Linear Dynamic System Simplification Using Genetic Algorithm

[1] Namrta Sharma, [2]Dr. C.B. Vishwakarma

[1] Research Scholar Mewar University, Rajasthan, India
[2]Assistant Professor, Department of Electrical Engineering, School of Engineering, Gautam Buddha University, Greater Noida, India
Email - [1] namratashmca@gmail.com, [2] cvishwakarma@gbu.ac.in

Abstract: *The authors suggested two approaches for getting optimum simplified models of any linear single-input and single-output (SISO) large-scale dynamic system. In the first approach, all coefficients of numerator and denominator of the simplified models are computed by using GA optimization technique. In the second approach, the denominator polynomial of the simplified model is obtained by using any classical method while the numerator coefficients are computed by applying Genetic Algorithm (GA) optimization tool of MATLAB software. The algorithm of the proposed method is illustrated by solving few examples and finally it is concluded that the algorithm is capable to yields better simplified dynamic models and to retain stability of the large-scale original system.*

Key Words: *Linear System, Dynamic System, Stability, GA, Simplification of Systems.*

1. INTRODUCTION:

Many classical methods [1]-[5] have been used to synthesize the simplified model having reduced denominator polynomial for ease of analysis and design. But almost all the conventional methods are little bit suffering from one or more limitations or drawbacks and therefore, simplified models obtained from an original high order system may not exactly match the time or frequency response of the original high order system. But, there is a scope to get more accurate models from the high order system and this can be done by using optimization techniques.

The various optimization techniques are available in the literatures. The optimization techniques can be classified as

1- Classical Optimization Techniques

2- Evolutionary Optimization Techniques

Nowadays, Evolutionary Optimization techniques are more popular because of fast and global search computing. Most of the real world optimization problems involve complexities like discrete, continuous or mixed variables, multiple conflicting objectives, non-linearity, discontinuity etc. The search space may be so large that the global optimum cannot be found in reasonable time. The classical methods may not be efficient to solve such problems. Various stochastic methods like simulated annealing or evolutionary optimization algorithms can be used in such situations. Few advance and evolutionary optimization techniques are listed here.

- Simulated Annealing (SA)
- Genetic Algorithm (GA)
- Ant Colony Optimization (ACO)
- Particle Swarm Optimization (PSO)
- Bacteria Forging Optimization (BFO)
- Memetic Algorithm (MA)
- Cuckoo Search (CS)
- Bat Algorithm (BA)
- Cultural Algorithm (CA)
- Differential Evolution (DE)

1.1 GENETIC ALGORITHM (GA):

Genetic algorithms are search algorithms based on the mechanics of natural selection and natural genetics. They combine survival of the fittest among string structures with a structured yet randomized information exchange to form a search algorithm with some of the innovative flair of human good measure. While randomized, genetic algorithms are no simple random walk. The GAs has been developed by John Holland, his colleagues, and his students at the University of Michigan. Genetic Algorithms are different from more normal optimization and search procedures in four ways [6]:

- GAs work with a coding of the parameter set, not the parameters themselves.
- GAs search from a population of points, not a single point.
- GAs use payoff (objective function) information, not derivatives or other auxiliary knowledge.
- GAs use probabilistic transition rules, not deterministic rules.

The algorithm of GAs can be understood with the help of the following flow chart.

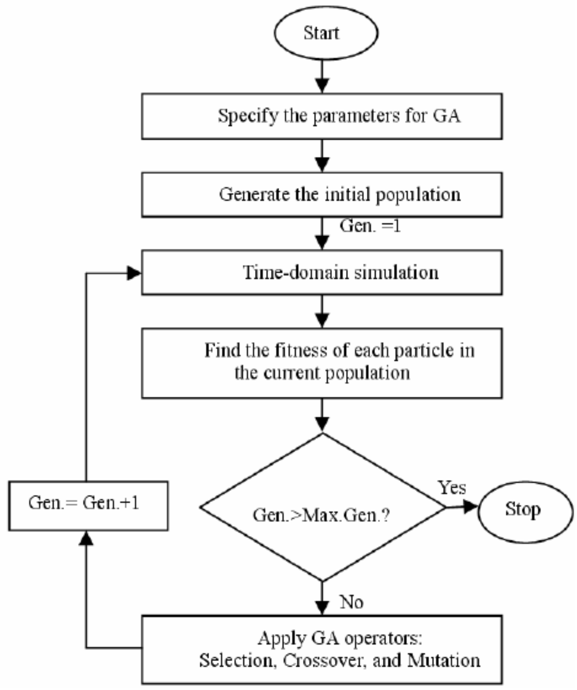

Fig.1. Flow Chart of Genetic Algorithm

2. PROBLEM STATEMENT:

Consider an original high-order linear dynamic system of the order $'n'$ given as

$$G(s)=\frac{N(s)}{D(s)}=\frac{a_0+a_1s+a_2s^2+...+a_{n-1}s^{n-1}}{b_0+b_1s+b_2s^2+...+b_ns^n} \quad (1)$$

