Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2017) 25: 4299 – 4310

doi:10.3906/elk-1701-294

Fractional-order controller design in frequency domain using an improved nonlinear adaptive seeker optimization algorithm

Mano Ranjan KUMAR*, Vishwa DEEPAK, Subhojit GHOSH
Department of Electrical Engineering, National Institute of Technology, Raipur, India

Received: 29.01.2017 • **Accepted/Published Online:** 28.04.2017 • **Final Version:** 05.10.2017

Abstract: Two nonlinear adaptive versions of the conventional seeker optimization algorithm (SOA) have been proposed for the design of fractional-order controllers using frequency domain specifications. The highly nonlinear and undetermined nature of equations resulting from controller design specifications rules out obtaining a closed-form solution. In this regard, the controller design task has been formulated as an optimization problem and solved using modified variants of the SOA. With the nonlinear adaptation of tuning parameters, the proposed variants of the SOA increase the probability of finding global optima along with improved convergence speed. This has been achieved by incorporating exponential weighting and chaotic behavior in the search process. The validation of the proposed techniques on a set of fractional-order controller design problems clearly exhibits their superiority over other algorithms and controller design techniques. The hardware implementation of the controllers on a DSP TMS320F2812 board ensures applicability for real-time applications.

Key words: Seeker optimization algorithm, exponential weighting, chaotic search, fractional-order controller, gain margin and phase margin

1. Introduction

Recent times have witnessed tremendous growth in the development of fractional-order (FO) control systems [1,2]. The use of arbitrary real numbers instead of fixed integers, for order of integration and differentiation, allows incorporating long-term memory and distributed behavior in controller dynamics. The development of computational tools in the field of fractional calculus has led to wide applications in different fields [3], including dynamic system modeling and control [4]. As compared to classical PI/PID controllers, FOPI/PID controllers provide greater flexibility in improving system performance due to the presence of extra tuning parameters, i.e. the noninteger orders of the integral and differential operators [1]. These extra parameters impart additional features to the fractional order controllers, such as robustness to plant uncertainties (iso-damping) [5], lesser sensitivity to load disturbances, high-frequency noise rejection [6], and flexibility on system stability [1]. FO controllers can achieve robustness similar to very high integer-order (IO) controllers. Some of the recent works on the application of FO controllers on real-time systems include controller design for coupled-tank liquid level systems [7] and magnetic levitation systems [8]. However, the extra flexibility arising out of the increase in the number of tuning parameters makes the controller design and synthesis task quite challenging, which warrants the use of nonlinear approximations or computational optimization tools [2].

*Correspondence: manornjn48@gmail.com

Optimization techniques have been widely adopted for designing FO controllers. In [6,7,9,10], local search-based techniques were applied for fast tuning of FO controllers. However, their performance was sensitive to the selection of the initial feed point and the solutions obtained were prone to getting trapped into local optima. In this context, a number of works have adopted population-based evolutionary algorithms such as particle swarm optimization [8], differential evolution (DE) [11], and the covariance matrix adaptation evolution strategy (CMAES) [12] to obtain global solutions for fractional controller design problems.

As compared to the widely used controller tuning approaches in the time domain, controllers designed in the frequency domain offer several advantages, such as robustness against system loop gain variations, high-frequency measurement noise rejection, disturbance suppression, and reduced size of the actuator because of the lesser magnitude of the control action [9]. The works reported on FO controller design in the frequency domain include a graphical technique [5], a design based on the conflicting multiple objectives of phase margin and gain crossover frequency using chaotic optimization [13], and the minimization of load disturbance and high-frequency measurement noise rejection [6].

In the only reported work on optimization-based design of FO controllers using gain and phase margin [14], a tuning procedure for the design of an FOID controller was proposed. The approach is applicable only to a specified class of plants. Motivated by the correlation between gain and phase margin with robustness, relative stability, and the dynamic response of the system, the present work aims at attaining a set of controller parameters that satisfies a predefined gain and phase margin for a generalized class of plants. In this regard, the controller design task has been formulated as an optimization problem.

Considering the ability of the evolutionary algorithm in solving complex multimodel functions, a relatively new evolutionary technique, i.e. the seeker optimization algorithm (SOA) [15], has been used in the present work. As compared to other evolutionary algorithms, the smaller number of control parameters in the SOA makes it easier to implement. Recently the SOA has been employed in several engineering applications [16] including identification of FO systems [17]. However, like other evolutionary techniques, the conventional SOA suffers from the limitation of getting trapped into local minima for multimodal functions and, quite often, the convergence is slow [18]. In this context, the present work proposes two modified versions of the SOA based on the nonlinear adaptation of inertia weight in the solution (seeker position) updating process. Unlike the linearly decaying inertia weight in the conventional SOA, an exponential decay is incorporated in the learning weight in the first proposed variant, whereas in the second variant, a random component is included to chaotically adapt the learning process. While the exponential weighting leads to faster convergence, the chaotic behavior maintains the diversity in the population by exploiting the search behavior more effectively and hence increases the probability of converging to a global solution.

