (NFGSA) able to search local optima with low complexity. The future challenge will be the application of NFGSA for designing optimal IIR filters with fuzzy stability constraints.

Table 9. Convergence rates of DE, GSA and FGSA for LP, HP, BP and SB IIR filters.

Algorithm	cr_{LP}	cr_{HP}	cr_{BP}	cr_{SB}
Differential Evolution	0.6900	0.6460	0.9900	0.8660
Gravitational Search Algorithm	0.4840	0.5640	0.5200	0.6360
Fuzzy Gravitational Search Algorithm	0.4560	0.5080	0.5580	0.6440

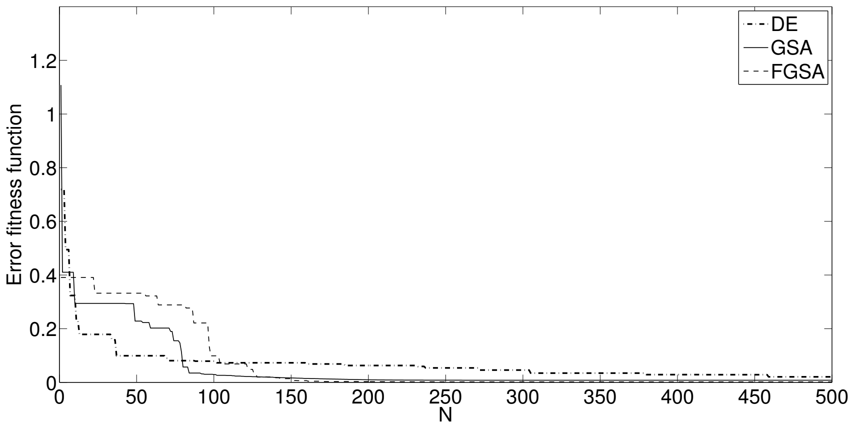

Figure 11. Error fitness function for LP IIR filter.

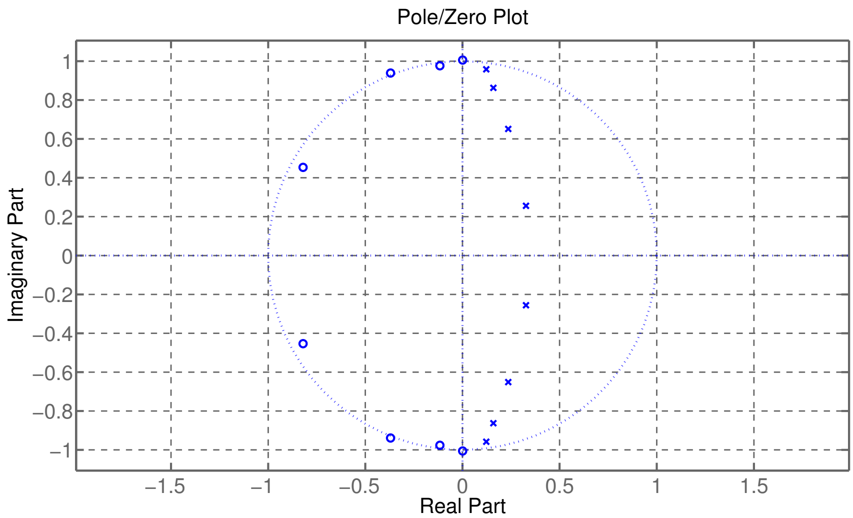

Figure 12. Pole-zero plot of LP IIR filter.

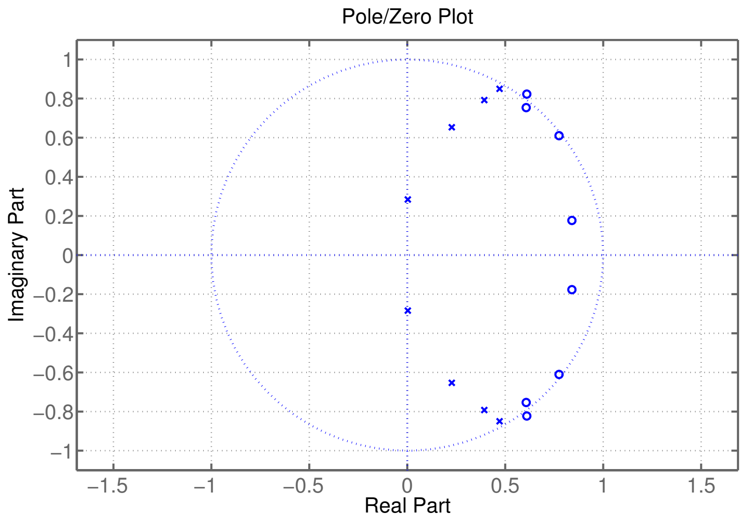

Figure 13. Pole-zero plot of HP IIR filter.

