Evolutionary Algorithm for Identification of a Flexible Single-link Manipulator System

ALI A. M. AL-KHAFAJI, NIK M.R. SHAHARUDDIN, INTAN Z. M. DARUS
Department of Applied Mechanics & Design,
Faculty of Mechanical Engineering,
Universiti Teknologi Malaysia,
81310, Johor Bahru
MALAYSIA
ali1977abdalhusain@gmail.com, nmridzuan2@hotmail.my, intan@fkm.utm.my

Abstract: - This study presents an investigation into dynamic modelling of a flexible single-link manipulator system using differential evolutionary technique (DE) and particle swarm optimization technique (PSO). Details of simulation study, Model Selection, optimization and result analysis are given in this study. The input-output data of the system were first acquired through the simulation study using finite element method (FDM) based on Lagrangian approach. A bang-bang torque was applied as an input and the dynamic responses of the system were investigated. Next, an appropriate model structure was chosen and optimized using DE and PSO. One Step Ahead (OSA) prediction, correlation tests and mean squared error (MSE) have been performed for validation and verification of the obtained model in characterizing the manipulator system. Furthermore, an unseen data was used to observe the prediction ability of the model. A comparative assessment of the two models in characterizing the manipulator system is presented in time and frequency domains. Results demonstrate the advantages of DE over PSO in parametric modeling of the flexible manipulator system used in this study.

Key-Words: - Differential evolutionary algorithm, particle swarm optimization, flexible single-link manipulator, system identification, finite element method, dynamic modelling.

1 Introduction

Modern flexible link manipulator systems are widely used in many areas such as space vehicles, medical, defense and automation industries. The flexible link manipulator system, compared with rigid link manipulator system offers many advantages including low inertia, light weight, few powerful actuators, cheaper construction, fast response, safer operation, higher payload carrying capacity and longer reach. In general, the flexibility in modern robot manipulators system is an unwanted feature since it causes a serious of vibration problem when subjected to disturbance forces. Therefore the control of the flexible link manipulator robot becomes much more complex than rigid link robots. Accordingly, there is a growing need to develop suitable control strategies for such kind of systems [1], [2].

To control the vibration of a flexible link system effectively, it is often required to obtain an accurate or approximate model of the system structure, which will result in good control. Thus, finding an appropriate model of a dynamic system, such as a flexible system, would be the key to design an effective controller to suppress the vibration in the flexible link manipulator system [1].

The process of constructing a structure that characterizes the dynamics of a system is called system modelling. Different modeling techniques for analysis the flexible manipulators are studied by a number of researchers [1]. Mathematical method and numerical analysis have been proposed, developed and used by researchers. Through mathematical analysis, the dynamic model of the flexible manipulator dynamics is represented in the form of a partial deferential equation (PDE). The exact solution of such systems is not feasible practically and the infinite dimensional model imposes severe constraints on the design of controllers as well. Hence, the numerical analysis method was used to represent the flexible manipulator dynamics by solving the PDE using assumed modes, finite elements or lumped parameter methods [1], [2], [3].

To avoid solving the dynamics of flexible single-link manipulator, it can be modelled using system identification approaches [4]. System identification is the process of developing an accurate or

approximate dynamic model of a physical system using a set of input-output data pairs collected at input-output terminals of the system using experimental or simulation study. Modeling from first principles requires an in-depth knowledge of the system. System identification techniques can represent several system dynamics without knowledge of the actual physics of the system. This is one of the advantages of using the system identification method to represent the model of physical systems.

System identification consists of parametric and nonparametric techniques. Parametric modelling constitutes building a linear mathematical function of a dynamic system based on measured data in form of partial differential/difference equation or transfer function. Non parametric modelling is an attempt to represent plant with an input-output behavioral box, not as an explicit mathematical function.

There are two stages for these methods. The first stage is known as characterization, in which the assembly of the model such as type and order of the differential/difference equation that relate the input to the output is defined. The second stage is called identification, in which the numerical values of the structural parameters are determined. Construction of a model based on input-output data pair consist of four main components [5]:

1. Collection of a set of input-output data from experiment or simulation. The accuracy of the model is highly dependent on the accuracy of the collected data.

2. Determination of a suitable model structure from a set of candidate models.

3. Model estimation, which means using an assessment mechanism to determine the best model. The estimated model must have similar properties to that of the true one, it should be able to simulate the dynamic system, and predict the future values of the output.

4. Model validation, which means verifying whether the model is sufficient and if it has fulfilled the necessary requirements in representing the system.

Several parametric and nonparametric estimation methods can be found in the literature. Researchers and engineers have been developing and applying hard computing methods and soft computing methods [6]. One of the parametric soft computing methodologies is differential evolutionary algorithm.

Many parametric and nonparametric estimation techniques have been employed as optimization tools in identification of flexible link manipulator such as least mean square (LMS) and recursive least squares (RLS) [7], [8]; genetic algorithm (GA) [7], [9]; particle swarm optimization (PSO) [10], [11]; neural networks (NNs) [12], bacterial foraging algorithms (PFA) [13]. It can be noted from the literature that the use of DE for modelling manipulators has not been reported yet.

In this study, DE is compared to PSO in term of identification of a single-link flexible manipulator. First, simulation environment characterizing the dynamic behavior of the flexible manipulator structure is developed using FEM based on Lagrangian approach. Then, DE and PSO based identification are carried out using the input-output data acquired. The results are obtained in both time and frequency domains. The validity of the obtained model also will be investigated using OSA prediction, correlation tests and MSE.

2 Parametric system identification

Parametric system identification constitutes building mathematical model of a dynamic system based on measured data. So, the first step is to obtain the input-output data pairs through simulation or experimental work. Then, define a suitable model structure that could represent the system. After determining the model structure, the main task of identification is to estimate the model parameters, which are usually determined on the basis of a global minimum criterion function. The final step is to verify whether the model is adequate or not using validation tests

2.1 Flexible Manipulator System

The flexible single-link manipulator system considered in this study is shown in Fig. 1. The link has been modelled as a pinned-free flexible beam. The pinned end of the flexible beam of length L is attached to the hub with inertia I_h, where the input torque $\tau(t)$ is applied at the hub by a motor, and a payload mass M_p is attached at the free end. E, I and ρ represent the Young's modulus, second moment of inertia and mass density per unit length of the flexible manipulator respectively. X_0Y_0 axis and XY axis represent the stationary and moving coordinate respectively. Both axes lie in a horizontal plane and all rotation occurs about a vertical axis passing through o.

The flexible link, viewed as an Euler-Bernoulli beam, is modeled based on the following assumptions and constraints:

- The flexible link is viewed as a pinned-free flexible beam.
- The flexible link is assumed to be moving in the horizontal plane. Thus, perpendicular deformation is neglected as the gravitational term is not considered.
- Cross-section area and material properties remain constant in each segment.
- The effect of rotary inertia and shear deformation are ignored.

The mathematical modeling of a flexible single-link manipulator was derived using a finite element method based on Lagrangian approach [14], [15], [16]. The main step in a finite element analysis can be divided into six sections as follows:

1. Decompose the structure into finite elements, which are assumed to be interconnected at certain points. Number of elements determines the accuracy of analysis.
2. For each element, select an approximating polynomials function describing the behavior of the element using an approximation technique to interpolate the result.
3. Formulate the element characteristic matrices and vectors. The equations can be derived from the properties of the material and kinetic and potential energies.
4. Assemble the element matrices and vectors and derive the system equation. This equation describes the dynamic behavior of the system.
5. Incorporate the boundary condition. These conditions will prevent the structure from unlimited body motion.
6. Solve the system equations with the inclusion of the boundary condition.

In this manner, the overall approach involves treating the link of the manipulator as an assemblage of N elements of equal length $l = \frac{L}{N}$. For each of these elements the kinetic energy T_i and potential energy P_i are computed in terms of a suitably selected system of five generalized variables Q and their rate of change $\dot{Q}$. The development of the algorithm can be divided into three main steps: finite element method and Lagrangian approach analysis, state space representation and obtaining the result. An outline of this process is given below:

For a small angular displacement $\theta(t)$ and small elastic deflection $v(x,t)$, the overall displacement $y(x,t)$ of a point along the link at a distance x from the hub can be described as a function of both the rigid body motion $\theta(t)$ and elastic deflection $v(x,t)$ measured from the line passing through o:

$$y(x,t)=x\theta(t)+v(x,t) \quad (1)$$

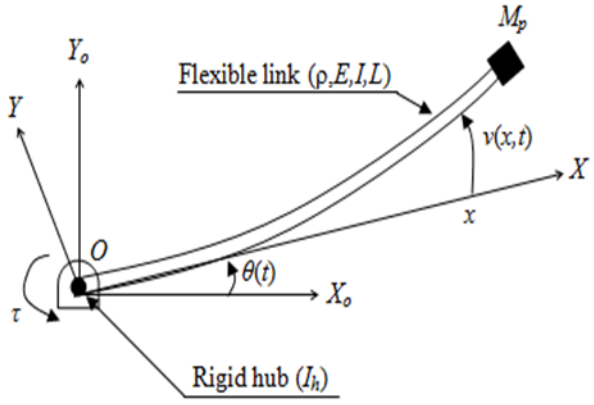


Fig.1: Schematic diagram of the flexible manipulator system

Using the standard FEM to solve dynamic problems, the flexible displacement $v_i(x,t)$ can be approximately expressed, as:

$$v_i(x,t)=N_i(s)\,Q_i(t) \quad (2)$$

where s, represent the local variable, $N_n(s)$ and $Q_n(t)$ represent the shape function and nodal displacement respectively. As a consequence of using the Euler-Bernoulli beam theory, the finite element method requires each the nodes through which we divide the beam into elements to possess two degrees of freedom, a transverse deflection and rotation at the same time when the beam is rotating. The displacements of two nodes of the ith element are denoted by v_n and v_{n+1}, and the rotations of the nodes by θ_n, θ_{n+1}. Because there are N elements in total, the number of generalized variables Q equal $[2N+2]$. These necessitate the use of Hermite cubic basis functions as the element shape function. Hence, the shape function can be obtained as:

$$N_n(s) = [\,\phi_1(s) \;\; \phi_2(s) \;\; \phi_3(s) \;\; \phi_4(s)]$$

where,

$$\phi_1(s) = 1 - \frac{(3s^2)}{l^2} + \frac{(2s^3)}{l^3} \qquad \phi_2(s) = s - \frac{(2s^2)}{l} + \frac{(s^3)}{l^2}$$

$$\phi_3(s) = \frac{(3s^2)}{l^2} - \frac{(2s^3)}{l^3} \quad , \quad \phi_4(s) = \frac{(s^3)}{l^2} - \frac{(s^2)}{l}$$

and the generalized variables

$$Q_i(t) = [v_n(t) \;\; \theta_n(t) \;\; v_{n+1}(t) \;\; \theta_{n+1}(t)\,]^T$$

Substituting equation (2) into (1) gives

$$y(x,t)=x\theta(t)+\sum_{i=1}^{N} N_i(s)\,Q_i(t) = N_i'(s)Q_i'(t) \tag{3}$$

where,

$$N_i'(s)=[x\ N_i(s)] \tag{4}$$

$$Q_i'(t)=[\theta(t)\ \ Q_i(t)\ ]^T \tag{5}$$

The new shape function $N_i'(s)$ in equation (4) and new nodal displacement vector $Q_i'(t)$ in equation (5) incorporate local and global variables. Among these, the angle $\theta(t)$ and the distance x are global variables while $N_i(s)$ and $Q_i(t)$ are local variables. Define, $s = x - \sum_{i=1}^{n-1} l_i$ as a local variable of the ith element, where l_i is the length of the ith element. The equations (4) and (5) can be expressed as:

$$N_i'(s)=[(s+l(N\text{-}1))\ N_i(s)] \tag{6}$$

$$y(s,t)=N_i'(s)Q_i'(t) \tag{7}$$

According to the basic finite element method and energy principle, the kinetic energy and potential energy of ith element of the link can be acquired according to equation (8) and (10) as follows:

$$\begin{aligned} T_i &= \frac{1}{2}\int_0^t \rho A\left[\frac{\partial y(s,t)}{\partial t}\right]^2 ds = \frac{1}{2}\int_0^t \rho A\dot{Y}^T\dot{Y} ds \\ &= \frac{1}{2}\dot{Q}_i'^T\left[\int_0^l \rho A(N_i'^T N_i')ds\right]\dot{Q}_i' \\ &= \frac{1}{2}\dot{Q}_i'^T[M_e]\dot{Q}_i' \end{aligned} \tag{8}$$

where

$$M_i = \int_0^l \rho A(N_i'^T N_i')ds \tag{9}$$

is element mass matrix

$$\begin{aligned} P_i &= \frac{1}{2}\int_0^t EI\left[\frac{\partial^2 y(s,t)}{\partial s^2}\right]^2 ds \\ &= \frac{1}{2}\int_0^t EI(B_iQ_i')^T(B_iQ_i')ds \\ &= \frac{1}{2}Q_i'^T\left[\int_0^l EI(B_i^T B_i)ds\right]Q_i' \\ &= \frac{1}{2}Q_i'^T[K_e]Q_i' \end{aligned} \tag{10}$$

where

$$K_i = \int_0^l EI(B_i^T B_i)ds \tag{11}$$

is element stiffness matrix

$$B_i = \frac{d^2 N_i'(s)}{ds^2}$$

For a flexible single-link manipulator system, the mass matrix consists of three terms; mass matrix due to the structural mass of the manipulator, mass matrix due to hub inertia and mass matrix due to tip payload mass. A brief outline of the mass matrices is given below. The kinetic energy of the ith element due to structural mass of the manipulator can be obtained using equation (8). Thus,

$$T_{is} = \frac{1}{2}\dot{Q}_i'[M_i]_s\dot{Q}_i' \tag{12}$$

the element structural mass matrix yields:

$$[M_i]_s = \frac{\rho_s A_s l}{420}\begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} & m_{15} \\ m_{21} & 156 & 22l & 54 & -13l \\ m_{31} & 22l & 4l^2 & 13l & -3l^2 \\ m_{41} & 54 & 13l & 156 & -22l \\ m_{51} & -13l & -3l^2 & -22l & 4l^2 \end{bmatrix} \tag{13}$$

where A_s is the cross section area of the manipulator, and ρ_s is the density of the material of the manipulator and

$m_{11} = 140l^2(3N^2 - 3N + 1)$, $m_{12} = m_{21} = 21l(10N - 7)$

$m_{13} = m_{31} = 7l^2(5N - 3)$, $m_{14} = m_{41} = 21l(10N - 3)$

$m_{15} = m_{51} = -7l^2(5N - 2)$

and the generalized variables are $Q_i'(t) = [\theta(t)\ \ \theta_i(t)\ \ v_i(t)\ \ v_{i+1}(t)\ \ \theta_{i+1}(t)]$. The kinetic energy of tip payload mass can be found using equation (14):

$$\begin{aligned} T_p &= \frac{1}{2}M_p\left[\frac{\partial y(s,t)}{\partial t}\right]^2 = \frac{1}{2}M_p\dot{Y}_i^T\dot{Y}_i \\ &= \frac{1}{2}M_p\dot{Q}_i'^T N_i'^T N_i'\dot{Q}_i' = \frac{1}{2}\dot{Q}_i'^T[M_p N_i'^T N_i']\dot{Q}_i' \\ &= \frac{1}{2}\dot{Q}_i'^T[M]_p\dot{Q}_i' \end{aligned} \tag{14}$$

where $[M]_p = M_p N_i'^T N_i'$. Hence, the inertia matrix of tip payload mass M_p is:

$$[M_n]_p = \begin{bmatrix} L^2M_p & 0 & 0 & LM_p & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ LM_p & 0 & 0 & M_p & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{15}$$

and the generalized variables are $Q_i'(t) = [\theta\ \ \theta_N\ \ v_N\ \ \theta_{N+1}\ \ v_{N+1}]$. Rotational kinetic energy of driving end can be found using equation (16):

$$T_h = \frac{1}{2}I_h\left[\frac{\partial\theta(t)}{\partial t}\right]^2 = \frac{1}{2}I_h\dot{Q}_i'^T\dot{Q}_i' \tag{16}$$

The inertia matrix of driving end is expressed as:

$$[M]_m = \begin{bmatrix} I_h & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (17)$$

and generalized variables are $Q_i'(t) = [\theta\ \theta_1\ v_1\ \theta_2\ v_2]$.

Similarly, the potential energy due to the elasticity of the FEM can be obtained using equation (10) and the element stiffness matrices can be expressed

$$[K_n]_i = \frac{EI}{l^3} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 12 & 6l & -12 & 6l \\ 0 & 6l & 4l^2 & -6l & 2l^2 \\ 0 & -12 & -6l & 12 & -6l \\ 0 & 6l & 2l^2 & -6l & 4l^2 \end{bmatrix} \quad (18)$$

The global elements mass matrix has been achieved by assembling the element mass matrices in equation (13) and adding the payload mass matrix (15) and hub inertia mass matrix (17) as in equation (19).

$$M_n^* = \begin{bmatrix} [M_n]_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & [M_n]_N \end{bmatrix} \quad (19)$$

The global elements stiffness matrix has been achieved by assembling the element stiffness matrices in equation (18) as in equation (20).

$$K_n^* = \begin{bmatrix} [K_n]_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & [K_n]_N \end{bmatrix} \quad (20)$$

Considering node coordinates overlapping between neighborhood elements when assembled, the general coordinate is introduced as $Q_z = [\theta\ \theta_1 v_1 \theta_2 v_2 \theta_3 v_3 \cdots \theta_{N-1} v_{N-1} \theta_N v_N \theta_{N+1} v_{N+1}]$. While taking boundary conditions into account, the beam is fixed on one end and there holds $\theta_1 = w_1 = 0$. Then θ_1 and w_1 can be removed from the general coordinates, while the corresponding row and column in the mass matrix and stiffness matrix can also be removed. The total kinetic and potential energies from equations (12), (14), (16) and (10) can be written as:

$$T = T_h + \sum_{i=1}^{N} T_{is} + \sum_{i=1}^{N} T_{ia} + T_p = \frac{1}{2} \dot{Q}_n^{*T} M_n^* \dot{Q}_n^* \quad (21)$$

$$P = \sum_{i=1}^{N} P_i = \frac{1}{2} Q_n^{*T} K_n^* Q_n^* \quad (22)$$

where the local coordinate $\dot{Q}_n = [[Q_n]_1\ [Q_n]_2 \cdots [Q_n]_N]^T$

The dynamic equations of motion of the flexible manipulator can be derived utilizing the Lagrange equation based on general coordinates:

$$\frac{d}{dt}\left\{\frac{\partial L}{\partial \dot{Q}}\right\} - \left\{\frac{\partial L}{\partial Q}\right\} = F \quad (23)$$

where $L = T - P$ is the Lagrantian and F is a vector of external forces. Considering the damping, the desired dynamic equations of motion of the system can be obtained as:

$$M^*\ddot{Q}^* + D^*\dot{Q}^* + K^*Q^* = b'\tau \quad (24)$$

where M^* is the global mass matrix which consist of the structural mass, added mass of fluid, payload mass and hub mass. K^* is global rigidity matrix and Q^* is the general coordinates when substituting boundary conditions. $b' = [1\ 0\ \cdots\ 0]^T$ and τ is input torque. $Q(t)$ is the nodal displacement given as:

$Q(t) = [\theta\ w_2\ \theta_2 \cdots\cdots w_N \theta_N]$

D^* is structural damping matrix due to structural material. For the flexible manipulator under consideration, the global mass matrix can be represented as:

$$M^* = \begin{bmatrix} M_{\theta\theta} & M_{\theta v} \\ M_{\theta v}^T & M_{vv} \end{bmatrix}$$

where M_{vv} is associated with the elastic degrees of freedom (residual motion), $M_{\theta v}$ represents the coupling between these elastic degrees of freedom and the hub angle θ and $M_{\theta\theta}$ is associated with the inertia of the system about the motor axis. Similarly, the global stiffness matrix can be written as:

$$K^* = \begin{bmatrix} 0 & 0 \\ 0 & K_{vv} \end{bmatrix}$$

where K_{vv} is associated with the elastic degrees of freedom (residual motion). It can be shown that the elastic degrees of freedom do not couple with the hub angle through the stiffness matrix. The global damping matrix D^* in equation (24) can be represented as

$$D^* = \begin{bmatrix} 0 & 0 \\ 0 & D_{vv} \end{bmatrix}$$

where D_{vv} denotes the sub-matrix associated with the material damping. The matrix is obtained by assuming that the beam exhibits the characteristics of Rayleigh damping. This proportional damping model has been assumed because it allows experimentally determined damping ratios of

individual modes to be used directly in forming the global matrix. It also allows assignment of individual damping ratios to individual modes, such that the total beam damping is the sum of the damping in the modes. Using this assumption, the damping can be obtained as

$$[D_{vv}] = \alpha[M_{vv}] + \beta[K_{vv}] \quad (25)$$

where α, and β, are the Rayleigh damping coefficients related to the modal damping and frequency of the manipulator by:

$$\alpha + \omega_i^2 \beta = 2\zeta_i \omega_i \quad (26)$$

where ζ_i and ω_i are the ith modal damping ratio and frequency, respectively. Equation (26) indicates that the more samples of the modal damping ratio and frequency, the more accurate estimation of the Rayleigh damping coefficients. However, it is a common practice in engineering application to use the lower and upper cutoff frequencies of the manipulator system and the corresponding modal damping ratios to define the values of the Rayleigh damping coefficients, such that,

$$\alpha = 2\frac{\omega_1\omega_2(\omega_2\zeta_1 - \omega_1\zeta_2)}{\omega_2^2 - \omega_1^2},\ \beta = 2\frac{\omega_2\zeta_2 - \omega_1\zeta_1}{\omega_2^2 - \omega_1^2}$$

where ω and ζ are the modal frequency and damping ratio and the subscripts '1' and '2' are the lower and upper bounds of the frequency region in interest, respectively.

The M^*, D^*, K^* matrices are of size $m_1 * m_1$ and $b' = [1 \ 0 \ \cdots \ 0]^T \tau$ has $m_1 * 1$ size and $m_1 = 2n+1$. For simplicity $D' = 0$ and $Q(0) = 0$

Now, the equation of motion is expressed in state-space form, so that it can be solved using control system approaches.

The state-space form of the equation of motion is:

$$\dot{v} = Av + Bu$$
$$y = Cv + Du$$

where

$$A = \begin{bmatrix} 0_{m1} & \vdots & I_{m1} \\ \cdots & \cdots & \cdots \\ -M^{-1}K & \vdots & -M^{-1}D \end{bmatrix}$$

$$B = \begin{bmatrix} 0_{m1*1} \\ \cdots \\ M^{-1}e \end{bmatrix}$$

$$C = [0_{m1} \ \vdots \ I_{m1}]$$

$$D = [0_{2m1*1}]$$

0_{m1} is an $m_1 * m_1$ null matrix, I_{m1} is an $m_1 * m_1$ identity matrix, 0_{2m1*1} is an $m_1 * 1$ null vector, and the vector e is the first column of the identity matrix.

$$u = [\tau \ \ 0 \ \cdots \ 0]^T$$

$$\mathrm{v} = [\theta \ \mathrm{u}_2 \ \theta_2 \cdots \mathrm{u}_{n+1} \ \theta_{n+1} \ \dot{\theta} \ \dot{\mathrm{u}}_2 \ \dot{\theta}_2 \cdots \dot{\mathrm{u}}_{n+1} \ \dot{\theta}_{n+1}]^{\mathrm{T}}$$

2.2 Model Structure

A variety of model structures are available to assist in modelling a system. The choice of model structure is based upon an understanding of the system identification method and insight and understanding into the system undergoing identification. The autoregressive moving average model with exogenous inputs (ARMAX) model is one of the most popular linear models [17], [18]. The ARMAX Model structure provides a complete model with disturbance properties modelled separately from system dynamics. If the model is good enough to identify the system without incorporating the noise term or considering the noise as additive at the output, the model can be represented in the ARX form. The ARX model is the simplest model incorporating the stimulus signal. The estimation of the ARX model is the most efficient of the polynomial estimation methods because it is the result of solving linear regression in analytic form. Moreover, the solution is unique. In other words, the solution always satisfies the global minimum of the loss function. The ARX model therefore is preferable [18].

As the simulation data would be collected by the sampling process from the simulation procedure, it is more straightforward to relate the observed data to a discrete time model. The schematic of ARX model is shown in discrete domain as in Fig. 2.

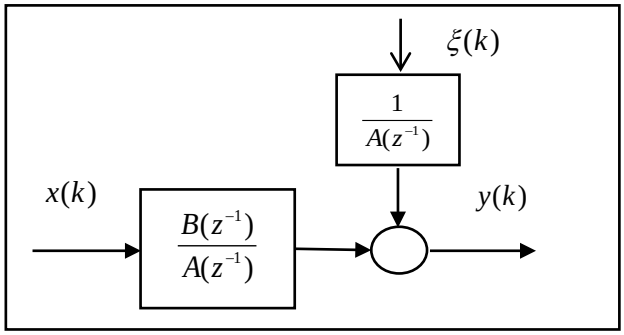


Fig.2: Schematic of ARX model [19]

Mathematically the ARX model is given by the following equation:

$$y(k)=\frac{B(z^{-1})}{A(z^{-1})}x(k)+\frac{1}{A(z^{-1})}\xi(k) \quad (27)$$

where

$A(z^{-1})=1+a_1z^{-1}+a_2z^{-2}+\cdots+a_nz^{-n}$

$B(z^{-1})=b_1z^{-1}+b_2z^{-2}+\cdots+b_nz^{-n}$ are polynomials with associated parameters. $x(k)$ is the input data, $y(k)$ is the output data, $\xi(k)$ is the zero mean white noise. z^{-1} is a back-shift operator, n is order of the model and a_1 until a_n and b_1 until b_n are the model parameters. The main objective of system identification is thus to estimate the model parameters as best as possible.

2.3 Parameters estimation

The main task of identification after determining the model structure is to estimate the model parameters. Optimization methodologies are techniques to estimate parameters in a given model structure essentially by finding (by numerical search) those numerical values of the parameters that give the best agreement between the simulated or predicted output of the model and the measured one. Fig. 3 shows the principle of system identification via optimization methodologies. As noted, optimization methodology uses error function to estimate the model parameters.

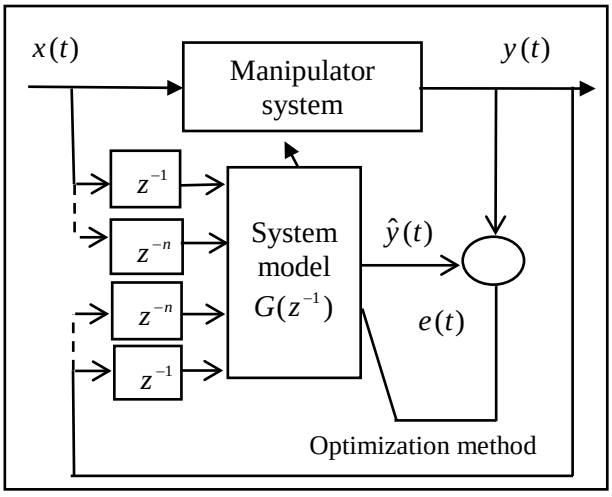


Fig.3.Diagrammatic representation of the principle of system identification via optimization methodologies

where $x(k)$ is the actual input, $y(k)$ is the actual output, $\hat{y}(k)$ is the predicted output and $e(k)$ is the prediction error. The predicted error $e(k)$ between the system output $y(k)$ and OSA estimated model output $\hat{y}(k)$ at time k is: $e(k)=y(k)-\hat{y}(k)$. The mean square error is defined as: $\mathrm{MSE}=\frac{1}{N}\sum_{k=1}^{N}[y(k)-\hat{y}(k)]^2$. An obvious approach is then to estimate the model parameters so as to fit the predicted output $\hat{y}(k)$ as best as possible to the real output $y(k)$. In other words, the parameters should be estimated so that MSE converges to zero. Therefore, MSE was employed as the fitness function of the optimization methodology and the optimization process of the optimization methodology was conducted to estimate the model parameters so that the value of MSE was reduced to a distinct level. Several optimization methodologies can be found in the literature. Two of these methods are DE and PSO.