The next section focuses on the formulation of the FO controller design as an optimization problem. The proposed variants of the SOA and their convergence are discussed in Section 3, the simulation and experimental results are dealt with in Section 4, and finally Section 5 provides conclusions.

2. Optimization-based FO controller design in frequency domain

2.1. FO controllers

FO controllers have been found to be suitable in rejecting high-frequency noise and ensuring good output disturbance rejection [6]. The methodologies developed for tuning the FOPI/FOPID can be broadly classified in terms of their specifications into time domain and frequency domain [9]. In the present work, frequency domain specifications (gain margin and phase margin) have been used for the design of FOPI controllers.

2.2. Optimization-based FOPI controller design

For a control system (Figure 1) consisting of a plant $G_p(s)$ and a controller $G_c(s)$, the gain margin (A_m) and the phase margin (φ_m) can be specified in the frequency domain using the following set of equations:

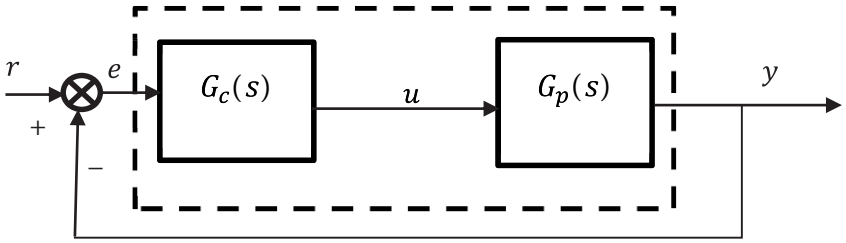


Figure 1. Unity gain negative feedback control system.

$$arg\left[G_c\left(j\omega_p\right)G_p\left(j\omega_p\right)\right]=-\pi \tag{1}$$

$$A_m=\frac{1}{\left|G_c\left(j\omega_p\right)G_p\left(j\omega_p\right)\right|} \tag{2}$$

$$\left|G_c\left(j\omega_g\right)G_p\left(j\omega_g\right)\right|=1, \tag{3}$$

$$arg\left[G_c\left(j\omega_g\right)G_p\left(j\omega_g\right)\right]+\pi=\varphi_m \tag{4}$$

where ω_p and ω_g represent the phase and gain crossover frequency, respectively.

For proportional gain K_p and FO integral gain K_i, the transfer function of the FOPI controller can be represented as:

$$G_c\left(s\right)=G_{FOPI}\left(s\right)=K_p\left(1+\frac{K_i}{s^\lambda}\right) \tag{5}$$

Considering a prototype plant with constant gain K and time delay L the plant can be represented as:

$$G_p\left(s\right)=\frac{K(1+\omega_{z_1}s)^{z_1}(1+\omega_{z_2}s)^{z_2}\cdots(1+\omega_{z_n}s)^{z_n}}{(1+\omega_{p_1}s)^{p_1}(1+\omega_{p_2}s)^{p_2}\cdots(1+\omega_{p_m}s)^{p_m}}e^{-Ls} \tag{6}$$

The open loop transfer function is obtained using Eqs. (5) and (6) as:

$$G_{OLTF}\left(s\right)=K_p\left(1+\frac{K_i}{s^\lambda}\right)\frac{K(1+\omega_{z_1}s)^{z_1}(1+\omega_{z_2}s)^{z_2}\cdots(1+\omega_{z_n}s)^{z_n}}{(1+\omega_{p_1}s)^{p_1}(1+\omega_{p_2}s)^{p_2}\cdots(1+\omega_{p_m}s)^{p_m}}e^{-Ls} \tag{7}$$

On substituting Eqs. (5) and (6) into Eqs. (1)–(4):

$$-\tan^{-1}\left(\frac{K_i\omega_p^{-\lambda}\sin\left(\lambda\frac{\pi}{2}\right)}{1+K_i\omega_p^{-\lambda}\cos\left(\lambda\frac{\pi}{2}\right)}\right)+z_1\tan^{-1}\left(\omega_p\omega_{z_1}\right)+...+z_n\tan^{-1}\left(\omega_p\omega_{z_n}\right)$$

$$-\omega_pL-p_1\tan^{-1}\left(\omega_p\omega_{p_1}\right)-\cdots-p_m\tan^{-1}\left(\omega_p\omega_{p_m}\right)=-\pi \tag{8}$$

$$A_mK_pK\left(\left(1+K_i\omega_p^{-\lambda}\cos\left(\lambda\frac{\pi}{2}\right)\right)^2+\left(K_i\omega_p^{-\lambda}\sin\left(\lambda\frac{\pi}{2}\right)\right)^2\right)^{0.5}$$

$$= \frac{\left(1+\omega_p^2\omega_{p_1}^2\right)^{0.5p_1}}{\left(1+\omega_p^2\omega_{z_1}^2\right)^{0.5z_1}} \cdots \frac{\left(1+\omega_p^2\omega_{p_m}^2\right)^{0.5p_m}}{\left(1+\omega_p^2\omega_{z_n}^2\right)^{0.5z_n}} \tag{9}$$

