Generating Lower Order System using Modified Truncation and PSO

S. R. Desai[#], Rajendra Prasad[*]
[#] Electrical Engineering Department, Indian Institute of Technology Roorkee, Roorkee, India
[*] Electrical Engineering Department, Indian Institute of Technology Roorkee, Roorkee, India

ABSTRACT

A novel method of generating lower order system is being introduced which grasps the advantages of Particle Swarm Optimization (PSO) technique and Modified Truncation (MT) Method. The proposed combination of PSO and MT is applied to original fourth and sixth order Linear Time Invariant (LTI) systems. The denominator polynomial is reduced using MT and numerator term using PSO. The step responses of the resultant lower order system and the original system are compared and their performances are justified.

Keywords— Lower order system, Particle Swarm Optimization, Modified Truncation, Linear Time Invariant System.

1. INTRODUCTION

The rapidly increasing system complexity along with miniaturization in sizes, has resulted in great demand for faster simulation process during the design validation stage [1]. Inspite of having high speed processors, lowering the system order is one way, which is generally practiced in systems and control engineering field and is under active research. This further ends up as a necessary procedure for simulating large complex systems. Currently a variety of order reduction algorithms are being used [2-9] but none can be judged as the universally best as it depends upon how well it satisfies the application specifications. Moreover, the best reduction method should also preserve the vital dynamic characteristics of the system under consideration; simplify the best available model to suit the purpose with less error as far as possible. Here, nature inspired approach which has proved to be fruitful is roped in to meet the requirements. Particle Swarm Optimization (PSO) in combination with Modified Truncation (MT) being a conventional method is proposed for lowering the system order effectively [10].

In the recent past, evolutionary techniques have been used in almost all fields successfully and have become popular. These techniques have proved to be effective in developing lower order approximations for systems having large dimension and controller design of the same [11-14]. The advantage of these optimization methods is that they help in the elimination /optimizing some of the state variables from the original or a transformed system representation, a task which cannot be accomplished easily. Finally, this results in reduction of storage and computation time without affecting the vital properties of the original system. This ensures that the resultant lower order system is viable for use. In spite of the current popular optimization methods, there is a great vehemence for the advancement of the so called global optimization methods [15]. Active research is going on to develop a universal optimization method that can be applied to all multifaceted problems with equal efficiency.

PSO, a subset of evolutionary computation technique is being used widely for quite some time. This technique has upper hand over GA in terms of fast convergence; simplicity, requires no rigid first guess algorithm, ease of implementation and exploration of majority of problem space are some of the additional features [13]. Also, it is uncomplicated to code and understand its most basic form. Hence, it is found to be useful in solving mixed integer optimization problems that are of typical complex engineering system [13].In this paper, the benefits of PSO and Modified Truncation method is used to generate the reduced order system [16]. The purpose of using PSO is to support in searching the best values among the available ones to suit the requirements. The proposed method turned out to be comparable with other conventional techniques.

2. PARTICLE SWARM OPTIMIZATION (PSO)

PSO, a subset of evolutionary computation has been popular in academia and industry, mainly because of its intuitiveness, handles both discrete and continuous variables. PSO works well with any dimension problem and finding the optimum for single objective and multi-objective functions (nonlinear and linear) even though the problem of being stuck in local minima exists. PSO is similar to Genetic Algorithms due to the stochastic population based nature, but is easier to implement with the same. Further, this stochastic population based method comes with a simple memory component. In conclusion, PSO has similar or better results than GA [17-18].

In 1995, Kennedy and Eberhart introduced PSO algorithm to the world in terms of social and cognitive behavior [19]. Till now, many researchers are benefited by utilizing the same to solve various problems belonging to varied disciplines; fairly simple computations and sharing of information within the algorithm as it derives its internal communications from the social behavior of individuals are some of the major attractions. These individuals referred to particles are flown through the multi-dimensional search space with each particle representing a possible solution to the multi-dimensional optimization problem [20]. Each solution's fitness is based on a performance function related to the optimization problem being solved.