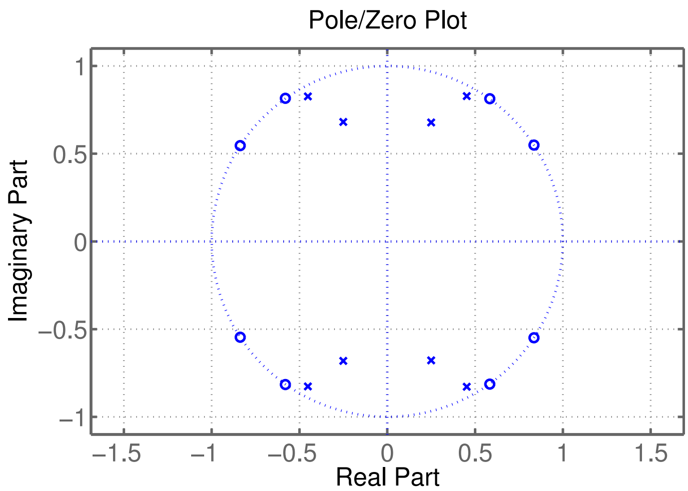

Figure 14. Pole-zero plot of BP IIR filter.

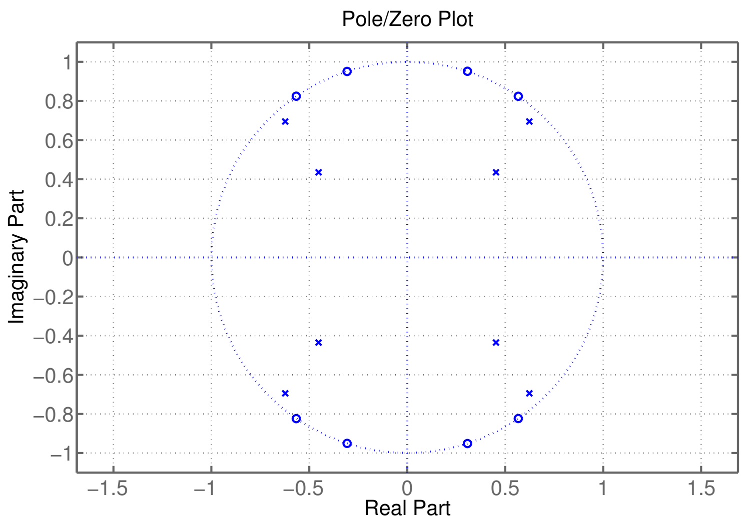

Figure 15. Pole-zero plot of SB IIR filter.

5. Conclusions

An intelligent algorithm able to optimize the design of 8th order IIR filters has been described. Because the quality of a filter depends on pass band ripple, stop band ripple and transition band, the target of the paper is the optimization of an error fitness function that depends on these parameters. Such task is accomplished through a suitable optimization algorithm. The proposed algorithm is a combination between fuzzy techniques and GSA. In particular, two fuzzy systems able to adjust some parameters of GSA have been designed. Moreover, to improve GSA, one of these parameters has been re-defined. Our algorithm has been compared with DE and GSA for the design of IIR filters. The results show that FGSA is the best algorithm to design 8th order IIR filters in terms goodness, robustness and convergence rate. Moreover, the proposed algorithm always gives a stable filter. Further research tasks will focus on: (1) the improvement of the fitness function definition; (2) the design of FISs for other GSA parameters assuring a good compromise between best solution and high convergence speed for the design of IIR filters; and (3) the comparison with other optimization algorithms such as Particle Swarm Algorithm and Genetic Algorithms. A future fascinating challenge will be the design of optimal IIR filters with fuzzy stability constraints.