$$K_pK\left(\left(1+K_i\omega_g^{-\lambda}\cos\left(\lambda\frac{\pi}{2}\right)\right)^2+\left(K_i\omega_g^{-\lambda}\sin\left(\lambda\frac{\pi}{2}\right)\right)^2\right)^{0.5}$$

$$= \frac{\left(1+\omega_g^2\omega_{p_1}^2\right)^{0.5p_1}}{\left(1+\omega_g^2\omega_{z_1}^2\right)^{0.5z_1}} \cdots \frac{\left(1+\omega_g^2\omega_{p_m}^2\right)^{0.5p_m}}{\left(1+\omega_g^2\omega_{z_n}^2\right)^{0.5z_n}}, \tag{10}$$

$$-\tan^{-1}\left(\frac{K_i\omega_g^{-\lambda}\sin\left(\lambda\frac{\pi}{2}\right)}{1+K_i\omega_g^{-\lambda}\cos\left(\lambda\frac{\pi}{2}\right)}\right)+z_1\tan^{-1}\left(\omega_g\omega_{z_1}\right)+..+z_n\tan^{-1}\left(\omega_g\omega_{z_n}\right)$$

$$-\omega_gL-p_1\tan^{-1}\left(\omega_g\omega_{p_1}\right)-\cdots-p_m\tan^{-1}\left(\omega_g\omega_{p_m}\right)=-\pi+\varphi_m \tag{11}$$

The FOPI controller design involves determination of K_p, K_i, and λ from specifications (A_m and ϕ_m) and process parameters ($K, \omega_{z_1}, \omega_{z_2}, \ldots, \omega_{z_n}, \omega_{p_1}, \omega_{p_2} \ldots \omega_{p_m} L$) using the above set of equations. The highly nonlinear nature of the equations and the larger number of parameters avoids obtaining a closed-form analytical solution and application of graphical techniques [5]. In this context, the present FOPI controller design task has been framed as an optimization problem that aims at obtaining an approximate solution for Eqs. (8)–(11).

3. Nonlinear adaptive seeker optimization algorithm

3.1. Seeker optimization algorithm

The SOA [15] is a human intelligence-based heuristic search algorithm that operates on a population of position vectors, with each vector (seeker) representing a possible solution to the optimization problem. The algorithm starts with a randomly generated population of seekers, which is grouped into a set of three subpopulations. After initialization, the position of each seeker is iteratively updated as:

$$x_{ij}(k+1) = x_{ij}(k) + \alpha_{ij}(k)\, d_{ij}(k) \tag{12}$$

where k is the iteration count, and $\alpha_{ij}(k)$ and $d_{ij}(k)$ represent the step length and search direction respectively of the jth dimension of the ith seeker with $\alpha_{ij}(k) \geq 0$ and $d_{ij}(k) = \{-1, 0, +1\}$. The search direction is dependent on the trade-off between the seeker's self-belief to follow its own personal best position, cooperative nature to share its information among neighbors, and personal past exploration experience.

A suitable step length along the estimated search direction for each seeker in the population S is obtained as:

$$\mu_i = \mu_{max} - \frac{S-I_i}{S-1}\left(\mu_{max}-\mu_{\min}\right) \tag{13}$$

where I_i is the sequence number of the ith seeker after sorting the fitness in ascending order and μ_i is known as the trust degree, which is proportional to the fitness value of the ith seeker. Its value ranges from $\mu_{max} = 1$ for the global best to $\mu_{\min} = 0.2$ for the worst seeker in the population. The conversion of fuzzy reasoning output to a crisp value provides the step length for the jth dimension of the ith seeker using a Gaussian member function as:

$$\alpha_{ij} = \delta_j\sqrt{-ln\left(rand(\mu_{ij})\right)} \tag{14}$$

with $\mu_{ij} < rand(\mu_{ij}) < 1$ and δ_j is the jth component of the vector δ given as:

$$\delta = \omega_{lin} abs\left(x_{best} - x_{rand}\right) \tag{15}$$

where ω_{lin} is the linearly descending inertia weight, whereas x_{best} and x_{rand} represent the best and a randomly selected seeker in the neighborhood of the ith seeker, respectively.

After updating the population using Eq. (12), the corresponding objective functions are evaluated and hence the positions for personal best, subpopulation best, and overall best are updated. Furthermore, to perform information exchange among the neighborhood, an inter-subpopulation learning strategy is applied, in which the worst two positions of each subpopulation are replaced by the best position of the other subpopulations as:

$$x_{p,j}^{nth\,worst} = \begin{cases} x_{q,j}^{best}; & \text{if } rand_j = 0.5 \\ x_{p,j}^{nth\,worst}; & \text{else} \end{cases} \tag{16}$$

where $rand_j$ represents a uniformly distributed random real number between 0 and 1, $x_{p,j}^{nth\,worst}$ represents the jth dimension of the nth worst position of the pth subpopulation, and $x_{q,j}^{best}$ denotes the jth dimension of the best position of the qth subpopulation. The learning strategy is an essential step for exchanging good information among the subpopulations, leading to diversity of the seeker population.

