ICSV13 - Vienna
The Thirteenth International Congress on Sound and Vibration
Vienna, Austria, July 2-6, 2006

MODELLING AND VIBRATION CONTROL OF A TWIN ROTOR SYSTEM: A PARTICLE SWARM OPTIMISATION APPROACH

M. S. Alam[1], M. H. Shaheed[2] and M. O. Tokhi[*1]

[1]Department of Automatic Control and Systems Engineering University of Sheffield, UK,
[2]Queen Mary College, University of London, UK
cop03msa@sheffield.ac.uk, o.tokhi@sheffield.ac.uk

Abstract
The construction and operation of twin rotor multi-input multi-output system (TRMS) in many aspects resemble that of a helicopter, with a significant cross-coupling between longitudinal and lateral directional motions. Moreover, flexible motion due to the unsymmetrical mass distribution of the system causes structural vibration while in operation. Command shaping is an effective control strategy to reduce vibration of flexible dynamic systems. Designing a conventional command shaper requires a priori knowledge of the system characteristics. This paper investigates a new method to extract parametric model of the system and to design a command shaper for vibration reduction of the system using particle swarm optimization (PSO) algorithm. Each parameter set, that forms the model or controller, is represented as a particle in the particle swarm. In order to control the global search and convergence to the global best solution, a derivative of particle swarm algorithm that uses time-varying inertia weight factor and time-varying acceleration coefficients is used in this work. The effectiveness of the proposed algorithms is verified and their performances in vibration suppression are assessed both in time and frequency domains.

INTRODUCTION

Recent advances in aircraft technology have led to the development of many new concepts in aircraft design. The twin rotor multi-input multi-output system (TRMS) is a laboratory platform designed for control experiments [3]. The TRMS can be perceived as an unconventional and complex "air vehicle" with a flexible main body. These system characteristics present formidable challenges in modelling, control design, and implementation. A number of techniques have been proposed and implemented to model and control structural vibration of such systems [1], [2], [8].

Particle swarm optimization (PSO) is a population-based, self-adaptive search optimization technique first introduced by Kennedy and Eberhart [4] in 1995. PSO has proved to be efficient at solving unconstrained global optimization and engineering problems. PSO has become increasingly popular mainly due to its simplicity, low memory requirement, low computational cost, fast convergence and its good overall performance. This paper investigates to extract parametric model of the vertical channel of TRMS and to design a command shaper [8], [9] to reduce structural vibration using PSO.

EXPERIMENTAL SET-UP

The TRMS consists of a beam pivoted on its base in such a way that it can rotate freely in both its horizontal and vertical planes producing two rotating movements around yaw and roll axes, respectively. At both ends of the beam, there are two rotors; main and tail rotors, driven by DC motors. A counterbalance arm with a weight at its end is fixed to the beam at the pivot[3]. The experimental TRMS and its schematic diagram are shown in Figures 1 and 2 respectively.

Figure 1- The twin rotor MIMO system.

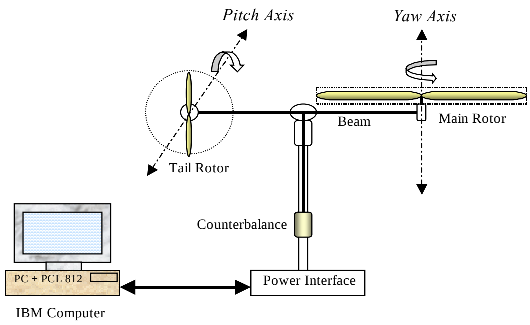

Figure 2: Schematic diagram of TRMS.