The PSO process is kicked off by randomly initializing the particle's position and velocity within the entire search range. Each candidate solution (particle's position) is expressed as a position within the dimensional space of the problem. The particles will move into the solution space under the influence of the information obtained from iteration-to-iteration as well as particle-to-particle. Between iteration-to-iteration, the best solution visited so far by a particle, is stored in its memory as *pbest* [21]. Likewise, the particle-to-particle information ensures that, the best solution visited by any particle is stored in its memory and experiences an attraction towards this solution, called *gbest*. The *pbest* and *gbest* are updated for each particle, after every iterations, till a better or more dominating solution (in terms of fitness) is found. These information's are called as social and cognitive components and this whole process continues till the desired result is found within the computational limit.

The heart of the PSO algorithm is the velocity equation and expresses individual particle's velocity as a balance between attraction to its own personal best position and the current global best position among all particles. The velocity of each particle is updated using the velocity update equation given by

$$v_{id} = v_{id} + c_1 r_1 (p_{id} - x_{id}) + c_2 r_2 (p_{gd} - x_{id})$$
(1)

and the position is updated using

$$x_{id} = x_{id} + v_{id} \quad (2)$$

Where, $i = 1, 2, \ldots, S$ represents the particle index, S is the size of the swarm, $V_i = (v_{i1}, v_{i2}, \ldots, v_{in})^T$ and $P_i = (p_{i1}, p_{i2}, \ldots, p_{in})^T$ is the associated velocity and previously best visited position of i^{th} the particle, 'g' is the index of the best particle in the swarm, c_1 and c_2 are constants, called cognitive and social scaling parameters respectively (usually, $c_1 = c_2$; r_1, r_2 are random numbers drawn from a uniform distribution).

Equations (1) and (2) define the classical version of PSO algorithm. Later, the concept of an inertia weight was developed to enhance control exploration and exploitation; introduced in the literature (1998) and the resulting velocity update equation [22,23] is given by

$$v_{id} = w * v_{id} + c_1 r_1 (p_{id} - x_{id}) + c_2 r_2 (p_{gd} - x_{id})$$
(3)

The initial value of w is set to 0.9 and reduced linearly to 0.4, allowing initial exploration followed by acceleration toward an improved global optimum. This is the difference between local and global searching and is one of the reasons, the algorithm is so resistant to getting stuck in local minima [23].The flowchart showing the process of PSO is as shown in Fig. 1.

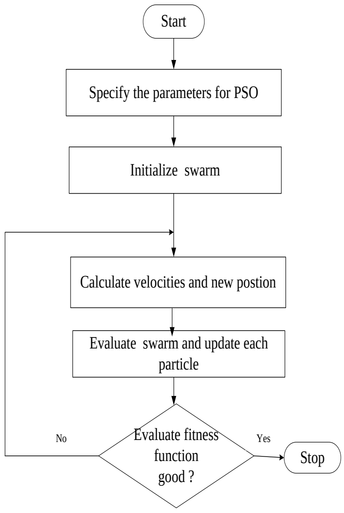

Fig 1. Optimization process

3. STATEMENT OF PROBLEM

Consider a nth order linear time invariant single input single output (LTI-SISO) system described by the transfer function

$$G(s) = \frac{\sum_{j=1}^{n} a_j s^{j-1}}{\sum_{j=1}^{(n+1)} b_j s^{j-1}} \quad (4)$$

Where a_j's and b_j's are scalar constants. The objective is to find the k^{th} (k<n) order reduced model R(s), comprising of scalar constants c_i's and d_i's represented in the form of

$$R(s) = \frac{\sum_{i=1}^{k} c_i s^{i-1}}{\sum_{i=1}^{(k+1)} d_i s^{i-1}} \; ; d_{i+1} = 1$$
(5)

4. GENERATION OF REDUCED ORDER MODELS

The following section describes the procedure to find out the denominator and numerator polynomials of the reduced order model.