3.2. Nonlinear adaptive seeker optimization algorithm

Any search-based optimization technique should involve high diversity during the early stage for wider exploration, whereas minor variation is required for fine tuning at the near optimal solution. Based on these concerns, a linearly decreasing inertia weight ω_{lin} is considered in the standard SOA. However, a linear time-varying adaptive weight in heuristic search-based methods has been found to be ineffective for real-world applications. In this context, the linear function has been replaced by an exponential decay function with the aim of faster reduction in the weight, leading to rapid convergence. For exponential decay, the linear function ω_{lin} is replaced as:

$$\omega_{exp}\left(k\right) = e^{-1.1\xi(1-k)} \tag{17}$$

where k represents the iteration count and ξ is the tuning parameter given as:

$$\xi = \frac{log(\omega_{exp,final})}{\text{Number of generations}} \tag{18}$$

Here, $\omega_{exp,final}$ is the predefined final weight value corresponding to the last iteration. With the weight function ω_{lin} in Eq. (15) replaced by ω_{exp} in Eq. (17), the resulting algorithm will henceforth be referred to as exponential SOA (ESOA).

Furthermore, with the aim of increasing the search space in addition to faster convergence, chaotic searching behavior has been incorporated by considering the chaotic evolution of variables for searching the global optimum. For achieving this, the following weight function has been used to replace ω_{lin} in Eq. (15):

$$\omega_{lin,\,chaotic}\left(k\right) = \omega_{lin}\left(k\right)\omega_{chaotic}\left(k\right) \tag{19}$$

The chaotic term $\omega_{chaotic}$ is implemented using the widely used logistic equation in chaos theory [19] as:

$$\omega_{chaotic}\left(k+1\right) = \mu(k)\,\omega_{chaotic}\left(k\right)\left(1 - \omega_{chaotic}\left(k\right)\right) \tag{20}$$

where $\mu(k)$ is the control parameter that determines the extent of the chaos. The initial condition for the above equation is chosen randomly, i.e. $0 \leq \omega_{chaotic}(k) \leq 1$, for $k = 1$ and 2. The chaotic search leads to inclusion of ergodicity, irregularity, and randomness in the SOA, so henceforth the algorithm is referred to as chaotic SOA (CSOA). The flow chart of the proposed CSOA is shown in Figure 2. If the chaotic behavior of Eq. (19) is replaced by the exponentially decaying behavior of Eq. (17) for updating the weighting factor in the flow chart (Figure 2), the same would correspond to the ESOA.

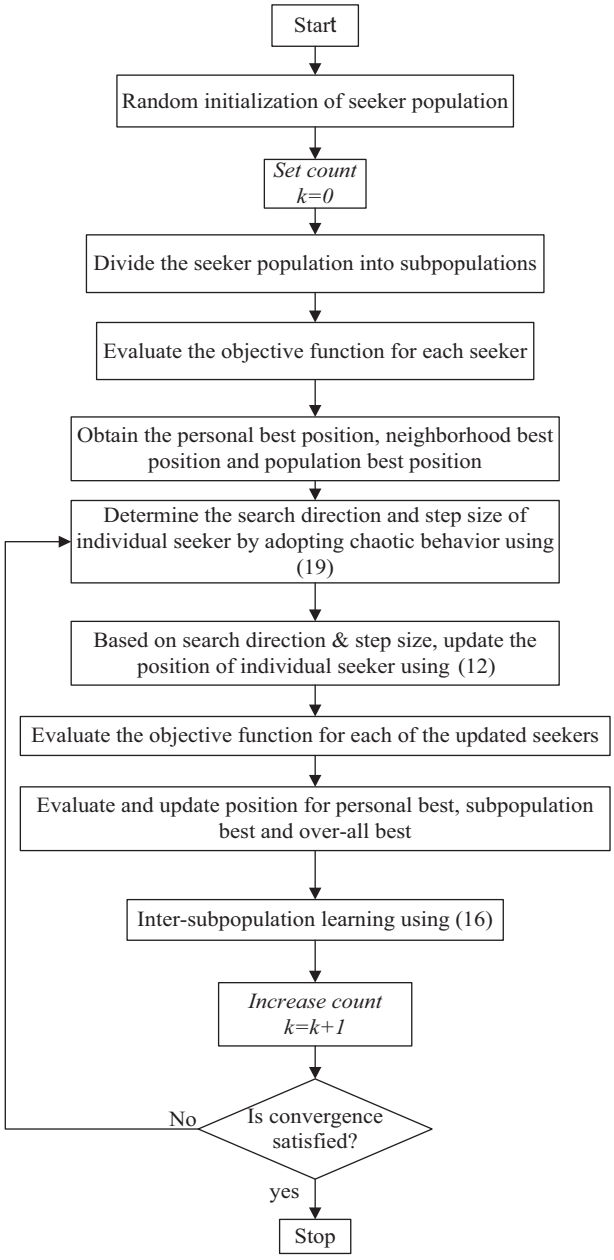


Figure 2. Flow chart of the chaotic seeker optimization algorithm (CSOA).