2.3.1 Differential Evolution Algorithm

Evolutionary algorithms are used effectively for solving difficult optimization problems such as nonlinear global optimization problems subjected to multiple nonlinear constraints [20]. The differential evolution algorithm (DE) proposed by Storn and Price is a population-based stochastic optimization algorithm developed to solve nonlinear and multi-modal global optimization problems. DE is an efficient, robust, and effective algorithm. It uses mutation and crossover operations as searching mechanisms and selection operations to direct the search to the most promising regions of the search space. New candidate individuals are made by combining the target individual and several other individuals by randomly choosing from the same population, where the candidate with a better fitness value replaces the target individual [21]. The DE can be outlined as shown in Fig. 4.

The basic operations of differential evolution can be described as:

Initialization: All the DE optimization parameters required for optimization process are listed below:

- Problem dimension, D.
- Population size, N.
- Number of generation /stopping condition, G.
- Boundary constraints, L-H.

The upper and lower bounds of each parameter is set based on the parameters range such that $X_j^L \le X_{j,i,G} \le X_j^U$ where $j=1,2,\ldots,D$ and $i=1,2,\ldots,N$. After that, the target (parent) vector is generated by a random number assigned to each parameter of every vector within the prescribed

range $[X_j^L, X_j^U]$. Equation (28) shows how a target (parent) vector is created.

$$X_{j,i,(0)} = \{X_{1,i,(0)}, X_{2,i,(0)}, \ldots X_{(D),i,(0)}\}$$

$$X_{j,i,0} = \text{rand}_j(0,1).(x_j^U - x_j^U) + x_j^L \quad (28)$$

After that, evaluation of fitness function of each vector is implemented. At the first generation $X_{j,i,(G)} = X_{j,i,(0)}$.

Mutation: Mutation can be viewed as an operation that generate Mutant individuals from the target vectors such that the weighted difference of two randomly selected individuals from the target vectors are multiplied by a constant factor and then added with a randomly selected individual from the target vectors. Equation (29) shows how a mutant (donor) vector is created.

$$V_{j,i,(G)} = \{V_{1,i,(G)}, V_{2,i,(G)}, \ldots V_{(D),i,(G)}\}$$

$$V_{j,i,(G+1)} = V_{j,r3,(G+1)} + F(V_{j,r1,(G+1)} - V_{j,r2,(G+1)}) \quad (29)$$

The mutation factor (or scale factor), F is a positive real constant that controls the rate of the differential variation between two individuals and its value lies between 0 and 2. This type of mutation is called de/rand/1. There are other forms of creating the mutant vector frequently used in the literature.

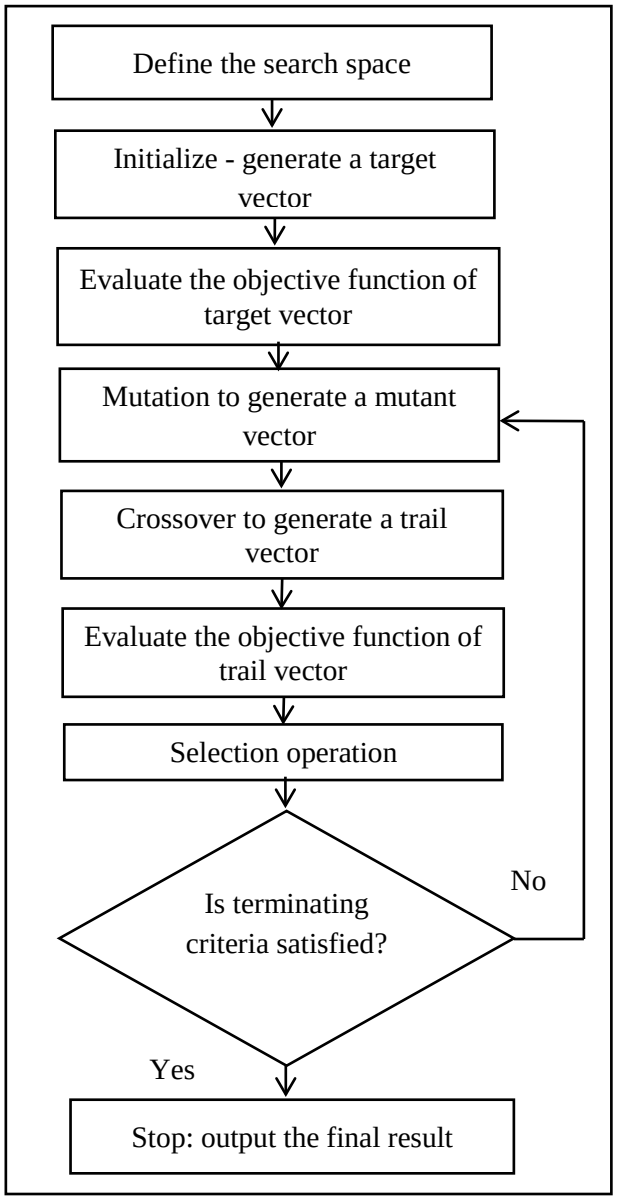


Fig.4: Working Principle of DE

Crossover: crossover is introduced to increase the diversity of the perturbed parameters for each individual in the population. The aim of applying the crossover is to achieve trial vectors by replacing a certain individual in the target vectors with corresponding individual in the donor vectors such that an individual is selected randomly from the mutant individuals when rand (D) is less than the value of the crossover probability. Equation (30) shows how a trail vector is created.

$$U_{j,i,(G)} = \{U_{1,i,(G)}, U_{2,i,(G)}, \ldots U_{(D),i,(G)}\}$$

$$U_{j,i,(G+1)} = \begin{cases} V_{j,i,(G+1)} & if\ rand(D) \le CR \\ X_{j,i,(G+1)} & otherwise \end{cases} \quad (30)$$

The crossover ratio controls the fraction of parameter values that are to be copied from the mutant vector. If the value of a first random number

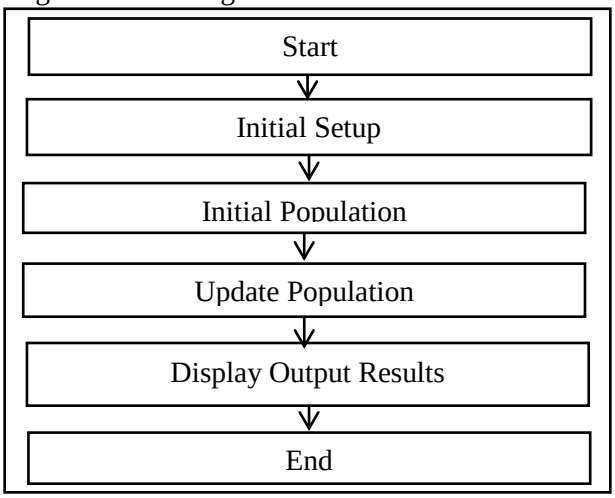
is less than the chosen CR then the corresponding element of mutant vector is passed on to the target vector otherwise it is copied from the trial vector. This process is repeated for all elements and for the entire population.

Selection: If the objective function value of the individual in the trial vector $U_{j,i,(G+1)}$ has an equal or lower corresponding individual in the target vectors $X_{j,i,(G)}$, it replaces the target vector in the next generation; otherwise, the target retains its place in the population for at least one generation. Equation (31) shows how a mutant (donor) vector is created.

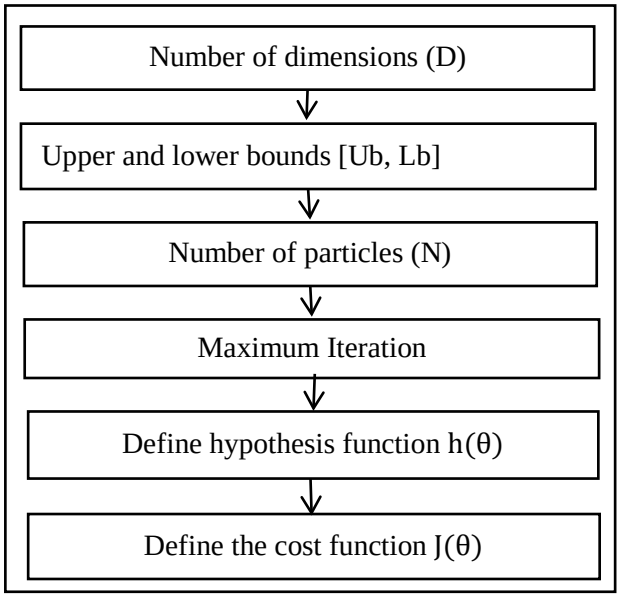
$$U_{j,i,(G+1)} = \begin{cases} U_{j,i,(G+1)} & if\ f(U_{j,i,(G+1)}) \le f(X_{j,i,(G+1)}) \\ X_{j,i,(G)} & otherwise \end{cases} \quad (31)$$

2.3.2 Particle Swarm Optimization

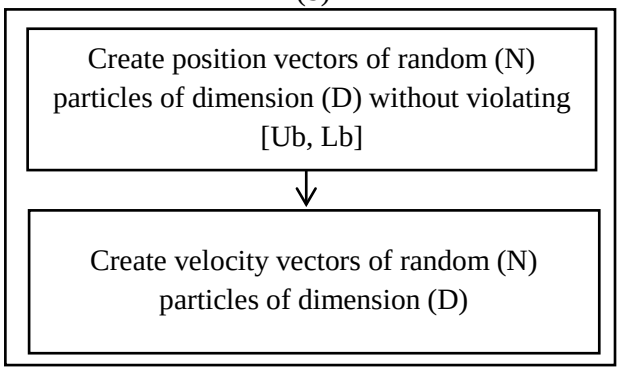
Particle swarm optimization is a population based, stochastic optimization technique introduced by Kennedy and Eberhart [22]. The operating procedure of a PSO can be described through the stages shown in Fig. 5.



