Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 September 2018 doi:10.20944/preprints201809.0606.v1

System identification in the delta domain: a unified approach using FAGWO algorithm

Souvik Ganguli[1*], Gagandeep Kaur[1], Prasanta Sarkar[2]

[1] Department of Electrical & Instrumentation Engineering, Thapar University, Patiala-147004, Punjab, India

[2]Department of Electrical Engineering, National Institute of Technical Teachers' Training & Research, Kolkata-700106, West Bengal, India

***Corresponding Author E-mail:** souvik.ganguli@thapar.edu

Abstract- The identification of linear dynamic systems with static nonlinearities in the delta domain has been presented in this paper applying a firefly based hybrid meta-heuristic algorithm integrating Firefly algorithm (FA) and Gray wolf optimizer (GWO). FA diversifies the search space globally while GWO intensifies the solutions through its local search abilities. A test system with continuous polynomial nonlinearities has been considered for hammerstein and wiener system identification in continuous, discrete and delta domain. Delta operator modelling unifies system identification of continuous-time systems with discrete domain at higher sampling frequency. Pseudo random binary sequence, contaminated with white noise, has been taken up as the input signal to estimate the unknown model parameters as well as static nonlinear coefficients. The hybrid algorithm not only outperforms the parent heuristics of which they are constituted but also proves better as compared to some standard and latest heuristic approaches reported in the literature.

Keywords:- System idenfication; delta operator modelling; Firefly algorithm gray wolf optimizer (FAGWO)

Abbreviations

ABC	Artificial bee colony
ALO	Ant lion optimization
BFA	Bacterial foraging algorithm
DA	Dragonfly algorithm
DE	Differential evolution
FA	Firefly algorithm
GOA	Grasshopper optimization algorithm
GWO	Gray wolf optimizer
FAGWO	Firefly algorithm gray wolf optimizer
MFO	Moth flame optimization

 doi:10.20944/preprints201809.0606.v1

MVO	Multi-verse optimization
PRBS	Pseudo random binary sequence
PSO	Particle swarm optimization
PSOGSA	Particle swarm optimization gravitational search algorithm
SCA	Sine cosine algorithm
SSA	Salp swarm algorithm
WOA	Whale optimization algorithm

1. Introduction

System identification is an approximate modeling for a specific application on the basis of observed data and prior system knowledge. The literature on the system identification problem is extensive [1]. Meta-heuristic algorithms and their hybridizations have also taken active participation in the literature of system identification and control [2-3]. Linear systems with static nonlinearities at the input termed as the Hammerstein model, and linear systems with static nonlinearities at the output known as the Wiener model are two widely prevailing models used for system identification [4]. Parameter estimation of these models has traditionally been carried out in discrete-time using either shift operator in the time domain and z-transformation in the complex domain via soft computing approaches [5-8]. Likewise, a huge volume of literature also exists in the continuous-time system [9].

In the literature of identification and control, there have been several methods developed over the last five decades on discrete time systems utilizing the potential of digital computers. Concurrently, there has been a similar attempt in developing methods in continuous time identification and control in system theory due to the very fact that the physical signals are continuous time in nature. Modelling, identification and control with the help of delta operator is a holistic approach in which the signals and systems are modelled in discrete domain and leads to converge to its corresponding continuous time signals and systems at a high sampling frequency thus unifying both discrete and continuous time signals and systems [10].

Though hammerstein and wiener model identification with meta-heuristic approaches are quite popular in the discrete-time domain, similar analyses are rarely investigated for continuous-time systems. Hence system identification with hybrid meta-heuristic techniques can be thought of to unify both continuous and discrete time systems leveraging the properties of delta operator. A hybrid algorithm namely FAGWO developed by Ganguli *et al.* [11] has been utilized to identify the unknown hammerstein and wiener model parameters in a unified delta operator framework. Continuous and discrete time analyses are carried out simultaneously to highlight upon the usefulness of delta operator modelling.

The rest of the paper is developed as follows. Section 2 discusses the problem of wiener and hammerstein models in the delta domain. Section 3 gives a brief overview of the parent algorithms FA and GWO algorithms. Section 4 discusses the hybrid FAGWO algorithm. Section 5 presents the results while Section 6 concludes the paper.

2. Statement of the problem

2.1 Delta operator modelling

The δ-operator, an alternative formulation of discrete-time system [10] is defined in the time domain as:

$$\delta = \frac{q-1}{\Delta} \tag{1}$$

where Δ denotes the sampling period while q is the forward shift operator. Operating δ on a differential signal $x(t)$ gives

$$\delta x(t) = \frac{x(t+\Delta) - x(t)}{\Delta} \tag{2}$$

It is straightforward to see that

$$\lim_{\Delta \to 0} \delta x(t) = \frac{d}{dt} x(t) \tag{3}$$

which indicate the close relationship between the discrete-time δ-operator and the continuous-time differential operator $\frac{d}{dt}$ at high sampling rate.

Similarly relation exists in the complex domain as well. The delta transform operator γ is defined as

$$\gamma = \frac{z-1}{\Delta} \tag{4}$$

2.2 Wiener system identification in delta domain

The wiener model is characterized by a linear dynamic part followed by a static nonlinearity shown in the Fig. 1.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 September 2018 doi:10.20944/preprints201809.0606.v1

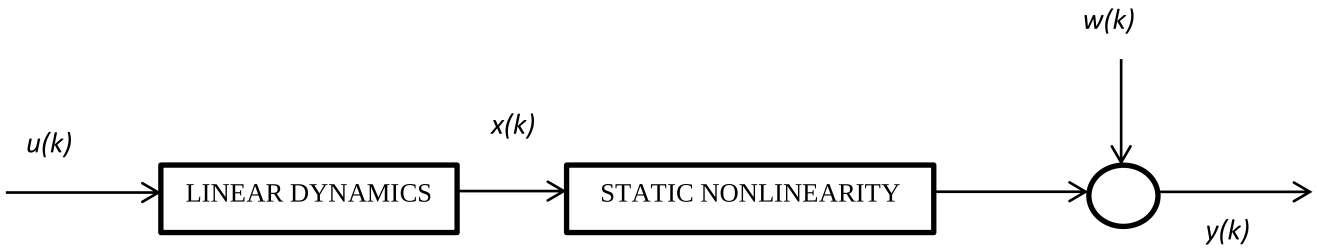

Fig. 1. Wiener model

The intermediate signal $x(k)$ is however not available for measurement. Assuming that the nonlinearity has a known structure with unknown parameters, the wiener model is represented by the following equations in the delta domain:

$$y(k) = \frac{B(\delta)}{A(\delta)} u(k) \tag{5}$$

where

$$B(\delta) = b_0\delta^m + b_1\delta^{m-1} + b_2\delta^{m-2} + \cdots + b_m$$

$$A(\delta) = a_0\delta^n + a_1\delta^{n-1} + a_2\delta^{n-2} + \cdots + a_n$$

B and A are two polynomials of unknown orders and coefficients, u and y represents system input and output respectively. The non-measured intermediate variable $x(k)$ is the input to the static nonlinearity given by-

$$y(k) = f(\theta, x(k)) + w(k) \tag{6}$$

$f\,()$ is any nonlinear function and θ is a set of parameters describing the nonlinearity. Thus, the problem of the wiener model identification is to estimate the unknown parameters $b_0, \cdots, b_m, a_1, \cdots, a_n$ from the input-output data. Further, 'w' represents the white gaussian noise of fixed signal-to-noise ratio (SNR). In case the structure of the nonlinear function $f\,()$ is not known, a polynomial of degree L can be used to approximate the nonlinearity as-

$$y(k) = c_1 x(k) + c_2 x^2(k) + \cdots + c_L x^L(k) + w(k) \tag{7}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 September 2018 doi:10.20944/preprints201809.0606.v1

2.3 Hammerstein model identification in the delta domain

Hammerstein model is a good example of nonlinear dynamic systems in which nonlinear static system and linear dynamic systems are separated in different order. The block diagram of Hammerstein model is shown below in Fig. 2.

