Optimal Nonlinear System Identification Using Fractional Delay Second-Order Volterra System

Manjeet Kumar, Apoorva Aggarwal, Tarun Rawat and Harish Parthasarathy

***Abstract*—The aim of this work is to design a fractional delay second order Volterra filter that takes a discrete time sequence as input and its output is as close as possible to the output of a given nonlinear unknown system which may have higher degree nonlinearities in the least square sense. The basic reason for such a design is that rather than including higher than second degree nonlinearities in the designed system, we use the fractional delay degrees of freedom to approximate the given system. The advantage is in terms of obtaining a better approximation of the given nonlinear system than is possible by using only integer delays (since we are giving more degrees of freedom via the fractional delays) and simultaneously it does not require to incorporate higher degree nonlinearities than two. This work hinges around the fact that if the input signal is a decimated version of another signal by a factor of M, then fractional delays can be regarded as delays by integers less than M. Using the well known formula for calculating the discrete time Fourier transform (DTFT) of a decimated signal, we then arrive at an expression for the DTFT of the output of a fractional delay system in terms of the unknown first and second order Volterra system coefficients and the fractional delays. The final energy function to be minimized is the norm square of the difference between the DTFT of the given output and the DTFT of the output of the fractional delay system. Minimization over the filter coefficients is a linear problem and thus the final problem is to minimize a highly nonlinear function of the fractional delays which is accomplished using search techniques like the gradient-search and nature inspired optimization algorithms. The effectiveness of the proposed method is demonstrated using two nonlinear benchmark systems tested with five different input signals. The accuracy of the stated models using the globally convergent metaheuristic, cuckoo-search algorithm (CSA) are observed to be superior when compared with other techniques such as real-coded genetic algorithm (RGA), particle swarm optimization (PSO) and gradient-search (GS) methods. Finally, statistical analysis affirms the potential of the proposed designs for its successful implementation.**

***Index Terms*—Fractional delay, second-order Volterra system, gradient-search method, stochastic search algorithm, mean square error**

I. Introduction

THE modeling of unknown systems is of significant importance in different fields of engineering[1]. Various linear systems have been utilized owing to the simplicity in solving the system identification problems and in developing different signal processing techniques[2−8]. Such linear systems have been extensively applied with the comprehensive mathematical analysis and simplified simulations. However, most of the practical systems exhibit nonlinear behaviour due to which the estimation using linear systems is not accurate. To state some, in estimating the saturation-type nonlinear systems[9], development of nonlinear behaviour due to brake sequel conditions in automotive industry[10], identification of nonlinear dynamical structures[11], using linear models can often give corrupt results. The application of nonlinear systems have been extensively researched by practitioners in various science and engineering fields such as in communication engineering, signal processing, biomedical engineering and system identification[12]. Some typical applications[13] in communication systems include amplifier nonlinearities, nonlinear satellite channel, compensation of nonlinearities, equalization of nonlinear channels, blind identification, nonlinearities in orthogonal frequency division multiplexing systems and digital magnetic recording. In speech and image processing, the nonlinear systems are employed for the compensation of loudspeaker nonlinearities, in adaptive quadratic filters, nonlinear echoes cancelation and many more.

In the past, much research has been carried out for estimating practical systems using a variety of nonlinear systems based on different models. These nonlinear models and functions include Volterra and Wiener series[14,15], Hammerstein model[16], Walsh functions[17], Kautz models[18], Laguerre transform[19], Uryson model[20] and neural networks[21] etc. The aforementioned models have been substantially implemented in nonlinear system identification problems. Conventionally, the modeling of unknown systems was practiced using the gradient based search methods. Based on the successful implementation of metaheuristic algorithms in the system identification problems, the trend has been shifted towards the use of these algorithms. In [14], Chang efficiently utilized the improved particle swarm optimization algorithm for the different memory size Volterra filter models of nonlinear discrete-time systems. The implementation of the gravitational search algorithm for the nonlinear and linear system identification problem was proposed by Rashedi et al. in [22]. Gotmare et al. applied the CSA for the improvement of nonlinear system identification of adaptive Hammerstein model[16].

The above referred techniques implemented the concept of integer delays to obtain a nonlinear system with significantly accurate estimations. In this paper, we propose to model a highly nonlinear system with quadratic, cubic and even higher order nonlinearities in the presence of noise using a fractional

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

This article was recommended by Associate Editor Antonio Visioli.

M. Kumar, A. Aggarwal, T. Rawat and H. Parthasarathy are with the Department of Electronics and Communication Engineering Division, Netaji Subhas Institute of Technology, Sector-3, Dwarka, New Delhi 110078, India (e-mail: manjeetchhillar@gmail.com, 16.apoorva@gmail.com, tarundsp@gmail.com, harisignal@yahoo.com).

Digital Object Identifier 10.1109/JAS.2016.7510184

delay second order Volterra nonlinear system. The input-output equation for such an approximating system is the usual relation for a system involving an FIR linear system and an FIR second order Volterra system but with fractional delays. Both the continuous time and the discrete time models have been addressed. The fractional delay Volterra system is an LIP (Linear in parameters) model as far as the filter coefficients are concerned, but it is an NLIP (nonlinear in the parameters) model as far as the fractional delays are concerned. Thus, using the standard least squares algorithm, the first and second order filter coefficient estimates from input-output data can be obtained using standard orthogonal projection theory, but with the orthogonal projection being a highly nonlinear function of the fractional delays. By substituting this expression for the filter coefficient estimation into the original least squares energy function, we obtain an energy function that is a nonlinear function of the time delays but not involving the filter coefficients. Then, a search algorithm is used to minimize this energy function w.r.t. the fractional delays and hence obtain good estimates for the latter. The computation has been carried out entirely in the frequency domain because time delays appear as exponentials which multiplies with the Fourier transform of the signals. These exponentials can be represented as steering vectors which depend on the fractional time delays and elegant expressions for the energy function in terms of these steering vectors can be derived. If however, we work in the time domain, then the fractional delays appear inside the time argument of the signals involved and hence optimization algorithms are impossible to carry out. For practical implementation using MATLAB the signals must be discrete time and we have formulated this discrete time version by representing the input signal as a decimated version of the original input by an integer factor of $M > 1$ and the fractional delays by integers in the range $0, 1, \dots, M-1$. The simulation results show that it is possible to approximate complicated nonlinear systems like the ratio of two nonlinear Volterra systems using this second order system involving fractional delays. The advantage of the proposed approach is that no extra filter coefficient energy is required. Indeed, fractional delays do not change the signal energy, they merely shift the signal and superpose. Here, we identify the parameters of a fractional delay second-order Volterra system from input data. This model gives a more accurate system identification with fewer filter coefficients, especially for nonlinear systems like multipath systems with interaction between the different paths shown in Fig. 1. Further, the gradient-search (GS) and stochastic-search approaches are employed to obtain a close approximation of the unknown nonlinear systems. The optimization algorithms utilized are, real-coded genetic algorithm (RGA), particle swarm optimization (PSO) and cuckoo-search algorithm (CSA). The results and analysis presented, demonstrate high accuracy using the proposed design methods.

The paper is organized in 6 sections. Following the literature survey in Section I, the nonlinear system identification problem is modeled as a second order Volterra system using fractional delays in Section II. Section III presents the gradient-search optimization technique articulated for the Volterra system identification problem. A brief overview of stochastic algorithms for the formulated problem is discussed in Section IV. In Section V, two design examples are illustrated and analyzed for different input signals. Finally, Section VI concludes the paper.

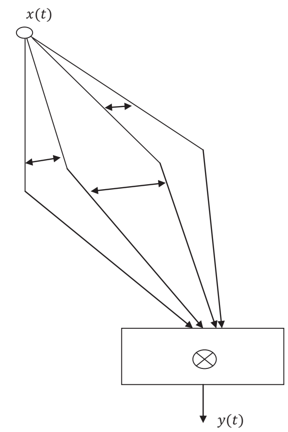

Fig. 1. Multipath system with interaction between the different paths.

II. Volterra System Modeling using Fractional Delay

Suppose $y_d(t)$ is the desired nonlinear system output and it is well approximated using a third order Volterra system with p integer delays, given by

$$\begin{aligned} y(t) = &\sum_{k=0}^{p} h(k)x(t-k\Delta) \\ &+ \sum_{k,m=0}^{p} g(k,m)x(t-k\Delta)x(t-m\Delta) \\ &+ \sum_{k,m,r=0}^{p} f(k,m,r)x(t-k\Delta)x(t-m\Delta)x(t-r\Delta) \end{aligned} \tag{1}$$

where, $x(t)$, $y(t)$ are the input and corresponding output of the Volterra system, $\{h(k)\}$ are the first order kernels of the linear system response with integer delays, $k\Delta$ and $\{g(k,m)\}$ are the second order kernels associated with the nonlinear system response with integer delays, $k\Delta, m\Delta$ and $\{f(k,m,r)\}$ are the third order kernels associated with the nonlinear system response with integer delays, $k\Delta, m\Delta$ and $r\Delta$.

To implement this filter, we require $\mathcal{O}(p^3)$ multiplications and further, the right hand side of the above expression is modeled as

$$M_0 \sum_{k=0}^{p} |h(k)| + M_0^2 \sum_{k,m=0}^{p} |g(k,m)| + M_0^3 \sum_{k,m,r=0}^{p} |f(k,m,r)| \tag{2}$$

where, $M_0 = \max_t |x(t)|$.

In this system identification problem, the aim is to estimate the filter coefficients of a second order Volterra system modeled using the fractional delays, such that it matches the response of an unknown system with higher order nonlinearities. In Fig. 2, this concept is demonstrated by applying the gradient-search and stochastic optimization algorithms. The Volterra system mathematically models the linear and nonlinear combinations of its input signal using the infinite

Volterra series expansion in the form of convolution integrals. The second order Volterra system can be expressed as [23]

$$y(t) = h(0) + \sum_{k=1}^{p} h(k)x(t-\tau_k) + \sum_{k,m=0}^{p} g(k,m)x(t-\tau_k)x(t-\tau_m) \tag{3}$$

where, $h(0)$ is the constant kernel, $\{h(k)\}$ are the first order kernels of the linear system response with fractional delays, τ_k and $\{g(k,m)\}$ are the second order kernels associated with the nonlinear system response with fractional delays, τ_m.

Here, τ_k is varied in addition to the $\{h(k)\}$ and $\{g(k,m)\}$, to get an equally good output match, with $\mathcal{O}(p^2)$ multiplications. The right hand side of eq. (3) is modeled as

$$M_0 \sum_{k=0}^{p} |h(k)| + M_0^2 \sum_{k,m=0}^{p} |g(k,m)| \tag{4}$$

which is likely to be much smaller than eq. (2). Thus, by spending less energy and fewer multiplications, we are able to obtain nearly the same output error.

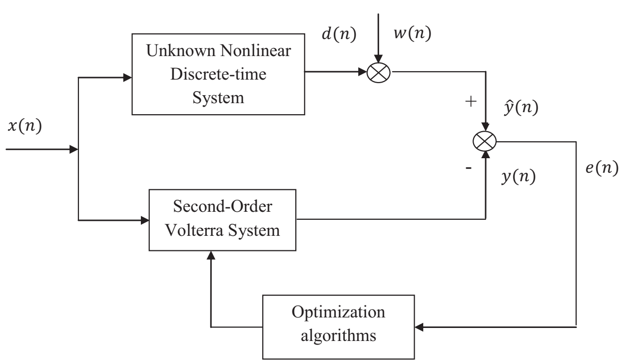

Fig. 2. Volterra System Modeling of nonlinear system using gradient search, RGA, PSO and CSA.

The objective is to optimize the parameters, $\{\tau_k\}$, $\{h(k)\}$ and $\{g(k,m)\}$, such that $\tau_k \in [k\Delta, (k+1)\Delta)$, $0 \le k,\ m \le p$ and

$$\begin{aligned}\xi(\mathbf{h},\mathbf{g},\tau) &= \int_0^{\mathrm{T}} \big(y_d(t) - y(t)\big)^2 \mathrm{d}t \\ &= \int_0^{\mathrm{T}} \Bigg(y_d(t) - \sum_{k=1}^{p} h(k)x(t-\tau_k) \\ &\quad - \sum_{k,m=0}^{p} g(k,m)x(t-\tau_k)x(t-\tau_m)\Bigg)^2 \mathrm{d}t \end{aligned} \tag{5}$$

is minimum. Here, $y_d(t)$ is the desired output. Let $\tau = \{\tau_k\}_{k=0}^{p}$, $\mathbf{h} = \{h(k)\}_{k=0}^{p}$, $\mathbf{g} = \mathrm{vec}(g(k,m))$ and

$$\zeta(t,\tau) = \begin{bmatrix} x(t-\tau_1) \\ x(t-\tau_2) \\ \vdots \\ x(t-\tau_p) \end{bmatrix} \tag{6}$$

then,

$$\begin{aligned}\xi(\mathbf{h},\mathbf{g},\tau) = \int_0^{\mathrm{T}} & (y(t) - \mathbf{h}^{\mathrm{T}}\zeta(t,\tau) \\ & - \mathbf{g}^{\mathrm{T}}(\zeta(t,\tau)\otimes\zeta(t,\tau)))^2 \mathrm{d}t\end{aligned} \tag{7}$$

where $\zeta(t,\tau)\otimes\zeta(t,\tau) = \mathrm{vec}(x(t-\tau_\alpha)x(t-\tau_\beta))$, $1 \le \alpha, \beta \le p$. The optimal equations are

$$\frac{\partial \xi}{\partial \mathbf{h}} = 0,\ \frac{\partial \xi}{\partial \mathbf{g}} = 0,\ \frac{\partial \xi}{\partial \tau} = 0 \tag{8}$$