A. Denominator Polynomials

The denominator polynomial D(s) of the original system G(s) can be rewritten as

$$D(s) = b_1 + b_2 s + b_3 s^2 + b_4 s^3 + \ldots + b_{n+1} s^n$$
(6)

The direct truncation of D(s), (n-k) times gives the r^{th} order model that tends to approximate the poles and zeros with a small modulus rather than those with large modulus. This results in good approximation for systems dominated by the poles lying near the imaginary axis. However the same method may perform badly for the dominant poles with large magnitudes. To overcome this drawback we use reciprocal transformation [16,24]. This technique helps in reversing the order of the denominator coefficients and thereby the small magnitude poles of D(s) of will become large magnitude poles of $\tilde{D}(s)$ and vice versa.

$$\tilde{D}(s) = s^n D\left(\frac{1}{s}\right)$$
$$= b_1 s^n + b_2 s^{n-1} + b_3 s^{n-2} + \ldots + b_{n+1}$$
(7)

This transformation enables that some dominant poles having small magnitude and some with large magnitude roots may be retained in the reduced denominator. This ensures good time response matching in both transient and steady state regions. The proposed modification consists of truncation of D(s). Truncate k_1 times to obtain $D_{k1}(s)$. Similarly truncate $\tilde{D}(s)$, k_2 times to obtain $\widetilde{D_{k2}}(s)$. The reduced denominator $D_k(s)$ is then obtained as

$$D_k(s) = D_{k1}(s).D_{k2}(s)$$
$$= d_1^* + d_2^* s + d_3^* s^2 + d_4^* s^3 + \ldots + d_{k+1}^* s^k$$
(8)

which is normalized to give

$$D_k(s) = d_1 + d_2 s + d_3 s^2 + d_4 s^3 + ... + d_{k+1} s^k$$
(9)

Where, d_{k+1} =1, $D_{k2}(s)$ is reciprocal of $\widetilde{D_{k2}}(s)$ and $k = k_1 + k_2$.

To illustrate the proposed method second order models are obtained for some problems taken from literature. The general form of second order model is taken as

$$R(s) = \frac{c_2 s + c_1}{s^2 + d_2 s + d_1} \qquad (10)$$

Where d_2,d_1 are obtained from (8).

B. Numerator Polynomial

Once the denominator polynomial is found, then the numerator coefficients c_1, c_2 are found by using the PSO algorithm by minimizing the fitness function f_k given by

$$f_k = \sum_{i=0}^{M-1} \left[y(i\Delta t) - y_r(i\Delta t) \right]^2 \qquad (11)$$

$y(i\Delta t)$ and $y_r(i\Delta t)$ are the unit step responses of the higher order and the reduced order models at time t=Δt. Usually time T is taken as 10 sec and Δt =0.1 sec. The parameter settings used are the initial population size (100 feasible solutions), the number of iterations is 50. Further the steady state error between the response of the original and the reduced system can be nullified as in any general case.

5. NUMERICAL EXAMPLES

Ex 1: Consider a 4th order system contrived by Shamash [25]

$$G(s) = \frac{81.691s^3 + 506.649s^2 + 99.843s + 5}{s^4 + 105.2s^3 + 521.01s^2 + 101.05s + 5}$$

Consider the denominator term

$$D(s) = s^4 + 105.2s^3 + 521.01s^2 + 101.05s + 5$$

The reciprocal of D(s) is

$$\tilde{D}(s) = 5s^4 + 101.05s^3 + 521.01s^2 + 105.2s + 1$$

By modified truncation the following reduced denominators are found for various values of k_1 and k_2

$$521.01s^2 + 101.05s + 5 \; ; \; k_1 = 2, \; k_2 = 0$$
$$s^2 + 105.2s + 521.01 \; ; \; k_1 = 0, \; k_2 = 2$$
$$s^2 + 105.25s + 5.205 \; ; \; k_1 = 1, \; k_2 = 1$$