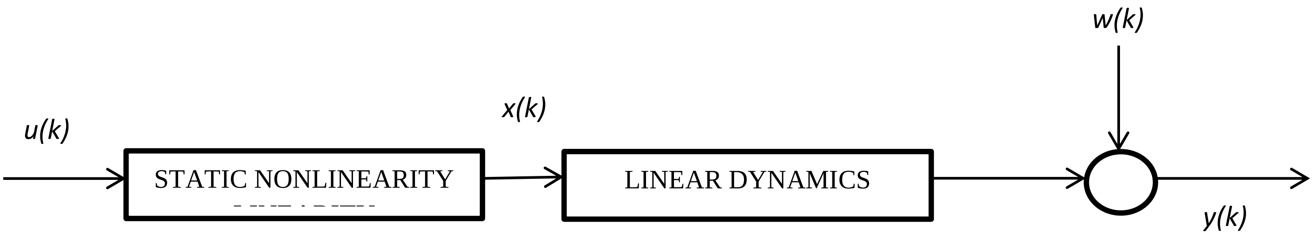

Fig. 2. Hammerstein model

The input/output relation of discrete-delta system is represented as:

$$y(k) = \frac{B(\delta)}{A(\delta)}x(k) + w(k) = G(\delta)x(k) + w(k) \tag{8}$$

with $$B(\delta) = b_0\delta^m + b_1\delta^{m-1} + b_2\delta^{m-2} + \cdots + b_m$$

$$A(\delta) = a_0\delta^n + a_1\delta^{n-1} + a_2\delta^{n-2} + \cdots + a_n$$

where $$x(k) = f(\theta, u(k))$$

The objective is to estimate the parameters $a_1, \cdots, a_n, b_0, \cdots, b_m, \theta$ through the minimization of mean square error (MSE) obtained by the difference between actual and estimated values defined as-

$$J = \frac{1}{N}\sum_{k=1}^{N}\left[y(k) - \hat{y}(k)\right]^2 \tag{9}$$

where $$\hat{y}(k) = \frac{\hat{B}(\delta)}{\hat{A}(\delta)}\hat{x}(k)$$

$$\hat{B}(\delta) = \hat{b}_0\delta^m + \hat{b}_1\delta^{m-1} + \hat{b}_2\delta^{m-2} + \cdots + \hat{b}_m$$

with

$$\hat{A}(\delta) = \hat{a}_0\delta^n + \hat{a}_1\delta^{n-1} + \hat{a}_2\delta^{n-2} + \cdots + \hat{a}_n$$

$$\hat{x}(k) = f(\hat{\theta}, u(k))$$

Here 'N' denotes the number of input-output data points used in the identification and the parameter estimates $\hat{a}_1, \cdots, \hat{a}_n, \hat{b}_0, \cdots, \hat{b}_m, \hat{\theta}$ are found by minimizing the fitness function defined in equation (9).

3. Brief overview of FA and GWO algorithms

In this section, the two parent algorithms viz. FA and GWO are introduced to set up an appropriate background for the hybrid method. The hybrid technique is then utilized to solve hammerstein and wiener model identification in the delta domain.

3.1 Firefly algorithm (FA)

Xin-She Yang developed the firefly algorithm [12] considering the following assumptions:

- All fireflies are unisex so that one firefly is attracted to other fireflies regardless of their sex
- Attractiveness is proportional to their brightness, thus for any two flashing fireflies, the less bright one will move towards the brighter one. The attractiveness is proportional to the brightness and they both decrease as their distance increases. If no one is brighter than a particular firefly, it moves randomly
- The brightness or light intensity of a firefly is affected or determined by the landscape of the objective function to be optimized.

In the firefly algorithm, there are two salient aspects: the variation of light intensity and formulation of attractiveness. For the sake of simplicity, it is assumed that the attractiveness of a firefly is determined by its brightness or light intensity which in turn is correlated with the encoded objective function. For maximum optimization problems, the brightness $I(\underline{x})$ of a firefly at a particular location $(\underline{x})$ is chosen as $I(\underline{x}) \alpha F(\underline{x})$. The attractiveness β is relative; it will be seen in the eyes of the beholder or to be judged by the other fireflies. So it will vary with the distance r_{ij} between firefly i and firefly j. As light intensity decreases with the distance from its source, the light is also absorbed in the media, hence it is concluded that the attractiveness should vary with the degree of absorption. The light intensity $I(r)$ varies with the distance *r* monotonically and exponentially as:

$$I = I_0 e^{-\gamma r} \tag{10}$$

where I_0 is the original light intensity and γ is the light absorption coefficient. As a firefly's attractiveness is proportional to the light intensity seen by adjacent fireflies, the attractiveness β of a firefly is defined as

$$\beta = \beta_0 e^{-\gamma r^2} \tag{11}$$

where β_0 is the attractiveness at r=0. It is worth mentioning that the exponent γr^2 can be replaced by other functions such as γr^m when $m > 0$. The distance between any two fireflies *i* and *j* at $\underline{x}_i$ and $\underline{x}_j$ respectively, is the Cartesian distance is calculated as

$$r_{ij} = \left\| \underline{x_i} - \underline{x_j} \right\| = \sqrt{\sum_{d=1}^{n} (\underline{x_{id}} - \underline{x_{jd}})^2} \tag{12}$$

where $\underline{x_{i,d}}$ is the *d*th component of the spatial coordinate $\underline{x_i}$ of *i*th firefly. The movement of a firefly *i* is attracted to another more attractive (brighter) firefly *j* is determined by

$$\underline{x_i} = \underline{x_j} + \beta_0 e^{-\gamma r^2} \left(\underline{x_i} - \underline{x_j} \right) + \alpha(rand - 0.5) \tag{13}$$

3.2 Gray wolf optimizer (GWO)

Gray wolf optimizer (GWO) is a population based meta-heuristic algorithm 7ehaviour the leadership hierarchy and hunting mechanism of gray wolves found in nature [13]. Gray wolves are considered as apex predators, belonging at the top of the food chain. They live in groups (packs), each group containing 5-12 members on average. All the members in the group maintain a strict social hierarchy as shown in Fig. 3.