PARTICLE SWARM ALGORITHMS

The PSO algorithm works on the social behaviour of particles in the swarm. The position vector and the velocity vector of the ith particle in the d-dimensional search space can be represented as $X_i=(x_{i1}, x_{i2}, x_{i3},\ldots,x_{id})$ and $V_i=(v_{i1}, v_{i2}, v_{i3},\ldots,v_{id})$ respectively. According to a user defined fitness function, let the best position of each particle (which corresponds to the best fitness value obtained by that particle at time, t) be , $P_i=(p_{i1}, p_{i2}, p_{i3},\ldots,p_{id})$, and the fittest particle found so far at time t be $P_g=(p_{g1}, p_{g2}, p_{g3},\ldots, p_{gd})$. Then, the new velocities and positions of the particles for the next fitness evaluation are calculated using the following two equations[4]:

$$v_{id}= v_{id} + c_1 \times rand(\bullet) \times (p_{id} - x_{id}) + c_2 \times Rand(\bullet) \times (p_{gd} - x_{id}) \quad (1)$$

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

where c_1 and c_2 are constants known as acceleration coefficients, and rand(•) and Rand(•)are two separately generated uniformly distributed random numbers in the range [0,1]. Generally, a maximum velocity ($Vmax_d$) for each modulus of the velocity vector of the particles ($\mathbf{v}_{id}$) is defined in order to control excessive roaming of particles outside the user defined search space. Whenever a v_{id} exceeds the defined limit, its velocity is set to Vmax_d.

In order to control the global search and convergence to the global best solution, a derivative of particle swarm algorithm that uses time-varying inertia weight factor (ω) and time-varying acceleration coefficients c_1 and c_2 [7] is used in this work. The mathematical representation of this modified PSO is given as:

$$v_{id} = \omega \times v_{id} + c_1 \times rand(\bullet) \times (p_{id} - x_{id}) + c_2 \times Rand(\bullet) \times (p_{gd} - x_{id}) \tag{3}$$

where ω, c_1 and c_2 are given by

$$\omega = \omega_2 + (\omega_1 - \omega_2) \times (\mathrm{MAXITER} - \mathrm{iter})/\mathrm{MAXITER} \tag{4}$$
$$c_1 = c_{1i} + (c_{1f} - c_{1i})\ \mathrm{iter}/\mathrm{MAXITER} \text{ and } c_2 = c_{2i} + (c_{2f} - c_{2i})\ \mathrm{iter}/\mathrm{MAXITER}$$

where ω_1 and ω_2 are the initial and final values of the inertia weight, respectively, c_{1i}, c_{1f}, c_{2i} and c_{2f} are constants, iter is the current iteration number and MAXITER is the maximum number of allowable iterations.[7]. The commonly used PSOs are either global version or local version of PSO. Kennedy and Mendes tested PSOs with regular shaped neighbourhoods, such as global version, local version, pyramid structure, star structure, "small" structure, and von Neumann, and PSOs with randomly generated neighbourhoods[6].

DYNAMIC MODELLING OF TRMS

The vertical channel is excited with a sequence of psudo-random binary signal (PRBS), within ±0.2 volts and bandwidth (0-10 Hz) in order to ensure that all system resonance modes are captured. An input data of 500 points and its corresponding response are recorded and out of that 300 are used for modelling and the rest 200 for validating the model. Auto-regressive moving average (ARMA) structure is chosen to model the vertical channel. This is expressed as [5]:

$$\hat{y}(k) = -\sum_{i=1}^{N} a_i \times y(k-i) + \sum_{j=0}^{M} b_j \times u(k-j) + \eta(k) \tag{5}$$

where a_i, b_j are denominator and numerator coefficients, N and M are number of coefficients in denominator and numerator, y, u, $\hat{y}$, and η are measured output, input, predicted output and noise respectively. The order of the transfer function depends on N. Taking the values of N and M as 4 and 3 and neglecting the noise term η, equation (5) can be simplified as:

$$\hat{y}(k) = -a_1 y(k-1) \ldots - a_4 y(k-4) + b_0 u(k-1) \ldots + b_3 u(k-4) \tag{6}$$

In matrix form, the above equation can be written as:

$$\begin{aligned} \hat{y}(k) &= -[a_1, a_2, a_3, a_4][y(k-1), y(k-2), y(k-3), y(k-4)]^T \\ &+ [b_0, b_1, b_2, b_3][u(k-1), u(k-2), u(k-3), u(k-4)]^T \end{aligned} \tag{7}$$