Calculating the first two terms, we get

$$\begin{aligned}\int_0^{\mathrm{T}} \zeta(t,\tau)y(t)\mathrm{d}t = & \left(\int_0^{\mathrm{T}} \zeta(t,\tau)\zeta(t,\tau)^{\mathrm{T}}\mathrm{d}t\right)\mathbf{h} \\ & + \left(\int_0^{\mathrm{T}} \zeta(t,\tau)(\zeta(t,\tau)\otimes\zeta(t,\tau))\mathrm{d}t\right)\mathbf{g}\end{aligned} \tag{9}$$

Calculating the third term, we obtain

$$\begin{aligned}&\int_0^{\mathrm{T}} (\zeta(t,\tau)\otimes\zeta(t,\tau))y(t)\mathrm{d}t \\ &= \left(\int_0^{\mathrm{T}} (\zeta(t,\tau)\otimes\zeta(t,\tau))(\zeta(t,\tau))^{\mathrm{T}}\mathrm{d}t\right)\mathbf{h} \\ &+ \left(\int_0^{\mathrm{T}} (\zeta(t,\tau)\otimes\zeta(t,\tau))(\zeta(t,\tau)\otimes\zeta(t,\tau))^{\mathrm{T}}\mathrm{d}t\right)\mathbf{g}\end{aligned} \tag{10}$$

Defining the nonlinear filter vector

$$\mathbf{k} = \begin{bmatrix} \mathbf{h} \\ \mathbf{g} \end{bmatrix} \in R^{p+p^2} \tag{11}$$

and the $(p+p^2)\times(p+p^2)$ matrix is given by eq. (12). Also define

$$\mathbf{b}(\tau) = \begin{bmatrix} \int_0^{\mathrm{T}} \zeta(t,\tau)y(t)\mathrm{d}t \\ \int_0^{\mathrm{T}} (\zeta(t,\tau)\otimes\zeta(t,\tau))y(t)\mathrm{d}t \end{bmatrix} \in R^{p+p^2} \tag{13}$$

Then, the optimal equations for $\mathbf{k} = [\mathbf{h}^{\mathrm{T}}, \mathbf{g}^{\mathrm{T}}]^{\mathrm{T}}$ are solved as

$$\hat{\mathbf{k}}(\tau) = \mathbf{A}(\tau)^{-1}\mathbf{b}(\tau) = \begin{bmatrix} \hat{\mathbf{h}}(\tau) \\ \hat{\mathbf{g}}(\tau) \end{bmatrix} \tag{14}$$

Further, τ is extended as

$$\hat{\tau} = \arg\min_{\tau} \xi(\hat{\mathbf{h}}(\tau), \hat{\mathbf{g}}(\tau), \tau) \tag{15}$$

$$= \arg\min_{\tau} \xi(\hat{\mathbf{k}}(\tau), \tau) \tag{16}$$

Now

$$\xi(\hat{\mathbf{k}}(\tau),\tau) = \left[\int_0^{\mathrm{T}} y_d^2(t)\mathrm{d}t - \hat{\mathbf{k}}(\tau)^{\mathrm{T}}\mathbf{b}(\tau)\right] \tag{17}$$

$$= [\sigma_y^2 - \mathbf{b}(\tau)^{\mathrm{T}}\mathbf{A}(\tau)\mathbf{b}(\tau)] \tag{18}$$

So the optimal fractional delays are

$$\hat{\tau} = \arg\min_{\tau} \mathbf{b}(\tau)^{\mathrm{T}}\mathbf{A}(\tau)\mathbf{b}(\tau) \tag{19}$$

The proposed method can be applied to better equalization of nonlinear channels with random delays, for better forecasting of system and better system identification. Less power loss is there since loss depends on the number of coefficients and

$$\mathbf{A}(\tau)=\left[\begin{array}{cc}\int_0^{\mathrm{T}}\zeta(t,\tau)\zeta(t,\tau)^{\mathrm{T}}\mathrm{d}t & \int_0^{\mathrm{T}}\zeta(t,\tau)(\zeta(t,\tau)\otimes\zeta(t,\tau))\mathrm{d}t\\ \int_0^{\mathrm{T}}(\zeta(t,\tau)\otimes\zeta(t,\tau))(\zeta(t,\tau))^{\mathrm{T}}\mathrm{d}t & \int_0^{\mathrm{T}}(\zeta(t,\tau)\otimes\zeta(t,\tau))(\zeta(t,\tau)\otimes\zeta(t,\tau))^{\mathrm{T}}\mathrm{d}t\end{array}\right] \tag{12}$$

not on the delay given to each one. Moreover, the Volterra fractional delay system can be made adaptive, resulting in better adaptive noise cancelation, when the noise is generated from nonlinearities with delays like hysteresis. The optimal values of these fractional delays and Volterra kernels of first and second order are computed using the gradient-search and metaheuristic algorithms, described in the following section.

III. Gradient Search Method

This section focusses on the implementation of the gradient-search method to approximate the response of the unknown nonlinear system. This optimization is carried out using a gradient descent approach explained as follows.

$$\zeta(t,\tau)=\left(\int_R X(\omega)e^{j\omega(t-\tau_k)}\mathrm{d}\omega\right)_{k=1}^{p} \tag{20}$$

where, $X(\omega)$ is the DTFT of input signal $x(t)$ and $R\in(0,T)$. Now,

$$\begin{aligned}&\zeta(t,\tau)\otimes\zeta(t,\tau)\\ =&\mathrm{vec}\Bigg(\int_R X(\omega_1)X(\omega_2)\\ &\times e^{j(\omega_1+\omega_2)t}e^{j(\omega_1\tau_k+\omega_2\tau_m)}\mathrm{d}\omega_1\mathrm{d}\omega_2\Bigg)_{k,m=0}^{p}\end{aligned} \tag{21}$$

Substituting eqs. (20) and (21) in eq. (13), we get (22) at the top of next page. The derivative of $\mathbf{b}(\tau)$ in eq. (22) w.r.t. the fractional delays, τ_k is expressed as (23) at the top of next page, where

$$\mathbf{e}_k=\left[\begin{array}{c}0\\0\\\vdots\\0\\1(\mathrm{k}^{th}\mathrm{row})\\0\\\vdots\\0\end{array}\right] \tag{24}$$

and $\frac{\partial\mathbf{A}(\tau)}{\partial\tau_k}$ can be calculated using eq. (20), we get

$$\begin{aligned}\frac{\partial\zeta(t,\tau)}{\partial\tau_k}&=\frac{\partial}{\partial\tau_k}\left(\int X(\omega)e^{j\omega(t-\tau_m)}\mathrm{d}\omega\right)_{m=0}^{p}\\ &=\left(-j\int\omega X(\omega)e^{j\omega(t-\tau_k)}\mathrm{d}\omega\right)\mathbf{e}_k\end{aligned} \tag{25}$$

$$\frac{\partial\mathbf{A}(\tau)}{\partial\tau_k}=\left[\begin{array}{cc}a_{11}&a_{12}\\a_{21}&a_{22}\end{array}\right] \tag{26}$$

where

$$a_{11}=\int_0^{\mathrm{T}}\left(\frac{\partial\zeta(t,\tau)}{\partial\tau_k}\zeta(t,\tau)^{\mathrm{T}})+\zeta(t,\tau)\left(\frac{\partial\zeta(t,\tau)}{\partial\tau_k}\right)^{\mathrm{T}}\right)\mathrm{d}t$$

$$\begin{aligned}a_{12}=\int_0^{\mathrm{T}}&\left(\frac{\partial\zeta(t,\tau)}{\partial\tau_k}\left(\zeta(t,\tau)\otimes\zeta(t,\tau)\right)^{\mathrm{T}}\right.\\ &+\zeta(t,\tau)\left(\frac{\partial\zeta(t,\tau)}{\partial\tau_k}\otimes\zeta(t,\tau)\right)^{\mathrm{T}}\\ &\left.+\zeta(t,\tau)\left(\zeta(t,\tau)\otimes\frac{\partial\zeta(t,\tau)}{\partial\tau_k}\right)^{\mathrm{T}}\right)\mathrm{d}t\\ a_{21}=\int_0^{\mathrm{T}}&\left(\left(\frac{\partial\zeta(t,\tau)}{\partial\tau_k}\otimes\zeta(t,\tau)\right)\left(\zeta(t,\tau)\right)^{\mathrm{T}}\right.\\ &+\left(\zeta(t,\tau)\otimes\frac{\partial\zeta(t,\tau)}{\partial\tau_k}\right)\left(\zeta(t,\tau)\right)^{\mathrm{T}}\\ &\left.+\left(\zeta(t,\tau)\otimes\zeta(t,\tau)\right)\left(\frac{\partial\zeta(t,\tau)}{\partial\tau_k}\right)^{\mathrm{T}}\right)\mathrm{d}t\\ a_{22}=\int_0^{\mathrm{T}}&\left(\left(\frac{\partial\zeta(t,\tau)}{\partial\tau_k}\otimes\zeta(t,\tau)\right)(\zeta(t,\tau)\otimes\zeta(t,\tau))^{\mathrm{T}}\right.\\ &+\left(\zeta(t,\tau)\otimes\frac{\partial\zeta(t,\tau)}{\partial\tau_k}\right)(\zeta(t,\tau)\otimes\zeta(t,\tau))^{\mathrm{T}}\\ &+(\zeta(t,\tau)\otimes\zeta(t,\tau))\left(\frac{\partial\zeta(t,\tau)}{\partial\tau_k}\otimes\zeta(t,\tau)\right)^{\mathrm{T}}\\ &\left.+(\zeta(t,\tau)\otimes\zeta(t,\tau))\left(\zeta(t,\tau)\otimes\frac{\partial\zeta(t,\tau)}{\partial\tau_k}\right)^{\mathrm{T}}\right)\mathrm{d}t\end{aligned}$$

Now,

$$F(\tau)=\mathbf{b}(\tau)^{\mathrm{T}}\mathbf{A}(\tau)^{-1}\mathbf{b}(\tau) \tag{27}$$

The designed system can be formulated using the above equations with

$$\tau_k[m+1]=\tau_k[m]-\mu\frac{\partial}{\partial\tau_k[m]}\left(\mathbf{b}(\tau[m])^{\mathrm{T}}\mathbf{A}(\tau[m])^{-1}\mathbf{b}(\tau[m])\right) \tag{28}$$

Eq. (28) updates the gradient-search algorithm for the fractional delay values.

IV. Stochastic Search Algorithms

The stochastic search algorithms are proven to produce optimal solutions to the complex problems in a reasonably practical time. These algorithms are characterized as heuristic, adaptive and learning with which they produce effective optimizations. Genetic algorithm, particle swarm optimization and cuckoo-search algorithm are population based, since they use a set of strings, particles and host nest, respectively to obtain the solution which are globally optimal. Further, these algorithms are briefly reviewed in this section.

$$\mathbf{b}(\tau) = \begin{bmatrix} \left(\int X(\omega)\overline{Y(\omega)}e^{-j\omega\tau_k}\mathrm{d}\omega\right)_{k=0}^{p} \\ \mathrm{vec}\left(\int X(\omega_1)X(\omega_2)\overline{Y(\omega_1+\omega_2)}e^{-j(\omega_1\tau_k+\omega_2\tau_m)}\mathrm{d}\omega_1\mathrm{d}\omega_2\right)_{k,m=0}^{p} \end{bmatrix} \tag{22}$$

$$\frac{\partial \mathbf{b}(\tau)}{\partial \tau_{\mathbf{k}}} = \begin{bmatrix} \left(-j\int \omega X(\omega)\overline{Y(\omega)}e^{-j\omega\tau_k}\mathrm{d}\omega\right)\mathbf{e}_k \\ \left(-j\int \omega_2 X(\omega_1)X(\omega_2)\overline{Y(\omega_1+\omega_2)}e^{-j(\omega_1\tau_m)}\mathrm{d}\omega_1\mathrm{d}\omega_2\right)_{m=0}^{p}\otimes\mathbf{e}_k \\ +\mathbf{e}_k\otimes\left(-j\int \omega_2 X(\omega_1)X(\omega_2)\overline{Y(\omega_1+\omega_2)}e^{-j(\omega_1\tau_m)}\mathrm{d}\omega_1\mathrm{d}\omega_2\right)_{m=0}^{p} \end{bmatrix} \tag{23}$$

A. Real-Coded Genetic Algorithm

The basic concept of GA was introduced by Holland in 1975[24] and it is an adaptive population based optimization method. This bio-inspired technique is based on the evolutionary ideas of natural selection and genetics, wherein a set of coefficient chromosomes is selected and encoded as binary strings. To avoid the precision problems, the final local tuning potential of a binary coded GA is improved with the use of RGA. Using real values, the natural form of the strings is maintained, thus, avoiding the coding and decoding processes. A considerable increase in the speed of operation, efficiency and precision in the results can be observed. RGA is universally employed to obtain the set of optimal solutions[25]. The algorithm undergoes three main processes after randomly generating the initial population. The selection process chooses better individual genotype chromosome depending on computing the fitness of each individual and produce a new generation of offspring chromosomes. The use of tournament operator allows a competition amongst the chromosomes on the grounds of their fitness values, where winners are selected with better fitness values. The crossover process is responsible for combining two chromosomes to produce new generations in search of a better fitness. A heuristic crossover operator aims towards determining the direction towards a better solution. Finally, the mutation process makes random changes to incorporate diversity in the results for achieving the global solution. The adaptive feasible mutation generates random variations adaptively with respect to the last successful or unsuccessful generation. The implementation steps of GA for the nonlinear system modeling using second order Volterra system model are adopted from [26].