Where the coefficients $'a_i'$ $(0\le i\le n-1)$ and $'b_i'$ $(0\le i\le n)$ are known and real.

$$R_k(s)=\frac{N_k(s)}{D_k(s)}=\frac{c_0+c_1s+c_2s^2+...+c_{k-1}s^{k-1}}{d_0+d_1s+d_2s^2+...+d_ks^k} \quad (2)$$

Where the coefficients $'c_i'$ $(0\le i\le k-1)$ and $'d_i'$ $(0\le i\le k)$ are unknown.

For finding the optimum values of unknown coefficients, the following fitness function has to be minimized using optimization technique.

Minimize $$e(t)=\sum_{k=0}^{m}|g(k)-r(k)| \quad (3)$$

Where $'m'$ is the number of samples and $g(k)$ and $r(k)$ are the discrete values of the unit step responses at the k^{th} sample.

The problem is to find the simplified model $R_k(s)$ from the original system $G(s)$ such that time and frequency response of the both must match as close as possible.

Here, two approaches have been suggested to realize optimum simplified models are as follows:

First Approach: The optimum numerator $N_k(s)$ and optimum denominator $D_k(s)$ of the simplified model $R_k(s)$ are computed by using GA.

Second Approach: The denominator $D_k(s)$ of the simplified model is computed by using any classical method and optimum numerator coefficients are computed using GA.

3. METHOD OF COMPARISON:

The simplified models obtained by these methods are compared with the original dynamic system. Two types of comparisons are shown here as graphical and performance index comparison. Two performance indices which are known as Integral Square Error (ISE) and Relative Integral Square Error (RISE) are defined as

$$\text{ISE}=\int_0^\infty [y(t)-y_k(t)]^2 dt$$

$$\text{RISE}=\int_0^\infty [y(t)-y_k(t)]^2 dt \Big/ \int_0^\infty [y(t)-y(\infty)]^2 dt$$

Where $y(t)$ and $y_k(t)$ are step responses of the original and k^{th} -order simplified model respectively. $y(\infty)$ is the steady-state value of the original system.

4. VALIDATION OF METHOD AND RESULTS:

Two dynamic systems are taken from literature and simplified into low order models by the proposed methods for the validation purpose. The results obtained here are compared with the original system.

Example-1: The 4^{th} –order transfer function [7] of a dynamic system is given as

$$G(s)=\frac{24+24s+7s^2+s^3}{24+50s+35s^2+10s^3+s^4}$$

Let the 2^{nd} –order simplified model is required in the form as

$$R_2(s)=\frac{c_0+c_1 s}{d_0+d_1 s+d_2 s^2}$$ here, $c_0=d_0$ for matching steady-state values and applying 1^{st} approach.

Taking Fitness function $'J'$ as

Minimize $J=\sum_{k=0}^{1000}\left|g(k)-r(k)\right|$

Using the GA optimization technique tool of MATLAB with the following parameters:

Population Type	Double Vector
Population Size	20
Cross over probability	0.8
Mutation Rate	0.05
Generation	200

The simplified 2nd order system is obtained as

$$R_2(s) = \frac{1.15 + 0491s}{1.15 + 1.688s + 0.61s^2}$$

The step response of the 2nd order system with the fourth order system is plotted and shown here

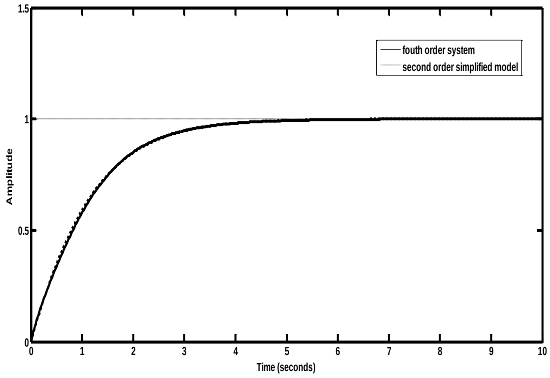

Fig 2: Step Response Comparison

Using 2nd approach of the method, the denominator is obtained by using simple clustering technique

$$D_2(s) = s^2 + 2.5s + 1.3125$$

The 2nd -order reduced model is written as

$$R_2(s) = \frac{c_1 s + 1.3125}{s^2 + 2.5s + 1.3125}$$

Using GA tool of MATAB, the unknown numerator coefficient are obtained and hence final simplified model is written as

$$R_2(s) = \frac{1.231s + 1.3125}{s^2 + 2.5s + 1.3125}$$

The step response of simplified model is compared with the original system and shown in fig.3