Parameter Optimization with PSO

PSO optimization process begins with a population of real numbers called swarm. Each row represents a solution set called particle. A swarm of ten particles having eight elements each, i.e., 10 × 8 is created randomly within the range of -2 to +2. The first four elements are assigned to $b_0,...,b_3$ and the next four to $a_1,...,a_4$ as indicated in equation (7). The predicted output $\hat{y}$, at any sample instant, is calculated based on equation (7) and taking the elements of first particle, actual input and output data. Subsequent predicted outputs are calculated in the same way with the same parameters while taking consecutive input and output data. The difference between the predicted and actual output is recorded as error, $e(k) = y(k) - \hat{y}(k)$, which in turn is used to form the objective function (*f(x)*) of the optimization process. In this work, sum of absolute error is chosen as the objective function. The objective function is as follows:

$$f(x) = \sum_{k=1}^{300} \left| y(k) - \hat{y}(k) \right| \tag{8}$$

This process continues for each particle and then based on 'global version' or 'local version' particles are updated according to equations (3) and (4) for the next generation. The PSO optimization process is run for 200 generations.

Modelling results of global version of PSO: Particles at different generations are shown in Figure 3, 4 and 5. It is observed that at generation 200, all particles converge to the best value, found so far. From the convergence curve (see Figure 6) it is evident that the average objective function almost coincides with best objective value at generation 140 and there is almost no further improvement in the best solution in subsequent generations.

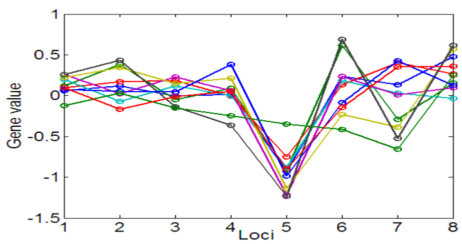

Figure 3- Particles at generation 25

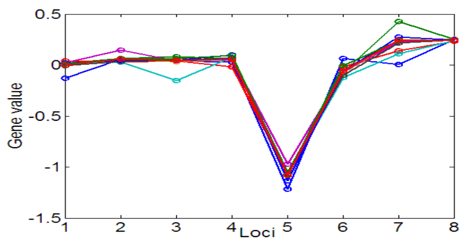

Figure 4- Particles at generation 75

Modelling results of local version of PSO: In this work, ring topology [6] is used to find the best guide for any particle while updating its velocity. Particles at different generations are shown in Figures 7, 8 and 9. It is observed that even at generation 500 there is some diversity among the particles and it does not converge to a single value. From the convergence curve (see Figure 10) it is evident that the mean objective function differs largely from the best objective value even at the end of 500 generations. Although the convergence is very slow, it may provide better solution due to diversity in the swarm. The discrete transfer function for vertical channel at a sampling time of 0.1sec is as follows:

$$H(z) = \frac{0.01021z^3 + 0.02482z^2 + 0.02499z + 0.007504}{z^4 - 1.77z^3 + 0.8893z^2 + 0.1039z - 0.03837} \tag{9}$$

Using MATLAB functions, this discrete transfer function can be converted into continuous form (s-domain) as:

$$H(s) = \frac{0.283s^4 - 0.6471s^3 + 312s^2 - 2110s + 145100}{s^5 + 47.99s^4 + 1774s^3 + 22030s^2 + 3.5290s + 397100} \tag{10}$$

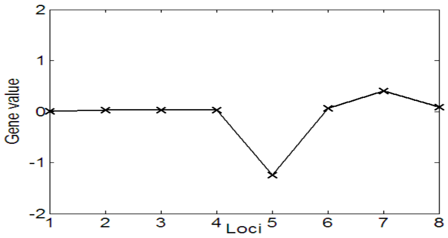

Figure 5- Particles at generation 200

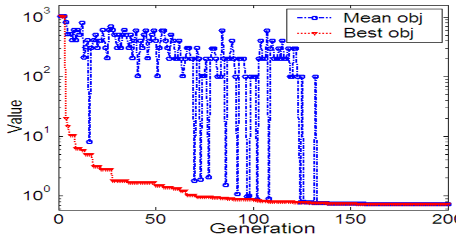

Figure 6- Convergence of PSO (global)

The derived model was validated with a separate input-output data set and the actual and one-step-ahead predicted output are shown in Figure 11. Time domain tracking reveals that the predicted output follows the actual output very well. The frequency domain plot (Figure 12) of the predicted and actual output indicates that the model has successfully captured the system dynamics, especially the main dominant modes at the low frequency region.