B. Particle-Swarm Optimization

The social behavior of certain animals within a team such as fish schooling, insect swarming and bird flocking is transformed into an artificial swarm and is mathematically modeled as the PSO algorithm. It is a robust, population-based stochastic search technique which is suitable for non-differentiable and multiple objective functions. It was developed in 1995[27], and is successfully being applied to many engineering applications. In this algorithm, each particle acts as agent and is a potential solution. It is characterized by its position in the solution space and velocity with which it moves towards the optimal solution evaluated by the best fitness value. At every iteration, each particle is attracted towards the position of the current global best location. The velocity of the ith particle in the current iteration (let l), is adapted by evaluating the sum of three terms: the global best position vector, $gbest$, its personal best value, $pbest$ and the particle's present velocity, v^l. This new velocity vector is determined by the following formula considering the initial velocity, $v_i^{l=0} = 0$.

$$\begin{aligned} v_i^{l+1} =& W * v_i^l + \alpha C_1[gbest^l - x_i^l] \\ &+ \beta C_2[pbest_i^l - x_i^l] \end{aligned} \tag{29}$$

where W is the inertia weight parameter that controls the tradeoff between $gbest$ and $pbest$ of the swarm. Its value is set less than one. C_1, C_2 are the learning parameters that indicates the relative attraction towards $gbest$ and $pbest$ and α, β are random numbers ranging between $[0, 1]$. Also, the new position, x_i^{l+1} of the ith particle is updated by using

$$x_i^{l+1} = x_i^l + v_i^{l+1} \tag{30}$$

v_i can be bounded with the range $[v_{min}, v_{max}]$. On calculation of the new position, the particle flies to that location and ultimately at the final iteration, the global best solution becomes the optimal solution searched by PSO. The implementation steps of PSO for the nonlinear system modeling using second order Volterra system model are adopted from [28].

C. Cuckoo-Search Algorithm

CSA is a mathematical conceptualization which simulates the breeding strategy of the cuckoo birds. It was developed in 2009 by Yang and Deb[29]. It is based on the unique parasitic behaviour of some cuckoo bird species in combination with the Lévy flight. These bird species reproduce and lay their eggs in the nests of other birds. The host birds sometimes belligerently throw away the foreign eggs to increase the probability of hatching their own eggs. Whereas, some host birds simply abandon their nests and build a new nest at a new location. In CSA, each cuckoo egg in the host's nest symbolizes to a potential solution of the design problem. Each solution is characterized by its fitness value. The objective of CSA is to exchange a low fitness value solution with a better solution. In the process of generating a new solution, the concept of random walk performed by Lévy flights is applied. In this, the next step of the random walk is based on the current location (solution) and the transition probability to the next location.

In order to simplify the algorithm, it is governed by three guiding rules[29]. (i) Each bird is allowed to lay only one egg at once, which is randomly placed among the host bird's nests. (ii) The nest with the high quality eggs (solutions with high fitness values) will be imparted over to the next generation. (iii) A predetermined number of host nests are available, in which the probability of identification of alien eggs by host birds is also fixed ($P_a \in [0, 1]$). In instance of discovery, the host bird can either throw the alien egg or abandon the nest.

While generating a new solution, the Lévy flight is performed, represented in eq. (31). It is a Markov chain in which the next step depends on the current location and the transition probability.

$$\mathbf{a}_{l+1} = \mathbf{a}_l + \delta \oplus \text{Lévy}(\lambda) \tag{31}$$

where $\mathbf{a}_l$ is the solution vector which is the location of current solution at iteration, l, δ ($\delta > 0$) is the step size that determines the distance of the random walk. If δ is too big, then $\mathbf{a}_{l+1}$ will be too far away from $\mathbf{a}_l$. Similarly, if δ is too small, then $\mathbf{a}_{l+1}$ will be very close to $\mathbf{a}_l$ to be of any importance. Lévy(λ) is adopted from the Lévy distribution with an infinite variance and infinite mean[29].

The steps involved in the optimization algorithm utilizing the strategy of cuckoo birds for the process of evolving their generations along with their parasitic behavior are as follows.
Step 1: Initialize the maximum number of iterations (N) and randomly generate an initial population of n_c host nests, $\mathbf{a}_l$.
Step 2: Compute the fitness value, say E_l, of randomly generated host nest, $\mathbf{a}_l$.
Step 3: Generate a new nest using the Lévy flights given in eq. (31) and compute the fitness value, say E_{l+1}, of the new nests.
Step 4: Compare the two fitness values. For a minimization problem, if $E_l > E_{l+1}$, the initial host nests $\mathbf{a}_l$ are replaced by new nests, $\mathbf{a}_{l+1}$, generated by Lévy flights.
Step 5: Abandon a fraction of worst nests depending on the probability parameter p_a and build new nests, $\mathbf{a}_n$ using the random flights.
Step 6: Calculate the fitness of all the new nests and update the best nest, $\mathbf{a}_b$ of the generation until the current iteration. Compare it with the fitness value of the nest of next iteration and update the best nest.
Step 7: Repeat Steps 2-6 till the maximum number of iterations has reached. The best solution, $\mathbf{a}_b$ gives the optimal solution to the problem.

V. Simulation and Analysis

In this section, the discrete time nonlinear system identification problem is formulated and the simulated results have been presented. In order to implement the above formulated continuous time Volterra system using MATLAB, the discrete time signals are incorporated, by decimating the original input with integer factor of $M > 1$ and the fractional delays by integers in the range $0, 1, \cdots, M-1$.

A. Fractional delay system in discrete time

Given an input signal $x[n] = z[Mn]$. It is delayed by a fraction of r/M, where r is an integer in $0, 1, \cdots, M-1$, given by $x[n-r/M] = z[Mn-r]$. Let r_k be an integer of the form $(Mk + s_k)$ where $s_k \in 0, 1, \cdots, M-1$, $k = 1, 2, \cdots, p$. The output generated by passing the input signal $x[n]$ through a second order Volterra filter with fractional delays of $r_1, \cdots, r_p$ is given by

$$\begin{aligned} y[n] &= h[0] + \sum_{k=1}^{p} h[k]x[n-r_k/M] \\ &\quad + \sum_{k,m=0}^{p} g[k,m]x[n-r_k/M]x[n-r_m/M] \\ &= h[0] + \sum_{k} h[k]z[Mn-r_k] \\ &\quad + \sum_{k,m} g[k,m]z[Mn-r_k]z[Mn-r_m] \end{aligned} \tag{32}$$

Considering a noisy signal, eq. (32) is an approximate relation. Now, the aim is to determine the coefficients $h[k]$, $g[k,m]$ and the integers $r_1, \cdots, r_p$ such that the difference between the left hand side and right hand side of eq. (32) has minimum error energy. The Fourier transform (DTFT) of $z[Mn-r]$ is given by

$$\begin{aligned} &\text{DTFT}\{z[Mn-r]\} = \\ &\quad M^{-1} \sum_{l=0}^{M-1} e^{(-jr(\omega-2\pi l)/M)} Z\left(\frac{\omega - 2\pi l}{M}\right) \end{aligned} \tag{33}$$

The Fourier transform of $y_1[n] = \sum_k h[k]z[Mn-r_k]$ in eq. (32) is

$$Y_1(\omega) = M^{-1} \sum_{k,l} h[k] e^{(-j(\omega-2\pi l)r_k/M)} Z\left(\frac{\omega - 2\pi l}{M}\right) \tag{34}$$

where, k ranges over $1, 2, \cdots, p$ and l ranges over $0, 1, \cdots, M-1$. The Fourier transform of $y_2[n] = \sum_{k,m} g[k,m]z[Mn-r_k]z[Mn-r_m]$ in eq. (32) is given by

$$\begin{aligned} Y_2(\omega) &= M^{-1} \sum_{k,m,l} g[k,m] \int_{-\pi}^{\pi} e^{(-j(\omega_1 r_k + ((\omega-2\pi l)/M - \omega_1) r_m))} \\ &\quad \times Z(\omega_1) Z\left(\frac{\omega - \omega_1 - 2l\pi}{M}\right) \mathrm{d}\omega_1 \end{aligned} \tag{35}$$

Let Ω be a discrete set of frequencies in $[-\pi, \pi]$ which are equispaced. For each integer, r, a column vector of size equal to the cardinality of Ω is defined by

$$\hat{e}(r) = (e^{(-j\omega r/M)})_{\omega \in \Omega} \tag{36}$$

Further the diagonal matrix is defined as

$$D_Z[\alpha] = M^{-1} \times \text{diag}\left[Z\left(\frac{\omega - \alpha}{M}\right), \omega \in \Omega\right] \tag{37}$$

Assume that the inter-frequency spacing of Ω is Δ. Then we have

$$\begin{aligned} Y_1 &= (Y_1(\omega))_{\omega \in \Omega} \\ &= \sum_{k,l} h[k] e^{(j2\pi l r_k/M)} D_Z[2\pi l] D(r_k) \hat{e}(r_k) \end{aligned} \tag{38}$$

where

$$D(r) = \text{diag}\left[e^{(-j\omega r/M)} : \omega \in \Omega\right] \tag{39}$$

$$Y_2 =(Y_2(\omega))_{\omega\in\Omega} =\Delta \times \sum_{k,m,l,\omega_1} g[k,m]e^{(-j\omega_1(r_k-r_m))} \times Z(\omega_1)e^{(j2\pi lr_m/M)}D_Z(\omega_1+2l\pi)\hat{e}(r_m) \quad (40)$$

Considering the vectors

$$Q[k,m|\mathbf{r}] = \Delta \times \sum_{l,\omega_1} e^{(-j\omega_1(r_k-r_m))} \times Z(\omega_1)e^{(j2\pi lr_m/M)}D_Z(\omega_1+2l\pi)\hat{e}(r_m) \quad (41)$$

and

$$P[k|\mathbf{r}] = \sum_l e^{(j2\pi lr_k/M)}D_Z(2\pi l)D(r_k)\hat{e}(r_k) \quad (42)$$

where

$$\mathbf{r} = (r_m)_{m=0}^p$$

Then,

$$Y_1 = \sum_k h[k]P[k|\mathbf{r}]$$
$$Y_2 = \sum_{k,m} g[k,m]Q[k,m|\mathbf{r}] \quad (43)$$

Further, in terms of the matrices

$$P[\mathbf{r}] = \mathrm{Col}[P[k|\mathbf{r}] : k = 1,2,\cdots,p]$$
$$Q[\mathbf{r}] = [Q[k,m|\mathbf{r}] : k,m = 1,2,\cdots,p] \quad (44)$$

Thus,

$$Y \approx Y_1 + Y_2 = P[\mathbf{r}]h + Q[\mathbf{r}]g \quad (45)$$

Here,

$$h = (h[k]) \in R^p, g = \mathrm{vec}(g[k,m]) \in R^{p^2} \quad (46)$$

$h, g, \mathbf{r}$ are estimated by minimizing

$$E[h,g,\mathbf{r}] =\| Y - P[\mathbf{r}]h - Q[\mathbf{r}]g \|^2 \quad (47)$$

Now, writing

$$\begin{pmatrix} h \\ g \end{pmatrix} = q \in R^{p^2+p} \quad (48)$$

and

$$[P[\mathbf{r}]|Q[\mathbf{r}]] = S[\mathbf{r}] \quad (49)$$

gives

$$E[q,\mathbf{r}] =\| Y - S[\mathbf{r}]q \|^2 \quad (50)$$

Eq. (50) has to be minimized w.r.t q, $\mathbf{r}$. Firstly, minimizing E w.r.t. q gives

$$\hat{q}(\mathbf{r}) = (S[\mathbf{r}]^{\mathrm{T}}S[\mathbf{r}])^{-1}S[\mathbf{r}]^{\mathrm{T}}Y \quad (51)$$

Substituting eq. (51) into the expression for E gives

$$E[\mathbf{r}] = E[\hat{q}(\mathbf{r}),\mathbf{r}] =\| Y \|^2 - \| P_{S[\mathbf{r}]}Y \|^2 \quad (52)$$

Minimizing this w.r.t. $\mathbf{r}$ is equivalent to maximizing

$$F(\mathbf{r}) =\| P_{S[\mathbf{r}]}Y \|^2 \quad (53)$$

w.r.t $\mathbf{r}$. Here, $P_{S[\mathbf{r}]}$ is the orthogonal projection onto $\mathcal{R}(S[\mathbf{r}])$:

$$P_{S[\mathbf{r}]} = S[\mathbf{r}](S[\mathbf{r}]^{\mathrm{T}}S[\mathbf{r}])^{-1}S[\mathbf{r}]^{\mathrm{T}} \quad (54)$$

The above result has been simulated using the MATLAB software and the results are illustrated in the next subsection.

TABLE I
CONTROL PARAMETERS FOR FILTER DESIGN.

Parameters	Symbol	RGA	PSO	CSA
Population Size	n_g, n_p, n_c	55	55	25
Max. Iteration Cycle	N	200	200	200
Tolerance		10^{-6}	10^{-6}	10^{-6}
Limits of System Coefficients		-10,+10	-10,+10	-10,+10
Selection	Tournament	Size: 4	-	-
Crossover Rate, Ratio	Heuristic	0.8, 1.2	-	-
Mutation rate	Adaptive feasible	0.01	-	-
Learning Parameters	C_1, C_2	-	2, 2	-
Particle Velocity	v_{min}, v_{max}	-	0.01, 1	-
Inertia Weight	W	-	0.4	-
Discovering Rate of alien eggs	P_a	-	-	0.25

B. Nonlinear System Modeling Examples

Extensive simulations have been conducted with two nonlinear system examples to evaluate the performance of the proposed method based on second order Volterra system using fractional delay. The unknown nonlinear system and a second order Volterra system are tested with five different input signals. The results obtained are presented in terms of the comparison between the actual system output and the estimated output using gradient search, RGA, PSO and CSA. Mean square error (MSE), accuracy and statistical data are investigated in order to demonstrate the effectiveness of the proposed nonlinear system modeling method. The fitness function is minimized such that the output of the estimated Volterra system closely approximates the actual nonlinear system output. The mean square error objective function is defined as

$$E = \frac{1}{M}\sum_{n=1}^{M}(\hat{y}[n]-y[n])^2 \quad (55)$$

where $\hat{y}[n]$ and $y[n]$ are the response of the actual nonlinear system and the second order Volterra system, respectively, M is the number of samples utilized to compute the fitness function. The two examples are expressed below.