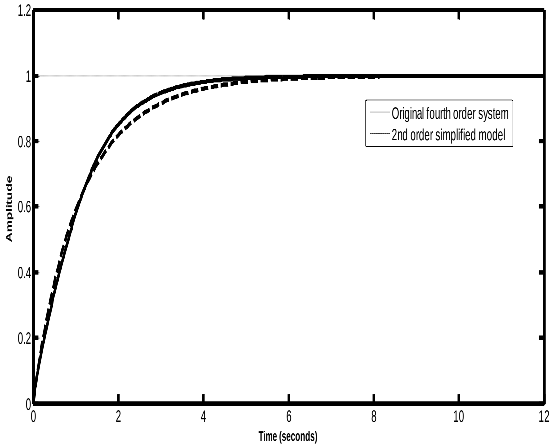

Fig.3: Step Response Comparison

The RISE is calculated by MATLAB simulink model and shown in the Table-1

TABLE I. COMPARISON WITH OTHER METHODS

Simplification Method	**Simplified Model**	**RISE**
Proposed Method (first Approach)	$\frac{1.15 + 0491s}{1.15 + 1.688s + 0.61s^2}$	0.00018

Proposed Method (second Approach)	$\frac{1.3125+1.231s}{1.3125+2.5s+s^2}$	0.02017
Prasad and Pal [7]	$\frac{34.2465+s}{34.2465+239.8082s+s^2}$	1.2050
Davison [8]	$\frac{2-s^2}{2+3s+s^2}$	0.1403

Example-2: Consider an eight-order system from the literature [9]

$$G(s)=\frac{N(s)}{D(s)}$$

$$N(s)=18s^7+514s^6+5982s^5+36380s^4+122664s^3$$
$$+222088s^2+185760s+40320$$
$$D(s)=s^8+36s^7+546s^6+4536s^5+22449s^4$$
$$+67284s^3+118124s^2+109584s+40320$$

Let 2nd –order simplified model is required in the form as

$$R_2(s)=\frac{c_0+c_1 s}{d_0+d_1 s+d_2 s^2}$$ here, $c_0=d_0$ for matching steady-state values.

The simplified 2nd -order system is obtained using 1st approach as

$$R_2(s)=\frac{0.312+0.966s}{0.312+0.397s+0.073s^2}$$

The step response of the 2nd order system with the fourth order system is plotted and shown here

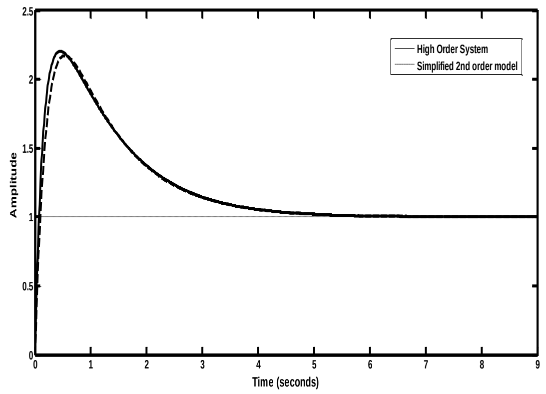

Fig.4: Step response Comparison for example 2

Using 2nd –approach of the proposed method, where the simplified denominator is obtained by simple pole clustering technique

$$D_2(s)=s^2+4.5s+4.0625$$

Using the Genetic Algorithm (GA) the numerator coefficients are obtained as

$c_0=4.0625 \quad c_1=11.827$

GA parameters are taken while running GA tool of MATLAB.
Population Type: Double Vector
Population Size: 20
Selection Function : Stochastic uniform
Mutation Function: Adaptive feasible
Crossover function: Scattered
Now the simplified modes can be compared with the original system