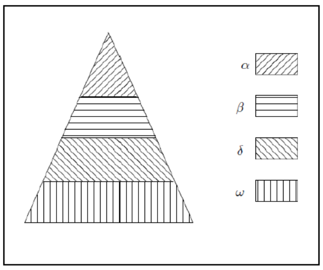

Fig. 3. Social hierarchy of gray wolves

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 September 2018 doi:10.20944/preprints201809.0606.v1

As seen from Fig. 3, four types of gray wolves such as alpha, beta, delta, and omega are employed for simulating the leadership hierarchy. In the hierarchy, alpha (α) is considered the most dominating member among the group. The rest of the subordinates to α are beta (β) and delta (δ), which help to control the majority of wolves in the hierarchy that are considered as omega (ω).The ω wolves are of the lowest ranking in the social hierarchy.

In GWO algorithm, the hunting is guided by α, β and δ. The ω solutions follow these three wolves. During hunting, the gray wolves encircle the prey. The mathematical model of the encircling 8ehaviour is presented as:

$$\vec{D} = \left|\vec{C}\vec{X}_p(t) - \vec{X}(t)\right| \quad (14)$$

$$\vec{X}(t+1) = \vec{X}_p(t) - \vec{A}.\vec{D} \quad (15)$$

where t indicates the current iteration, $\vec{A}$ and $\vec{C}$ are the coefficient vectors, $\vec{X}_p$ denotes the position vector of the prey while $\vec{X}$ represents the position vector of a gray wolf. The vectors $\vec{A}$ and $\vec{C}$ are computed using the following equations:

$$\vec{C} = 2.rand_2 \quad (16)$$

$$\vec{A} = 2\vec{a}.rand_1 - \vec{a} \quad (17)$$

where $\vec{a}$ is linearly decreased from 2 to 0 over the course of iterations while $rand_1$ and $rand_2$ denote random numbers lying in the range $(0,1)$. The hunting operation of the gray wolves is usually guided by the alpha wolves. The beta and delta wolves occasionally participate in the hunting process. Thus, in the mathematical model for the hunting 8ehaviour of gray wolves, it is assumed that the alpha, beta and delta type gray wolves have better knowledge about the potential location of prey. Hence, the first three best solutions acquired are saved and the other search agents are obliged to update their positions according to the location of the best search agents. The following mathematical equations are thus framed as:

$$\vec{D}_\alpha = \left|\vec{C}_1\vec{X}_\alpha - \vec{X}\right| \quad (18)$$

$$\vec{D}_\beta = \left|\vec{C}_2\vec{X}_\beta - \vec{X}\right| \quad (19)$$

$$\vec{D}_\gamma = \left|\vec{C}_3\vec{X}_\gamma - \vec{X}\right| \quad (20)$$

Thus

$$\vec{X}_1 = \vec{X}_\alpha - \vec{A}_1.\vec{D}_\alpha \quad (21)$$

$$\vec{X}_2 = \vec{X}_\beta - \vec{A}_2.\vec{D}_\beta \quad (22)$$

$$\vec{X}_3 = \vec{X}_\gamma - \vec{A}_3.\vec{D}_\gamma \quad (23)$$

and finally

$$\vec{X}(t+1) = \frac{(\vec{X}_1 + \vec{X}_2 + \vec{X}_3)}{3} \quad (24)$$

The gray wolves complete the hunt by attacking the prey when it stops moving. In this phase, the value of $\vec{a}$ is decreased and thereby the fluctuation range of $\vec{A}$ is reduced. When $\vec{A}$ has random values in the range $[-1,1]$, the search agent's next location will be in anywhere between its current position and the position of the prey.

4. FAGWO algorithm

Ganguli *et al.* [11] integrated FA with GWO as a low level relay type heterogenous hybrid topology, coined as FAGWO algorithm. From the literature it has been found that FA can subdivide the whole population into subgroups automatically in terms of the attraction mechanism with the variation of light intensity. Further, FA can also escape from the local minima by virtue of long-distance mobility via Lévy flight. Such advantages clearly indicate that FA has good exploration capabilities. Thus FA is used to explore the solution vector globally whereas GWO is employed to exploit the solutions through its local search abilities. The search operation begins with FA with the help of initialization through a group of random agents. The computation continues with FA for a certain number of iterations to search for the global best position in the specified search domain. The search process then shifts to GWO to speed up the convergence for global optimum. Thus the hybrid algorithm finds an optimum more accurately and precisely. The pseudocode of the hybrid algorithm to solve identification problem is provided in Fig. 4 given below.

Begin:

Global search phase

Initialize the algorithm parameters:

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 September 2018 doi:10.20944/preprints201809.0606.v1

Max_iter: Maximum number of iterations

n: number of fireflies

γ: the light absorption coefficient

β_0: the initial brightness of a given firefly

D: the search domain

Define the objective function f(X) where $X=(x_1,x_2,\ldots\ldots\ldots x_d)^T$

Generate the initial population of fireflies X_i $(i=1,2,\ldots\ldots n)$

Determine the light intensity I_i *of the* i^{th} *firefly* X_i *via the fitness function computed using the following steps:*

Step1: Excite the static nonlinear system by PRBS sequence.
Step2: Generate random initial solutions for zeros and poles of the linear part, and the parameters of the nonlinearity in the appropriate search space.
Step3: Evaluate the fitness function defined in equation (9) for all possible solutions generated in Step 2.

t=1

while t<Max_iter do

for i = 1 to n (all 'n' fireflies) do

for j = 1 to n (all 'n' fireflies) do

if $(I_j > I_i)$ *then*

Move firefly i towards j

end if

Vary attractiveness with distance 'r'

Evaluate new solutions and update light intensity

end for j

end for i

Rank the fireflies and find the current best

t = t + 1

end

Local search phase

Consider the best solution obtained by FA as the initial guess for GWO

Apply GWO algorithm to search around global best, which is found by FA

Output the solution obtained from GWO

Post process results and visualization

End

Fig. 4. Pseudo code for identification algorithm using FAGWO

5. Results and discussions

The input-output relation of the continuous-time plant model to be identified [9] is given by-

$$G(\rho) = \frac{-\rho + 5}{\rho^4 + 23\rho^3 + 185\rho^2 + 800\rho + 2500} \quad (25)$$

The symbol ρ means $\frac{d}{dt}$ operator. The zero order hold discrete-time approximation of this system parameterized using shift operator with 100 Hz sampling rate is stated as-

$$G(q) = \frac{-10^{-6} \times \left(0.15537q^3 + 0.41605q^2 - 0.47402q - 0.142\right)}{q^4 - 3.7777q^3 + 5.3506q^2 - 3.3674q + 0.79453} \quad (26)$$

The delta operator form of the transfer function at the same sampling time is represented as-

$$G(\delta) = \frac{-0.8\delta + 4.5}{\delta^4 + 22\delta^3 + 180\delta^2 + 760\delta + 2200} \quad (27)$$

Two separate continuous nonlinearities viz.

$$y(k) = x(k) + 0.5x^2(k) + 0.25x^3(k) \quad (28)$$

and
$$y(k) = \frac{x(k)}{\sqrt{0.10 + 0.90x^2(k)}} \tag{29}$$

are considered respectively for identifying wiener and hammerstein model parameters in continuous, discrete and delta domain. As observed in equations (25) and (27), continuous time and delta domain parameters are in close proximity, such is also the case after estimating these parameters in the respective domains. Discrete time parameter estimation shows a slight deviation. A sample size of 255 has been taken up for conducting the experiments. The input signal in both cases have been contaminated with white gaussian noise of SNR 50 dB. A population size of 20 and maximum no. of iterations of 100 are taken up for each of the experiments. Standard parameter values available in the literature are considered for all the algorithms mentioned for comparison with the hybrid method. Since heuristic algorithms are stochastic processes, they have to be run at least more than 10 times to generate meaningful statistical measures. Therefore around twenty independent test runs are carried out for each of the algorithms to get meaningful statistical results. Wilcoxon rank sum test [14] is also performed to validate the results. The actual and estimated parameters for wiener and hammerstein model identification in continuous, discrete and delta domain are shown respectively in Tables 1-6. The best estimates in these tables are marked in bold letters.

