Research Article

A Robust Computational Technique for Model Order Reduction of Two-Time-Scale Discrete Systems via Genetic Algorithms

Othman M. K. Alsmadi[1] and Zaer S. Abo-Hammour[2]

[1]Department of Electrical Engineering, The University of Jordan, Amman 11942, Jordan
[2]Department of Mechatronics Engineering, The University of Jordan, Amman 11942, Jordan

Correspondence should be addressed to Othman M. K. Alsmadi; othman_mk@yahoo.com

A robust computational technique for model order reduction (MOR) of multi-time-scale discrete systems (single input single output (SISO) and multi-input multioutput (MIMO)) is presented in this paper. This work is motivated by the singular perturbation of multi-time-scale systems where some specific dynamics may not have significant influence on the overall system behavior. The new approach is proposed using genetic algorithms (GA) with the advantage of obtaining a reduced order model, maintaining the exact dominant dynamics in the reduced order, and minimizing the steady state error. The reduction process is performed by obtaining an upper triangular transformed matrix of the system state matrix defined in state space representation along with the elements of B, C, and D matrices. The GA computational procedure is based on maximizing the fitness function corresponding to the response deviation between the full and reduced order models. The proposed computational intelligence MOR method is compared to recently published work on MOR techniques where simulation results show the potential and advantages of the new approach.

1. Introduction

Model order reduction (MOR) of multi-time-scale systems has been an important subject area in control engineering for many years [1, 2]. In many industrial control systems, simple controllers are preferable. However, derivation of the mathematical model often leads to detailed description of a complex model in the form of high order differential equations [2]. Due to this point of view along with other different design objectives, model order reduction has been an active research area in the control society since the 1960s where a large number of model order reduction methods have been introduced in literature for single input single output (SISO) as well as MIMO type systems. The reduction operation is to search for the possibility of finding some lower order equations of the same type that may be considered to adequately reflect the dominant characteristics of the original system. The objective of simplification is to obtain a low order model of the existing high order model such that both are equivalent in terms of system response and being close to each other in some physical representation means. Model reduction problems have attracted much attention in recent years; for example, the model reduction problem has been investigated using artificial neural networks [3], genetic algorithms [4], and invasive weed optimization [5]. It was also used in nonlinear systems [6], gain scheduling [7], linear time-varying systems [8, 9], and linear parameter-varying systems [10].

To obtain a model of lower order, a significant number of methods have been proposed in recent and earlier years, some for continuous time systems [3–5] and some for discrete-time systems [1, 11–15]. Some methods, such as model order reduction by matching Markov parameters [16], were introduced to ensure stability of the reduced order model. A popular technique for obtaining reduced order models is the Krylov subspace [17]; however, stability of the reduced model is not guaranteed. Another important group of reduction algorithm is the eigenvalue preservation technique [3–5, 11] where important eigenvalues of the system are retained to find suitable lower order models.

A numerous number of MOR methods are available for continuous systems, but very few have been devoted to the discrete-time systems MOR. The discrete-time system MOR may be performed in two different ways. The first one is performed based on transforming a continuous time model into another form using different types of transformation as seen in [6, 18]. In this group of MOR, the reduction process is completely performed in the continuous time form. The discrete reduced order model is then obtained by the corresponding inverse transformation of the continuous time reduced model. The second method for obtaining a discrete reduced order model, which is known as a direct method [14], is deriving the discrete reduced order model directly without using any type of transformation. Some of these methods perform the MOR using canonical expansion of z-transfer function and stable optimal methods [13, 19], power decomposition and system identification [20], and multipoint step response matching [21]. New optimization techniques, particle swarm optimization [22], and artificial neural networks [11] have also been introduced for MOR of discrete-time systems.

GA-based MOR, on the other hand, has received some of the researchers' attention as well. Recently, Ponda et al. [23] employed a particular swarm optimization technique to obtain a reduced order model of SISO large scale linear systems. Their technique is based on the integral square error (ISE). Vishwakarma and Prasad [24] proposed a mixed method for reducing the order of large-scale linear systems. They have synthesized the denominator of the reduced order transfer function using modified pole clustering while the coefficients of the numerator elements are computed using GA. Parmar et al. [25] presented a technique for model order reduction using GA for SISO linear time systems. They have focused on obtaining a reduced order model that maintains stability and retains the steady state value. In spite of the methods available in literature, each method has advantages and disadvantages when tried on a particular system. In addition to that, no approach always gives the best results for all systems. It is important to mention that GAs have also been used for model system identification, where order and parameters are set to be determined, as we have investigated in [26, 27]. In this paper, however, and as motivated by the singular perturbation method which has the characterization of multi-time-scale systems, the GA procedure is performed with the advantages of retaining the exact dominant dynamics in the designed model, obtaining a new robust model with a lower order, and maintaining a minimum steady state response error.

The work in this paper is organized as follows: Section 2 presents problem formulation of the discrete full and reduced order models. In Section 3, the genetic algorithm approach for MOR of multi-time-scale discrete systems is presented. Illustrative examples utilizing the new approach along with simulation comparative results of different MOR techniques are presented in Section 4. Section 5 presents an overall conclusion of the proposed MOR method.

2. Problem Formulation

In this paper, MOR is investigated for discrete LTI systems of both SISO and MIMO type models. For SISO systems, a transfer function model is used, while the state space representation is used for MIMO systems.