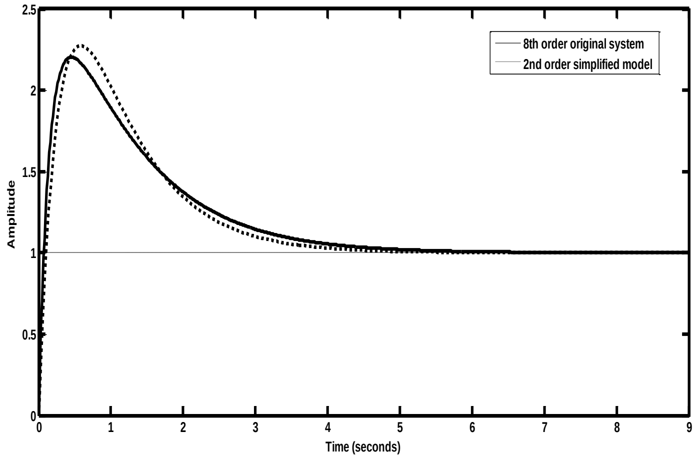

Fig.5: Step response Comparison for example 2

TABLE II. COMPARISON WITH OTHER METHODS

Simplification Methods	Reduced Models	ISE
Proposed Method (1st Approach)	$\frac{0.312+0.966s}{0.312+0.397s+0.073s^2}$	0.0142
Proposed method (2nd Approach)	$\frac{11.827s+4.0625}{s^2+4.5s+4.0625}$	0.03204

From the step responses comparison from fig. 4 and 5 and from table 1 and 2, it may be concluded that 1st approach of the proposed method gives more better response than 2nd approach because in 1st approach GA has more flexibility as all unknown coefficients are to be computed without any constraint while in 2nd approach only numerator coefficients are to be found by GA algorithm. It is also seen from the table-I and II that GA optimization has capability to give better simplified model than any classical method available in the literature.

5. CONCLUSION AND FUTURE SCOPE:

The authors proposed an optimum method for simplification of linear dynamic system using Genetic Algorithm optimization technique. In this paper, two approaches have been suggested based on optimization of numerator and denominator coefficients. The 1st approach deals the optimization of all coefficients of the numerator and denominator of the simplified model of 2nd order while 2nd approach is a mixture of the classical and optimization based computation.

Two systems are taken from literature and proposed methods have been applied to get 2nd order simplified model. The results are compared graphically and qualitatively. From the figures and tables, finally it is concluded that the 1st approach is better than 2nd approach of the proposed method. The GA optimization technique is most popular because of its global optimum point search capability and that is why, authors have chosen GA technique to synthesize the simplified models. The proposed method is also capable to retain stability of the original system and able to capture the transient and steady –state behavior of the original system in the simplified models.

The proposed method may be extended to linear and nonlinear multivariable systems as well. Some other evolutionary optimization techniques may be applied to synthesize the simplified models.

REFERENCES:

1. Sharad Kumar Tiwari, Gagandeep Kaur, (2017): Model reduction by new clustering method and frequency response matching, *J Control Autom. Electr. Syst.* Vol. 28, pp. 78-85.
2. G. Parmar *et. al, (*2007) : A mixed method for large-scale systems modelling using eigen spectrum analysis and cauer second form, *IETE Journal of Research*, Vol. 53, No. 2, pp. 93-102.
3. Jay Singh, Kalyan Chatterjee, C.B. Vishwakarma, (2015): Reduced order modelling of linear dynamic systems, *ASME-Journals-2015-series: Advances C* Vol.70, pp. 71-85.
4. C.B. Vishwakarma,, R. Prasad, (2014): Time domain model order reduction using Hankel matrix approach, *Journal of Franklin Institute*, Vol. 351, pp. 3445-3456.
5. V. Singh, D. Chandra and H. Kar, (2004): Improved Routh-Pade Approximants: A Computer Aided Approach, *IEEE Trans. Automat Control*, Vol 49, No.2, pp. 292-296.
6. David E. Goldberg, (2006): *Genetic Algorithms: in serach, optimization & machine learning* Pearson Education.
7. Prasad R and Pal J, (1991) : Stable reduction of linear systems by continued fractions, *Journal of Institution of Engineers, India*, Vol. 72, (pp. 113-116).
8. Davison E.J., (1991): A method of simplification of linear dynamic systems. *IEEE Trans Autom Control*, Vol. 11, pp. 93-101.
9. Shamash Y., (1975): Linear System reduction uisng Pade approximation to allow retention of dominants poles, *International Journal of Control*, Vol. 21, No. 2, pp. 257-272.