Table 1. Actual and estimated values of wiener system in continuous time

Types of values	Algorithms	b_0	b_1	a_1	a_2	a_3	a_4	c_1	c_2	c_3
Actual		-1	5	23	185	800	2500	1	0.5	0.25
Estimated	FAGWO	**-0.9939**	**4.9964**	**22.7272**	**184.8096**	**799.0473**	**2502.8094**	**0.9934**	**0.4979**	**0.2459**
	FA [12]	-1.0476	5.4998	22.1545	196.7104	808.4689	2661.3213	1.0550	0.5214	0.2250
	BFA [15]	-0.9452	5.2816	20.7916	195.4189	845.0546	2640.7955	1.0563	0.5282	0.2641
	FPA [12]	-0.9007	4.5021	20.7087	166.5247	720.0390	2550.2607	0.9005	0.4523	0.2262
	GWO [13]	-1.0099	5.0738	22.0321	188.6011	810.1951	2307.3943	0.9762	0.4631	0.2679
	PSOGSA[16]	-0.9553	4.7903	25.1954	189.6628	783.8390	2317.8042	0.9805	0.5347	0.2250
	PSO	-1.0608	4.5152	23.9994	174.4268	767.0963	2519.7977	1.0164	0.4867	0.2698
	DE	-0.9745	5.3161	21.8083	199.2662	848.2653	2294.8964	1.0626	0.4842	0.2368
	ABC	-0.9328	4.7244	21.8578	174.1058	750.2416	2597.0666	0.9433	0.4680	0.2360
	ALO [17]	-0.9155	4.7343	25.2560	180.2339	780.1791	2277.2918	1.0461	0.5500	0.2673
	DA [18]	-1.0266	4.9227	21.0482	176.3234	752.4213	2477.9154	1.0438	0.4802	0.2384

MFO [19]	-0.9610	4.8545	20.8164	195.1274	877.9816	2721.4627	0.9757	0.5488	0.2624
GOA [20]	-1.0613	4.8990	24.1544	197.4717	776.7262	2749.9633	1.0685	0.4546	0.2670
SCA [21]	-1.0397	5.0248	25.2236	196.0021	817.6265	2472.3345	0.9273	0.5448	0.2742
SSA [22]	-0.9115	4.7288	20.7738	203.5000	814.4726	2750.0000	0.9872	0.5212	0.2737
WOA [23]	-1.0980	4.5930	20.7347	170.6249	872.2001	2738.4178	0.9006	0.5467	0.2287

Table 2. Actual and estimated values of wiener system in discrete time

Types of values	Algorithms	b_0	b_1	b_2	b_3	a_1	a_2	a_3	a_4	c_1	c_2	c_3
Actual		-1.5537E-07	-4.1605E-07	4.7402E-07	1.4200E-07	-3.7777	5.3506	-3.3674	0.7945	1.0000	0.5000	0.2500
Estimated	FAGWO	**-1.5423E-07**	**-4.1278E-07**	**4.7391E-07**	**1.4114E-07**	**-3.7427**	**5.3019**	**-3.3579**	**0.7826**	**0.9865**	**0.4826**	**0.2418**
	FA [12]	-1.5047E-07	-4.5551E-07	4.4501E-07	1.3112E-07	-3.5003	4.9063	-3.2602	0.8658	1.0834	0.4620	0.2250
	BFA [15]	-1.4912E-07	-3.7492E-07	4.2868E-07	1.3014E-07	-3.5152	4.8642	-3.1124	0.7771	0.9020	0.4702	0.2254
	FPA [12]	-1.4680E-07	-3.7440E-07	4.2828E-07	1.2792E-07	-3.5197	4.8267	-3.0416	0.7406	0.9024	0.4616	0.2216
	GWO [13]	-1.5314E-07	-4.0623E-07	4.8399E-07	1.4740E-07	-3.6231	5.1298	-3.3731	0.8680	0.9222	0.4554	0.2480
	PSOGSA [16]	-1.4302E-07	-3.7663E-07	4.4130E-07	1.3053E-07	-3.6912	5.2341	-3.3824	0.8340	1.0483	0.5500	0.2382
	PSO	-1.4179E-07	-3.8254E-07	4.9885E-07	1.3149E-07	-3.5770	4.9643	-3.1541	0.7657	0.9000	0.4566	0.2251
	DE	-1.7065E-07	-4.5651E-07	5.2088E-07	1.4745E-07	-3.5860	4.9318	-3.0843	0.7412	1.0074	0.4511	0.2310
	ABC	-1.7062E-07	-4.3803E-07	4.6534E-07	1.3952E-07	-3.5728	4.9100	3.0711	0.7368	0.9383	0.4838	0.2292
	ALO [17]	-1.4559E-07	-4.5770E-07	5.2160E-07	1.4295E-07	-3.5007	4.9224	-3.2623	0.8740	0.9410	0.5500	0.2322
	DA [18]	-1.3980E-07	-4.5418E-07	4.5521E-07	1.3763E-07	-3.5816	5.0248	-3.3094	0.8671	1.0235	0.4925	0.2389
	MFO [19]	-1.7090E-07	-4.5770E-07	4.9987E-07	1.5620E-07	-3.5370	4.9075	-3.1868	0.8173	1.0291	0.5467	0.2250
	GOA [20]	-1.5767E-07	-3.7481E-07	5.2156E-07	1.3568E-07	-3.6838	5.2425	-3.4181	0.8605	0.9342	0.4662	0.2363
	SCA [21]	-1.6221E-07	-4.1481E-07	4.4600E-07	1.4877E-07	-3.5385	4.8155	-3.0307	0.7538	0.9646	0.4500	0.2750
	SSA [22]	-1.4971E-07	-4.4296E-07	4.7581E-07	1.4923E-07	-3.7207	5.2746	-3.3531	0.7968	1.0535	0.4581	0.2544
	WOA [23]	-1.6254E-07	-3.9890E-07	5.0421E-07	1.3137E-07	-3.5487	5.1182	-3.4519	0.8451	0.9617	0.5335	0.2745

Table 3. Actual and estimated values of wiener system in delta domain

Types of values	Algorithms	b_0	b_1	a_1	a_2	a_3	a_4	c_1	c_2	c_3
Actual		-0.8	4.5	22	180	760	2200	1	0.5	0.25
Estimated	FAGWO	**0.8007**	**4.9189**	**21.9993**	**179.8898**	**759.8564**	**2196.8982**	**1.0008**	**0.4993**	**0.2506**
	FA [12]	-0.7572	4.3832	21.7958	172.1841	763.0585	2258.0174	1.0442	0.4592	0.2620
	BFA [15]	-0.7489	4.4711	20.8981	185.6811	784.0277	1988.0011	0.9440	0.4778	0.2700
	FPA [12]	-0.8344	4.5947	22.0104	175.9721	766.0472	2370.6045	1.0576	0.5189	0.2565
	GWO [13]	-0.7610	4.4137	22.7008	175.3825	714.7724	2223.9991	0.9913	0.4693	0.2601