For the SISO systems, consider the discrete-time system described by

$$
\begin{aligned}
& y(k) + a_1 y(k-1) + a_2 y(k-2) + \cdots + a_n y(k-n) \\
& \quad = b_0 u(k) + b_1 u(k-1) + \cdots + b_{\bar{n}} u(k - \bar{n}),
\end{aligned}
\tag{1}
$$

where $u(k)$ is the input and $y(k)$ is the output of the system at the kth sampling instant. Equation (1) can be written in the form of a pulse transfer function as

$$
G(z) = \frac{Y(z)}{U(z)} = \frac{b_0 z^{\bar{n}} + b_1 z^{\bar{n}-1} + \cdots + b_{\bar{n}}}{z^n + a_1 z^{n-1} + \cdots + a_n}
\tag{2}
$$

with $\bar{n} \leq n$. The characteristic polynomial contains the system dominant and nondominant poles (distinct, repeated, or complex) where their number, n, is referred to as the model order. The corresponding desired reduced rth order model is given by

$$
G_r(z) = \frac{Y(z)}{U(z)} = \frac{b_0 z^{\bar{r}} + b_1 z^{\bar{r}-1} + \cdots + b_{\bar{r}}}{z^r + a_1 z^{r-1} + \cdots + a_r},
\tag{3}
$$

where some of the coefficients a_i ($i = 1, 2, \ldots, r$) and b_i ($i = 0, 1, 2, \ldots, \bar{r}$) may be zeros as long as $\bar{r} \leq r$. For the MIMO systems, consider the following nth order discrete-time system:

$$
x(k+1) = Ax(k) + Bu(k),
\tag{4}
$$

$$
y(k) = Cx(k) + Du(k),
\tag{5}
$$

where k is the time index, $x \in \mathfrak{R}^n$ is the state vector, $u \in \mathfrak{R}^p$ and $y \in \mathfrak{R}^m$ are the input and output vectors, respectively, and $A \in \mathfrak{R}^{n \times n}$, $B \in \mathfrak{R}^{n \times p}$, $C \in \mathfrak{R}^{m \times n}$, $D \in \mathfrak{R}^{m \times p}$ are matrices of appropriate dimensions with n, p, and m being the system order, number of inputs, and number of outputs, respectively. The corresponding desired reduced rth order model is obtained as follows:

$$
x_r(k+1) = A_r x_r(k) + B_r u(k),
\tag{6}
$$

$$
y_r(k) = C_r x_r(k) + D_r u(k),
\tag{7}
$$

where $x_r(k)$ is r-state vector, $y_r(k)$ is the reduced order model output, and A_r, B_r, C_r, and D_r are matrices with appropriate dimensions.

3. Genetic Algorithms with MOR

GAs are based on principles inspired from the genetic and evolution mechanisms observed in natural systems. Their basic principle is the maintenance of a population of solutions to the problem that evolves in time. They are based on the triangle of genetic reproduction, evaluation, and selection [12].

Genetic reproduction is performed by means of two basic genetic operators: crossover and mutation. Evaluation is performed by means of the fitness function that depends on the specific problem. Selection is the mechanism that selects parent individuals with probability proportional to their relative fitness.

In this paper, using the computational intelligence of GA, we will obtain the reduced order model based on only the dominant dynamics of the system. For dynamic decoupling, the GA will set the reduced order model state matrix A_r in the modal form where all of the selected eigenvalues (system dynamics), real and/or complex, are placed on the diagonal. Thus, the reduced order model state matrix A_r, in (6), is designed to have the following decoupling format:

$$A_r = \begin{bmatrix} \lambda_1 & a_{12} & a_{13} & a_{14} & & \cdots & & a_{1r} \\ 0 & \lambda_2 & a_{23} & a_{24} & & \cdots & & \\ & \ddots & \ddots & \vdots & & & & \\ . & & 0 & \lambda_{\bar{b}} & & & & \vdots \\ . & & & 0 & \sigma_1 & \alpha_1 & & \\ . & & 0 & & -\alpha_1 & \sigma_1 & & \\ & 0 & & & 0 & & \ddots & a_{(r-2)r} \\ & & & & & 0 & \sigma_{\bar{p}} & \alpha_{\bar{p}} \\ 0 & . & . & . & & 0 & -\alpha_{\bar{p}} & \sigma_{\bar{p}} \end{bmatrix}, \quad (8)$$

where the original system dominant poles (real and/or complex) are preserved in the diagonal, seen as λ_i, $i = 1, 2, \ldots, \bar{b}$ (real) and $\sigma_i \pm \alpha_i$, $i = 1, 2, \ldots, \bar{p}$ (complex). Notice that, for this reduced order model, $r = (\bar{b} + 2\bar{p}) < n$. To insure that the dominant poles are preserved in the reduced order model and for further order reduction, the following condition is satisfied:

$$\lambda_{\text{dominant}} := |\,|\lambda_1| > |\lambda_2| > \cdots > |\lambda_{\bar{b}}|, \quad |\lambda| > |\sigma|,$$
$$|\sigma_1 \pm \alpha_1| > |\sigma_2 \pm \alpha_2| > \cdots > |\sigma_{\bar{p}} \pm \alpha_p|. \quad (9)$$

Take into account that if $|\lambda_i| < |\sigma_i \pm \alpha_i|$, then (9) is to be redefined accordingly if necessary. For simplicity, the modal form is chosen, which implies that all elements seen in (8) as a_{ij} (i and $j = 1, \ldots, r$) are set to zero.

The GA will determine the parameters of the B_r and C_r (and D_r if necessary) in (6) and (7). Hence, the total number of elements that the GA will need to find is given by

$$n_p = n_r \cdot m + p \cdot n_r. \quad (10)$$

It is to be noted that all of the parameters that the GA will have to find are restricted to be real values. Now, based on the number of unknown parameters (n_p), the GA creates a population of individuals, where each parameter is basically an individual in this population. The population consists of different "sets" of solutions. Each solution set is called a chromosome, which contains n_p individuals. Given a population size (n_{pop}), a matrix consisting of n_{pop} rows is
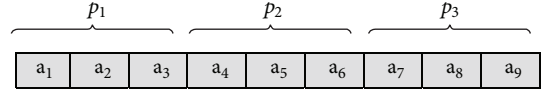

FIGURE 1: A set of individuals in a GA chromosome.

formed with each row containing one set of solutions for the unknown parameter values. This would result in a matrix containing $n_{\text{pop}} \times n_p$ elements. Each element in this matrix contains a value pertaining to one unknown parameter, where each row presents one set of solutions.