Considering the denominator obtained for k1=0 and k2=2, and using BBBC algorithm, the numerator polynomial of the reduced system will be

$$79.75s + 521.01$$

The reduced second order system is then given by

$$R(s) = \frac{79.75s + 521.01}{s^2 + 105.2s + 521.01}$$

The reduced second order system obtained by [18]

$$R_{CAUER}(s) = \frac{25.241561s + 5.2034}{s^2 + 26.4979s + 5.2034}$$

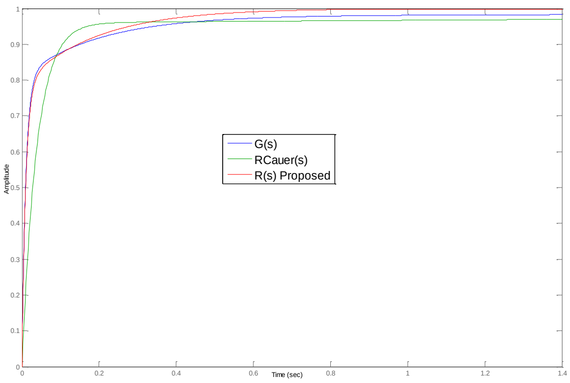

Fig 2 Comparison of step responses

The step responses of the original system G(s), proposed system R(s) and $R_{cauer}(s)$ are shown in figure 2 and are comparable. It is observed that the proposed method performs much better than R(s).

Ex 2: Consider another 6th order system having transfer function [26]

$$G(s) = \frac{s^5 + 1014s^4 + 14069s^3 + 69140s^2 + 140100s + 1000000}{s^6 + 222s^5 + 14541s^4 + 248420s^3 + 1454100s^2 + 2220000s + 1000000}$$

Consider the denominator term

$$D(s) = s^6 + 222s^5 + 14541s^4 + 248420s^3 + 1454100s^2 + 2220000s + 1000000$$

The reciprocal of D(s) is

$$\tilde{D}(s) = 1000000s^6 + 2220000s^5 + 1454100s^4 + 248420s^3 + 14541s^2 + 222s + 1$$

By modified truncation the following reduced denominators are found for various values of k_1 and k_2

$$s^3 + 5.853s^2 + 8.9365s + 4.0254 \; ; k_1 = 3, \; k_2 = 0$$
$$s^3 + 222s^2 + 14541s + 248420 \; ; k_1 = 0, \; k_2 = 3$$
$$s^3 + 222.45s^2 + 14641s + 6550 \; ; k_1 = 1, \; k_2 = 2$$
$$s^3 + 223.53s^2 + 339.62s + 152.67 \; ; k_1 = 2, \; k_2 = 1$$

Considering the denominator obtained for k_1=2, k_2=1 and using the BBBC optimization algorithm, the numerator polynomial of the reduced system will be

$$9.518s^2 + 23.08s + 152.7$$

Therefore the third order reduced model obtained will be

$$R(s) = \frac{9.61s^2 + 23.1s + 152.7}{s^3 + 223.53s^2 + 339.62s + 152.7}$$

According to the method [10], the reduced third order model is

$$R_{DP}(s) = \frac{4.3638s^2 + 31.0845s + 490.077}{s^3 + 56.554s^2 + 736.8139s + 490.07747}$$

Figure 3 shows the step responses of the original system G(s), proposed reduced system R(s) and the reduced system using dominant pole[10] $R_{DP}(s)$. It is seen that the responses of the G(s) and R(s) are matching both in steady and transient states, whereas the response of $R_{DP}(S)$ performs weakly.

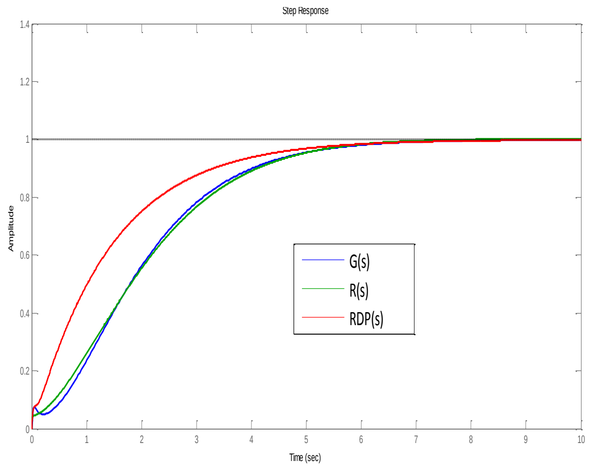

Fig 3 Comparison of step responses

Ex 1: Consider a fourth-order system [10] described by the transfer function as

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

Consider the denominator term

$$D(s)=s^4+10s^3+35s^2+50s+24$$

The reciprocal of D(s) is

$$\tilde{D}(s)=24s^4+50s^3+35s^2+10s+1$$

By modified truncation the following reduced denominators are found for various values of k_1 and k_2