Author Contributions: Danilo Pelusi and Raffaele Mascella conceived and designed the experiments; Danilo Pelusi performed the experiments; Danilo Pelusi, Raffaele Mascella and Luca Tallini analyzed the data; Danilo Pelusi contributed reagents/materials/analysis tools; and Danilo Pelusi wrote the paper.

Conflicts of Interest: We declare that there is not conflict of interest.

References

1. Antoniou, A. *Digital Signal Processing: Signals, Systems, and Filters*; McGraw-Hill: New York, NY, USA, 2005.
2. Lu, W.S.; Antoniou, A. Design of digital filters and filter banks by optimization: A state of the art review. In Proceedings of the 2000 10th European Signal Processing Conference, Tampere, Finland, 4–8 September 2000; Volume 1, pp. 351–354.
3. Mikhael, R.; Agathoklis, P.; Xiao, C. Design of linear-phase recursive filters by optimization of model reduced non-recursive filters. In Proceedings of the 2003 PACRIM IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, Ictoria, BC, Canada, 28–30 August 2003.
4. Chen, B.S.; Peng, S.C.; Chiou, B.-W. IIR filter design via optimal Hankel–norm approximation. *IEE Proc. G Circuits Devices Syst.* **1992**, *139*, 586–590.
5. Lu, W.S.; Hinamoto, T. Optimal Design of IIR Frequency-Response-Masking Filters Using Second-Order Cone Programming. *IEEE Trans. Circuits Syst*. **2003**, *50*, 1401–1412.
6. Lu, W.S.; Hinamoto, T. Optimal design of IIR digital filters with robust stability using conic-quadratic-programming. *IEEE Trans. Signal Process.* **2003**, *51*, 1581–1592.
7. Lai, X.; Lin, Z. Minimax Phase Error Design of IIR Digital Filters With Prescribed Magnitude and Phase Responses. *IEEE Trans. Signal Process.* **2012**, *60*, 980–986.
8. Nongpiur, R.C.; Shpak, D.J.; Antoniou, A. Design of IIR Digital Differentiators Using Constrained Optimization. *IEEE Trans. Signal Process.* **2014**, *62*, 1729–1739.
9. Lai, X.; Lin, Z. Iterative Reweighted Minimax Phase Error Designs of IIR Digital Filters With Nearly Linear Phases. *IEEE Trans. Signal Process.* **2016**, *64*, 2416–2428.
10. Lang, M.C. Least-squares design of IIR filters with prescribed magnitude and phase responses and a pole radius constraint. *IEEE Trans. Signal Process.* **2000**, *48*, 3109–3121.
11. Duarte, C.; Barner, K.E.; Goossen, K. Design of IIR Multi-Notch Filters Based on Polynomially-Represented Squared Frequency Response. *IEEE Trans. Signal Process.* **2016**, *64*, 2613–2623.
12. Pei, S.C.; Shyu, J.J.; Huang, Y.D.; Chan, C.H. Improved Methods for the Design of Variable Fractional-Delay IIR Digital Filters. *IEEE Trans. Circuits Syst.* **2012**, *59*, 989–1000.
13. Storn, R.; Price, K. Differential Evolution: A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. *J. Glob. Optim.* **1997**, *11*, 341–359.