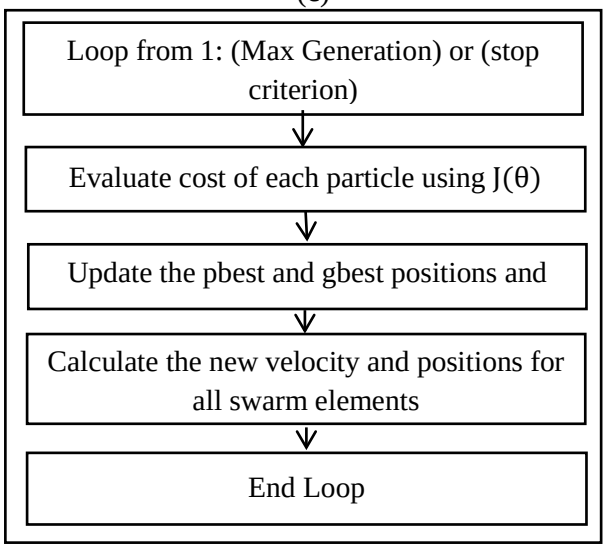
(a)



(b)



(c)



(d)

Fig. 5: Flowchart of PSO algorithm: (a) PSO procedure, (b) Initial setup procedure of PSO, (c) Initial population procedure of PSO, (d) Update population procedure of PSO

PSO is initialized with a group of random particles, 'fly' in the search space of an optimization problem. Particles are updated with two 'best' values every iteration. The first one is called *pbest*, which is the best position a particle has visited so far and is memorized. Another 'best' value is the global best or *gbest*, obtained so far by any particle in the population. Using their memories of the best positions of *pbest* and *gbest*, particle is then accelerated toward those two best values by updating the particle position and velocity using the following set of equations:

$$v_{id}(t) = wv_{id}(t-1) + c_1 rand(p_{id} - x_{id}(t-1)) + c_2 rand(p_{gd} - x_{id}(t-1))$$

$$x_{id}(t) = x_{id}(t-1) + v_{id}(t)$$

where $v_{id}(t)$ and $x_{id}(t)$ are the current velocity and position vector of the *i*-th particle in the *d*-dimensional search space respectively. C_1 and C_2 are acceleration coefficients usually $C_1 = C_2 = 2$ and *rand* is a random number between 0 and 1. w is the inertia which serves as memory of the previous direction, preventing the particle from drastically changing direction. High values of w promote global exploration and exploitation while low values of w lead to a local search. The common approach is to provide balance between global and local search by linearly decrease w during the search process. Decreases the inertia over time can be expressed as:

$$w(t) = w_{start} - \frac{w_{start} - w_{end}}{T_{\max}} t$$

where w_{start} and w_{end} are the starting and end point of inertia weight set as 0.9 to 0.25 respectively and $T_{\max}$ is the maximum number of time step the swarm is allowed to search.

2.4 Model Validation

Model validity tests are procedures designed to detect the adequacy of a fitted model. Once a model of the system is obtained, it is required to validate the model whether it is good enough to represent the system or not. Many model validation tests are available in the literature. In this study, mean squared error, one step-ahead prediction and correlation tests were used to validate the model. Furthermore, an unseen data used to observe the prediction ability of the model and pole-zero plots test was used to check the stability of the obtained models.

2.4.1 One Step-Ahead Prediction (OSA)

A common measure of predictive accuracy used in control and system identification is to compute the OSA prediction of the system output. This is expressed as:

$$\hat{y}(k) = -a_n y(k-n_x) \cdots\cdots - a_1 y(k-1) + b_n y(k-n_y) \cdots\cdots + b_1 y(k-1)$$

Often $\hat{y}(k)$ will be a relatively good prediction of $y(k)$ over the estimation set, even if the model is biased, because the model was estimated by minimizing the prediction errors.

2.4.2 Mean Squared Error

Mean squared error is one of the most common variables used in the validations. The mean-squared error is difference between actual output of the system and the predicted output produced from the input to the system and the optimized parameters.

2.4.3 Correlation Tests

Correlation Tests is a more convincing method to be used in model validation. Correlation test is a statistical test that shows the degrees of the relationship between two variables. There are two types of correlation test:

1. Auto correlation test is representing as a vector.
2. Cross correlation test is representing as matrix.

In the theory of linear systems, the usual statistical approach to validate identified linear and nonlinear models consists of computing the autocorrelation test examines the correlation among the residuals themselves while the cross-correlation test surveys the correlation between the residuals and past inputs. It has been shown that acceptable predictions over different data sets are produced only if the model is unbiased. If the model structure and the estimated parameters are correct then the prediction error sequence e(t) should be uncorrelated with all linear and nonlinear combinations of past inputs and outputs (unbiased) and this will hold if and only if the following conditions are satisfied :

$$\phi_{ee}(\tau) = E[e(t-\tau)e(t)] = \delta(t)$$
$$\phi_{ue}(\tau) = E[u(t-\tau)e(t)] = 0, \quad \forall \tau$$
$$\phi_{u^2e}(\tau) = E[(u^2(t-\tau) - u^2(t))e(t)] = 0, \quad \forall \tau$$
$$\phi_{u^2e^2}(\tau) = E[(u^2(t-\tau) - u^2(t))e^2(t)] = 0, \quad \forall \tau$$
$$\phi_{(e)(eu)}(\tau) = E[e(t)e(t-1-\tau)u(t-1-\tau)] = 0 \quad \tau \geq 0$$

In practice, the correlation will never be exactly zero for all lags but the model is considered as adequate if the correlation tests lay within 95% confidence limits, defined as $1.96/\sqrt{N}$, where N is the data length. Autocorrelation of the residual will also never be an ideal delta function but will be considered as adequate if the autocorrelation plot enters the 95% confidence limits before lag one.

Ideally the model validity test should detect all the deficiencies in algorithm performance including bias due to internal noise. Consequently the full five tests defined by equation should be satisfied. For linear system only first two conditions must be satisfied.

3. Implementation and Results

To study the dynamic behavior of the flexible manipulator system, a computer program was written within Matlab environment to simulate the state space matrices derived from the mathematical modeling (referred to section 2.1). A thin aluminium alloy with the specifications shown in Table 1 is considered [3].

For simplicity purposes, the effect of mass payload is neglected. Throughout this simulation, a bang-bang torque input with an amplitude ±0.3 Nm was applied at the hub of the manipulator as shown in Fig. 6. The response of the flexible manipulator at the hub angle and end-point residual is monitored for duration of 3.0 seconds with sampling time 0.37 ms and is observed and recorded as shown in Figs. 7 (a), (b) and 8 (a), (b) in both time and frequency domain respectively. Previous researches demonstrated the validity of FEM in modeling flexible single link manipulator system []. Therefore, the same FEM is adopted in this study.

The flexible manipulator is a single-input multiple-output (SIMO) system. The input, is the torque of the motor and the outputs are the hub-angle and end-point residual. In this study, two single-input single-output (SISO) models are developed. The first model represents system behavior from input torque to hub-angle output and the second model represents system behavior from input torque to the end-point residual output.

Table 1
Physical parameters of the flexible manipulator

Components	**value**
Length (l)	960 mm
Width (w)	19.008 mm
Thickness (h)	3.2004 mm
Mass density (ρ)	2710 $\mathrm{kg\,m^{-3}}$
second moment of inertia (I)	$5.19 \times 10^{-11}\ \mathrm{m^4}$
Young's modulus (E)	$71\times10^{9}\ \mathrm{N\,m^{-2}}$
hub inertia (I_h)	$5.86\times10^{-4}\ \mathrm{kg\,m^2}$

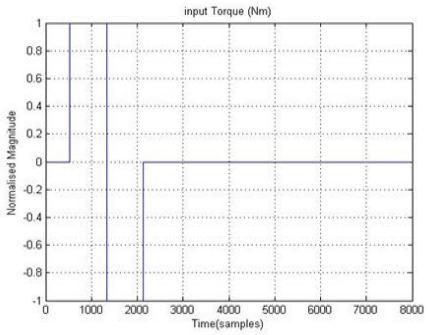
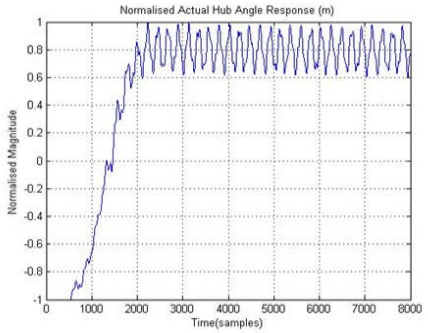
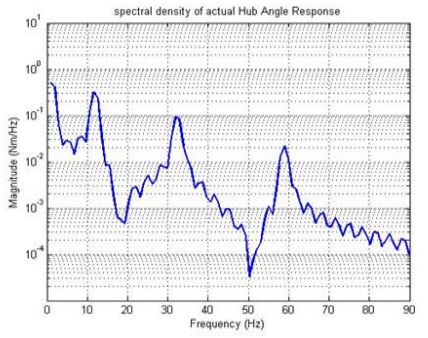