$$s^2+1.42857s+0.6857\ ;\ k_1=2,\ k_2=0$$
$$s^2+10s+35\ ;\ k_1=0,\ k_2=2$$
$$s^2+0.48s+14.8\ ;\ k_1=1,\ k_2=1$$

Considering the denominator obtained for k_1=2 and k_2=0, and using BBBC algorithm, the numerator polynomial of the reduced system will be

$$0.7241s + 0.6857$$

The reduced second order system is then given by

$$R(s)=\frac{0.7241s + 0.6857}{s^2+1.42857s + 0.6857}$$

The reduced second order system obtained by [10]

$$R_{DP}(s)=\frac{0.9315s + 1.6092}{s^2+2.75612s + 1.6092}$$

Figure 4 shows the step responses of the original system G(s), the proposed reduced system R(s) and the reduced system using dominant pole[10] $R_{DP}(s)$. It is seen that the responses are matching both in steady and transient states.

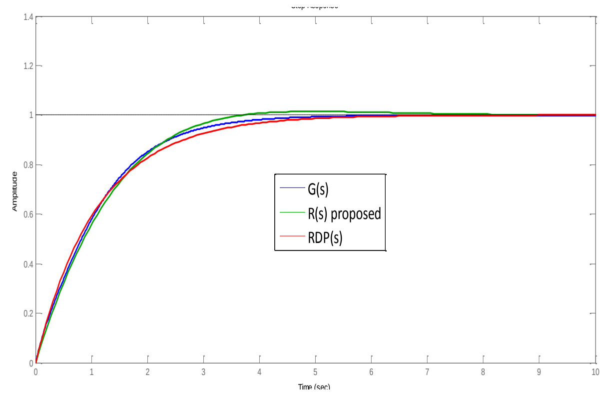

Fig 4 Comparison of step responses

6. CONCLUSIONS

A new method of reducing the order of the original system is discussed. The task is accomplished using the mixed method (PSO and modified truncation method). The denominator of the reduced method is obtained by the modified truncation method and the numerator of the reduced system is generated using PSO. The application of the proposed method is justified by comparing the step responses in the above examples. It is observed that, the step response of the original and the proposed reduced system are closely matching and the results are better/comparable to that of the other methods. Further, the proposed method holds good for multiple input and multiple output and discrete system. Various combinations of conventional technique and PSO can also be tried for lowering the order of the system.

7. REFERENCES

[1] Dia Abu-Al-Nadi, Othman MK Alsmadi, Zaer S Abo-Hammour," Reduced order modeling of linear mimo systems using particle swarm optimization", *The Seventh International Conference on Autonomic and Autonomous Systems ICAS 2011*,. ISBN: 978-1-61208-134-2, pp. 62-66, IARIA, 2011.

[2] Obinata, G., and Anderson, B.D.O., *Model Reduction for Control System Design*, London, Springer-Verlag, 2001.

[3] M. S. Mahmoud and M. G. Singh, *Large Scale Systems Modelling*, Pergamon Press, International Series on Systems and Control 1st ed., Vol. 3, 1981.

[4] M. Jamshidi, *Large Scale Systems Modelling and Control Series*, New York, Amsterdam, Oxford, North Holland, Vol. 9, 1983.

[5] R. Prasad and J. Pal, "Use of continued fraction expansion for stable reduction of linear multivariable systems", *Journal of Institution of Engineers, India, IE(I) Journal – EL*, Vol. 72, pp. 43-47, June 1991.

[6] R. Prasad, A. K. Mittal and S. P. Sharma, "A mixed method for the reduction of multi-variable systems", *Journal of Institution of Engineers, India, IE(I) Journal – EL*, Vol. 85, pp. 177-181, March 2005.