1) Example 1: A standard nonlinear model is considered to carry out the simulations as utilized by Chang in [14]. This system is input with the discrete-time signal, $x[n]$ and the system output is given as

$$d[n] = \frac{0.3d^2[n-1]+0.8x[n-1]+0.6d[n-2]}{1+x^2[n-1]+d^2[n-1]} \quad (56)$$

The eq. (56) is considered as the actual output which is approximated with the discrete-time output of the second-order Volterra system, $y[n]$ given in eq. (32). Table 1 summarizes the control parameters of the stochastic algorithms to perform the system identification task. Several simulation runs have been performed with different initial conditions in order to obtain an accurate approximation to the nonlinear system under consideration.

Computations are performed with the Volterra kernel size, $p = 5$ and with following five different discrete-time input signals, (i) sinusoidal signal, $x[n] = 0.8\sin(\frac{\pi}{9}n)$, (ii) noisy sinusoidal signal, $x[n] = 0.8\sin(\frac{\pi}{9}n) + w[n]$, (iii) square input, $x = 0.4 \times square(n)$, (iv) noisy square input, $x =$

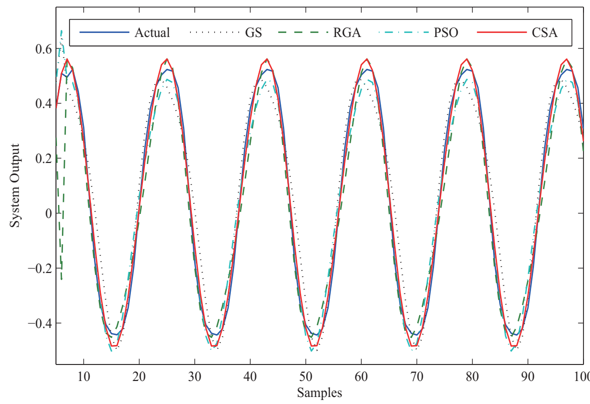

Fig. 3. Comparison of actual nonlinear system output with second order fractional delay Volterra system model output using gradient search, RGA, PSO and CSA for sinusoidal input signal $x(n) = 0.8\sin(\frac{\pi}{9}n)$ in example 1.

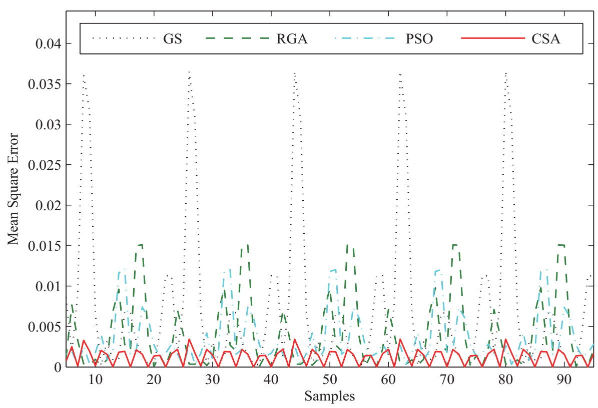

Fig. 4. Comparison of MSE for second order fractional delay Volterra system model output using gradient search, RGA, PSO and CSA for sinusoidal input signal $x(n) = 0.8\sin(\frac{\pi}{9}n)$ in example 1.

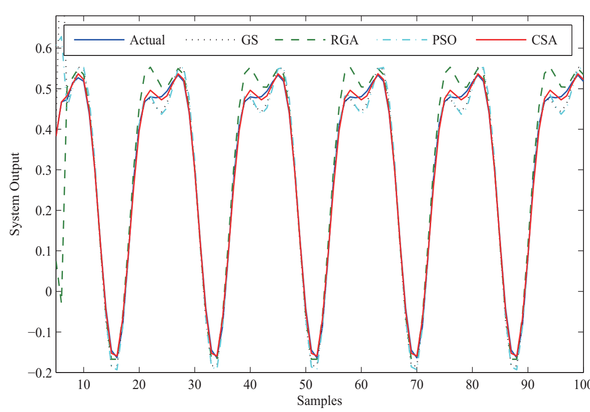

Fig. 5. Comparison of actual nonlinear system output with second order fractional delay Volterra system model output using gradient search, RGA, PSO and CSA for noisy sinusoidal input signal $x(n) = 0.8\sin(\frac{\pi}{9}n) + w(n)$ in example 1.

TABLE II
KERNEL PARAMETERS OF SECOND ORDER FRACTIONAL DELAY VOLTERRA SYSTEM MODEL USING GRADIENT SEARCH, RGA, PSO AND CSA BASED METHODS FOR SINUSOIDAL INPUT SIGNAL $x(n) = 0.8\sin(\frac{\pi}{9}n)$ FOR EXAMPLE 1.

Kernel Parameters	Gradient search	RGA	PSO	CSA
$h(0)$	4.7819	-2.6580	-0.2483	-1.1851
$h(1)$	6.9289	7.0705	0.9464	0.3521
$h(2)$	-1.6173	-6.4367	-0.6477	0.3159
$h(3)$	-7.6525	0.3392	-0.1653	-0.2498
$h(4)$	-0.4248	-3.5871	0.3371	-0.1913
$h(5)$	6.3502	5.5849	0.4426	0.6771
$g(0,0)$	-2.6945	10.0000	-0.2351	-0.2155
$g(0,1)$	-8.0000	-6.5660	0.5631	2.7324
$g(0,2)$	7.4322	0.3116	-1.4216	-0.1614
$g(0,3)$	1.7955	0.4281	0.3063	0.5090
$g(0,4)$	6.2381	-0.3697	-0.9874	-3.9133
$g(1,1)$	-6.6592	-6.8174	-0.3265	-1.5858
$g(1,2)$	-0.4924	9.3514	1.5749	0.5226
$g(1,3)$	-1.4650	-0.3788	0.1068	1.5532
$g(1,4)$	-4.5295	-4.4736	-1.0289	1.0851
$g(2,2)$	7.9095	-2.0326	1.0759	0.9137
$g(2,3)$	-7.3655	-2.9788	-1.0691	-0.5968
$g(2,4)$	-6.1501	-0.4481	0.3015	0.1160
$g(3,3)$	7.7976	9.8700	0.3764	-0.9234
$g(3,4)$	1.9988	-0.0004	1.2163	0.8949
$g(4,4)$	-7.6527	0.0220	-0.8601	1.0284

TABLE III
KERNEL PARAMETERS OF SECOND ORDER FRACTIONAL DELAY VOLTERRA SYSTEM MODEL USING GRADIENT SEARCH, RGA, PSO AND CSA BASED METHODS FOR NOISY SINUSOIDAL INPUT SIGNAL $x(n) = 0.8\sin(\frac{\pi}{9}n) + w(n)$ FOR EXAMPLE 1.

Kernel Parameters	Gradient search	RGA	PSO	CSA
$h(0)$	-0.0517	-0.2039	-0.0371	0.6549
$h(1)$	2.7845	-1.6059	0.7056	-2.6148
$h(2)$	-2.3396	1.3792	0.2225	0.2798
$h(3)$	0.0679	-0.7032	0.4377	2.8849
$h(4)$	1.2805	0.0966	-1.1508	1.5909
$h(5)$	0.3900	-1.0010	1.6167	-2.9161
$g(0,0)$	-1.3171	1.7034	0.6819	-0.4359
$g(0,1)$	-0.0356	0.8047	-0.4647	0.8494
$g(0,2)$	-0.2043	0.0536	-0.3457	1.9477
$g(0,3)$	-0.5186	2.8425	0.0924	-2.9480
$g(0,4)$	1.7869	0.1401	0.0518	1.6660
$g(1,1)$	0.3328	-0.9125	-0.8096	2.1944
$g(1,2)$	1.4369	-2.3005	-1.1969	-2.5532
$g(1,3)$	2.1203	-2.4689	-1.0682	0.9425
$g(1,4)$	-2.7382	-1.5798	1.9820	-0.9348
$g(2,2)$	-0.2732	1.5571	1.6017	-2.8826
$g(2,3)$	-3.4732	-0.8132	1.3062	0.1963
$g(2,4)$	-0.3665	2.1578	-2.5803	0.0648
$g(3,3)$	1.1350	0.7772	0.5590	0.7749
$g(3,4)$	0.8235	-1.1588	-1.3976	4.0653
$g(4,4)$	0.0337	1.9707	0.5623	-2.7554

$0.4\times square(n)+w[n]$ and (v) random input signal. The noise factor, $w[n]$ is taken to be 0.5. Fig. 3 shows the comparison of the actual system output by simulating eq. (56) with the sinusoidal input signal and the estimated signal using gradient-search, RGA, PSO and CSA. The mean square error between the actual and estimated system output with sinusoidal input signal is depicted in Fig. 4 for gradient-search, RGA, PSO and CSA. The Volterra system coefficients, $h(k)$ and $g(k,m)$ with kernel memory size, $p = 5$, optimized using aforementioned algorithms are listed in Table II. The mean value of MSE with

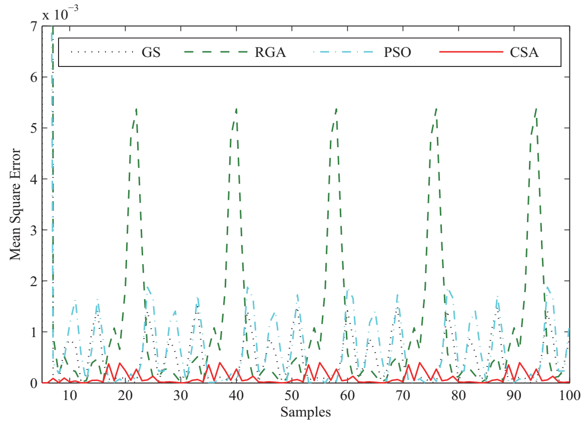

Fig. 6. Comparison of MSE for second order fractional delay Volterra system model output using gradient search, RGA, PSO and CSA for noisy sinusoidal input signal $x(n) = 0.8\sin(\frac{\pi}{9}n) + w(n)$ in example 1.

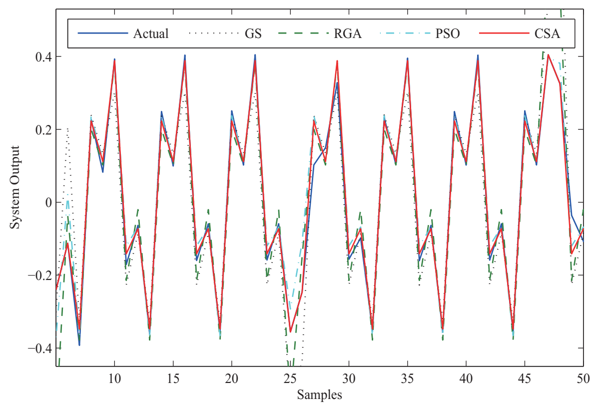

Fig. 7. Comparison of actual nonlinear system output with second order fractional delay Volterra system model output using gradient search, RGA, PSO and CSA for square input signal in example 1.

a sinusoidal signal using gradient-search, RGA, PSO and CSA is observed to be 0.0028, 0.0036, 0.0016, and 8.6450×10^{-4}, respectively. Based on the observations of MSE values and the graphical comparison in Figs. 3 and 4, it is inferred that CSA gives a better approximation to the nonlinear system coefficients. The performance of the employed methodologies is sequenced as, CSA $>$ PSO $>$ GS $>$ RGA. The comparison of output response of the system when tested with noisy sinusoidal signal is demonstrated in Fig. 5. The MSE obtained when the system is subjected to noisy sinusoidal signal using gradient-search, RGA, PSO and CSA is shown in Fig. 6. Table III indicates the kernel parameters of Volterra system with noisy sinusoidal input signal. The mean MSE values obtained are 0.0013, 0.0020, 9.5133×10^{-4} and 5.3905×10^{-4}, respectively, with gradient-search, RGA, PSO and CSA when the system is tested with noisy sinusoidal input signal. Thus, a better approximation to the nonlinear system coefficients is achieved with CSA and optimization techniques can be arranged according to the performance as, CSA $>$ PSO $>$ GS $>$ RGA.

TABLE IV
KERNEL PARAMETERS OF SECOND ORDER FRACTIONAL DELAY VOLTERRA SYSTEM MODEL USING GRADIENT SEARCH, RGA, PSO AND CSA BASED METHODS FOR SQUARE INPUT SIGNAL FOR EXAMPLE 1.

Kernel Parameters	Gradient search	RGA	PSO	CSA
$h(0)$	0.2427	-0.0773	-0.3601	-0.3919
$h(1)$	0.3044	0.2828	-0.0349	0.0153
$h(2)$	1.1082	0.7179	0.5309	0.5852
$h(3)$	-0.1549	-0.1826	-0.1097	-0.1126
$h(4)$	0.5619	0.6813	0.3180	0.3600
$h(5)$	0.4230	-0.0164	-0.1591	-0.1024
$g(0,0)$	0.1047	1.2752	0.1685	0.5475
$g(0,1)$	0.1606	0.4552	0.6614	0.6311
$g(0,2)$	0.2924	0.3694	-0.0932	-0.5869
$g(0,3)$	0.7192	0.1017	0.2101	0.4265
$g(0,4)$	1.1304	0.7404	0.6378	0.4574
$g(1,1)$	0.1574	-0.0658	0.7991	0.0763
$g(1,2)$	0.5883	0.5414	0.9405	0.5744
$g(1,3)$	-0.0925	0.5049	-0.0661	-0.0203
$g(1,4)$	0.3228	0.1119	0.0264	0.0726
$g(2,2)$	0.9729	-1.0543	0.6691	1.4622
$g(2,3)$	-0.4436	0.4137	0.5556	0.0912
$g(2,4)$	1.2680	0.3468	0.4035	-0.0375
$g(3,3)$	-0.5785	0.0981	0.1056	-0.1419
$g(3,4)$	0.2559	0.9561	0.6068	0.4822
$g(4,4)$	-0.4156	0.3841	0.3029	0.5164

TABLE V
KERNEL PARAMETERS OF SECOND ORDER FRACTIONAL DELAY VOLTERRA SYSTEM MODEL USING GRADIENT SEARCH, RGA, PSO AND CSA BASED METHODS FOR NOISY SQUARE INPUT SIGNAL FOR EXAMPLE 1.