Fig.6: Bang-bang input torque

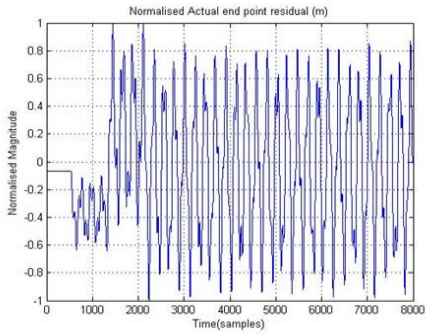


(a)

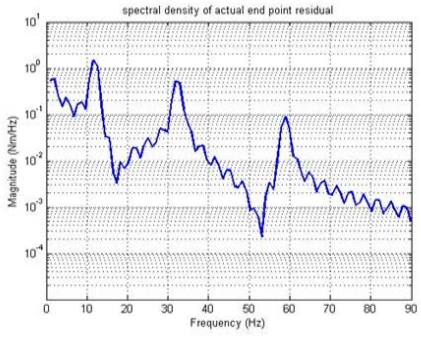


(b)

Fig.7: Hub angle response, (a) Time domain, (b) Frequency domain



(a)



(b)

Fig.8: End-point residual response, (a) Time domain, (b) Frequency domain

3.1 Modeling of Hub Angle Using DE

A Matlab program has been created based on DE algorithm and used to estimate the parameters of ARX model using torque input and hub angle output data obtained from the simulation study. Since there was no a priori knowledge regarding the suitable order of the model, the structure realization was performed by a trial-and-error method. Randomly selected parameters were optimized for different, arbitrarily chosen order to fit into the system by applying the working mechanism of DE based on One Step-Ahead prediction. Table 2 shows the work that carried out to select the best model order.

It can be noted from Table 2 that the best result was achieved with model order = 4 for 4000 data length. From the work carried out for the model order = 4, the satisfactory results were achieved with the following set of parameters:

Population Initial Range: [-3; 3]
Number of Generations: 2800
Population Size: 10

The data set, comprising 7986 data points was divided into two sets of 4000 and 3986 data points respectively. The first set was used to estimate the model parameters while the second set was used to validate the model. Both output and estimated output of the hub angle in time and frequency domains are plotted in Figs. 9 and 10 respectively. The error between actual and predicted DE output are plotted in Fig. 11. The division between the trained data and the unseen data is indicated as a vertical line located at point 4000 as shown in Figs. 9 and 11. The best mean square error of DE algorithm is 2.504×10^{-4}. Using the proposed identification procedure, the parameters of the model were estimated as follows:

a_1	-1.998052
a_2	0.998075
b_1	0.000156
b_2	-0.000172

Figs. 9, 10 and 10 show that the model was able to mimic the measured output well. The pole-zero diagram and correlation tests are depicted in Figs. 12 and 13 respectively. It is noticed from Fig. 12 that the model was stable and the poles of the transfer function were inside the unit circle and the zero was outside the unit circle indicating non-minimum phase behavior. The correlation functions were carried out for 1000 samples to determine the effectiveness of the DE-based model. The results were found to be within 95% confidence level thus confirming the accuracy of the results.

Table 2
Comparative assessment for hub angle best order

	MSE	Correlation test	time	No. of generation	stability
four	2.5×10^{-4}	unbiased	25.05	2800	stable
six	3.75×10^{-4}	unbiased	57.8	6000	stable
eight	5.02×10^{-4}	unbiased	133.4	12800	stable
ten	6.32×10^{-4}	unbiased	262.4	23000	unstable
twelve	7.61×10^{-4}	unbiased	323	36000	unstable
fourteen	8.83×10^{-4}	unbiased	424.8	42000	unstable
Sixteen	0.0010	unbiased	447.5	48000	unstable

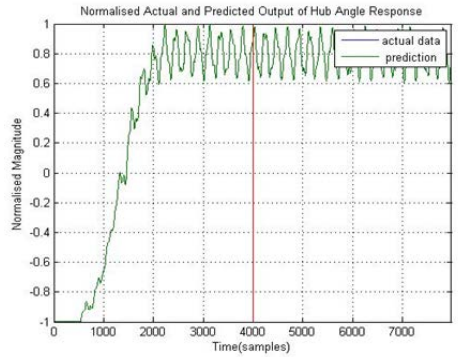


Fig.9: Actual and estimated output of the hub angle in time domain using DE

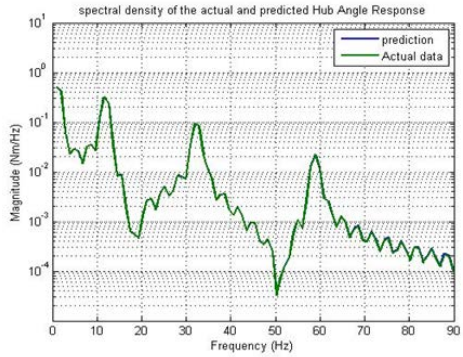


Fig.10: Actual and estimated output of the hub angle in frequency domain using DE

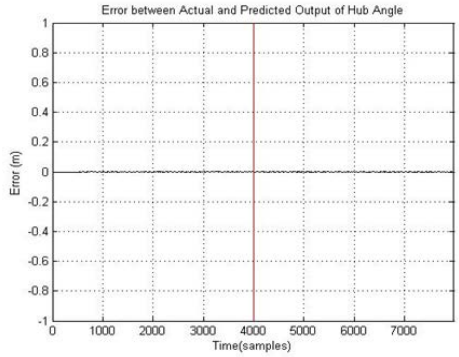


Fig.11: Error between actual and estimated output of hub angle using DE

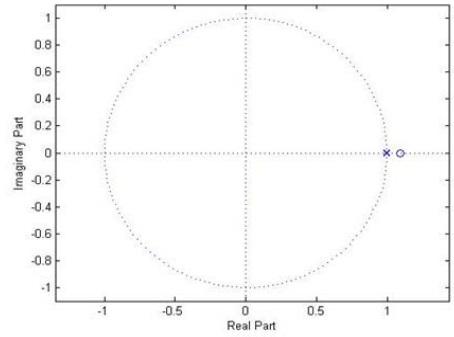
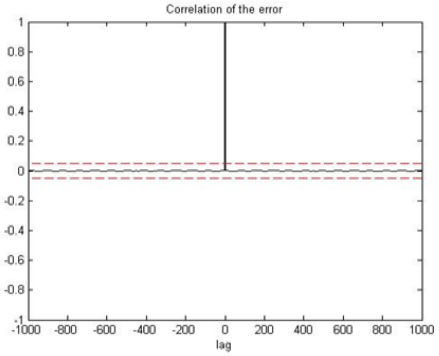
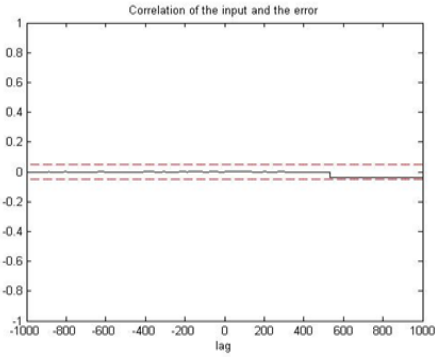


Fig.12: Pole-zero diagram of hub angle using DE



a) Auto corelation



b) Cross correlation

Fig.13: Correlation tests for hub angle using DE

3.2 Modeling End-point residual Using DE

A Matlab program has been created based on the DE algorithm, and used to estimate the parameters of ARX model based on the torque input and end-point residual output data obtained from the simulation study. Since there was no a priori knowledge about the suitable model order of the flexible manipulator system, the structure realization was performed using a heuristic method. Randomly selected parameters were optimized for different, arbitrarily chosen order to fit into the system by applying the working mechanism of DE based on OSA prediction. Table 3 shows the work carried out to select the best model order

It can be noted from Table 3 the best result was achieved with model order = 6, for 4000 data length. For the best model order, DE was designed with 10 individuals with maximum number of iterations set to 6000. The data set, comprising 7986 data points divided into two sets of 4000 and 3986 data points respectively. The first set was used to compute the model parameters whilst the second set was used to validate the model. In addition, the best mean square error of DE algorithm is 1.587×10^{-6}. Using the proposed identification procedure, the parameters of the model were estimated as follows:

a_1	-2.990312

a_2	2.9849733
a_3	−0.9946538
b_1	−0.000530
b_2	0.0000267
b_3	0.000502

Both output and estimated output of end-point residual in time and frequency domains are plotted in Figs. 14 and 15 respectively and the error between actual and predicted DE output are plotted in Fig. 16. The division between the trained data and the unseen data is indicated as a vertical line located at point 4000 as shown in Figs. 14 and 16. The pole-zero diagrams and correlation tests are depicted in Figs. 17 and 18 respectively. Fig. 17 indicated that the model is stable. The correlation test functions results confirm that the model is acceptable.