[7] Antonio Lepschy and Umberto Viaro," A Note on the Model Reduction Problem", *IEEE Transactions On Automatic Conixol*, Vol. Ac-28, No. 4, April 1983.

[8] R. Genesio and M. Milanese,"A note on the derivation and use of reduced order models", *IEEE Trans. Automat. Control*, Vol. AC-21, No. 1, pp. 118-122, February 1976.

[9] A. C. Antoulas, D. C. Sorensen, S. Gugercin ,"A survey of model reduction methods for large-scale systems", *Contemporary Mathematics*, Vol. 280, pp. 193-219, 2001.

[10] Boby Philip, Jayanta Pal, "An Evolutionary Computation Based Approach for Reduced Order Modelling of Linear Systems", *IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), 2010,* Coimbatore 28-29 Dec. 2010.

[11] Dia Abu-Al-Nadi, Othman MK Alsmadi and Zaer S.Abo-Hammour,"Reduced order modeling of linear MIMO systems using particle swarm optimization", *The Seventh International Conference on Autonomic and Autonomous Systems(ICAS 2011)*,2011.

[12] S.N. Sivanandam and S.N.Deepa,"A Comparative Study Using Genetic Algorithm and Particle Swarm Optimization for Lower Order System Modelling ", *International Journal of the Computer, the Internet and Management* Vol. 17. No.3 pp 1 -10 Sept - Dec, 2009.

[13] S. R. Desai, and Rajendra Prasad," Design of PID Controller using Particle Swarm Optimized Reduced Order Model", Eighth Control Instrumentation System Conference (An International Conference) CISCON-2011, Manipal, Nov. 3- 6, 2011.

[14] S. R. Desai and Rajendra Prasad,"Genetically Optimized Model Order Reduction for PID Controller ", *International Conference on System Dynamics and Control –ICSDC 2010.*

[15] http://www.mat.univie.ac.at/~neum/glopt.html

[16] Prasad R., Pant A. K. and Pal. J.,"Model Order Reduction using Modified Truncation", National System Conference, REC Kurukshetra, Dec.22-24,1987.

[17] Hassan, R., Cohanim, B., de Weck, O., "A Comparison of Particle Swarm Optimization and the Genetic Algorithm", 46th IAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics & Materials Conference, 18-21 April 2005,

[18] M. Clerc, "The swarm and the queen: towards a deterministic and adaptive particle swarm optimization", in *Proc. 1999 ICEC*, Washington, DC, 1999, pp. 1951-1957.

[19] J. Kennedy and R. Eberhart, "Particle swarm optimization", in *Proc. IEEE International Conference Neural Networks*, vol. 4, 1995, pp. 1942,1948.

[20] Kennedy, J., Eberhart, R., "Swarm Intelligence", Academic Press, ISBN 1-55860-595-9, 2001.

[21] Majid Bahrepour, Elham Mahdipour, Raman Cheloi, and Mahdi Yaghoobi," SUPER-SAPSO: A new sa-based pso algorithm", *Applications of Soft Computing, AISC* 58, pp. 423- 430,2009.

[22] Y. Shi and R. C. Eberhart, "A modified particle swarm optimizer", *Proc. IEEE International Conference on Evolutionary Computation*, Piscataway, NJ, IEEE Press, pp. 69–73, 1998.

[23] Jagdish C. Bansal, Kusum Deep, Kalyan Veeramachaneni, Lisa Osadciw,"Information Sharing Strategy among Particles in Particle Swarm Optimization Using Laplacian Operator", IEEE Swarm Intelligence Symposium, pp. 30-36, 2009.

[24] Prasad Rajendra,"Analysis and design of control Systems using reduced order models ", Ph. D Thesis, University of Roorkee, Roorkee,India, 1989.

[25] Shamash Y.,"Viability of methods for generating stable reduced order models",IEEE. Trans. On Automatic Control, vol. AC-26, No. 6, pp. 1285-1286,1981.

[26] S. Mukherjee, and R.N. Mishra, "Order reduction of linear systems using an error minimization technique", Journal of Franklin Inst., Vol. 323,No. 1, pp. 23-32, 1987.