3.3. Convergence of proposed variants of the SOA

Theorem 1 *The stability and convergence of individual seekers for the ESOA with the exponential weighting function of Eq. (17) are ensured if* $0 < \omega_{exp,final} < 1$.

Proof Consider Eq. (12) as a first-order discrete time system, with $x_{ij}(k)$ as the state variable and $d_{ij}(k)$ as input. Comparing Eq. (12) with the standard state space representation, $X(k+1) = AX(k) + Bu(k)$, we have:

$$\begin{gathered} X(k+1) = x_{ij}(k+1); X(k) = x_{ij}(k) \\ A = 1; B = \alpha_{ij}(k); u(k) = d_{ij}(k) \end{gathered} \tag{21}$$

Using Eq. (21), the deviation in the state variable between successive iterations is given by:

$$\Delta x_{ij}(k) = x_{ij}(k+1) - x_{ij}(k) = \alpha_{ij}(k) d_{ij}(k) \tag{22}$$

The stability of the system and hence the convergence of the state variable at steady state to a finite quantity demands [20]

$$\Delta x_{ij}(k) = 0 \tag{23}$$

for $d_{ij} \in \{-1, 0, 1\}$, and the above condition is achievable with

$$\exp^{-1.1\xi(1-k)} < 1 \ \ \forall k \ \ \& \ \ log(\omega_{exp,final}) < 0, \tag{24}$$

which leads to

$$0 < \omega_{exp,final} < 1. \tag{25}$$

□

Theorem 2 *The stability and the convergence of the individual seeker for the CSOA with chaotic search are guaranteed if* $\omega_{chaotic}(k) > 1 - \frac{1}{\mu}$.

Proof Rewrite Eq. (20) as:

$$\omega_{chaotic}(k+1) = \bar{\mu}(k) . \omega_{chaotic}(k) \tag{26}$$

where $\bar{\mu}(k) = \mu(k)(1 - \omega_{chaotic}(k))$ is an iteration-dependent tuning parameter. The above assumption converts the nonlinear discrete system of Eq. (20) into a set of an N piecewise linear discrete system, with linear dynamics valid over a given iteration only. Using successive iteration with a finite initial condition $\omega_{chaotic}(0)$, the chaotic search component at instant k is given as:

$$\omega_{chaotic}(N) = \prod_{k=0}^{N} \bar{\mu}(k) . \omega_{chaotic}(k) \tag{27}$$

The stability of the piecewise linear autonomous system is ensured if all the linear systems constituting the overall system are stable, i.e. $\bar{\mu}(k) < 1, \ \ \forall k$.

$$\mu\ (1 - \omega_{chaotic}(k)) < 1 \tag{28}$$

$$\omega_{chaotic}(k) > 1 - \frac{1}{\mu} \tag{29}$$

□

4. Experimental results

4.1. Fractional-order controller design

The ability of the proposed variants of the SOA in designing a FO controller that meets a desired set of specifications has been evaluated in this subsection. A first-order-plus-dead-time (FOPDT) plant model has been considered to validate the proposed controller design procedure, as it has been widely used in the literature to model/approximate a large number of higher order industrial plant dynamics and for the validation of control strategies [21]. The plant considered in the present problem is given as:

$$G_p = \frac{e^{-0.1s}}{(1+s)} \tag{30}$$

Initially, all three algorithms (SOA, ESOA, and CSOA) are compared in achieving a wide range of specifications. The mean and standard deviation of the objective function values achieved by 30 independent runs for different sets of gain and phase margin specifications are depicted in Table 1. It can be observed that for all the cases, proposed variants of the SOA are able to minimize the deviations in the desired gain and phase margin and hence the objective function to a significant extent as compared to the classical SOA.

Table 1. Mean and standard deviation of the objective function of the FOPI controlled closed-loop system for the desired set of specifications for 30 independent runs.