Table 3
Comparative assessment for end-point residual best order

	MSE	Correlation test	time	No. of generation	stability
four	1.8×10^{-6}	biased	15.5	4000	stable
six	2.3×10^{-6}	unbiased	55.9	6000	stable
eight	2.1×10^{-6}	unbiased	105.5	10400	stable
ten	6.4×10^{-6}	biased	330	32000	unstable
twelve	4.5×10^{-6}	biased	353.0	36000	unstable
fourteen	7.2×10^{-6}	biased	495.5	42000	unstable
Sixteen	6.9×10^{-6}	biased	523.3	48000	unstable

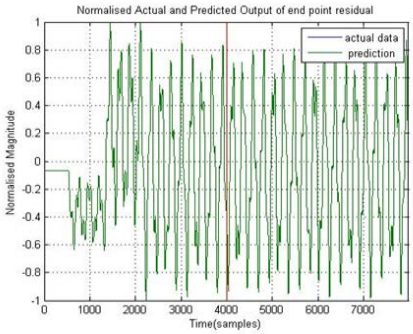


Fig.14: Actual and estimated output of end-point residual in time domain using DE

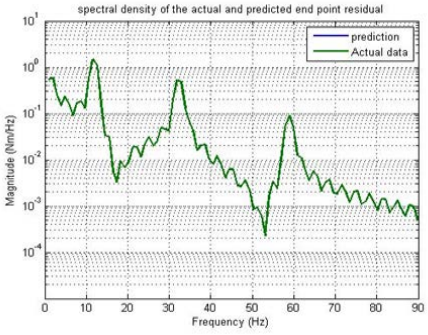


Fig.15: Actual and estimated output of end-point residual in frequency domain using DE

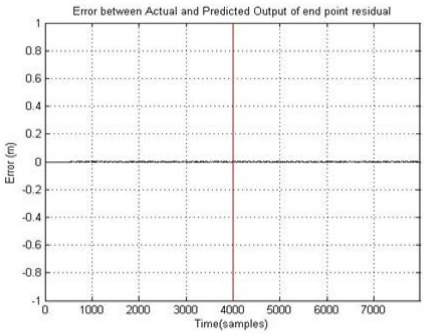


Fig.16: Error between actual and estimated output for end-point residual using DE

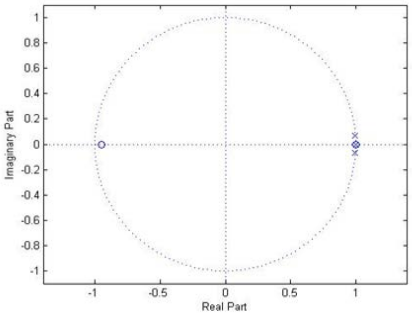
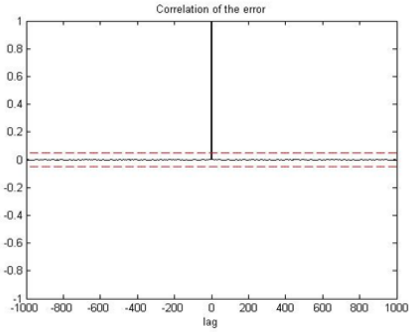
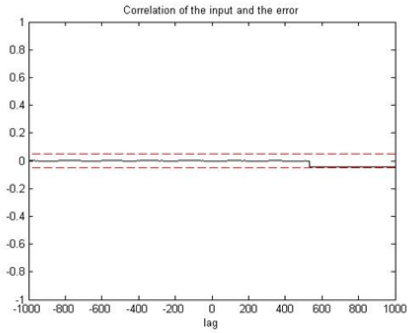


Fig.17: Pole-zero diagram of end-point residual using DE



a) Auto corelation



b) Cross corelation

Fig.18: Correlation tests of end-point residual using DE

3.3 Modeling Hub Angle Using PSO

Investigations were then carried out with PSO algorithm to model the hub angle using the same input-output data and same model order achieved with DE. PSO was designed with 50 individuals with maximum number of iterations set to 30000. The best mean square error of PSO is 2.519×10^{-4}. Using the proposed identification procedure, the parameters of the model were estimated as follows:

a_1	-1.643623
a_2	0.643645
b_1	0.003884
b_2	-0.003815

Both output and estimated output of end-point residual in time and frequency domains are plotted in Figs. 19 and 20 respectively and the error between actual and predicted PSO output are plotted in Fig. 21. The division between the trained data and the unseen data is indicated as a vertical line located at point 4000 as shown in Figs. 19 and 21. The pole-zero diagrams and correlation tests are depicted in Figs. 22 and 23 respectively. Fig. 22 indicating that the model is stable with non-minimum phase behavior. The correlation test functions results confirm that the model is acceptable.

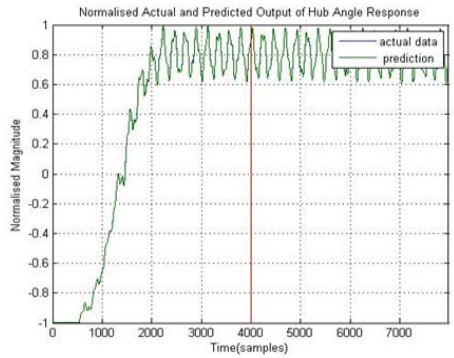


Fig.19: Actual and estimated output of the hub angle in time domain using PSO

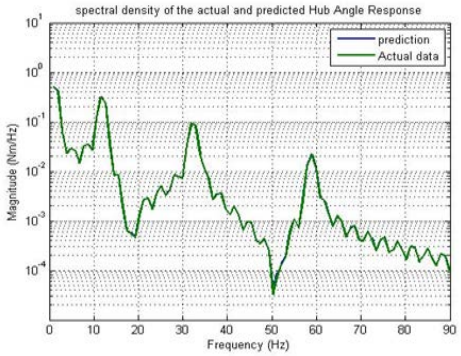


Fig.20: Actual and estimated output of the hub angle in frequency domain using PSO

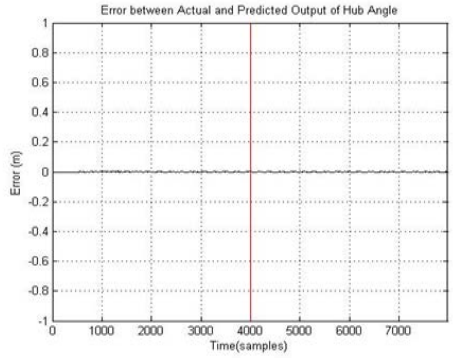


Fig.21: Error between actual and estimated output for hub angle using PSO

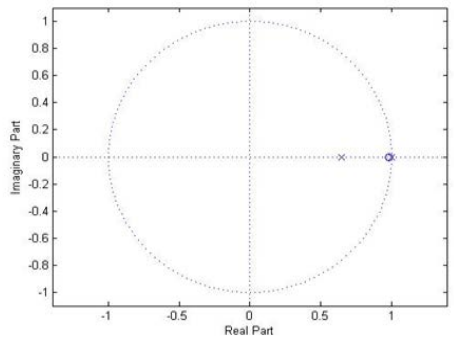
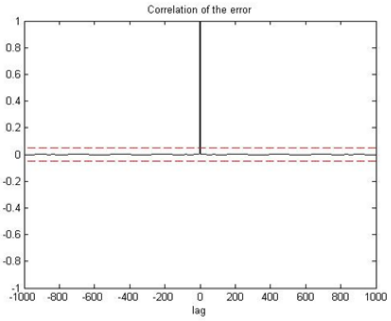
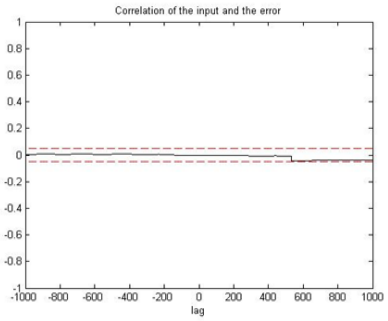


Fig.22: Pole-zero diagram for hub angle using PSO



a) Auto corelation



b) Cross corelation

Fig.23: Correlation tests for hub angle using PSO

3.4 Modeling End-point residual Using PSO

Investigations were then carried out with the PSO algorithm to model the end-point residual using the same input-output data and same model order achieved for end-point residual used in DE. PSO was designed with 150 individuals with maximum number of iterations set to 30000. The best mean square error of PSO algorithm is 1.014×10^{-4}. Using the proposed identification procedure, the parameters of the model were estimated as follows:

a_1	-0.448655
a_2	-2.375043
a_3	1.828526
b_1	-0.036951
b_2	0.341808
b_3	-0.305859

Both output and estimated output of end-point residual in time and frequency domains are plotted in Figs. 24 and 25 respectively and the error between actual and predicted PSO output are plotted in Fig. 26. The division between the trained data and the unseen data is indicated as a vertical line located at point 4000 as shown in Figs. 24 and 26. The pole-zero diagrams and correlation tests are depicted in Figs. 27 and 28 respectively. Fig. 27 indicated that the model is unstable. The correlation test functions results showed that the model is biased.