14. Storn, R. Differential evolution design of an IIR-filter with requirements for magnitude and group delay. In Proceedings of the IEEE International Conference on Evolutionary Computation (ICEC), Nagoya, Japan, 20–22 May 1996; pp. 268–273,
15. Storn, R. Designing nonstandard filters with differential evolution. *IEEE Signal Process. Mag.* **2005**, *22*, 103–106.
16. Pan, S.T. Evolutionary Computation on Programmable Robust IIR Filter Pole-Placement Design. *IEEE Trans. Instrum. Meas.* **2011**, *60*, 1469–1479.
17. Karaboga, N. Digital IIR Filter Design Using Differential Evolution Algorithm. *EURASIP J. Appl. Signal Process.* **2005**, *8*, 1269–1276.
18. Dai, C.; Chen, W.; Zhu, Y. Seeker Optimization Algorithm for Digital IIR Filter Design. *IEEE Trans. Ind. Electron.* **2010**, *57*, 1710–1718.
19. Kennedy, J.; Eberhart, R. Particle swarm optimization. *Proc. Int. Conf. Neural Netw.* **1995**, *4*, 1942–1948.
20. Hartmann, A.; Lemos, J.M.; Costa, R.S.; Vinga, S. Identifying IIR filter coefficients using particle swarm optimization with application to reconstruction of missing cardiovascular signals. *Eng. Appl. Artif. Intell.* **2014**, *34*, 193–198.
21. Aimi, H.; Suyama, K. Multi-Constrained IIR Filter Design Using PSO. In Proceedings of the International Symposium on Electronics and Smart Devices, Bandung, Indonesia, 29–30 November 2016; pp. 258–263.
22. Wang, Y.; Li, B.; Chen, Y. Digital IIR filter design using multi-objective optimization evolutionary algorithm. *Appl. Soft Comput.* **2011**, *12*, 1851–1857.
23. Tsai, J.T.; Chou, J.H. Design of Optimal Digital IIR Filters by Using an Improved Immune Algorithm. *IEEE Trans. Signal Process.* **2006**, *54*, 4582–4596.
24. Holland, J. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*; Bradford Books; MIT Press: Cambridge, MA, USA, 1992.
25. Zhang, G.X.; Jin, W.D.; Jin, F. Multi-criterion satisfactory optimization method for designing IIR digital filters. In Proceedings of the ICCT 2003 International Conference on Communication Technology, Beijing, China, 9–11 April 2003; pp. 1484–1490.
26. Liu, Y.; Shao, S.Q.; Zhao, H.;Liao, X.F.; Yu, J.B. An application of genetic algorithms with guiding strategy in the design of digital filters. In Proceedings of the Communications, Circuits and Systems, Chengdu, China, 27–29 June 2004; pp. 1141–1145.
27. Chang, W.-D. Coefficient estimation of IIR filter by a multiple crossover genetic algorithm. *Comput. Math. Appl.* **2006**, *51*, 1437–1444.
28. Yu, Y.; Yu, X. Cooperative Coevolutionary Genetic Algorithm for Digital IIR Filter Design. *IEEE Trans. Ind. Electron.* **2007**, *54*, 1311–1318.
29. Pan, S.T. Design of Robust D-Stable IIR Filters Using Genetic Algorithms With Embedded Stability Criterion. *IEEE Trans. Signal Process.* **2009**, *57*, 3008–3016.
30. Rashedi, E.; Nezamabadi-Pour, H.; Saryazdi, S. GSA: A Gravitational Search Algorithm. *Inf. Sci.* **2009**, *179*, 2232–2248.
31. Saha, S.K.; Kar, R.; Mandal, D.; Ghoshal, S.P. Gravitation search algorithm: Application to the optimal IIR filter design. *J. King Saud Univ. Eng. Sci.* **2014**, *26*, 69–81.
32. Saha, S.K.; Kar, R.; Mandal, D.; Ghoshal, S.P. Optimal IIR filter design using Gravitational Search Algorithm with Wavelet Mutation. *J. King Saud Univ. Comput. Inf. Sci.* **2015**, *27*, 25–39.
33. Precup, R.E.; David, R.C.; Petriu, E.M.; Preitl, S.; Radac, M.B. Fuzzy logic-based adaptive gravitational search algorithm for optimal tuning of fuzzy controlled servo systems. *IET Control Theory Appl.* **2013**, *7*, 99–107.
34. Precup, R.E.; David, R.C.; Petriu, E.M.; Preitl, S.; Radac, M.B. Adaptive GSA-based optimal tuning of PI controlled servo systems with reduced process parametric sensitivity, robust stability and controller robustness. *IEEE Trans. Cybern.* **2014**, *44*, 1997–2009.
35. Gonzalez, B.; Valdez, F.; Melin, P.; Prado-Arechiga, G. Fuzzy logic in the gravitational search algorithm enhanced using fuzzy logic with dynamic alpha parameter value adaptation for the optimization of modular neural networks in echocardiogram recognition. *Appl. Soft Comput.* **2015**, *37*, 245–254.
36. Gonzalez, B.; Valdez, F.; Melin, P.; Prado-Arechiga, G. Fuzzy logic in the gravitational search algorithm for the optimization of modular neural networks in pattern recognition. *Exp. Syst. Appl.* **2015**, *42*, 5839–5847.
37. Pelusi, D.; Mascella, R.; Tallini, L. Revised gravitational search algorithms based on evolutionary-fuzzy systems. *Algorithms* **2017**, *10*, 44.