The genetic algorithm used in this work will operate as follows.

3.1. Initialization. An initial population comprising N_p individuals is randomly generated. The GA type used in this paper is a binary genetic algorithm, where each value in the solution set consists of a number of bits (genes). The number of bits used to encode each numerical value depends on three variables, lower parameter bound (p_l), upper parameter bound (p_u), and accuracy (ϕ). Hence, the number of bits used is defined as follows [30]:

$$n_{\text{bits}} = \frac{\log_{10}\left(\left(p_u - p_l\right)/\phi\right)}{\log_{10}(2)}. \quad (11)$$

Given that each parameter value will consist of n_{bits} number of bits, each solution set will consist of $n_{b,t} = n_p \cdot n_{\text{bits}}$ bits. This represents one row of the entire $n_{\text{pop}} \times n_{b,t}$ matrix.

The GA starts by randomly initializing a binary matrix with n_{pop} rows and $n_{b,t}$ columns. Each row (set of solutions) is made up of multiple values decoded into binary and placed next to each other as illustrated in Figure 1 for one chromosome. This chromosome consists of three parameters (individuals) with each individual being made of three genes. These genes (bits) can be later decoded back into decimal values, which in return provide the desired parameters' values.

The initial population matrix consists of random numbers within the lower and the upper bounds of the parameters. The population will be split into rows (chromosomes), each constituting one solution set. These solution sets are each taken to have their fitness evaluated, as seen next.

3.2. Evaluation. The fitness, a nonnegative measure of quality, is used as a measure to reflect the degree of goodness of the individual and is calculated for each individual in the population. This measure of quality is calculated based on minimizing the following related cost function:

$$e(k) = y(k) - y_r(k) \quad (12)$$

which is the deviation between the full and reduced order models' responses given in (5) and (7). The fitness for each chromosome (solution set) is then evaluated as follows [31]:

$$\text{Fitness} = \frac{1}{1 + \text{MSE}} \times 100\%, \quad (13)$$

where

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^{N} \left[y(i) - y_r(i)\right]^2, \quad (14)$$
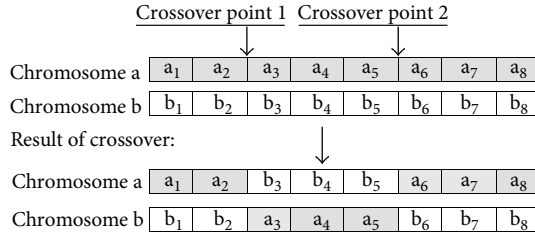
FIGURE 2: Chromosomes random crossover operation.

where MSE is the mean-square-error and N is the number of elements in the output vector(s). The higher this fitness is, the closer the reduced model is to mimic the original model.

3.3. Selection. In the selection process, chromosomes are chosen from the current population to enter a mating pool devoted to the creation of new children (offspring) for the next generation. The chance of a given individual to be selected to mate is proportional to its relative fitness That is, the larger the fitness value of a chromosome, the higher the probability of the chromosome to contribute one or more children in the next generation. First, the population fitness and associated chromosomes are ranked from highest fitness to lowest fitness. Then, only the best are selected to continue, while the rest are left. The selection rate (X_r) is a rank based ratio, which is the percentage of the best rank individual that should move to the mating pool where pairs of the mating pool are selected for crossover process. Deciding how many chromosomes to keep is somewhat arbitrary. Letting only a few chromosomes survive to the next generation limits the available genes in the offspring (children). Keeping too many chromosomes gives bad performers a chance to contribute their traits to the next generation. In the proposed algorithm, an initial selection rate $X_r = 0.60$ was used and the top 60% fitness chromosomes were placed in a mating pool.

There are several methods for choosing the chromosome pairs to be mated from the mating pool. In the proposed algorithm, random pairing was chosen, which uses a uniform random number generator to select chromosomes that enter the mating pool from kept chromosomes. The algorithm randomly chooses chromosomes from the mating pool for mating, while making sure that all pairs are unique. After the chromosome pairs are chosen, the next operation will be crossover.

3.4. Crossover. Crossover provides the means by which valuable information is shared among the population. In the crossover operation, a pair of chromosomes is mated to produce two offspring in the process that inherit genes from their parents. To perform crossover, the chromosomes need to be in their gene format (i.e., binary representation). The two parent chromosomes are crossed over at random points to segment each chromosome into a number of parts. These parts of each chromosome pair are swapped between each other to produce two new chromosomes, the offspring. This is shown in Figure 2.

Crossover is not applied to all pairs of chromosomes selected for mating. For every parent chromosome pair,

a crossover rate decides whether or not crossover occurs. If crossover is not applied, children are produced simply by duplicating the parents. This gives each chromosome a chance of passing on its genes without the disruption of crossover. In the proposed algorithm, the crossover rate is set to 0.80, meaning that 80% of parent pairs produce offspring. The number of crossover points for the parent chromosomes is chosen as a random integer for each generation. As a crossover result, a new children population will be performed with the same size as the parent population.