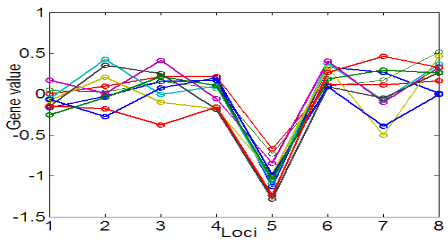

Figure 7- Particles at generation 100

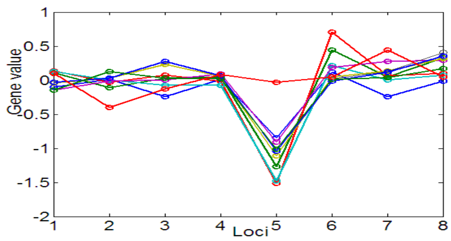

Figure 8- Particles at generation 200

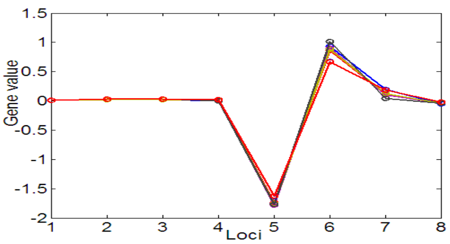

Figure 9- Particles at generation 500

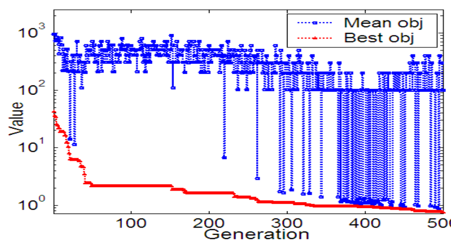

Figure 10- Convergence of PSO (local)

FEEDFORWARD VIBRATION CONTROL

Since its introduction [8] the method of command shaping has been applied to the control of different types of flexible systems. The design rules result in the amplitudes (A_i) and time locations (t_i) for ZV (zero vibration)-based impulses as [8], [9]

$$t_1 = 0, \quad t_1 = \frac{\pi}{\omega_d}, \quad A_1 = \frac{1}{1+K}, A_2 = \frac{K}{1+K} \tag{11}$$

where, $K = e^{-\zeta\pi/\sqrt{1-\zeta^2}}$, $\omega_d = \omega_n\sqrt{1-\zeta^2}$, ω_n is the natural frequency and ζ is the damping ratio of the system. Higher impulse command shapers are designed with the view to increase robustness to errors in natural frequencies of the system. The time locations and amplitudes of ZVD (zero vibration derivative)-based impulse and EI (extra insensitive)-based command shaper are as:

$$t_1 = 0, \quad t_1 = \frac{\pi}{\omega_d}, \quad t_3 = \frac{2\pi}{\omega_d}, A_1 = \frac{1}{1+2K+K^2}, A_2 = \frac{2K}{1+2K+2K^2}, A_3 = \frac{K^2}{1+2K+K^2} \tag{12}$$

$$t_1 = 0, \quad t_2 = \frac{\pi}{\omega_d}, \quad t_3 = \frac{2\pi}{\omega_d}, \quad t_4 = \frac{3\pi}{\omega_d}, A_1 = \frac{1}{1+3K+3K^2+K^3}, \tag{13}$$

$$A_2 = \frac{3K}{1+3K+3K^2+K^3}, \ A_3 = \frac{3K^2}{1+3K+3K^2+K^3}, \ A_4 = \frac{K^3}{1+3K+3K^2+K^3}$$

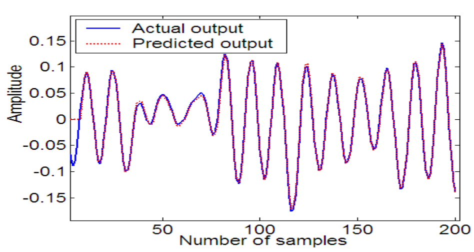

Figure 11- Actual vs predicted output (Time domain)