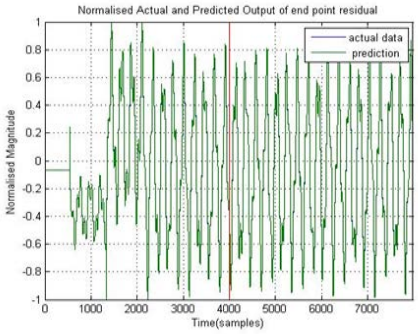


Fig.24: Actual and estimated output of end-point residual in time domain using PSO

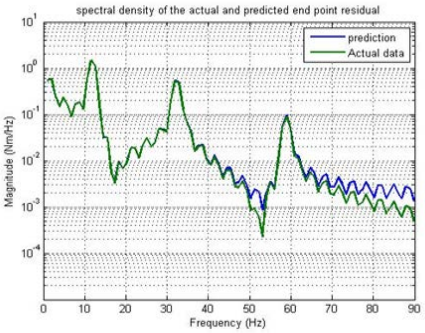


Fig.25: Actual and estimated output of end-point residual in frequency domain using PSO

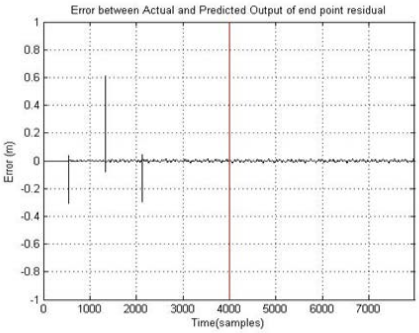


Fig.26: Error between actual and estimated output for end-point residual using PSO

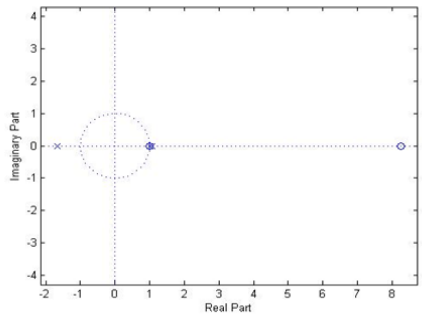
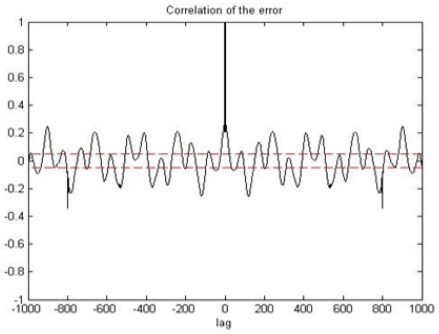
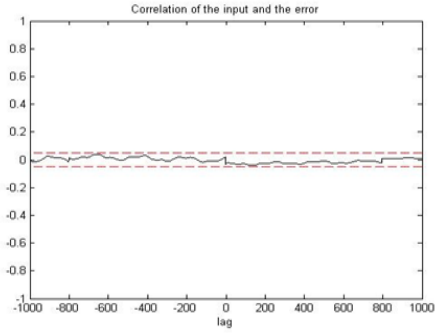


Fig.27: Pole-zero diagram for end-point residual using PSO



a) Auto corelation



b) Cross corelation

Fig.28: Correlation tests for end-point residual using PSO

5 Comparative Assessment

The overall comparative performance of DE and PSO modelling approaches in terms of mean squared error, stability and correlation tests is summarised in Table 4.

From Table 4, it can be seen that the stability and the correlation test results by DE are better than PSO especially for the end-point residual modelling. However, a major advantage of the PSO is that the algorithm is simple. Table 4 also shows that the performance of DE in terms of mean-squared error is better than the PSO with the same model structure. Therefore, high performance computing power could provide better solutions in the real time implementation of the DE based identification.

6 Conclusion

In this work, DE and PSO have been adopted for modelling a single-link flexible manipulator system. Input-output data pairs have been collected from a simulation study and used in developing linear models of the system from input torque to hub-angle and end-point acceleration. The performances of DE-ARX and PSO-ARX models have been assessed through many validation tests. It has been demonstrated that the DE and PSO modelling technique has performed well in approximating the system response and DE is better than PSO in modelling a single-link flexible manipulator system.

This simulation platform forms the basis to implement different control structures to control a single-link flexible manipulator system. Moreover, this study can be extended to use system identification techniques for modeling a single-link flexible manipulator system using a real plant to implement different control structures before online control can be implemented.

Acknowledgment

The authors would like to express their gratitude to Minister of Education Malaysia (MOE) and Universiti Teknologi Malaysia (UTM) for funding and providing facilities to conduct this research. This research is supported using FRGS Vote No. 4F395 and UTM Research University grant, Vote No. 05H71.

Table 4
Overall comparative assessment

	Modelling domain	MSE	Stability	Correlation test
End-point residual	DE	1.58×10^{-6}	stable	unbiased
	PSO	1.01×10^{-4}	unstable	biased
hub angle	DE	2.50×10^{-4}	stable	unbiased
	PSO	2.52×10^{-4}.	stable	unbiased

References

[1] S. K. Dwivedy and P. Eberhard, Dynamic analysis of a flexible manipulator, a literature review, *Mech. Mach. Theory., Vol. 41, pp. 749-777, 2006.*

[2] W.J. Book, Modelling, design and control of flexible link manipulator arms: a tutorial review, *Proceedings of IEEE conference on decision and control, pp. 500-506, 1990.*

[3] H. Supriyono, Novel Bacterial Foraging Optimisation Algorithms with Application to

Modelling and Control of Flexible Manipulator Systems, *PhD thesis, Dept. Automatic Control and Systems Eng, Sheffield Univ, 2012*

[4] M.H. Shaheed, and M.O Tokhi, Dynamic modelling of a single-link flexible manipulator: parametric and non-parametric approaches. *Robotica, 20, pp. 93 –109, 2002*

[5] L. Ljung, *System identification: theory for the user*, 2nd edition. Prentice hall. 1999

[6] I. Z. Mat Darus and A. A. M. Al-Khafaji,. Nonparametric Modelling of a Rectangular Flexible Plate Structure, *International Journal of Engineering Application of Artificial Intelligence, 25, pp. 94-106. 2012*

[7] M. S. Alam, M. O. Tokhi and I. A. Latiff, , Dynamic modelling of a single-link flexible manipulator using particle swarm optimisation. *The Second International Conference on Control, Instrumentation and Mechatronic Engineering (CIM09), Malacca, Malaysia, June 2 -3, 2009a.*

[8] M.H. Shaheed and M.O. Tokhi, Dynamic modelling of a single-link flexible manipulator: parametric and non-parametric approaches. *Robotica, 20, pp. 93 –109,2002*

[9] M. H. Shaheed, M. O. Tokhi, A. J. Chipperfield, and A. K. M Azad, Modelling and open-loop control of a single-link flexible manipulator with genetic algorithms, *Journal of Low Frequency Noise, Vibration and Active Control, 20(1), pp. 39 – 55.*

[10] M. S. Alam, and M.O Tokhi, Dynamic modelling of a single-link flexible manipulator system: a particle swarm optimisation approach. *Journal of Low Frequency Noise, Vibration and Active Control, 26 (1), pp. 57 – 72. 2007*

[11] B.A. Md Zain, M.O. Tokhi, and S. Md Salleh, Dynamic modelling of a single-link flexible manipulator using parametric techniques with genetic algorithms. *Proceedings of Third UKSim European Symposium on Computer Modelling and Simulation, Athens, Greece, November 25 – 27, 2009, pp. 373 – 378. 2009*

[12] H.A. Talebi, R.V. Patel and H.Asmer, Dynamic modelling of flexible-link manipulators using neural networks with application to the SSRMS. Proceedings of the 1998 IEEE/RSJ International Conference on Intelligent Robots and Systems, Victoria, B.C., Canada October 1998, pp. 673 – 678 (1998).

[13] H. Supriyono, M.O. Tokhi and B.A. Md Zain, Modelling of flexible manipulator systems using bacterial foraging algorithms, Proceedings of CLAWAR 2011: The 14th International Conference on Climbing and Walking Robots and the Support Technologies for Mobile Machines (CLAWAR2011), Paris, France, 6-8 September 2011, pp. 157-165. 2011

[14] M. O. Tokhi, A. K. M. Azad, Flexible Robot Manipulators Modelling, simulation and control, the Institution of Engineering and Technology, London, United Kingdom, 2008.

[15] M. O. Tokhi, Z. Mohamed, H. M. Shaheed, Dynamic characterisation of a flexible manipulator system. Robotica / Volume 19 / Issue 05 / September 2001, pp 571 - 580.

[16] M.O. Tokhi, Z. Mohamed, A. K. M. Azad, Finite difference and finite element approaches to dynamic modelling of a flexible manipulator, Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering 1997, 211: 145.

[17] R. Ismail, A. Y. Ismail, I. Z. Mat Darus, Identification Algorithms of Flexible Structure Using Neural Networks, Proceeding of the IEEE 4th Student Conference on Research and Development (SCOReD2006), Shah Alam, Selangor, MALAYSIA, 27-28 June, 2006, pp 162 – 168.

[18] D. Koulocheris, V. Dertimanis, and H. Vrazopoulos, Evolutionary parametric identification of dynamic systems, Forschung im Ingenieurwesen , Springer Verlag, 68, pp. 173-181.

[19] A. R. Tavakolpour, I. Z. Mat Darus, M. Mailah, and M.O. Tokhi, Genetic Algorithm-based Identification of a Rectangular Flexible Plate System for Active Vibration Control, International Journal of Engineering Application of Artificial Intelligence, 23, pp; 1388 - 1397. Science Direct, Elsevier Lt, 2010.

[20] Z. Michalewicz, and M. Shoenauer, Evolutionary Algorithms for Constrained Parameter Optimization Problems. Evolutionary Computation 4(1):1-32, 1996.

[21] L. Ching-Hung, K. Che-Ting and C. Hao-Han, Performance enhancement of the differential evolution algorithm using local search and a self-adaptive scaling factor, International Journal of Innovative, Computing, Information and Control, Volume 8, Number 4, April 2012

[22] M. Settles, An Introduction to Particle Swarm Optimization. Department of Computer Science, University of Idaho,2005