3.5. Mutation. Mutation is often introduced to guard against premature convergence. Generally, over a period of several generations, the gene pool tends to become more and more homogeneous. Therefore, further mutation is introduced to the offspring to guarantee that the offspring will have new qualities while retaining a similar overall structure. Mutation is applied to each child individually after crossover, where it randomly flips any bit (gene) with a small mutation probability (between 0.1 and 0.001). Mutation provides a small amount of random search to guard against premature convergence. In the proposed methodology, the mutation rate is set to 0.0125.

3.6. Replacement. Replacement operation takes place once the crossover of all parent chromosome pairs is performed. The parent population is totally or partially replaced by the children population depending on the replacement scheme used. This completes the life cycle of the population. At this stage, the population is ready to enter the next generation and undergo a new round genetic operation. The decision in which offspring replaces which population individual is made based on the fitness evaluation. Fitness is evaluated for all the resulting offspring as well as for all of the original population individuals and a replacement factor decides how many of the offspring with the highest fitness values are to replace the main population individuals. In the proposed approach, a replacement factor of 0.70 is used. This process continues until reaching the end of the population size or best desired fitness.

3.7. Termination. The GA is terminated when some convergence criterion is met. The termination condition could be considered as specified fitness value, reaching maximum number of generations, or a set progress limit. As one generation has gone by, depending on the genetic algorithm's termination condition, the algorithm could stop any time and identifies the chromosome with the highest fitness value as the optimal solution set. On the other hand, it may repeat the entire process from the selection procedure to continue another generation of the genetic algorithm. The termination condition for the proposed algorithm is reaching a fitness value of 99.999% and an average fitness value of at least 99.9% in the entire population set. When the algorithm terminates, the highest fitness chromosome is distributed across the B_r, C_r, and D_r matrices to produce the reduced order model.

4. Illustrative Examples

In this section, we consider a few discrete system examples, which have been investigated by different researchers in recent years. The examples are given as SISO and MIMO linear time invariant systems. Results of investigations are compared with the existing ones to show the potential of the new approach.

Example 1. As a first example, we consider the following 7th order SISO system investigated by Telescu et al. [32] given as

$$G(z) = \left(2.0434z^7 - 4.98255z^6 + 6.57z^5 - 5.8189z^4\right.$$

$$+ 3.636z^3 - 0.00088z^2 - 1.4105z + 0.2997\right)$$

$$\cdot \left(z^7 - 2.46^6 + 3.433z^5 - 3.333z^4\right.$$

$$\left. + 2.546z^3 - 1.584z^2 + 0.7478z - 0.252\right)^{-1} \tag{15}$$

with system dynamics (poles): [0.8913, 0.6843 ± 0.5820i, 0.2988 ± 0.7574i, −0.1987 ± 0.6993i]. Telescu et al. [32] used the Laguerre functions for their proposed method and obtained a 5th order reduced model:

$$G_r(z)$$

$$= \frac{2.043z^5 - 3.057z^4 + 2.195z^3 - 1.545z^2 + 0.8617z}{z^5 - 1.518z^4 + 1.270z^3 - 1.032z^2 + 0.7539z - 0.3156} \tag{16}$$

with poles given as [0.8320, −0.2318 ± 0.7612i, 0.5748 ± 0.5183i], which are unrelated to the original system dynamics. On the other hand, our first advantage of the proposed technique is that a lower dimension (3rd order) than the Telescu reduced model was obtained:

$$G_r(z) = \frac{1.51511z^3 - 3.12377z^2 + 2.40413z - 0.63117}{z^3 - 2.2598z^2 + 2.0266z - 0.71915} \tag{17}$$

with poles given as [0.8913, 0.6843 ± 0.5820i], which are exactly the dominant dynamics of the full order system as seen above, and this is our second advantage. The result was obtained for a population of 1000 and 300 generations with a fitness of 99.992. To investigate the system behavior, the full and reduced order models were excited by an impulse input (as performed by Telescu) with results of simulation shown in Figure 3, which shows our third advantage as the propose 3rd order reduced model response converges in about one second, which is much faster than the Telescu's response.

Our fourth advantage is clearly seen when simulating the full and reduced models to a step input. As seen in Figure 4, Telescu et al. [32] 5th order model has a huge steady state error while the proposed 3rd order model's error can barely be seen at the steady state which shows the robustness of the proposed method.
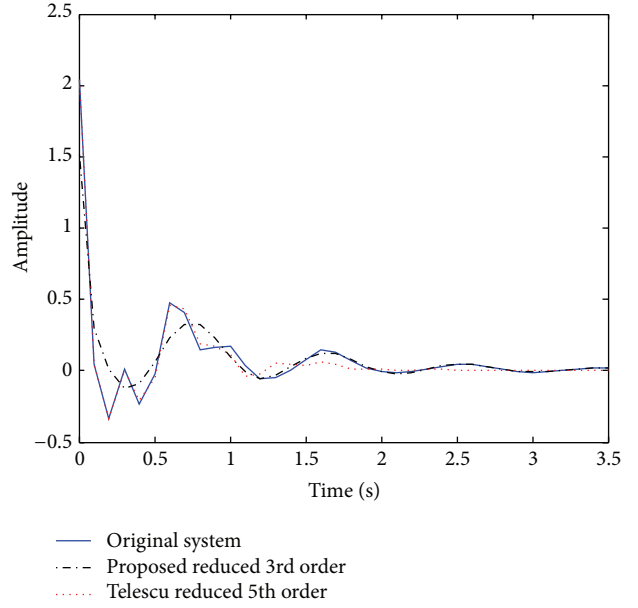