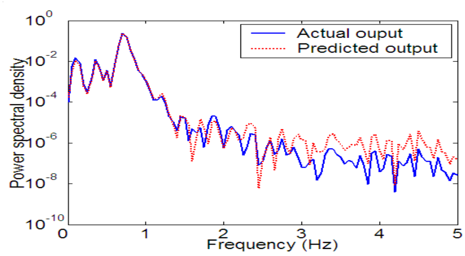

Figure 12- Actual vs predicted output (Frequency domain)

The command shaping method involves convolving a desired command with a sequence of impulses whose amplitudes and time locations are related to system parameters such as damping ratio, natural frequency, resonance mode/modes etc. So, designing a conventional command shaper requires a priori knowledge of the system such as, resonance modes and its associated damping ratios. Genetic algorithm (GA) was used to design unimodal and multimodal command shapers for flexible systems to reduce end point vibration [2]. In this work, global version of PSO is utilised to find the amplitudes and time locations of the impulses which are in turn convolved with the reference input to form the shaped command. The schematic diagram of the PSO based command shaper is shown in Figure 13. The PSO optimisation process is initialised with a random population consisting of 20 particles each having two elements within a range of 0.01 to 1. The first element of each particle, assigned to K, and is used to calculate the amplitude of impulses according to equation (11) (for ZV based command shaper). The next term is converted to a sample number by a conversion factor of 0.01 and used to calculate the time locations of the corresponding sequences. Equation (11) is used with random values of K in view of maintaining the same magnitude ratios among the impulses as provided by the theoretical method. Different types of objective function were tested but weighted sum of normalised rise time and settling time provided the best results. Similar techniques are used with equations (12) and (13) to design ZVD and EI based command shapers.

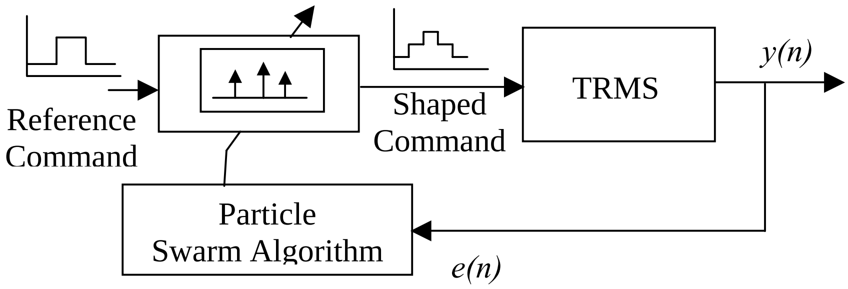

Figure 13- Command shaping for vibration control

RESULTS AND DISCUSSION

The leading edges of the shaped command based on ZV, ZVD and EI techniques are shown in Figure 14 and the responses of vertical channel due to shaped commands are shown in Figure 15. Compared to system's response due to unshaped bang-bang signal, significant amount of reduction in vibration has been achieved. For clarity, the system's response for leading edge of shaped input is shown in Figure 16. For ZV and ZVD based command shapers, wide range of oscillates are observed at the beginning whereas for EI based command shaper(CS), response is steady and stable without any oscillation. For ZV based CS, faster response (0.4 sec) causes higher overshoot (4.59%) and oscillation which in turn result in very large settling time (27.1 sec). For EI based CS, due to non-oscillatory behaviour, the rise time is quite satisfactory (1.7 sec) and it settles quickly (1.9 sec) with a small overshoot (1.7%). The frequency domain responses of the system due to unshaped bang-bang and different shaped commands are shown in Figure 17. For clarity, only the low frequency region is shown to highlight the vibration reduction at resonance mode (0.683Hz). Table-1 shows the reduction of vibration at resonance mode in dB for different shaping techniques. The reduction is -14.44, -34.74 and -34.81 dB for ZV, ZVD and EI based CS respectively. From the time and frequency domain plots and performance parameters, it is evident that, EI based CS outperformed the other two types.