Kernel Parameters	Gradient search	RGA	PSO	CSA
$h(0)$	0.5125	-0.1586	-0.0891	0.4181
$h(1)$	0.5834	-0.6495	0.4696	0.6257
$h(2)$	-0.4553	1.8403	-0.5114	-0.1225
$h(3)$	-0.9315	0.9366	-0.1402	-0.1988
$h(4)$	-0.4992	-0.5597	0.2433	-0.9789
$h(5)$	0.6559	-0.5733	0.0443	-0.1059
$g(0,0)$	-0.4693	0.4794	0.1119	-0.6883
$g(0,1)$	-0.6476	-0.4107	-0.1892	0.3784
$g(0,2)$	-0.7153	-0.0229	0.0493	-0.2357
$g(0,3)$	1.5708	1.0239	-0.4435	-0.0758
$g(0,4)$	-0.5353	0.7520	-0.5391	0.0521
$g(1,1)$	1.0706	-0.5719	0.9744	-2.6585×10^{-4}
$g(1,2)$	0.7109	-0.6802	-0.4078	0.6443
$g(1,3)$	-1.0828	-0.1425	-0.0566	-0.1139
$g(1,4)$	0.1929	0.0527	0.2139	-0.2014
$g(2,2)$	0.7066	-0.5288	0.4387	-0.2642
$g(2,3)$	0.9443	-0.3442	0.0375	0.3473
$g(2,4)$	-0.6769	0.0722	-0.3825	0.0691
$g(3,3)$	-0.3896	0.5909	0.2997	0.9077
$g(3,4)$	0.5375	0.6186	-0.3601	0.2017
$g(4,4)$	-0.5179	0.2909	0.4694	0.0131

Fig. 7 shows the comparison of the actual system output with square input signal and the estimated signal using gradient-search, RGA, PSO and CSA. Fig. 8 depicts the MSE observed when the system is tested with square input signal using gradient-search, RGA, PSO and CSA. The kernel parameters of Volterra system with squared input are reported in Table IV. The mean value of MSE noticed with gradient-search, RGA, PSO and CSA is 0.0042, 0.0026, 8.7709×10^{-4} and 5.4547×10^{-4}, respectively when squared signal is applied at the input of the system. From the graphical results and numerical values of MSE, one can conclude that CSA provides

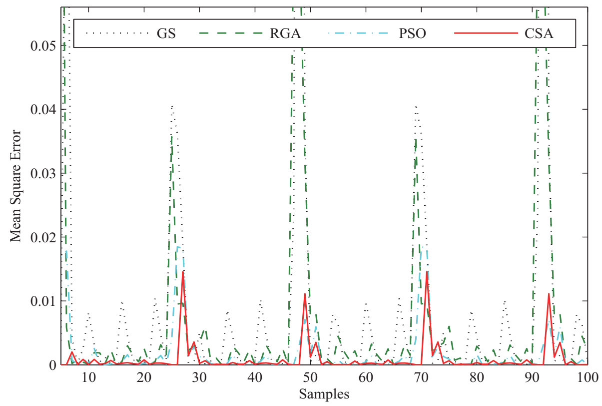

Fig. 8. Comparison of MSE for second order fractional delay Volterra system model output using gradient search, RGA, PSO and CSA for square input signal in example 1.

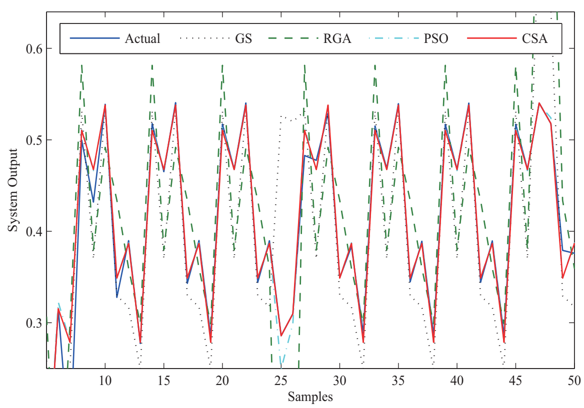

Fig. 9. Comparison of actual nonlinear system output with second order fractional delay Volterra system model output using gradient search, RGA, PSO and CSA for noisy square input signal in example 1.

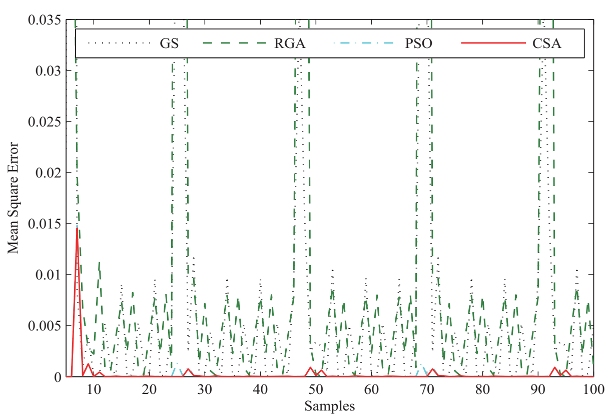

Fig. 10. Comparison of MSE for second order fractional delay Volterra system model output using gradient search, RGA, PSO and CSA for noisy square input signal in example 1.

a good approximation to the nonlinear fractional delay second order Volterra system coefficients compared to other applied

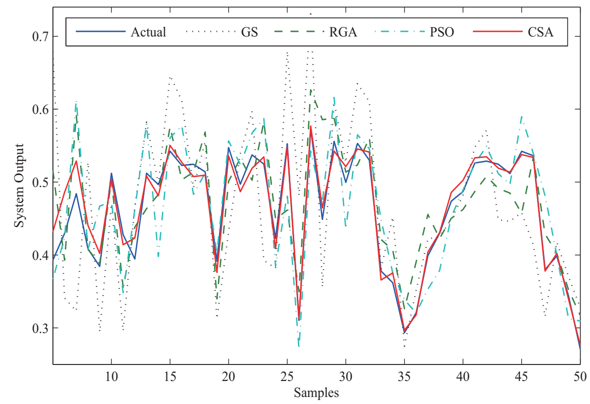

Fig. 11. Comparison of actual nonlinear system output with second order fractional delay Volterra system model output using gradient search, RGA, PSO and CSA for the random input signal in example 1.

TABLE VI

KERNEL PARAMETERS OF SECOND ORDER FRACTIONAL DELAY VOLTERRA SYSTEM MODEL USING GRADIENT SEARCH, RGA, PSO AND CSA BASED METHODS FOR RANDOM INPUT SIGNAL FOR EXAMPLE 1.

Kernel Parameters	Gradient search	RGA	PSO	CSA
$h(0)$	0.2872	0.3162	0.1696	0.2202
$h(1)$	0.0922	0.0613	0.3087	0.0095
$h(2)$	0.1827	0.5617	0.1116	0.5814
$h(3)$	-0.2102	-0.4402	0.0351	-0.0396
$h(4)$	0.0715	0.1864	0.4296	0.3219
$h(5)$	0.2372	0.0375	0.1394	0.0298
$g(0,0)$	0.0045	0.0853	-0.0468	0.0147
$g(0,1)$	0.0971	-0.1805	-6.4717×10^{-5}	-0.0171
$g(0,2)$	0.6207	0.0495	-0.3325	-0.0077
$g(0,3)$	0.0275	-0.2599	-0.2727	-0.0041
$g(0,4)$	-0.8997	0.0516	0.1541	-0.0085
$g(1,1)$	0.2362	-0.2754	-0.1177	-0.3871
$g(1,2)$	-0.4174	0.1923	0.2645	0.0192
$g(1,3)$	0.2129	-0.1189	-0.0345	-0.0073
$g(1,4)$	-0.1775	-0.0659	0.1635	0.0210
$g(2,2)$	-0.1632	0.0866	-0.0579	0.0247
$g(2,3)$	-0.1397	0.0479	0.1556	-0.0484
$g(2,4)$	0.7185	0.3258	-0.1632	0.0172
$g(3,3)$	-0.2123	0.2361	-0.0389	-0.1671
$g(3,4)$	0.3809	-0.2795	-0.3564	-0.0522
$g(4,4)$	-0.2813	-0.0921	-0.0363	-0.0268

optimization algorithms. The performance of these algorithms is arranged as, CSA $>$ PSO $>$ GS $>$ RGA. Fig. 9 exhibits the comparison of output response of the system analyzed with noisy square input using gradient-search, RGA, PSO and CSA. The MSE remarked for the system under consideration when examined with noisy square input is shown in Fig. 10. Table V summarizes the kernel parameters of Volterra system with noisy square input signal. The MSE values for second order fractional delay Volterra system with gradient-search, RGA, PSO and CSA are 0.0033, 0.0057, 6.0527×10^{-4} and 5.9464×10^{-4}, respectively. Based on these MSE values, it can be finally deduced that nonlinear system identification with the second order Volterra system using CSA surpasses the other employed optimization methods. The performance can be ranked as CSA $>$ PSO $>$ RGA $>$ GS. The comparison of output response of the system with random signal using

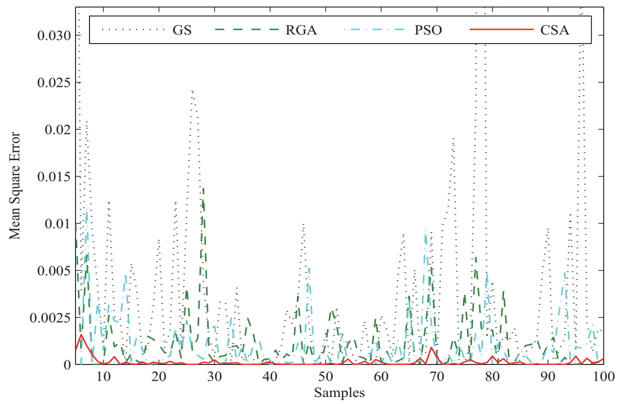

Fig. 12. Comparison of MSE for second order fractional delay Volterra system model output using gradient search, RGA, PSO and CSA for the random input signal in example 1.

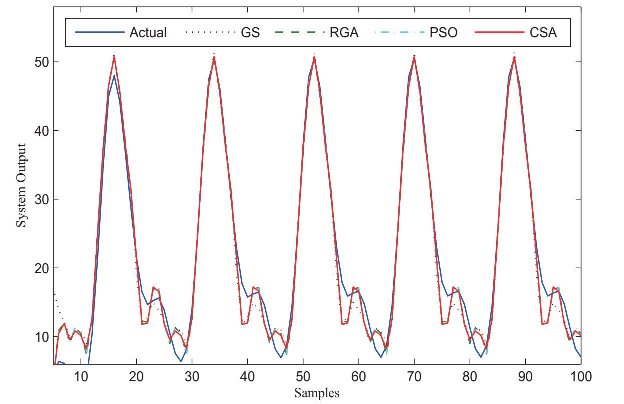

Fig. 13. Comparison of actual nonlinear system output with second order fractional delay Volterra system model output using gradient search, RGA, PSO and CSA for sinusoidal input signal $x(n) = 0.8\sin(\frac{\pi}{9}n)$ in example 2.

gradient-search, RGA, PSO and CSA is demonstrated in Fig. 11. The observed values of MSE and kernel parameters of Volterra system with random signal are exhibited in Fig. 12 and Table VI, respectively. The mean MSE values obtained are 0.0027, 8.5199×10^{-4}, 7.7135×10^{-4} and 1.9658×10^{-4}, respectively, with gradient-search, RGA, PSO and CSA when the system is tested with random signal. It can be concluded from the aforementioned results that the CSA based nonlinear system identification outperforms all other reported algorithms in terms of MSE. The order of the algorithm based on its performance is given as CSA > PSO > RGA > GS.

Furthermore, the statistical analysis in terms of maximum, minimum, mean, variance and standard deviation of the MSE is performed to evaluate the performance of the proposed method. Table VII shows the comparative numerical values of different characteristics like maximum, minimum, mean, variance and standard deviation of mean square error of the proposed second order fractional delay Volterra system for different input signals using gradient-search, RGA, PSO and CSA algorithms. This analysis provides a detailed comparison amongst the performance of estimated Volterra systems employing all four optimization techniques. It is observed that the MSE value obtained with CSA is lower as compared to other algorithms with all input signals. From Figs. 3-12 and statistically analyzed results from Table VII, it is evident that with all input signals, the proposed nonlinear system modeling method based on fractional delay second order Volterra system produced minimum MSE compared to that of the gradient-search, RGA and PSO. Finally, it can be concluded that CSA based second order fractional delay Volterra system identification method gives superior results compared to other reported algorithms with all the input signals.

2) Example 2: In this example, the mathematical model of heat exchanger used in [14] is considered. The system can be expressed as

$$w[n] = x[n] - 1.3228x^2[n] + 0.7671x^3[n] - 2.1755x^4[n] \quad (57)$$

$$d[n] = 1.608d[n-1] - 0.6385d[n-2] - 6.5306w[n-1] + 5.5652w[n-2] \quad (58)$$

where $x[n]$ be the input to the system, $w[n]$ is the static nonlinearity and $d[n]$ be the output of the system.

TABLE VIII
KERNEL PARAMETERS OF SECOND ORDER FRACTIONAL DELAY VOLTERRA SYSTEM MODEL USING GRADIENT SEARCH, RGA, PSO AND CSA BASED METHODS FOR SINUSOIDAL INPUT SIGNAL $x(n) = 0.8\sin(\frac{\pi}{9}n)$ FOR EXAMPLE 2.