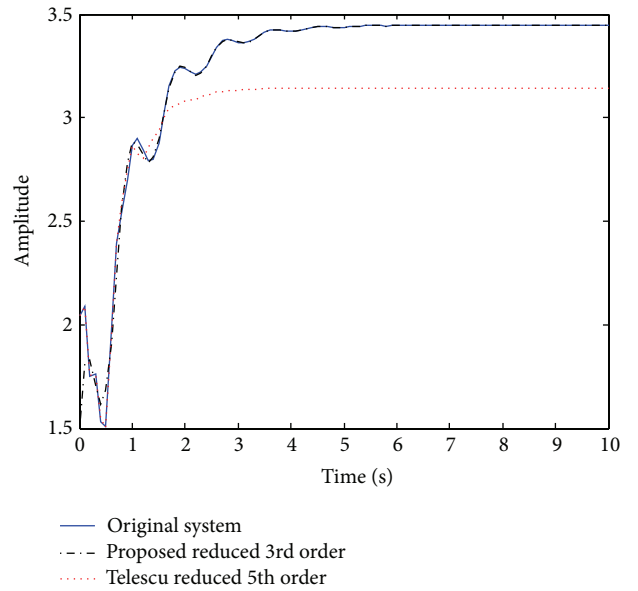FIGURE 3: Impulse responses for the full and reduced order models.

FIGURE 4: Step responses for the full and reduced order models.

Example 2. In this example, we consider the following 8th order SISO system investigated by Yadav et al. [29]:

$$G(z) = \left(0.1625z^7 + 0.125z^6 - 0.0025z^5 + 0.00525z^4\right.$$

$$- 0.02263z^3 - 0.00088z^2 + 0.003z - 0.000413\right)$$

$$\cdot \left(z^8 - 0.6307z^7 - 0.4185z^6 + 0.078z^5 - 0.057z^4\right.$$

$$\left. + 0.1935z^3 + 0.09825z^2 - 0.0165z + 0.00225\right)^{-1} \tag{18}$$

TABLE 1: Model order reduction method comparison.

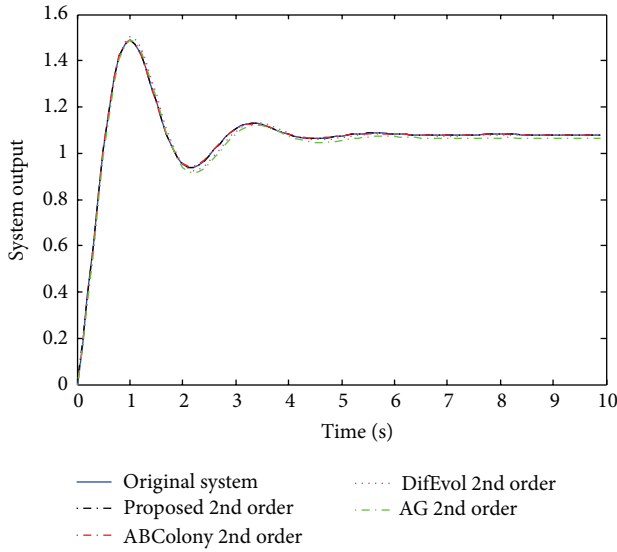| System	Dominant dynamics
Original 8th order	$0.8797 \pm 0.2442i, -0.5875 \pm 0.0959i$
	$0.0773 \pm 0.1078i, -0.0542 \pm 0.6558i$
Proposed 2nd order	$0.8797 \pm 0.2442i$
Artificial bee colony [24] 2nd order	$0.8778 \pm 0.2432i$
Differential evolution [28] 2nd order	$0.8768 \pm 0.2484i$
Genetic algorithm [29] 2nd order	$0.8854 \pm 0.2398i$

FIGURE 5: Step responses for the 8th and reduced 2nd order models.

FIGURE 6: Error comparison of the reduced order models with the full order.

with system dynamics: $[0.8797 \pm 0.2442i, -0.0542 \pm 0.6558i, -0.5875 \pm 0.0959i, 0.0773 \pm 0.1078i]$. Using the proposed method, the dominant dynamics $[0.8797 \pm 0.2442i]$ are preserved in the reduced order model which is given by

$$G_r(z) = \frac{0.021129z^2 + 0.12259z - 0.063666}{z^2 - 1.7594z + 0.83347}. \tag{19}$$

For result evaluation, the proposed reduced order model was compared with recent research for MOR, that is, artificial bee colony [28], differential evolution optimization algorithm, and real coded genetic algorithm [33]. Hence, we first compare for dominant dynamics of the reduced order models with results obtained as in Table 1. As can be seen, the proposed method provides the dominant dynamics of the full order model retained exactly in the reduced order, while in the other methods, the reduced order dynamics are close to the full order model, but not as close as our proposed method's results.

As a second comparison, the full and reduced order models were simulated for a step input with results as seen in Figure 5. Observing the resul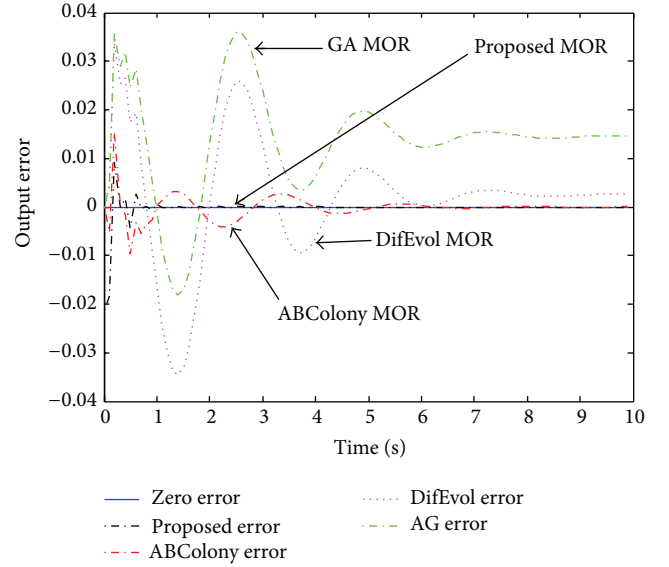ts in Figure 5, we can see that the 2nd order reduced model of all methods seems to be close to the original full order model. However, when we take a closer look, we see the differences very clearly as present in Figure 6. In this figure, it is seen that the proposed approach provides the least error among all. The error is seen a little at the beginning and then, at about one second, the error becomes very close to zero.