Table 1- Performance measures of different command shapers

TRMS Channel	Shaping technique	Overshoot (%)	Rise time (Sec)	Settling time (Sec)	Steady state error	Vibration reduction (dB)
Vertical	ZV	4.59	0.4	27.1	0	-14.44
	ZVD	1.4	2.5	4.3	0	-34.74
	EI	1.7	1.3	1.9	0	-34.81

CONCLUSION

The PSO has been successfully used to model TRMS and to design different types of command shapers to reduce structural vibration. The modelling part was in fact, a highly multimodal problem with 8-D searching space whereas the controller design was relatively simpler, 2-D searching problem. Local version of PSO performed better in the modelling problem due to its inherent diversity, local searching capability and slow convergence. For controller design, global version of PSO produced quite satisfactory results. Although only the vertical channel has been explored in this work, these algorithms and techniques can be extended to include other channels and to solve multi-input multi-output (MIMO) modelling and control problems.

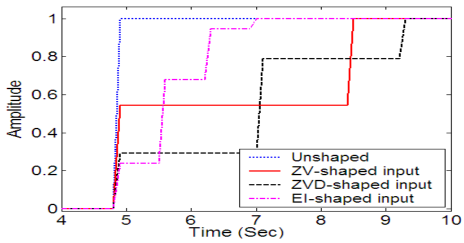

Figure 14- Unshaped and shaped command (leading edge)

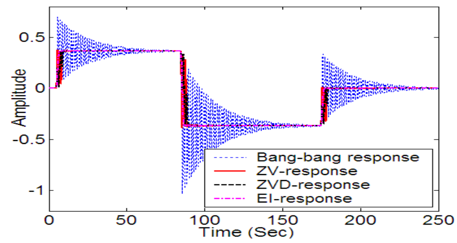

Figure 15- Time domain response

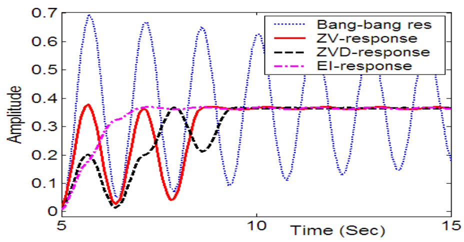

Figure 16- Time domain response (leading edge)

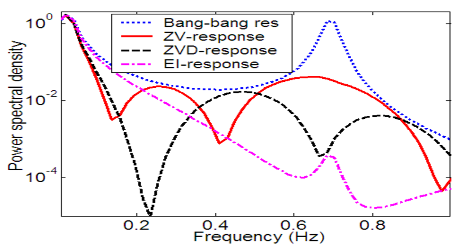

Figure 17- Frequency domain response

References

[1] Ahmad S. M., Chipperfield A. J. and Tokhi M. O., "Dynamic modelling and linear quadratic Gaussian control of a twin rotor MIMO system", Proceedings of IMechE-I: Journal of Systems and Control Engineering, **217**(I3), 203-227 (2003).

[2] Alam M. S., Aldebrez F. M. and Tokhi M. O., "Adaptive Command Shaping using Genetic Algorithms for Vibration Control", Proceedings of IEEE SMC UK-RI 3rd Workshop on Intelligent Cybernatic Systems, Ulster, United Kingdom, 7-8 Sept. (2004).

[3] Feedback Instruments Ltd. "*Twin Rotor MIMO System Manual 33-007-0*", Sussex, UK, (1996).

[4] Kennedy J. and Eberhart R. C. Particle swarm optimization, Proc. of IEEE International Conference on Neural Networks (ICNN), IV, 1942-1948, Perth, Australia, (1995).

[5] Ljung L, *System identification: Theory for the User*, 2nd ed. Englewood Cliffs, NJ: Prentice Hall, 1999.

[6] Mendes, R., Kennedy, J. and Neves J., "The fully informed particle swarm: simpler, maybe better", IEEE Transactions on Evolutionary Computation, **8**(3), 204 – 210 (2004).

[7] Ratnaweera A., Halgamuge S. K., and Watson H. C. Self-organizing hierarchical particle swarm optimizer with time varying accelerating Coefficients. IEEE Transactions on Evolutionary Computation, **8**(3), 240-255 (2004).

[8] Singer N. C. and Seering W. P., "Preshaping command inputs to reduce system vibration", Trans. ASME, J. Dynamic Systems, Measurement and Control, **112**(1), 76–82 (1990).

[9] Singhose W. E., Singer N. C. and Seering W. P., "Comparison of command shaping methods for reducing residual vibration", Proc. European Control Conference, Rome, pp. 1126–1131 (1995).