Kernel Parameters	Gradient search	RGA	PSO	CSA
$h(0)$	9.4709	16.2089	16.6626	14.4537
$h(1)$	-0.6698	25.5888	25.1569	19.6381
$h(2)$	-19.3316	11.4107	13.3984	15.6001
$h(3)$	8.6093	-11.0001	-11.9957	-14.5285
$h(4)$	-5.0777	43.6126	44.6720	42.9363
$h(5)$	-4.6264	13.3334	14.0811	11.2961
$g(0,0)$	-9.7089	5.2952	4.6671	1.2645
$g(0,1)$	15.3888	7.2002	7.6748	8.8481
$g(0,2)$	4.2497	-0.2573	4.3471	0.6401
$g(0,3)$	9.2702	-6.0859	-6.9853	-10.2706
$g(0,4)$	4.8762	-6.8910	-10.3297	-10.0909
$g(1,1)$	-3.8765	-0.6058	-2.8362	4.3473
$g(1,2)$	-16.4299	0.7216	-0.0073	-2.2931
$g(1,3)$	-17.3760	-3.6154	-1.2420	-6.1239
$g(1,4)$	-0.6727	-46.8916	-47.5947	-47.6447
$g(2,2)$	-0.4807	16.0521	15.5387	15.1564
$g(2,3)$	35.0206	28.9075	29.4013	31.9329
$g(2,4)$	-2.0420	-19.8110	-20.3997	-18.8809
$g(3,3)$	-10.0147	27.3951	28.3579	30.8179
$g(3,4)$	-15.3354	-17.4811	-17.5201	-22.3759
$g(4,4)$	11.2373	19.7659	19.8043	19.8045

In order to evaluate the performance of this system Volterra kernel size is selected as $p = 5$ and the input to the system is tested with five different input signals. Fig. 13 shows the comparison of actual output and the estimated output using gradient-search, RGA, PSO and CSA, when the sinusoidal input signal is applied. The Volterra system coefficients obtained with sinusoidal input using gradient-search, RGA, PSO and CSA are listed in Table VIII. The MSE error noticed with sinusoidal input is exhibited in Fig. 14. The mean MSE values obtained are 0.0220, 0.0155, 0.0154 and 0.0151, respectively, with gradient-search, RGA, PSO and CSA, when the system

TABLE VII
STATISTICAL COMPARISON OF MEAN SQUARE ERROR FOR THE IDENTIFICATION OF NONLINEAR SYSTEM WITH DIFFERENT INPUT SIGNAL USING GRADIENT SEARCH, RGA, PSO AND CSA BASED METHODS FOR EXAMPLE 1.

Input Signal	Algorithm	Mean Square Error (MSE)				
		Max	Min	Mean	Variance	Standard Deviation
$x(n) = 0.8\sin(\frac{\pi}{9}n)$	GS	0.0446	4.8916×10^{-5}	0.0028	2.8003×10^{-5}	0.0053
	RGA	0.1788	8.6341×10^{-6}	0.0036	3.4486×10^{-4}	0.0186
	PSO	0.0446	2.0815×10^{-5}	0.0016	2.1110×10^{-5}	0.0046
	CSA	0.0446	1.3196×10^{-9}	8.6450×10^{-4}	2.0281×10^{-5}	0.0045
$x(n) = 0.8\sin(\frac{\pi}{9}n) + w(n)$	GS	0.0623	5.0772×10^{-7}	0.0013	6.4219×10^{-5}	0.0080
	RGA	0.0787	2.4005×10^{-6}	0.0020	9.5760×10^{-5}	0.0098
	PSO	0.0494	1.9535×10^{-8}	9.5133×10^{-4}	2.7945×10^{-5}	0.0053
	CSA	0.0494	2.1471×10^{-11}	5.3905×10^{-4}	2.5107×10^{-5}	0.0050
$x(n) = 0.4\text{square}(n)$	GS	0.0945	1.3653×10^{-7}	0.0042	1.9755×10^{-4}	0.0141
	RGA	0.0409	1.2848×10^{-9}	0.0026	4.2058×10^{-5}	0.0065
	PSO	0.0242	6.8875×10^{-7}	8.7709×10^{-4}	7.6877×10^{-6}	0.0028
	CSA	0.0242	1.0647×10^{-16}	5.4547×10^{-4}	6.5939×10^{-6}	0.0026
$x(n) = 0.4\text{square}(n) + w(n)$	GS	0.0836	9.3438×10^{-7}	0.0033	1.0524×10^{-4}	0.0103
	RGA	0.0615	7.7230×10^{-6}	0.0057	1.7510×10^{-4}	0.0132
	PSO	0.0504	2.4164×10^{-8}	6.0527×10^{-4}	2.6298×10^{-5}	0.0051
	CSA	0.0504	5.7384×10^{-13}	5.9464×10^{-4}	2.6299×10^{-5}	0.0051
$x(n) = \text{rand}(n)$	GS	0.0341	4.6628×10^{-10}	0.0027	2.3409×10^{-5}	0.0048
	RGA	0.0110	1.1880×10^{-7}	8.5199×10^{-4}	2.2017×10^{-6}	0.0015
	PSO	0.0110	2.9739×10^{-9}	7.7135×10^{-4}	2.0691×10^{-6}	0.0014
	CSA	0.0110	3.9740×10^{-8}	1.9658×10^{-4}	1.2529×10^{-6}	0.0011

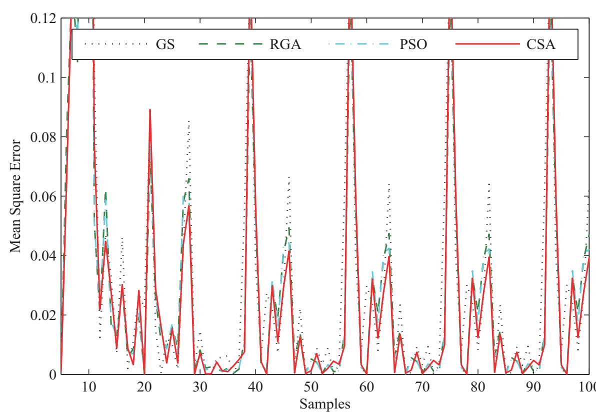

Fig. 14. Comparison of MSE for second order fractional delay Volterra system model output using gradient search, RGA, PSO and CSA for sinusoidal input signal $x(n) = 0.8\sin(\frac{\pi}{9}n)$ in example 2.

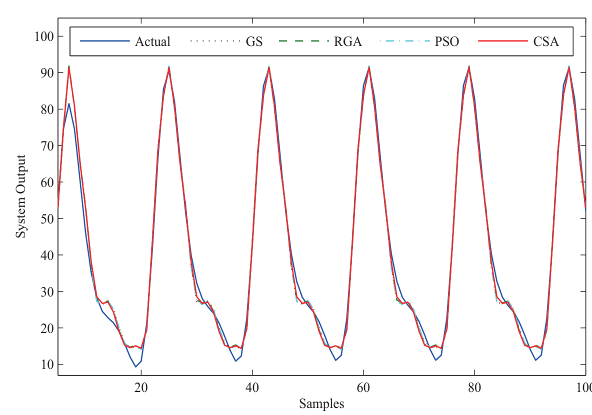

Fig. 15. Comparison of actual nonlinear system output with second order fractional delay Volterra system model output using gradient search, RGA, PSO and CSA for noisy sinusoidal input signal $x(n) = 0.8\sin(\frac{\pi}{9}n) + w(n)$ in example 2.

is tested with sinusoidal input. Based on the observations of MSE values and the graphical comparison in Figs. 13 and 14, it is inferred that CSA gives a better approximation to the nonlinear system coefficients. The performance of the employed methodologies is sequenced as, CSA $>$ PSO $>$ RGA $>$ GS.

The comparison of output response of the system when tested with noisy sinusoidal signal is demonstrated in Fig. 15. The MSE obtained when the system is subjected to noisy sinusoidal signal using gradient-search, RGA, PSO and CSA is shown in Fig. 16. Table IX lists the kernel parameters of Volterra system with noisy sinusoidal input signal. The mean value of MSE with noisy sinusoidal signal using gradient-search, RGA, PSO and CSA is observed to be 0.0154, 0.0158, 0.0154, and 0.0137, respectively. Thus, a better approximation to the nonlinear system coefficients is achieved with CSA and optimization techniques can be arranged according to the performance as, CSA $>$ PSO $=$ GS $>$ RGA.

The kernel parameters of Volterra system with squared input are reported in Table X. Fig. 17 shows the comparison of the actual system output with square input signal and the

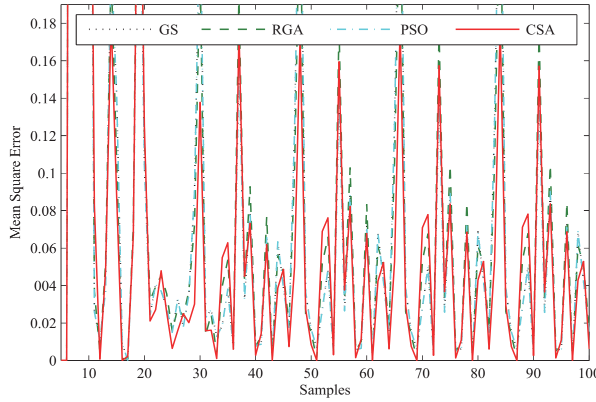

Fig. 16. Comparison of MSE for second order fractional delay Volterra system model output using gradient search, RGA, PSO and CSA for noisy sinusoidal input signal $x(n) = 0.8\sin(\frac{\pi}{9}n) + w(n)$ in example 2.

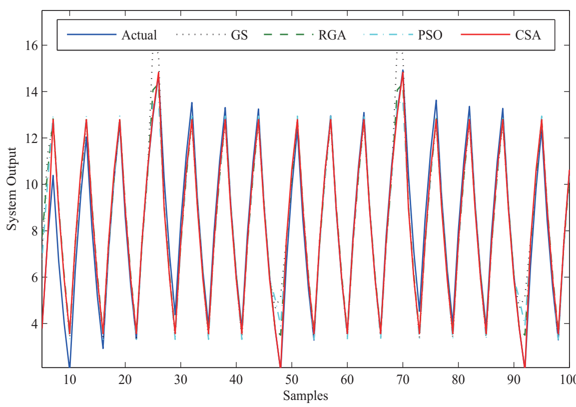

Fig. 17. Comparison of actual nonlinear system output with second order fractional delay Volterra system model output using gradient search, RGA, PSO and CSA for square input signal in example 2.

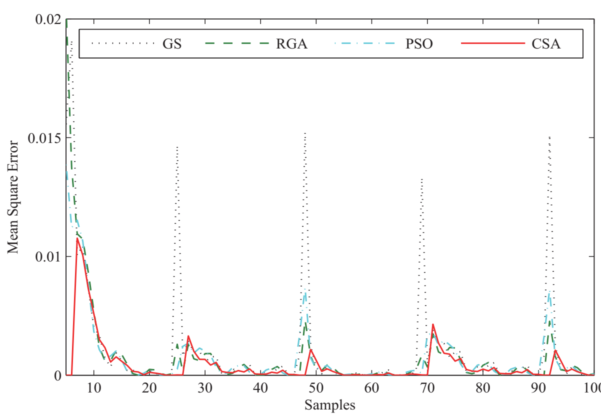

Fig. 18. Comparison of MSE for second order fractional delay Volterra system model output using gradient search, RGA, PSO and CSA for square input signal in example 2.

estimated signal using gradient-search, RGA, PSO and CSA.

TABLE IX

KERNEL PARAMETERS OF SECOND ORDER FRACTIONAL DELAY VOLTERRA SYSTEM MODEL USING GRADIENT SEARCH, RGA, PSO AND CSA BASED METHODS FOR NOISY SINUSOIDAL INPUT SIGNAL $x(n) = 0.8\sin(\frac{\pi}{9}n) + w(n)$ FOR EXAMPLE 2.

Kernel Parameters	Gradient search	RGA	PSO	CSA
$h(0)$	19.7175	19.9878	19.7175	19.6256
$h(1)$	2.5659	2.0529	2.5659	3.4482
$h(2)$	-24.1582	-23.6178	-24.1582	-26.4619
$h(3)$	-2.0403	-0.4954	-2.0403	-0.3887
$h(4)$	19.9393	20.0624	19.9393	23.4229
$h(5)$	-7.6989	-7.4160	-7.6989	-10.5589
$g(0,0)$	-7.0703	-7.2762	-7.0703	-6.6456
$g(0,1)$	11.4793	12.0153	11.4793	10.4377
$g(0,2)$	1.4172	2.3618	1.4172	1.6714
$g(0,3)$	-1.8849	-0.4439	-1.8849	-0.8707
$g(0,4)$	1.5957	0.1224	1.5957	0.9816
$g(1,1)$	-3.3497	-4.2333	-3.3497	-3.1366
$g(1,2)$	-5.7131	-6.3393	-5.7131	-5.6986
$g(1,3)$	0.4449	-1.0819	0.4449	1.1859
$g(1,4)$	4.8335	6.2813	4.8335	4.3979
$g(2,2)$	-2.1866	-1.1669	-2.1866	-1.8611
$g(2,3)$	6.8904	7.1536	6.8904	4.9551
$g(2,4)$	-1.2842	-1.4654	-1.2842	-1.7116
$g(3,3)$	2.6831	2.7344	2.6831	3.8625
$g(3,4)$	-10.9289	-11.3184	-10.9289	-9.3152
$g(4,4)$	7.6593	7.9471	7.6593	6.4721

TABLE X

KERNEL PARAMETERS OF SECOND ORDER FRACTIONAL DELAY VOLTERRA SYSTEM MODEL USING GRADIENT SEARCH, RGA, PSO AND CSA BASED METHODS FOR SQUARE INPUT SIGNAL FOR EXAMPLE 2.