GM↓	PM→	30°	50°	80°
3	*SOA*	4.03E-03 ± 8.14E-04	2.72E-03 ± 1.28E-03	3.22E-04 ± 2.33E-04
	ESOA	2.10E-03 ± 8.39E-04	1.81E-03 ± 9.36E-04	2.12E-04 ± 9.60E-05
	CSOA	7.32E-04 ± 3.89E-04	8.07E-04 ± 5.19E-04	1.20E-04 ± 3.79E-05
5	*SOA*	2.24E-03 ± 7.19E-04	2.79E-03 ± 9.99E-04	2.48E-03 ± 1.07E-03
	ESOA	2.04E-03 ± 9.87E-04	1.82E-03 ± 7.24E-04	1.40E-03 ± 5.75E-04
	CSOA	5.42E-04 ± 3.43E-04	6.03E-04 ± 3.71E-04	5.23E-04 ± 3.60E-04
12	*SOA*	2.15E-03 ± 1.15E-03	3.13E-03 ± 1.13E-03	3.80E-03 ± 1.41E-03
	ESOA	1.85E-03 ± 9.47E-04	2.06E-03 ± 1.07E-03	3.28E-03 ± 1.37E-03
	CSOA	5.23E-04 ± 3.11E-04	3.93E-04 ± 2.95E-04	4.76E-04 ± 3.24E-04

Further, a nonparametric test, i.e. the Wilcoxon signed rank test [22], has been carried out to statistically justify the improved performance of the proposed algorithms over the classical SOA. The null hypothesis in the test states that two samples under test belong to the same population and there exists no difference in the behavior of two algorithms. Table 2 depicts the corresponding P-values associated with the test conducted at a significance level of 5% (i.e. 0.05) for both the proposed algorithms over the standard SOA for 30 independent runs. A value of $P < 0.05$ for most of the cases indicates rejection of the null hypothesis. In other words, it reflects that the superior performance of the CSOA and ESOA over the conventional SOA is statistically significant and has not been achieved by chance.

The appropriateness of the proposed variants of the SOA in designing the FOPI controller is also compared with other global optimization techniques (DE [11] and CMAES [12]) and popular classical local search techniques (Nelder–Mead simplex search (NM-SS) [9] and interior point method (IPM) [10]) in Table 3. It can be observed that the CSOA outperforms the other algorithms under consideration, while the ESOA is better than the other algorithms except for the CSOA and is comparable to DE.

The effectiveness of the proposed FOPI controller design technique as compared to a previously reported technique [14] in terms of achieving desired gain and phase margin specifications is exhibited in Table 4.

Table 2. P-values obtained by Wilcoxon signed rank test for objective function values obtained using proposed variants with respect to the conventional SOA.

GM↓	*ESOA Vs SOA*			*CSOA Vs SOA*		
PM→	30°	50°	80°	30°	50°	80°
3	1.73E-06	2.18E-02	2.06E-02	1.73E-06	1.36E-05	2.37E-05
5	3.82E-01	1.36E-04	7.51E-05	3.18E-06	1.73E-06	1.73E-06
12	3.09E-01	7.71E-04	1.31E-01	3.41E-05	1.73E-06	1.73E-06

Table 3. Mean of the objective function of FOPI controlled closed-loop system, obtained using different algorithms for 30 independent runs with the desired set of specifications.

Specifications		Algorithms					
GM	PM	IPM [10]	NMSS [9]	DE [11]	CMAES [12]	ESOA	CSOA
3	30°	6.02E-01	6.90E-01	1.10E-03	1.16E-01	2.10E-03	7.32E-04
5	50°	6.29E-01	4.57E-01	9.96E-03	1.86E-01	1.82E-03	6.03E-04
12	80°	7.37E-01	1.91E+00	1.50E-03	2.19E-01	3.28E-03	4.76E-04

To the best of the authors' knowledge, the FOID design technique proposed in [14] is the only reported work on optimization-based FO controller design using the gain and phase margin. The noninclusion of any approximation for nonlinear functions ensures the low deviation in achieving gain and phase margin specifications with the proposed technique.

Table 4. Comparison of the proposed FOPI controller with the FOID controller for achieving desired gain and phase margin.

Specifications		Proposed FOPI					FOID [14]			
		Controller parameters			Deviation		Controller parameters		Deviation	
GM	*PM*	K_p	K_i	λ	*GM*	*PM*	$K_i = K_d$	$\alpha = 1 - \beta$	*GM*	*PM*
3	30°	5.19	5.47	1.27	2.8E-4	1.8E-3	7.71	1.38	4.1E-5	9.3E-5
3	80°	5.49	7.88	2.53	1.8E-5	2.4E-4	3.54	0.80	2.5E-5	2.8E-4
5	50°	3.21	2.75	1.26	9.9E-6	3.1E-3	4.32	1.23	1.6E-5	1.3E-4
12	30°	1.35	3.14	1.33	4.2E-5	1.8E-3	1.64	1.57	1.6E-4	7.7E-5
12	80°	1.36	0.79	1.29	5.4E-5	2.1E-5	1.37	1.02	1.3E-4	8.2E-5

4.2. DSP implementation of the FOPI controller

To experimentally validate the proposed controller design scheme, the designed FOPI controllers have been implemented on a digital platform using the TMS320F2812 DSP board. The Texas Instruments Inc. DSP chip is a 32-bit fixed-point high-end processor, which executes instructions at a rate of 150 MIPS. The processor has been interfaced with a host PC running with MATLAB 2010a and Code Composer Studio (CCS) Integrated Development Environment (IDE) v3.3 software. The embedded IDE link and the fixed-point tool of Simulink are used to generate a CCSv3.3 compatible code, which is built in the DSP. The FOPI controller and time delay plant (30) have been executed in closed loop for step input. The experimental set-up is shown in Figure 3.