38. Olivas, F.; Valdez, F.; Ostillo, O. Gravitational Search Algorithm with parameter adaptation through a fuzzy logic system. In *Nature-Inspired Design of Hybrid Intelligent Systems*; Studies in Computational Intelligence; Springer: New York, NY, USA, 2016; Volume 667, pp. 391–405.
39. Gonzalez, B.; Valdez, F.; Melin, P. A Gravitational Search Algorithm Using Type-2 Fuzzy Logic for Parameter Adaptation. In *Nature-Inspired Design of Hybrid Intelligent Systems*; Studies in Computational Intelligence; Springer: New York, NY, USA, 2016; Volume 667, pp. 127–138.
40. Askari, H.; Zahiri, S.H. Data Classification Using Fuzzy-GSA. In Proceedings of the International Conference on Computer and Knowledge Engineering, Mashhad, Iran, 13–14 October 2011; pp. 6–11.
41. Zahiri, S.H. Fuzzy gravitational search algorithm an approach for data mining. *Iran. J. Fuzzy Syst.* **2012**, *9*, 21–37.
42. Sombra, A.; Valdez, F.; Melin, P.; Castillo, O. A new gravitational search algorithm using fuzzy logic to parameter adaptation. In Proceedings of the IEEE Congress on Evolutionary Computation, Cancun, Mexico, 20–23 June 2013; pp. 1068–1074.
43. Khabisi, F.S.; Rashedi, E. Fuzzy gravitational search algorithm. In Proceedings of the International E-Conference on Computer and Knowledge Engineering, Mashhad, Iran, 18–19 October 2012; pp. 156–160.
44. Proakis, J.G.; Manolakis, D.G. *Digital Signal Processing*; Prentice-Hall: Upper Saddle River, NJ, USA, 1996.
45. Karaboga, N.; Cetinkaya, B. Design of minimum phase digital IIR filters by using genetic algorithm. In Proceedings of the 2004 NORSIG 6th Nordic IEEE 6th Nordic Signal Processing Symposium, Espoo, Finland, 9–11 June 2004; pp. 29–32.
46. Luitel, B.; Venayagamoorthy, G.K. Particle swarm optimization with quantum infusion for the design of digital filters. In Proceedings of the IEEE Swarm Intelligence Symposium, St. Louis, MO, USA, 21–23 September 2008; pp. 1–8.
47. Luitel, B.; Venayagamoorthy, G.K. Differential evolution particle swarm optimization for digital filter design. In Proceedings of the IEEE Congress on Evolutionary Computation, Hong Kong, China, 1–6 June 2008; pp. 3954–3961.
48. Dumitrescu, B.; Niemisto, R. Multistage IIR filter design using convex stability domains defined by positive realness. *IEEE Trans. Signal Process.* **2004**, *52*, 962–974.
49. Tseng, C.; Lee, S. Minimax design of stable IIR digital filter with prescribed magnitude and phase responses. *IEEE Trans. Circuits Syst. I Fundam. Theory Appl.* **2002**, *49*, 547–551.
50. Jiang, A.; Kwan, H.K. IIR digital filter design with new stability constraint based on argument principle. *IEEE Trans. Circuits Syst.* **2009**, *56*, 583–593.
51. Azizipanah-Abarghooee, R.; Terzija, V.; Golestaneh, F. Multiobjective Dynamic Optimal Power Flow Considering Fuzzy-Based Smart Utilization of Mobile Electric Vehicles. *IEEE Trans. Ind. Inform.* **2016**, *12*, 503–514.
52. Mamdani, E.H. Application of fuzzy logic to approximate reasoning using linguistic synthesis. *IEEE Trans. Comput.* **1977**, *26*, 1182–1191.
53. Precup, R.-E.; David, R.-C.; Petriu, E.M.; Radac, M.-B.; Preitl, S.; Fodor, J. Evolutionary optimization-based tuning of low-cost fuzzy controllers for servo systems. *Knowl. Based Syst.* **2013**, *38*, 74–84.
54. Pelusi, D.; Mascella, R.; Tallini, L.; Nayak, J.; Naik, B.; Abraham, A. Neural Network and Fuzzy System for the tuning of Gravitational Search Algorithm parameters. *Exp. Syst. Appl.* **2018**, doi:10.1016/j.eswa.2018.02.026.

© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).