Kernel Parameters	Gradient search	RGA	PSO	CSA
$h(0)$	5.3951	3.8427	5.2620	-1.4383
$h(1)$	6.6501	1.0145	1.2080	1.8399
$h(2)$	4.5466	-2.4147	-2.3915	-3.0599
$h(3)$	4.7278	0.5306	-1.4995	-0.6803
$h(4)$	5.1116	-0.8780	1.8397	5.3169
$h(5)$	7.1848	-1.2398	1.5111	6.1945
$g(0,0)$	3.4676	1.8247	-0.8796	3.7098
$g(0,1)$	8.5345	1.1147	2.1813	0.6935
$g(0,2)$	4.6133	2.4419	0.5851	-0.5799
$g(0,3)$	0.9336	-1.4757	0.4633	-0.0428
$g(0,4)$	8.2602	0.3734	0.4705	-0.4795
$g(1,1)$	6.7420	3.0117	0.9524	5.2397
$g(1,2)$	3.4231	0.6770	1.5778	4.5719
$g(1,3)$	5.9950	2.9727	0.6525	1.9149
$g(1,4)$	4.2657	-0.8576	1.3171	-5.3846
$g(2,2)$	6.8185	4.0588	1.4162	1.2048
$g(2,3)$	2.3524	3.0529	3.6863	7.1106
$g(2,4)$	6.0669	0.1657	-1.3885	-3.1878
$g(3,3)$	10.1462	0.9998	4.1036	5.8609
$g(3,4)$	3.8599	2.8173	2.1098	8.6596
$g(4,4)$	3.4468	-1.8829	1.6816	15.5536

Fig. 18 depicts the MSE observed when the system is tested with square input signal. The mean value of MSE noticed with gradient-search, RGA, PSO and CSA is 0.0023, 0.0016, 0.0016 and 8.9512×10^{-4}, respectively when squared signal is applied at the input of the system. From the graphical results and numerical values of MSE, one can conclude that CSA provides a good approximation to the nonlinear fractional delay second order Volterra system coefficients compared to other applied optimization algorithms. The performance of these algorithms is arranged as, CSA $>$ PSO $=$ RGA $>$ GS.

Table XI summarizes the kernel parameters of Volterra

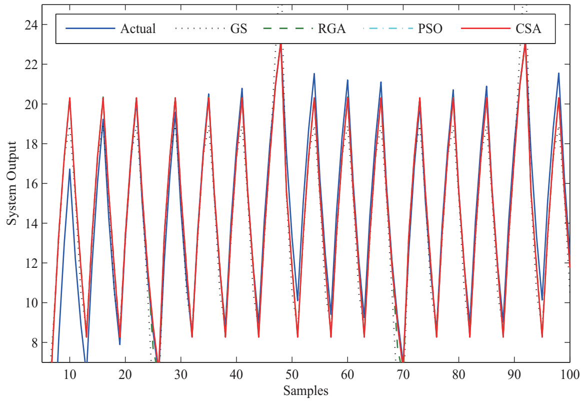

Fig. 19. Comparison of actual nonlinear system output with second order fractional delay Volterra system model output using gradient search, RGA, PSO and CSA for noisy square input signal in example 2.

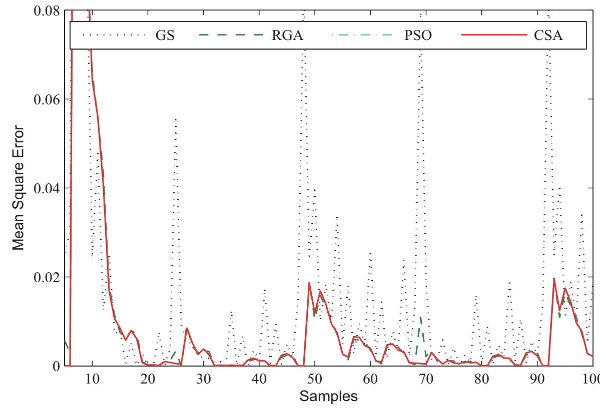

Fig. 20. Comparison of MSE for second order fractional delay Volterra system model output using gradient search, RGA, PSO and CSA for noisy square input signal in example 2.

TABLE XI
KERNEL PARAMETERS OF SECOND ORDER FRACTIONAL DELAY VOLTERRA SYSTEM MODEL USING GRADIENT SEARCH, RGA, PSO AND CSA BASED METHODS FOR NOISY SQUARE INPUT SIGNAL FOR EXAMPLE 2.

Kernel Parameters	Gradient search	RGA	PSO	CSA
$h(0)$	2.0734	2.0628	6.1067	3.5476
$h(1)$	1.2933	-0.4618	4.7040	-0.7788
$h(2)$	1.3408	1.0626	2.0808	-2.4513
$h(3)$	3.3256	-0.0636	2.7889	0.0166
$h(4)$	0.0297	0.2113	6.8997	0.8947
$h(5)$	1.8156	1.5619	4.4455	-1.2868
$g(0,0)$	3.4476	2.6711	3.7141	1.8125
$g(0,1)$	4.0397	-0.8521	3.0184	0.1079
$g(0,2)$	-1.3687	-1.2592	1.6243	-0.4851
$g(0,3)$	-1.3194	-0.4355	-2.9203	-1.7564
$g(0,4)$	1.1708	-2.5746	0.7713	-3.0847
$g(1,1)$	5.1978	8.6357	5.4871	3.6384
$g(1,2)$	5.3892	1.6294	3.9281	2.8376
$g(1,3)$	0.0668	-1.2428	3.6944	1.6550
$g(1,4)$	1.6461	-0.8980	-3.9143	-3.6949
$g(2,2)$	3.4492	3.1523	1.0847	1.3733
$g(2,3)$	3.5700	4.9677	6.3690	3.5562
$g(2,4)$	0.9387	-4.7134	-0.9484	-1.3355
$g(3,3)$	2.6421	0.4263	3.3774	0.7597
$g(3,4)$	2.3118	3.6257	6.6161	2.9279
$g(4,4)$	4.4797	6.1514	5.4867	3.3231

TABLE XII
KERNEL PARAMETERS OF SECOND ORDER FRACTIONAL DELAY VOLTERRA SYSTEM MODEL USING GRADIENT SEARCH, RGA, PSO AND CSA BASED METHODS FOR RANDOM INPUT SIGNAL FOR EXAMPLE 2.

Kernel Parameters	Gradient search	RGA	PSO	CSA
$h(0)$	1.0418	1.1669	1.1405	1.1775
$h(1)$	0.2133	-0.5722	0.1031	-0.2277
$h(2)$	-7.4315	-7.5149	-6.8923	-7.5111
$h(3)$	-4.4032	-4.4664	-4.3393	-4.4983
$h(4)$	-2.5474	-2.5632	-2.9064	-2.7099
$h(5)$	-0.0057	-0.6688	-1.5127	-1.3608
$g(0,0)$	-0.0514	-1.1548	-0.3505	-0.6698
$g(0,1)$	-0.1563	0.5335	-0.2269	-0.3422
$g(0,2)$	2.6338	2.7008	2.5176	2.5249
$g(0,3)$	0.7263	1.2250	0.8140	0.8262
$g(0,4)$	2.1507	1.7291	1.6591	1.5885
$g(1,1)$	-1.5875	-1.5097	-0.7013	-1.4393
$g(1,2)$	0.6341	-0.3207	-0.3789	-0.3649
$g(1,3)$	1.7894	1.6511	1.4016	1.3383
$g(1,4)$	2.2614	1.6739	0.8213	1.2354
$g(2,2)$	0.1802	0.2774	0.3187	0.1962
$g(2,3)$	-0.7735	-1.1470	-0.6745	-0.8926
$g(2,4)$	1.4143	1.2553	1.4124	1.2015
$g(3,3)$	0.9802	1.0123	0.5143	0.8377
$g(3,4)$	-0.7264	-0.3468	-0.1677	-0.7615
$g(4,4)$	2.6381	1.9273	0.9942	1.2558

system with noisy square input signal. The comparison of output response of the system analyzed with noisy square input using gradient-search, RGA, PSO and CSA is demonstrated in Fig. 19. The MSE remarked for the system under consideration when examined with noisy square input is shown in Fig. 20. The MSE values for second order fractional delay Volterra system with gradient-search, RGA, PSO and CSA are 0.0068, 0.0040, 0.0039 and 0.0039, respectively. Based on these MSE values, it can be finally deduced that nonlinear system identification with the second order Volterra system using CSA surpass the other employed optimization methods. The performance can be ranked as CSA $=$ PSO $>$ RGA $>$ GS. The comparison of output response of the system with random signal using gradient-search, RGA, PSO and CSA is depicted in Fig. 21. The noted values of MSE with random signal is exhibited in Fig. 22 and Table XII lists the kernel parameters of Volterra system with random input. The mean MSE values obtained are 0.0023, 0.0023, 0.0023 and 0.0022, respectively. It can be concluded from the aforementioned results that the CSA based nonlinear system identification outperforms all other reported algorithm in terms of MSE. The order of the algorithm based on its performance is given as CSA $>$ PSO $=$ RGA $=$ GS. Table XII shows the comparative numerical values of different characteristics like maximum, minimum, mean, variance and standard deviation of mean square error of proposed second order Volterra system using fractional delay for five different input signals using gradient-search, RGA, PSO and CSA algorithms. It is observed that the MSE value observed with CSA is lower as compared to other algorithms with all input signals. From Figs. 13-22 and statistically analyzed results from Table XIII, it is evident that

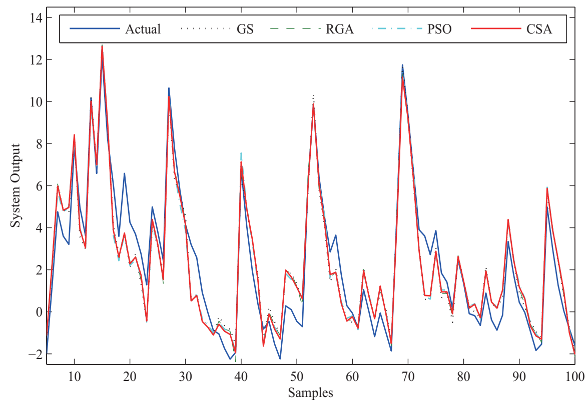

Fig. 21. Comparison of actual nonlinear system output with second order fractional delay Volterra system model output using gradient search, RGA, PSO and CSA for random input signal in example 2.

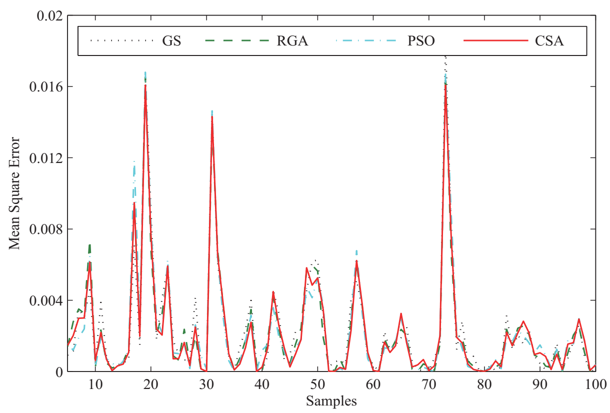

Fig. 22. Comparison of MSE for second order fractional delay Volterra system model output using gradient search, RGA, PSO and CSA for the random input signal in example 2.

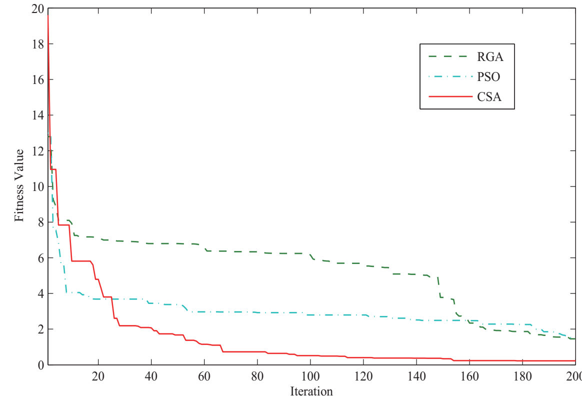

Fig. 23. Convergence profile for RGA, PSO and CSA for nonlinear system identification using second order fractional delay Volterra system model in example 1.

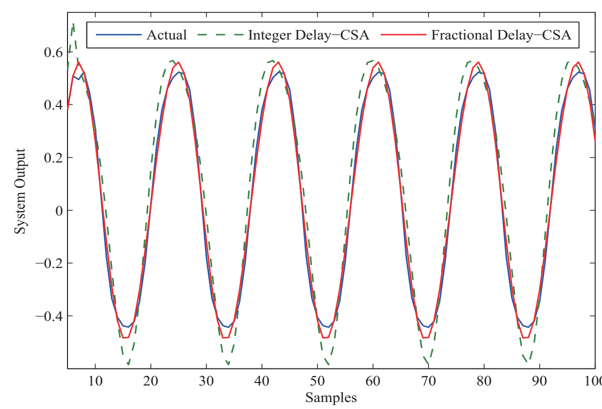

Fig. 24. Comparison of third order integer delay Volterra system output with second order fractional delay Volterra system model output using CSA for sinusoidal input signal in example 1.

with all input signals, the proposed nonlinear system modeling method based on fractional delay second order Volterra system produced minimum MSE compared to that of the gradient-search, RGA and PSO. Finally, it can be concluded that CSA based second order fractional delay Volterra system identification method gives superior results compared to other reported algorithms with all the input signals. In order to demonstrate the effectiveness of the proposed method in terms of convergence rate, Fig. 23 shows the convergence of MSE obtained, for example 1 tested with sinusoidal input. Similar plots have also been obtained for the example 1 and 2 with different input signals which are not shown here.

C. Comparative Analysis

1) Comparison with a Third Order Integer Delay Volterra System: The superiority of the proposed Volterra system identification method is demonstrated by comparing the results with a nonlinear Volterra system using an integer delay. Fig. 24 shows the comparison of approximated output of proposed second order fractional delay Volterra system in example 1, eq. (56) with the output of a third order integer delay Volterra system when both the systems are subjected to the sinusoidal input signal. From the visual analysis of Fig. 24, it can be inferred that a better approximation of the nonlinear unknown system is achieved using the proposed second order fractional delay Volterra system to its integer counterpart of third order. The mean values of MSE for integer and fractional delay system are obtained to be 3.2914×10^{-3} and 8.6450×10^{-4}. Thus, the introduction of fractional delay in the Volterra system identification technique leads to a better approximation with the involvement of less number of multipliers (due to order reduction) and low energy consumption in comparison to the integer delay systems. Similar graphical results are obtained for example 1 and example 2 with different input signals, which are not reported here.