For the digital implementation of the FOPI controller, discretization is a key step, which approximates the continuous-time FO system into a discrete-time system of higher order [23]. In the present work, a direct discretization scheme, i.e. an impulse-response-invariant method (also used in [5]) with fifth order has been

used to approximate the FO operator, while the plant has been discretized using the Tustin rule. A sampling rate of 0.01 s has been selected based on the Nyquist sampling theorem, stability in discrete domain, and the hardware limitations. Figure 4 displays the simulated and experimental step response of the FOPI controlled closed-loop system for gain margin = 5 and phase margin = 50°. The lesser deviation between the simulated and experimental response is attributed to the effectiveness of the employed discretization technique. The closed-loop step response of the system with FOPI controllers designed for different sets of gain and phase margin specifications is depicted in Figures 5a, 5b, and 5c. The transient responses show the variation in the degree of relative stability for different gain and phase margin requirements. The variation is because of the difference in damping achieved for different cases, i.e. high damping (low overshoot) is observed for high phase margin specification and vice versa.

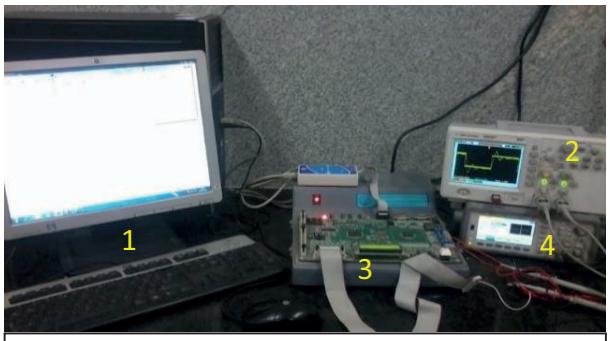


Figure 3. Experimental set-up.

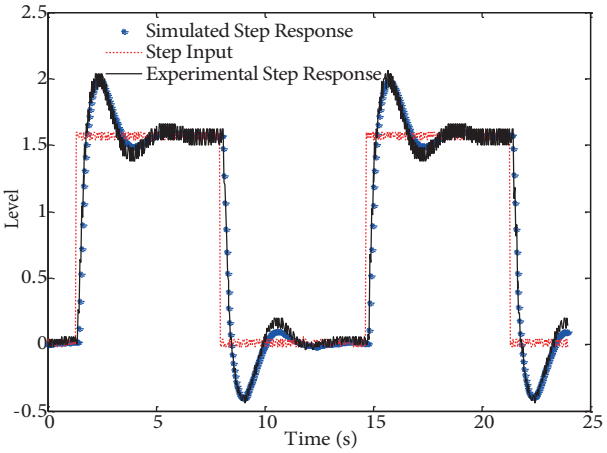
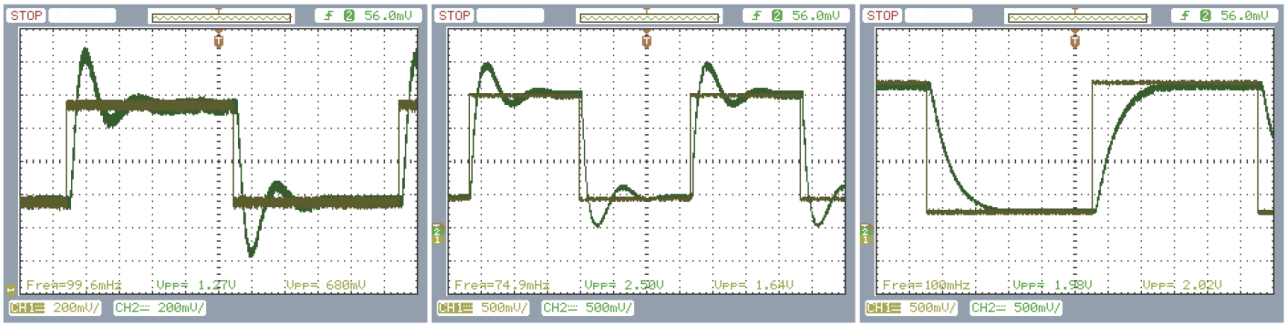


Figure 4. Comparison of simulated (continuous domain) and experimental (discrete domain) step response of controlled system for GM = 5; PM = 50°.



(a) GM = 3; PM = 30° (b) GM = 5; PM = 50° (c) GM = 12; PM = 80°

Figure 5. Step response of FOPI controlled system for different specifications: a) GM = 3, PM = 30°; b) GM = 5, PM = 50°; GM = 12, PM = 80°.