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 September 2018 doi:10.20944/preprints201809.0606.v1

PSOGSA [16]	-0.7988	4.9418	20.9575	188.6559	820.1872	2273.5409	1.0198	0.5167	0.2275
PSO	-0.8332	4.2612	21.7696	179.7970	796.8625	2332.6948	0.9634	0.5188	0.2359
DE	-0.8121	4.7793	20.2434	178.9678	774.0527	2257.0789	1.0289	0.5238	0.2357
ABC	-0.6239	3.7989	21.5476	120.8981	767.6003	3097.9705	1.0312	0.4816	0.2878
ALO [17]	-0.7228	4.0736	22.0121	167.1997	729.4998	2190.0464	1.0135	0.5210	0.2345
DA [18]	-0.8748	4.0742	23.5855	170.7276	730.4107	2019.8426	1.0022	0.5102	0.2300
MFO [19]	-0.7265	4.7379	23.1151	169.4181	734.5583	2251.8777	0.9507	0.4694	0.2538
GOA [20]	-0.7368	4.7694	23.1672	176.8427	760.3113	1997.8028	0.9101	0.5443	0.2314
SCA [21]	-0.7753	4.2673	21.7239	164.1472	741.4551	2312.0279	0.9608	0.4675	0.2636
SSA [22]	-0.8429	4.4526	22.2516	172.5593	699.8514	2309.6354	1.0326	0.5385	0.2485
WOA [23]	-0.8732	4.8243	22.6977	188.5943	802.2568	2378.2787	0.9588	0.5189	0.2326

Table 4. Actual and estimated values of hammerstein system in continuous time

Types of values	Algorithms	c_1	c_2	b_0	b_1	a_1	a_2	a_3	a_4
Actual		0.1	0.9	-1	5	23	185	800	2500
Estimated	FAGWO	**0.1005**	**0.9010**	**-1.0007**	**4.9919**	**22.8451**	**185.1138**	**798.6565**	**2489.7281**
	FA [12]	0.0975	0.9708	-1.0106	5.2112	25.0142	197.9079	730.8170	2543.4192
	BFA [15]	0.0927	0.9452	-1.0016	4.8382	23.3631	167.8127	821.0134	2652.2858
	FPA [12]	0.1025	0.8730	-0.9617	5.4506	21.7504	201.2880	790.5437	2644.1293
	GWO [13]	0.1017	0.8227	-0.9635	4.7647	23.5477	170.0642	775.2239	2611.2419
	PSOGSA[16]	0.0962	0.9492	-1.0791	5.1010	21.7703	184.0202	871.9438	2441.2051
	PSO	0.1036	0.8701	-0.9070	5.1071	23.3836	200.0911	866.1118	2420.9381
	DE	0.0972	0.9715	-0.9359	4.8392	25.2715	181.1624	849.5250	2378.9182
	ABC	0.1036	0.8589	-1.0112	5.2038	22.0595	189.9783	751.0919	2694.6895
	ALO [17]	0.1012	0.9356	-1.0748	4.5286	20.9742	191.6522	725.3687	2263.9133
	DA [18]	0.0924	0.8217	-1.0244	4.7467	25.1438	195.9269	804.6110	2517.1808
	MFO [19]	0.0904	0.9020	-1.0874	4.5681	23.3205	172.4943	763.8203	2612.8700
	GOA [20]	0.1021	0.8817	-1.0634	4.9887	23.7851	196.0922	739.6902	2656.9626
	SCA [21]	0.1045	0.9719	-0.9029	5.1086	21.0825	184.8791	809.2040	2689.4568
	SSA [22]	0.1041	0.9867	-0.9710	5.0217	24.7271	187.5363	794.4100	2730.7054
	WOA [23]	0.0933	0.8345	-0.9442	5.2642	23.6772	833.5913	833.5913	2254.1685

Table 5. Actual and estimated values of hammerstein system in discrete time

Types of values	Algorithms	c_1	c_2	b_0	b_1	b_2	b_3	a_1	a_2	a_3	a_4
Actual		0.1	0.9	-1.5537E-07	-4.1605E-07	4.7402E-07	1.4200E-07	-3.7777	5.3506	-3.3674	0.7945
Estimated	FAGWO	**0.1003**	**0.9008**	**-1.5541E-07**	**-4.1851E-07**	**4.7396E-07**	**1.4184E-07**	**-3.7691**	**5.3126**	**-3.3593**	**0.7927**
	FA [12]	0.10094	0.87585	-1.5084E-07	-4.4440E-07	4.9604E-07	1.4640E-07	-3.8517	4.9503	-3.0569	0.8142
	BFA [15]	0.10274	0.85244	-1.5650E-07	-4.5295E-07	4.7360E-07	1.4740E-07	-3.9742	4.9089	-3.1844	0.76297
	FPA [12]	0.10976	0.84761	-1.5341E-07	-3.7444E-07	4.3340E-07	1.3693E-07	-3.3999	4.9172	-3.1193	0.83146
	GWO [13]	0.10122	0.84377	-1.4053E-07	-4.3568E-07	4.8855E-07	1.4810E-07	-3.7526	4.8229	-3.2459	0.84807
	PSOGSA [16]	0.10794	0.98615	-1.6592E-07	-3.9185E-07	4.8987E-07	1.3833E-07	-3.6397	4.8718	-3.0485	0.74243
	PSO	0.096331	0.86196	-1.6882E-07	-4.2564E-07	4.4745E-07	1.5320E-07	-3.4873	4.9545	-3.6092	0.81398
	DE	0.10751	0.95917	-1.4477E-07	-3.8589E-07	5.0570E-07	1.3806E-07	-3.5264	4.8155	-3.1441	0.81657
	ABC	0.10055	0.97293	-1.6966E-07	-4.2249E-07	4.6219E-07	1.3488E-07	-4.1338	4.9899	-3.0795	0.77282
	ALO [17]	0.093972	0.97703	-1.5263E-07	-4.5385E-07	4.9892E-07	1.4639E-07	-3.6932	5.0305	-3.1525	0.78367
	DA [18]	0.10089	0.83585	-1.5085E-07	-4.4540E-07	4.7606E-07	1.4542E-07	-3.8611	4.9606	-3.0679	0.8478
	MFO [19]	0.10274	0.85244	-1.5650E-07	-4.5295E-07	4.7360E-07	1.4740E-07	-3.9742	4.9089	-3.1844	0.76297
	GOA [20]	0.10976	0.84761	-1.5341E-07	-3.7444E-07	4.3340E-07	1.3693E-07	-3.3999	4.9172	-3.1193	0.83146
	SCA [21]	0.10122	0.84377	-1.4053E-07	-4.3568E-07	4.8855E-07	1.4810E-07	-3.7526	4.8229	-3.2459	0.84807
	SSA [22]	0.10794	0.98615	-1.6592E-07	-3.9185E-07	4.8987E-07	1.3833E-07	-3.6397	4.8718	-3.0485	0.74243
	WOA [23]	0.096331	0.86196	-1.6882E-07	-4.2564E-07	4.4745E-07	1.5320E-07	-3.4873	4.9545	-3.6092	0.81398