2) Comparison with the Existing Techniques: The comparison of the proposed second order fractional delay Volterra system with the other reported nonlinear system modeling method has been presented in Table XIV. The observations are made on the MSE values of the existing methodologies

TABLE XIII
STATISTICAL COMPARISON OF MEAN SQUARE ERROR FOR THE IDENTIFICATION OF NONLINEAR SYSTEM WITH DIFFERENT INPUT SIGNAL USING GRADIENT SEARCH, RGA, PSO AND CSA BASED METHODS FOR EXAMPLE 2.

Input Signal	Algorithm	Mean Square Error (MSE)				
		Max	Min	Mean	Variance	Standard Deviation
$x(n) = 0.8\sin(\frac{\pi}{9}n)$	GS	0.3326	2.9491×10^{-6}	0.0220	0.0020	0.0443
	RGA	0.1787	6.3511×10^{-5}	0.0155	7.7422×10^{-4}	0.0278
	PSO	0.1815	5.0935×10^{-5}	0.0154	7.7765×10^{-4}	0.0279
	CSA	0.1605	2.0301×10^{-8}	0.0151	7.4953×10^{-4}	0.0274
$x(n) = 0.8\sin(\frac{\pi}{9}n) + w(n)$	GS	0.2117	1.3487×10^{-5}	0.0154	7.4549×10^{-4}	0.0273
	RGA	0.2129	8.0414×10^{-7}	0.0158	7.4020×10^{-4}	0.0272
	PSO	0.2117	1.3487×10^{-5}	0.0154	7.4549×10^{-4}	0.0273
	CSA	0.1947	5.4686×10^{-8}	0.0137	6.0762×10^{-4}	0.0246
$x(n) = 0.4\text{square}(n)$	GS	0.0282	6.2620×10^{-7}	0.0023	2.7538×10^{-5}	0.0052
	RGA	0.0311	3.3186×10^{-8}	0.0016	1.6176×10^{-5}	0.0040
	PSO	0.0177	2.5478×10^{-7}	0.0016	8.8820×10^{-6}	0.0030
	CSA	0.0115	4.8790×10^{-14}	8.9512×10^{-4}	3.4831×10^{-6}	0.0019
$x(n) = 0.4\text{square}(n) + w(n)$	GS	0.0975	5.2938×10^{-6}	0.0068	1.6614×10^{-4}	0.0129
	RGA	0.0880	1.8202×10^{-9}	0.0040	1.3367×10^{-4}	0.0116
	PSO	0.0865	1.0412×10^{-14}	0.0039	1.3078×10^{-4}	0.0114
	CSA	0.0865	3.7582×10^{-11}	0.0039	1.3078×10^{-4}	0.0114
$x(n) = \text{rand}(n)$	GS	0.0184	2.3431×10^{-6}	0.0023	1.0509×10^{-5}	0.0032
	RGA	0.0166	6.7229×10^{-7}	0.0023	9.7658×10^{-6}	0.0031
	PSO	0.0168	7.0138×10^{-8}	0.0023	1.0681×10^{-5}	0.0033
	CSA	0.0161	1.5406×10^{-9}	0.0022	9.6443×10^{-6}	0.0031

TABLE XIV
COMPARISON OF THE PROPOSED FRACTIONAL DELAY BASED NONLINEAR SYSTEM IDENTIFICATION WITH OTHER REPORTED METHODS.

Method	Example	Algorithm	Input signal	Memory size (p)	MSE
Rashedi et al. [22]	Example 1	GSA	White noise sequence $x(k)$ + noise $\eta(k) \in [-0.001, 0.001]$	-	3.91×10^{-7}
		GSA	White noise sequence $x(k)$ + noise $\eta(k) \in [-0.01, 0.01]$	-	4.23×10^{-5}
Chang [14]	Example 1	IPSO	$x(n) = 0.8\cos(\frac{\pi}{9}n)$	5	0.00929002
		IPSO	$x(n) = 0.8\cos(\frac{\pi}{9}n)$	8	0.00491307
		IPSO	$x(n) = rand(n)$	5	0.00556229
		IPSO	$x(n) = rand(n)$	8	0.00260959
Present Study	Example 1	CSA	$x(n) = 0.8\sin(\frac{\pi}{9}n)$	5	8.6450×10^{-4}
		CSA	$x(n) = 0.8\sin(\frac{\pi}{9}n) + w(n)$	5	5.3905×10^{-4}
		CSA	$x(n) = 0.4\text{square}(n)$	5	5.4547×10^{-4}
		CSA	$x(n) = 0.4\text{square}(n) + w(n)$	5	5.9464×10^{-4}
		CSA	$x(n) = \text{rand}(n)$	5	1.9658×10^{-4}
	Example 2	CSA	$x(n) = 0.8\sin(\frac{\pi}{9}n)$	5	7.4953×10^{-4}
		CSA	$x(n) = 0.8\sin(\frac{\pi}{9}n) + w(n)$	5	0.0137
		CSA	$x(n) = 0.4\text{square}(n)$	5	8.9512×10^{-4}
		CSA	$x(n) = 0.4\text{square}(n) + w(n)$	5	0.0039
		CSA	$x(n) = \text{rand}(n)$	5	0.0022

for nonlinear system identification problem.

VI. CONCLUSION

The objective of this work is to design an efficient method for nonlinear system approximation with the use of fractional delays. The novelty is that in implementing the fractional order delays, the higher order nonlinearities are estimated using a low order Volterra model with higher accuracy by using adept optimization methodologies. A discrete model of the estimation problem is formulated in order to simulate the proposed method in MATLAB. The Gradient-search method is developed for the system identification problem and optimizing the Volterra system parameters. To further optimize the system coefficients, different stochastic algorithms are applied. Two design examples are presented using nonlinear benchmark models with five different input signals and close approximations of the unknown system are analyzed in figures and tables, comparing the proposed gradient-search, RGA, PSO and CSA techniques. The statistical analysis of the estimated results is portrayed by computing the mean, variance and standard deviation of the computed error while performing multiple simulations. The accuracy in results is achieved with the globally convergent and widely applied metaheuristic optimization, CSA. A comparison between the various optimization technique is made. It can be concluded that the proposed method incorporating the fractional delay systems, delivers an effective approximation to an unknown nonlinear system modeled using a second order Volterra function.

References

[1] Widrow B, Strearns S D. *Adaptive signal processing*. NJ, Prentice-Hall: Englewood Cliffs, 1985

[2] Panda G, Pradhan P M, Majhi B. IIR system identification using cat swarm optimization. *Expert Systems with Applications*, 2011, **38**(10): 12671-12683

[3] Majhi B, Panda G, Choubey A. Efficient scheme of pole-zero system identification using particle swarm optimization technique. In *Proc. IEEE Congress on Evolutionary Computation*, 2008, 446-451

[4] Saha S K, Kar R, Mandal D, Ghoshal S P. Harmony search algorithm for infinite impulse response system identification. *Computers & Electrical Engineering*, 2014, **40**(4): 1265-1285

[5] Patwardhan AP, Patidar R, George NV. On a cuckoo search optimization approach towards feedback system identification. *Digital Signal Processing*, 2014, **32**: 156-163

[6] Upadhyay P, Kar R, Mandal D, Ghoshal S P. Craziness based particle swarm optimization algorithm for IIR system identification problem. *International Journal of Electronics and Communications*, 2014, **68**(5): 369-378

[7] Upadhyay P, Kar R, Mandal D, Ghoshal S P, Mukherjee V. A novel design method for optimal IIR system identification using opposition based harmony search algorithm. *Journal of the Franklin Institute*, 2014, **351**(5): 2454-2488

[8] Jiang S, Wang Y, Ji Z. A new design method for adaptive IIR system identification using hybrid particle swarm optimization and gravitational search algorithm. *Nonlinear Dynamics*, 2015, **79**(4): 2553-2576

[9] Mathews V J. Adaptive polynomial filters. *IEEE Signal Processing Magazine*, 1991, **8**: 10-26

[10] Rhee S H, Tsang P H S, Yen S W. Friction-induced noise and vibrations of disc brakes. *Wear*, 1989, **133**: 39-45

[11] Kerschen G, Worden K, Vakakis A F, Golinval J C. Past, present and future of nonlinear system identification in structural dynamics. *Mechanical Systems and Signal Processing*, 2006, **20**(3): 505-592

[12] Nelles O. *Nonlinear System Identification: From Classical Approaches to Neural Networks and Fuzzy Models*. Springer, 2001

[13] Giannakis G B, Serpedin E. A bibliography on nonlinear system identification. *Signal Processing*, 2001, **81**: 533-580

[14] Chang W D. Volterra filter modeling of nonlinear discrete-time system using improved particle swarm optimization. *Digital Signal Processing*, 2012, **22**: 1056-1062

[15] Boyd S, Chua L O. Fading memory and the problem of approximating nonlinear operators with Volterra series. *IEEE Transactions on Circuits and Systems*, 1985, **32**(11): 1150-1161

[16] Gotmare A, Patidar R, George N V. Nonlinear system identification using a cuckoo search optimized adaptive Hammerstein model. *Expert Systems with Applications*, 2015, **42**: 2538-2546

[17] Sloss B G, Blyth W F. A Walsh function method for a non-linear Volterra integral equation. *Journal of the Franklin Institute*, 2003, **340**(1): 25-41

[18] Wahlberg B O. System identification using Kautz models. *IEEE Transactions on Automatic Control*, 1994, **39**(6): 1276-1282

[19] Aadaleesan P, Miglan N, Sharma R, Saha P. Nonlinear system identification using Wiener type Laguerre avelet network model. *Chemical Engineering Science*, 2008, **63**(15): 3932-3941

[20] Gallman P G. An iterative method for the identification of nonlinear systems using a Uryson model. *IEEE Transactions on Automatic Control*, 1975, **20**(6): 771-775

[21] Narendra K S, Parthasarathy K. Identification and control of dynamical systems using neural networks. *IEEE Transactions on Neural Networks*, 1990, **1**(1): 4-27

[22] Rashedi E, Hossien N P, Saeid S. Filter modeling using gravitational search algorithm. *Engineering Applications of Artificial Intelligence*, 2011, **24**(1): 117-122

[23] Volterra V. *Theory of functionals and of integral and integrodifferential equations*. Dover, New York, 2005

[24] Holland JH. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor, 1975

[25] Aggarwal A, Rawat T K, Kumar M, Upadhyay D K. Optimal design of FIR high pass filter based on L_1 error approximation using real coded genetic algorithm. *Engineeing Science and Technology, an International Journal*, 2015, **18**(4): 594-602

[26] Kumar M, Rawat T K. Optimal design of FIR fractional order differentiator using cuckoo search algorithm. *Expert Systems with Applications*, 2015, **42**(7): 3433-3449

[27] Kennedy J, Eberhart R. Particle Swarm Optimization. In *Proceedings of IEEE International Conference on Neural Network*, 1995, **4**: 1942-1948

[28] Kumar M, Rawat T K. Optimal Fractional Delay-IIR Filter Design Using Cuckoo Search Algorithm. *ISA Transactions*, 2015, **59**: 39-54

[29] Yang X S, Deb S. Cuckoo search via Lévy flights. In *Proceedings of World Congress on Nature and Biologically Inspired Computing, IEEE Publications*, USA, 2009, 210-214

Manjeet Kumar graduated from Kurukshetra University, Haryana with B.Tech in Electronics & Communication Engineering in 2008. He received his M.Tech from Ambedkar Institute of Technology, GGSIPU, Delhi in 2011. He is currently pursuing Ph.D. in the area of optimal design of fractional order digital filter from Netaji Subhas Institute of Technology, University of Delhi. His research interests include digital signal processing, fractional calculus, filter design and optimization, fractional order differentiator and fractional delay filter.

Apoorva Aggarwal graduated from GGSIPU, Delhi with B.Tech in Electrical & Electronic Engineering in 2011. She received her M.Tech from GGSIPU, Delhi in 2013. She is currently pursuing Ph.D. in the area of optimal design of digital filters from Netaji Subhas Institute of Technology, University of Delhi. Her research interests include filter design and optimization, digital signal processing, nature-inspired metaheuristic optimization techniques, digital electronics and innovative projects.

Tarun Rawat received his M.Tech and Ph.D. from NSIT, University of Delhi in 2003 and 2010, respectively. Presently, he is attached with NSIT, Delhi, India, as Assistant Professor in the Department of Electronics and Communication Engineering. He has published more than 30 research papers in international journals and two books: Signal and System, Digital Signal Processing in Oxford University Press. His research interests include VLSI signal processing, digital signal processing, statistical signal processing and filter optimization via nature-inspired computational techniques.

Harish Parthasarathy received B.Tech. in 1990 (from Indian Institute of Technology Kanpur, India) and Ph.D. (from Indian Institute of Technology Delhi, India) in 1994, both in Electrical Engineering. He worked as Assistant Professor during 1994-1997 at Indian Institute of Tehnology Bombay in Electrical Engineering Department and during 1997-1998 at Indian Institute of Technology Kanpur in Electrical Engineering Department. He worked as Application Engineer at ST MicroElectronics, Delhi, during 1998-2000 before joining Netaji Subhas Institute of Technology, New Delhi, in 2000, where he worked as Assistant Professor till June 2007. Since July 2007, he has been working as Professor in the Division of Electronics and Communication Engineering at Netaji Subhas Institute of Technology, New Delhi. His teaching and research interests are in the areas of circuits and systems, signal processing, stochastic nonlinear filters, electromagnetics and group representations and he has published 40 papers in various international journals of repute.