5. Conclusions

An optimization-based methodology has been proposed for the design of FO controllers based on gain margin and phase margin. The high nonlinearity and larger number of parameters in the design problem hinder the use

of classical gradient-based techniques. Hence, for solving the FO controller design as an optimization problem, two nonlinear adaptive variants of the conventional SOA have been developed. With the aim of overcoming the limitations of the original SOA, modifications have been carried out by incorporating exponential weighting and chaotic search behavior in the solution updating process. The superiority of the proposed FO controller design scheme has been justified by a comprehensive comparison with other design methods and optimization techniques. The effectiveness of the proposed controller design technique has been validated by implementing the FOPI controller in the digital domain on the TMS320F2812 DSP board. Both the simulation and experimental results validate the ability of the controller design technique in stabilizing the system and meeting a wide range of robustness specifications. Future work in this direction is planned on extending the proposed FOPI controller design technique for the fractional order plants.

References

[1] Chen YQ, Petráš I, Xue D. Fractional order control - a tutorial. In: 2009 American Control Conference; 10–12 June 2009; St. Louis, USA. New York, NY, USA: IEEE. pp. 1397-1411.

[2] Shah P, Agashe S. Review of fractional PID controllers. Mechatronics 2016; 38: 29-41.

[3] Machado JT, Galhano AM, Trujillo JJ. On development of fractional calculus during the last fifty years. Scientometrics 2014; 98: 577-582.

[4] Chen YQ, Xue D, Visioli A. Guest editorial for special issue on fractional order systems and controls. IEEE/CAA J Autom Sin 2016; 3: 255-256.

[5] Luo Y, Zhang Y, Lee BJ, Kang C, Chen YQ. Fractional order proportional derivative controller synthesis and implementation of hard-disk-drive servo system. IEEE T Control Syst T 2014; 22: 281-289.

[6] Monje CA, Vinagre BM, Feliu V, Chen YQ. Tuning and auto-tuning of fractional order controllers for industry applications. Control Eng Pract 2008; 16: 798-812.

[7] Roy P, Kar B, Roy BK. Fractional order PI-PD control of liquid level in coupled two tank system and its experimental validation. Asian J Control (in press).

[8] Chopade AS, Khubalkar SW, Junghare AS, Aware MV, Das S. Design and implementation of digital fractional order PID controller using optimal pole-zero approximation method for magnetic levitation system. IEEE/CAA J Autom Sin 2016; 99: 1-12.

[9] Das S, Saha S, Das S, Gupta A. On the selection of tuning methodology of FOPID controllers for the control of higher order processes. ISA Trans 2011; 50: 376-388.

[10] Kehsarkar AA, Selvaganesan N, Priyadarshan H. A novel framework to design and compare limit cycle minimizing controllers: demonstration with integer and fractional-order controllers. Nonlinear Dyn 2014; 78: 2871-2882.

[11] Biswas A, Das S, Abraham A, Dasgupta S. Design of fractional-order $PI^{\lambda}D^{\mu}$ controllers with an improved differential evolution. Eng Appl Artif Intell 2009; 22: 343-350.

[12] Sivananaithaperumal S, Baskar S. Design of multivariable fractional order PID controller using covariance matrix adaptation evolution strategy. Arch Control Sci 2014; 24: 235-251.

[13] Pan I, Das S. Frequency domain design of fractional order PID controller for AVR system using chaotic multi-objective optimization. Int J Electr Power Energy Syst 2013; 51: 106-118.

[14] Ahn H, Bhambhani V, Chen YQ. Fractional-order integral and derivative controller for temperature profile tracking. Sadhana 2009; 34: 833-850.

[15] Dai C, Zhu Y, Chen W. Seeker optimization algorithm. In: Wang Y, Cheung Y, Liu H, editors. Computational Intelligence and Security. Berlin, Germany: Springer, 2007. pp. 167-176.

[16] Zhu Y, Dai C, Chen W. Seeker optimization algorithm for several practical applications. Int J Comput Intell Syst 2014; 7: 353-359.

[17] Kumar MR, Ghosh S, Das S. Identification of fractional order circuits from frequency response data using seeker optimization algorithm. In: IEEE 2015 International Conference on Industrial Instrumentation and Control; 28–30 May 2015; Pune, India. New York, NY, USA: IEEE. pp. 197-202.

[18] Tuba M, Brajevic I, Jovanovic R. Hybrid seeker optimization algorithm for global optimization. Appl Math Inf Sci 2013; 7: 867-875.

[19] May RM. Simple mathematical models with very complicated dynamics. Nature 1976; 261: 459-467.

[20] Ogata K. Discrete-Time Control Systems. 2nd ed. Upper Saddle River, NJ, USA: Prentice Hall, 1995.

[21] Sánchez HS, Padula F, Visioli A, Vilanova R. Tuning rules for robust FOPID controllers based on multi-objective optimization with FOPDT models. ISA Trans 2017; 66: 344-361.

[22] Derrac J, García S, Molina D, Herrera F. A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm Evol Comput 2011; 1: 3-18.

[23] Monje CA, Chen Y, Vinagre BM, Xue D, Feliu V. Fractional-Order Systems and Controls: Fundamentals and Applications. New York, NY, USA: Springer, 2010.