Table 6. Actual and estimated values of hammerstein system in delta domain

Types of values	Algorithms	c_1	c_2	b_0	b_1	a_1	a_2	a_3	a_4
Actual		0.1	0.9	-0.8	4.5	22	180	760	2200
Estimated	FAGWO	**0.1002**	**0.9016**	**0.8022**	**4.5056**	**21.8819**	**179.8914**	**759.7545**	**2203.2891**
	FA [12]	0.1009	0.9099	-0.8623	4.7801	20.6720	188.4102	763.1495	2031.5233
	BFA [15]	0.0911	0.9512	-0.7238	4.8713	23.9882	196.6399	799.4870	2102.4735
	FPA [12]	0.0909	0.8992	-0.8045	4.4909	23.1408	164.1544	825.5786	2031.3154
	GWO [13]	0.1082	0.8646	-0.7699	4.6708	20.0113	194.1521	761.2539	2038.4687
	PSOGSA [16]	0.0921	0.8388	-0.7737	4.7146	21.9818	168.9356	802.2432	2308.1866
	PSO	0.0972	0.9804	-0.8666	4.5083	21.5148	164.97335	815.51509	2163.3531
	DE	0.0950	0.8618	-0.8449	4.3414	22.9583	169.9138	825.1344	2147.6116
	ABC	0.1033	0.8404	-0.7705	4.4753	22.5379	188.7374	730.1771	2035.7064
	ALO [17]	0.1068	0.8987	-0.8577	4.2764	20.0365	172.6891	777.9958	1989.3751
	DA [18]	0.0925	0.8897	-0.8767	4.6237	22.9182	179.7771	711.7970	2301.3454

MFO [19]	0.0992	0.8659	-0.7829	4.5010	21.4638	188.7328	804.1881	2237.43
GOA [20]	0.0998	0.9695	-0.8658	4.6001	21.7052	174.5740	833.4633	2075.5985
SCA [21]	0.0983	0.9848	-0.7929	4.6235	20.1478	165.2356	793.3553	2275.4090
SSA [22]	0.1063	0.920612	-0.8153	4.4029	20.0600	166.9278	808.3730	2076.1786
WOA [23]	0.1035	0.8573	-0.8390	4.8825	21.8649	181.1502	706.6982	2195.8506

The hybrid method employed outperforms the parent algorithms as well as some standard heuristics in all the three domains. In addition, the results are better than a popular hyrbrid algorithm PSOGSA. Further, the continuous-time and discrete-delta parameters show close resemblance. The statistical measures of the test system in respective domains are narrated in Table 7.

Table 7. Statistical measures of test systems

Test Systems	Algorithms	Best	Worst	Average	Std.
Wiener system in continuous-time	FAGWO	**0.0016**	**0.0017**	**0.0016**	**3.0779e-05**
	FA [12]	0.0018	0.0020	0.0019	5.8361e-05
	BFA [15]	0.0018	0.0021	0.0020	6.7289e-05
	FPA [12]	0.0019	0.0021	0.0020	7.4325e-05
	GWO [13]	0.0018	0.0019	0.0019	5.4884e-05
	PSOGSA [16]	0.0017	0.0020	0.0018	7.6356e-05
	PSO	0.0018	0.0019	0.0019	5.8215e-05
	DE	0.0018	0.0020	0.0019	5.6053e-05
	ABC	0.0019	0.0021	0.0020	5.8263e-05
	ALO [17]	0.0018	0.0021	0.0019	8.5345e-05
	DA [18]	0.0018	0.0020	0.0019	5.8430e-05
	MFO [19]	0.0018	0.0020	0.0019	5.8296e-05
	GOA [20]	0.0018	0.0019	0.0019	6.5716e-05
	SCA [21]	0.0018	0.0019	0.0019	6.7876e-05
	SSA [22]	0.0018	0.0020	0.0019	5.5030e-05
	WOA [23]	0.0018	0.0019	0.0018	5.8973e-05
Wiener system in	FAGWO	**0.0018**	**0.0020**	**0.0018**	**4.1039e-05**

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 30 September 2018 doi:10.20944/preprints201809.0606.v1

discrete-time					
	FA [12]	0.0020	0.0022	0.0021	5.9134e-05
	BFA [15]	0.0020	0.0023	0.0022	6.8297e-05
	FPA [12]	0.0021	0.0023	0.0022	7.6349e-05
	GWO [13]	0.0020	0.0025	0.0023	1.6068e-04
	PSOGSA [16]	0.0019	0.0022	0.0021	7.7783e-05
	PSO	0.0020	0.0024	0.0022	1.0138e-04
	DE	0.0020	0.0024	0.0022	1.3089e-04
	ABC	0.0021	0.0023	0.0022	6.1283e-05
	ALO [17]	0.0020	0.0023	0.0021	6.9473e-05
	DA [18]	0.0020	0.0022	0.0021	8.1356e-05
	MFO [19]	0.0020	0.0022	0.0021	5.9185e-05
	GOA [20]	0.0020	0.0021	0.0021	7.3561e-05
	SCA [21]	0.0020	0.0021	0.0021	7.8124e-05
	SSA [22]	0.0020	0.0022	0.0021	9.7183e-05
	WOA [23]	0.0020	0.0021	0.0021	6.7377e-05
Wiener system in delta domain	FAGWO	**0.0016**	**0.0017**	**0.0016**	**3.0779e-05**
	FA [12]	0.0018	0.0020	0.0019	5.8361e-05
	BFA [15]	0.0018	0.0021	0.0020	6.7289e-05
	FPA [12]	0.0019	0.0021	0.0020	7.4325e-05
	GWO [13]	0.0018	0.0019	0.0019	5.4884e-05
	PSOGSA [16]	0.0017	0.0020	0.0018	7.6356e-05
	PSO	0.0018	0.0019	0.0019	5.8215e-05
	DE	0.0018	0.0020	0.0019	5.6053e-05
	ABC	0.0019	0.0021	0.0020	5.8263e-05
	ALO [17]	0.0018	0.0021	0.0019	8.5345e-05
	DA [18]	0.0018	0.0020	0.0019	5.8430e-05
	MFO [19]	0.0018	0.0020	0.0019	5.8296e-05
	GOA [20]	0.0018	0.0019	0.0019	6.5716e-05