Example 3. In this example, we consider the 5th order MIMO discrete system investigated by Li [2] as given by the following state space model:

$$x(k+1)$$

$$= \begin{bmatrix} 0.5034 & 0.1768 & -0.2340 & -0.1406 & 0.5814 \\ 0.0096 & 0.5498 & -0.0362 & -0.6744 & 2.2496 \\ 0.0337 & 0.2546 & 0.0984 & -0.4051 & 1.3599 \\ -0.2709 & 0.1470 & 0.3249 & 0.0484 & 0.6356 \\ -0.0909 & 0.0491 & 0.1075 & -0.1019 & 0.5681 \end{bmatrix} x(k)$$

$$+ \begin{bmatrix} 0.3306 & 0.1700 \\ 0.8951 & 0.3442 \\ 0.5487 & 0.2143 \\ 0.8748 & 0.8821 \\ 0.5217 & 0.4479 \end{bmatrix} u(k),$$

$$y(k)$$

$$= \begin{bmatrix} 3.0622 & -0.9986 & -0.7126 & 6.4339 & -10.4291 \\ 3.0396 & -0.9913 & -0.7073 & 5.2369 & -8.4887 \end{bmatrix} x(k)$$

$$+ \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} u(k).$$

$$\tag{20}$$

This system is given with the dynamics [0.5614, 0.5008, 0.3389, 0.2295, 0.1375]. Li performed the Coprime Factor MOR and obtained the following 4th order reduced model:

$$x_r(k+1) = \begin{bmatrix} 0.3830 & 0.0865 & -0.3771 & 0.7548 \\ -0.1633 & 0.6255 & -0.0157 & 0.1556 \\ -0.6731 & 0.5150 & 0.3777 & -0.2194 \\ -0.3692 & 0.2832 & 0.0094 & 0.2388 \end{bmatrix} x_r(k)$$

$$+ \begin{bmatrix} 1.0572 & 0.0349 \\ -0.2642 & -0.5105 \\ -0.6643 & -1.5790 \\ 0.9864 & 0.0412 \end{bmatrix} u(k),$$

$$y(k) = \begin{bmatrix} -0.2745 & -0.0669 & -0.6423 & -0.2425 \\ -0.2725 & -0.0664 & -0.5228 & -0.1973 \end{bmatrix} x_r(k)$$

$$+ \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} u(k) \tag{21}$$

with poles [0.5567, 0.5004, 0.3385, 0.2294], which are close to the full order model, but not the exact values. Using the proposed approach, the following 4th order reduced model was obtained:

$$x_r(k+1) = \begin{bmatrix} 0.5614 & 0 & 0 & 0 \\ 0 & 0.5008 & 0 & 0 \\ 0 & 0 & 0.3389 & 0 \\ 0 & 0 & 0 & 0.2295 \end{bmatrix} x_r(k)$$

$$+ \begin{bmatrix} 0.4285 & 0.2628 \\ 0.4157 & 0.2551 \\ 0.3405 & 0.2348 \\ 0.3111 & 0.2671 \end{bmatrix} u(k),$$

$$y(k) = \begin{bmatrix} 29.7314 & -20.0311 & -37.9382 & 27.0723 \\ 26.5158 & -20.4912 & -28.5003 & 21.6793 \end{bmatrix} x_r(k)$$

$$+ 10^{-3} \begin{bmatrix} 0.0061 & -0.1580 \\ 0.0046 & -0.1191 \end{bmatrix} u(k) \tag{22}$$

with poles [0.5614, 0.5008, 0.3389, 0.2295], which are the exact dominant dynamics of the original model. Hence, the superiority of the proposed method is clearly seen. Next, we investigate for quality performance of the new method compared with the Li approached reduced order model. Simulating the full and reduced order models for a mixed type signal, the two system output responses are presented in Figure 7. As seen in this figure, the two reduced order models seem to perform with same quality. Therefore, in order to observe the differences between the proposed and existed methods, we present the response accuracy error, as seen in Figures 8 and 9 for the two output responses.
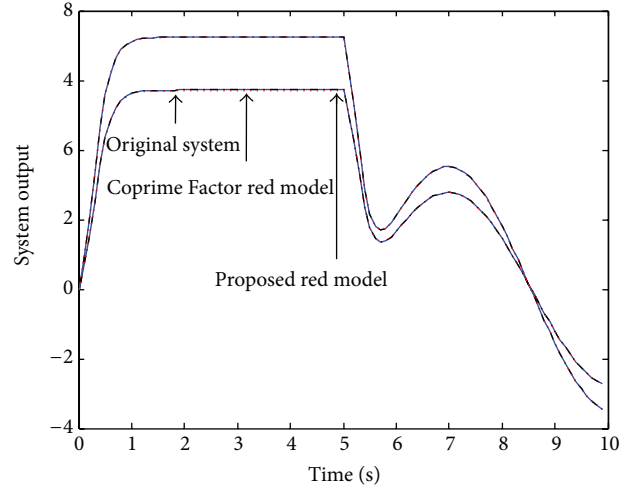

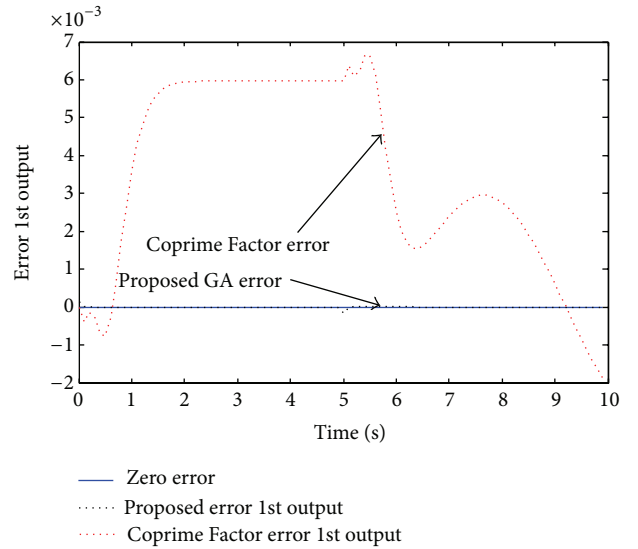FIGURE 7: Full and reduced order model's responses for mixed input signals.

FIGURE 8: Error of the reduced order models for the system first output response.

In Figures 8 and 9, notice that the Coprime Factor method response errors are much far from zero when compared with the proposed method response errors. The response errors of the proposed GA approach can barely be seen as they are almost zeros, which clearly shows the superiority of the proposed approach.

As seen in the three previous examples, the advantages of the new approach are present in all three of them. That is, the reduction process provides new reduced order models, maintains the exact dominant dynamics of the original model in the reduced order model, and provides reduced order models with the least response errors as compared with other methods. In addition to that, an advantage that the new approach is applicable to SISO and MIMO discrete type systems.
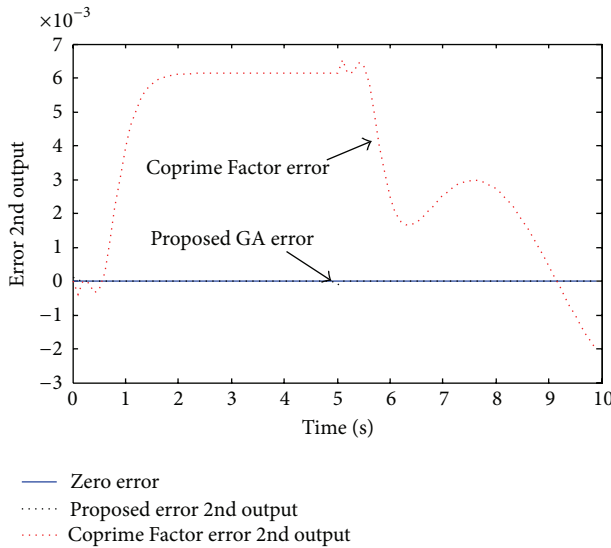
Figure 9: Error of the reduced order models for the system second output response.

5. Conclusion

In this paper, a robust computational intelligence approach is proposed using GA for MOR of discrete MIMO type systems with the advantage of dominant dynamic preservation. The reduction process is performed based on transforming the system state matrix (in a state space model) while decoupling the multi-time-scale dynamics. The dominant dynamic preservation is performed by retaining the dominant poles of the original system as a subset in the reduced order model utilizing the transformed system state matrix. Once the reduced order model state matrix is designed, the GA MOR new technique will search for the B, C, and D matrices of the reduced order model. Different examples are presented with a comparison of recently published work, such as differential evolution, artificial bee colony, and the Coprime Factor MOR methods. Comparison results show the robustness of the proposed method, as it outperforms such techniques, which points out the potential of the new approach.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] P. Karimaghaee and N. Noroozi, "Frequency weighted discrete-time controller order reduction using bilinear transformation," *Journal of Electrical Engineering*, vol. 62, no. 1, pp. 44–48, 2011.

[2] L. Li, "Coprime factor model reduction for discrete-time uncertain systems," in *Proceedings of the 49th IEEE Conference on Decision and Control (CDC '10)*, pp. 6213–6218, Atlanta, Ga, USA, December 2010.

[3] O. M. K. Alsmadi, Z. S. Abo-Hammour, A. M. Al-Smadi, and D. I. Abu-Al-Nadi, "Genetic algorithm approach with frequency selectivity for model order reduction of MIMO systems," *Mathematical and Computer Modelling of Dynamical Systems*, vol. 17, no. 2, pp. 163–181, 2011.

[4] D. I. Abu-Al-Nadi, O. M. Alsmadi, Z. S. Abo-Hammour, M. F. Hawa, and J. S. Rahhal, "Invasive weed optimization for model order reduction of linear MIMO systems," *Applied Mathematical Modelling*, vol. 37, no. 6, pp. 4570–4577, 2013.

[5] O. M. K. Alsmadi, Z. S. Abo-Hammour, and A. M. Al-Smadi, "Robust model order reduction technique for MIMO systems via ANN-LMI-based state residualization," *International Journal of Circuit Theory and Applications*, vol. 40, no. 4, pp. 341–354, 2012.

[6] Y.-C. Chu and K. Glover, "Bounds of the induced norm and model reduction errors for systems with repeated scalar nonlinearities," *IEEE Transactions on Automatic Control*, vol. 44, no. 3, pp. 471–483, 1999.

[7] G. Scorletti and L. E. Ghaoui, "Improved LMI conditions for gain scheduling and related control problems," *International Journal of Robust and Nonlinear Control*, vol. 8, no. 10, pp. 845–877, 1998.

[8] S. Lall and C. Beck, "Error-bounds for balanced model-reduction of linear time-varying systems," *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 946–956, 2003.

[9] H. Sandberg and A. Rantzer, "Balanced truncation of linear time-varying systems," *IEEE Transactions on Automatic Control*, vol. 49, no. 2, pp. 217–229, 2004.