	SCA [21]	0.0018	0.0019	0.0019	6.7876e-05
	SSA [22]	0.0018	0.0020	0.0019	5.5030e-05
	WOA [23]	0.0018	0.0019	0.0018	5.8973e-05
Hammerstein system in continuous-time	FAGWO	**0.0183**	**0.0186**	**0.0184**	**4.7946e-05**
	FA [12]	0.0190	0.0196	0.0193	5.9423e-05
	BFA [15]	0.0191	0.0198	0.0195	6.8792e-05
	FPA [12]	0.0190	0.0196	0.0193	7.7482e-05
	GWO [13]	0.0189	0.0194	0.0192	5.4519e-05
	PSOGSA [16]	0.0188	0.0192	0.0189	5.8183e-05
	PSO	0.0190	0.0195	0.0193	5.0138e-05
	DE	0.0190	0.0196	0.0193	5.3128e-05
	ABC	0.0191	0.0196	0.0194	6.8249e-05
	ALO [17]	0.190	0.0197	0.0193	6.9985e-05
	DA [18]	0.0189	0.0194	0.0191	5.2386e-05
	MFO [19]	0.0191	0.0197	0.0193	5.9376e-05
	GOA [20]	0.0190	0.0196	0.0193	7.4566e-05
	SCA [21]	0.0191	0.0195	0.0193	7.8523e-05
	SSA [22]	0.0190	0.0195	0.0193	9.7813e-05
	WOA [23]	0.0188	0.0194	0.0192	6.7287e-05
Hammerstein system in discrete-time	FAGWO	**0.0187**	**0.0190**	**0.0188**	**4.9827e-05**
	FA [12]	0.0192	0.0198	0.0195	6.1683e-05
	BFA [15]	0.0193	0.0200	0.0197	6.7632e-05
	FPA [12]	0.0192	0.0198	0.0195	6.1986e-05
	GWO [13]	0.0191	0.0196	0.0194	6.1243e-05
	PSOGSA [16]	0.0190	0.0194	0.0191	5.5235e-05
	PSO	0.0192	0.0197	0.0195	5.9223e-05
	DE	0.0192	0.0198	0.0195	5.7135e-05
	ABC	0.0193	0.0198	0.0196	5.9394e-05
	ALO [17]	0.193	0.0200	0.0197	8.8759e-05

	DA [18]	0.0192	0.0201	0.0196	5.5637e-05
	MFO [19]	0.0194	0.0201	0.0197	5.8909e-05
	GOA [20]	0.0193	0.0199	0.0196	6.1181e-05
	SCA [21]	0.0193	0.0198	0.0195	5.3989e-05
	SSA [22]	0.0194	0.0199	0.0196	5.8979e-05
	WOA [23]	0.0193	0.0198	0.0196	5.6291e-05
Hammerstein system in delta domain	FAGWO	**0.0183**	**0.0186**	**0.0184**	**4.7946e-05**
	FA [12]	0.0190	0.0196	0.0193	5.9423e-05
	BFA [15]	0.0191	0.0198	0.0195	6.8792e-05
	FPA [12]	0.0190	0.0196	0.0193	7.7482e-05
	GWO [13]	0.0189	0.0194	0.0192	5.4519e-05
	PSOGSA [16]	0.0188	0.0192	0.0189	5.8183e-05
	PSO	0.0190	0.0195	0.0193	5.0138e-05
	DE	0.0190	0.0196	0.0193	5.3128e-05
	ABC	0.0191	0.0196	0.0194	6.8249e-05
	ALO [17]	0.190	0.0197	0.0193	6.9985e-05
	DA [18]	0.0189	0.0194	0.0191	5.2386e-05
	MFO [19]	0.0191	0.0197	0.0193	5.9376e-05
	GOA [20]	0.0190	0.0196	0.0193	7.4566e-05
	SCA [21]	0.0191	0.0195	0.0193	7.8523e-05
	SSA [22]	0.0190	0.0195	0.0193	9.7813e-05
	WOA [23]	0.0188	0.0194	0.0192	6.7287e-05

The statistical assessments viz. best, worst, average and standard deviation of the fitness function for each algorithm are provided in this table. In addition to this, the best results obtained with respect to the best and worst values, average, standard deviations are highlighted with bold letters. The hybrid method has the least value in all the statistical measures in all the domains. The continuous-time and discrete-delta results are in close match. Since standard deviation turns out to be least with the hybrid method, it can be concluded the algorithm is more stable than those considered for comparison.

 doi:10.20944/preprints201809.0606.v1

Additionally, some more statistical tests need to be performed to validate the significance of the results obtained justifying the very fact that the results did not come by chance. Hence, the non-parametric Wilcoxon rank sum test [14] is carried out to validate the significance of the results obtained and the calculated p-values are quoted in two parts in Table 8 and Table 9 respectively as metrics of significance. $p > 0.05$ in this test turns to be non-significant values.

Table 8. p-values for Wilcoxon rank sum test (part-1)

Test systems	Hybrid Algorithm	PSOGSA	FA	BFA	FPA	GWO	PSO	DE	ABC
Wiener system in continuous-time	FAGWO	3.6067E-06	1.9323E-05	3.6384E-05	7.0135E-06	3.6556E-06	1.9552E-09	1.8675E-09	1.1329E-07
Wiener system in discrete-time	FAGWO	1.0550E-07	9.3989E-06	2.2600E-05	9.8000E-03	2.6862E-08	1.0725E-08	9.3694E-11	1.6365E-08
Wiener system in delta domain	FAGWO	3.6067E-06	1.9323E-05	3.6384E-05	7.0135E-06	3.6556E-06	1.9552E-09	1.8675E-09	1.1329E-07
Hammerstein system in continuous-time	FAGWO	1.9514E-05	4.6708E-05	4.9824E-05	5.2319E-05	7.6790E-07	7.3220E-07	5.7997E-10	2.3675E-06
Hammerstein system in discrete-time	FAGWO	5.6379E-05	4.9103E-05	5.6522E-05	4.7214E-05	1.3350E-06	2.0024E-06	7.8642E-10	7.9023E-06
Hammerstein system in delta domain	FAGWO	1.9514E-05	4.6708E-05	4.9824E-05	5.2319E-05	7.6790E-07	7.3220E-07	5.7997E-10	2.3675E-06

Table 9. p-values for Wilcoxon rank sum test (part-2)

Test systems	Hybrid Algorithm	ALO	DA	MFO	GOA	SCA	SSA	WOA
Wiener system in continuous-time	FAGWO	1.2128E-06	1.0414E-05	3.6067E-06	1.2147E-06	1.0414E-05	1.2030E-06	1.0186E-05
Wiener system in discrete-time	FAGWO	2.1324E-08	1.9780E-04	1.3247E-05	1.5643E-04	8.3573E-05	4.7941E-08	1.0572E-07
Wiener system in delta domain	FAGWO	1.2128E-06	1.0414E-05	3.6067E-06	1.2147E-06	1.0414E-05	1.2030E-06	1.0186E-05
Hammerstein system in continuous-time	FAGWO	6.5000E-03	1.0700E-02	5.9000E-03	1.7000E-03	5.4000E-03	3.0230E-06	1.9536E-05
Hammerstein system in discrete-time	FAGWO	1.3100E-02	2.0300E-02	4.5000E-03	1.1300E-02	1.2000E-02	6.5556E-06	5.6434E-05
Hammerstein system in delta domain	FAGWO	6.5000E-03	1.0700E-02	5.9000E-03	1.7000E-03	5.4000E-03	3.0230E-06	1.9536E-05

Tables 8 and 9 clearly indicate that the results obtained by the hybrid technique are significant with respect to all other algorithms. The convergence characteristics showing the normalized value of the fitness function versus the number of iterations for the proposed method are plotted in Fig. 5(a)-(b) for the test system used for wiener and hammerstein model identification in the delta domain respectively.

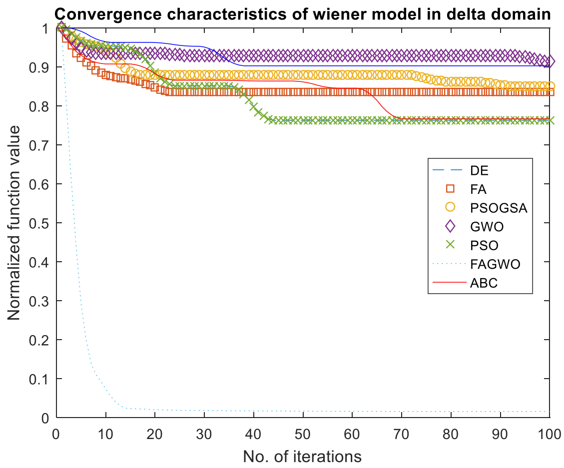

Fig. 5(a). Convergence characteristics of wiener model in delta domain

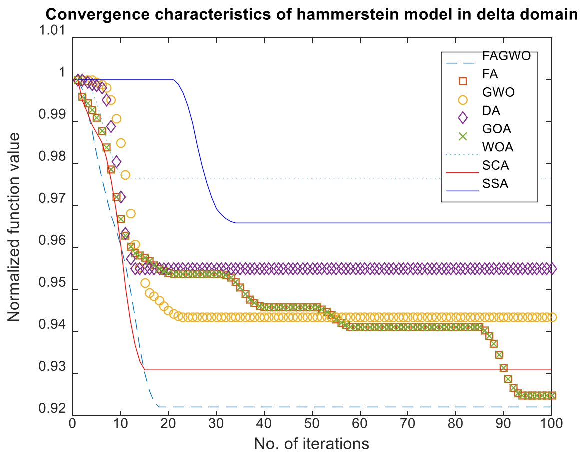

Fig. 5(b). Convergence characteristics of hammerstein model in delta domain

It is clearly evident from Fig. 5 that the proposed method converge faster as compared to the standard and parent algorithms considered.

6. Conclusions

A hybrid technique referred to as FAGWO is applied to identify hammerstein and wiener systems in the delta domain. Parameter estimation has been carried out in continuous, discrete and delta domain respectively. Delta operator modelling provides unification of continuous and discrete-delta results. The unknown model parameters are estimated through the minimization of mean square error (MSE). The fitness value of the hybrid technique not only surpasses that obtained by some popular metaheuristic algorithms but also the parent heuristics of which they are constituted for the test system under consideration. Wilcoxon test also validates the significance of the results obtained by the hybrid approach. The hybrid method also exhibits better convergence in the delta domain as compared to other algorithms.

 doi:10.20944/preprints201809.0606.v1

The algorithm can further be applied for identification of systems with time delay as well as multi-input multi-output system in the delta domain.

Conflict of Interest Statement: The authors declare that they have no conflict of interest.

References

[1] Keesman, K. J. (2011). *System identification: an introduction*. Springer Science & Business Media.

[2] Bhushan, B., & Singh, M. (2011). Adaptive control of DC motor using bacterial foraging algorithm. *Applied Soft Computing*, *11*(8), 4913-4920.

[3] Xu, X., Wang, F., & Qian, F. (2011, March). Study on method of nonlinear system identification. In *International Conference on Intelligent Computation Technology and Automation (ICICTA)*, (Vol. 1, pp. 944-947).

[4] Al-Duwaish, H. N. (2000). A genetic approach to the identification of linear dynamical systems with static nonlinearities. *International Journal of Systems Science*, *31*(3), 307-313.

[5] Nanda, S. J., Panda, G., & Majhi, B. (2010). Improved identification of Hammerstein plants using new CPSO and IPSO algorithms. *Expert systems with applications*, *37*(10), 6818-6831.

[6] Sun, J., & Liu, X. (2013). A novel APSO-aided maximum likelihood identification method for Hammerstein systems. *Nonlinear Dynamics*, *73*(1-2), 449-462.

[7] Raja, M. A. Z., Shah, A. A., Mehmood, A., Chaudhary, N. I., & Aslam, M. S. (2016). Bio-inspired computational heuristics for parameter estimation of nonlinear Hammerstein controlled autoregressive system. *Neural Computing and Applications*, 1-20, https://doi.org/10.1007/s00521-016-2677-x.

[8] Pal, P. S., Kar, R., Mandal, D., & Ghoshal, S. P. (2016). Identification of NARMAX Hammerstein models with performance assessment using brain storm optimization algorithm. *International Journal of Adaptive Control and Signal Processing*, *30*(7), 1043-1070.

[9] Sinha, N. K., & Rao, G. P. (Eds.). (2012). *Identification of continuous-time systems: Methodology and computer implementation* (Vol. 7). Springer Science & Business Media.

[10] Middleton, R. H., & Goodwin, G. C. (1990). *Digital Control and Estimation: A Unified Approach (Prentice Hall Information and System Sciences Series)*. Englewood Cliffs, NJ: Prentice Hall.

[11] Ganguli, S., Kaur, G., & Sarkar, P. (2018) A novel hybrid metaheuristic algorithm for model order reduction in the delta domain: a unified approach. *Neural Computing and Applications*, 1-15, https://doi.org/10.1007/s00521-018-3440-2.

[12] Yang, X. S. (2014). *Nature-inspired optimization algorithms*. Elsevier Inc.

[13] Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey wolf optimizer. *Advances in engineering software*, *69*, 46-61.

[14] Wilcoxon, F., Katti, S. K., & Wilcox, R. A. (1970). Critical values and probability levels for the Wilcoxon rank sum test and the Wilcoxon signed rank test. *Selected tables in mathematical statistics*, *1*, 171-259.

 doi:10.20944/preprints201809.0606.v1

[15] Passino, K. M. (2002). Biomimicry of bacterial foraging for distributed optimization and control. *IEEE control systems*, *22*(3), 52-67.

[16] Mirjalili, S., & Hashim, S. Z. M. (2010, December). A new hybrid PSOGSA algorithm for function optimization. In *International conference on Computer and information application (ICCIA)*, (pp. 374-377).

[17] Mirjalili, S. (2015). The ant lion optimizer. *Advances in Engineering Software*, *83*, 80-98.

[18] Mirjalili, S. (2016). Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems. *Neural Computing and Applications*, *27*(4), 1053-1073.

[19] Mirjalili, S. (2015). Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm. *Knowledge-Based Systems*, *89*, 228-249.

[20] Saremi, S., Mirjalili, S., & Lewis, A. (2017). Grasshopper optimisation algorithm: Theory and application. *Advances in Engineering Software*, *105*, 30-47.

[21] Mirjalili, S. (2016). SCA: a sine cosine algorithm for solving optimization problems. *Knowledge-Based Systems*, *96*, 120-133.

[22] Mirjalili, S., Gandomi, A. H., Mirjalili, S. Z., Saremi, S., Faris, H., & Mirjalili, S. M. (2017). Salp swarm algorithm: a bio-inspired optimizer for engineering design problems. *Advances in Engineering Software*, *114*, 163-191.

[23] Mirjalili, S., & Lewis, A. (2016). The whale optimization algorithm. *Advances in Engineering Software*, *95*, 51-67.