[10] W. M. Haddad and V. Kapila, "Robust, reduced-order modeling for state-space systems via parameter-dependent bounding functions," *IEEE Transactions on Automatic Control*, vol. 42, no. 2, pp. 248–253, 1997.

[11] O. M. K. Alsmadi, Z. S. Abo-Hammour, and A. M. Al-Smadi, "Artificial neural network for discrete model order reduction with substructure preservation," *Applied Mathematical Modelling*, vol. 35, no. 9, pp. 4620–4629, 2011.

[12] S. C. Chuang, "Homographic transformation for the simplification of discrete-time transfer functions by Padé approximation," *International Journal of Control*, vol. 22, no. 5, pp. 721–728, 1975.

[13] C. Hwang and C.-S. Hsieh, "A new canonical expansion of z-transfer function for reduced-order modeling of discrete-time systems," *IEEE Transactions on Circuits and Systems*, vol. 36, no. 12, pp. 1497–1509, 1989.

[14] R. Y. Hwang and Y. P. Shih, "Combined methods of model reduction via discrete Laguerre polynomials," *International Journal of Control*, vol. 37, no. 3, pp. 615–622, 1983.

[15] V. P. Singh and D. Chandra, "Reduction of discrete interval systems based on pole clustering and improved Padé approximation: a computer-aided approach," *Advanced Modeling and Optimization*, vol. 14, no. 1, pp. 45–56, 2012.

[16] W. Krajewski, A. Lepschy, and U. Viaro, "Model reduction by matching Markov parameters, time moments, and impulse-response energies," *IEEE Transactions on Automatic Control*, vol. 40, no. 5, pp. 949–953, 1995.

[17] B. Salimbahrami, B. Lohmann, T. Bechtold, and J. Korvink, "A two-sided Arnoldi-algorithm with stopping criterion and an application in order reduction of MEMS," *Mathematical and Computer Modelling of Dynamical Systems*, vol. 11, no. 1, pp. 79–93, 2005.

[18] Y. P. Shih and W. T. Wu, "Simplification of z-transfer functions by continued fraction," *International Journal of Control*, vol. 17, no. 5, pp. 1089–1094, 1973.

[19] N. N. Puri and M. T. Lim, "Stable optimal model reduction of linear discrete-time systems," *Journal of Dynamic Systems, Measurement and Control*, vol. 119, no. 2, pp. 300–304, 1997.

[20] C. M. Liaw and M. Ouyang, "Model reduction of discrete system using the power decomposition method and the system identification method," *IEE Proceedings D: Control Theory and Applications*, vol. 133, no. 1, pp. 30–34, 1986.

[21] S. Mukherjee, Satakshi, and R. C. Mittal, "Discrete system order reduction using multipoint step response matching," *Journal of Computational and Applied Mathematics*, vol. 170, no. 2, pp. 461–466, 2004.

[22] S. N. Deepa and G. Sugumaran, "MPSO based Model order formulation scheme for discrete time linear system in state space form," *European Journal of Scientific Research*, vol. 58, no. 4, pp. 444–454, 2011.

[23] S. Ponda, J. S. Yadav, N. P. Padidar, and C. Ardil, "Evolutionary techniques for model order reduction of large scale linear systems," *International Journal of Applied Science, Engineering and Technology*, vol. 5, no. 1, pp. 22–28, 2009.

[24] C. B. Vishwakarma and R. Prasad, "MIMO system reduction using modified pole clustering and genetic algorithm," *Modelling and Simulation in Engineering*, vol. 2009, Article ID 540895, 5 pages, 2009.

[25] G. Parmar, M. K. Pandey, and V. Kumar, "System order reduction using ga for unit impulse input and a comparative study using ISE & IRE," in *Proceedings of the International Conference on Advances in Computing, Communication and Control*, pp. 23–24, Mumbai, India, 2009.

[26] Z. Abo-Hammour, O. Alsmadi, S. Momani, and O. Abu Arqub, "A genetic algorithm approach for prediction of linear dynamical systems," *Mathematical Problems in Engineering*, vol. 2013, Article ID 831657, 12 pages, 2013.

[27] Z. S. Abo-Hammour, O. M. Alsmadi, A. M. Al-Smadi, M. I. Zaqout, and M. S. Saraireh, "ARMA model order and parameter estimation using genetic algorithms," *Mathematical and Computer Modelling of Dynamical Systems: Methods, Tools and Applications in Engineering and Related Sciences*, vol. 18, no. 2, pp. 201–221, 2012.

[28] G. Vasu and G. Sandeep, "Design of PID controller for higher order discrete systems based on order reduction employing ABC algorithm," *Control Theory and Informatics*, vol. 2, no. 4, pp. 4–15, 2012.

[29] J. S. Yadav, N. P. Patidar, and J. Singhai, "Controller design of discrete systems by order reduction technique employing differential evolution optimization algorithm," *International Journal of Computational Intelligence*, vol. 6, no. 1, pp. 43–49, 2010.

[30] Z. Michalewicz, *Genetic Algorithms + Data Structure = Evolution Programs*, Springer, New York, NY, USA, 1996.

[31] W.-D. Chang, "Coefficient estimation of IIR filter by a multiple crossover genetic algorithm," *Computers & Mathematics with Applications*, vol. 51, no. 9-10, pp. 1437–1444, 2006.

[32] M. Telescu, N. Iassamen, P. Cloastre, and N. Tanguy, "A simple algorithm for stable order reduction of z-domain Laguerre models," *Signal Processing*, vol. 93, no. 1, pp. 332–337, 2013.

[33] J. S. Yadav, N. P. Patidar, and J. Singhai, "Model order reduction and controller design of discrete systems technique employing real coded genetic algorithm," *International Journal of Advanced Engineering Technology*, vol. 1, no. 3, pp. 134–144, 2010.