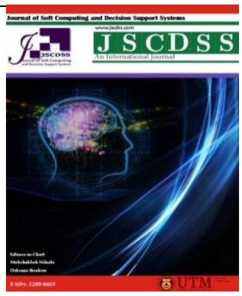
An International Journal
JSCDSS
http://www.jscdss.com
Vol.3 No.1 February 2016: 44-52
Article history:
Accepted 19 November 2015
Published online 19 November 2015

Journal of Soft Computing and Decision Support Systems



E-ISSN: 2289-8603

System Identification and Intelligent Control of Flexible Manipulator System

Mohsen Gol Zardian [a,*], Intan Z. Mat Darus [a], Mohammad Reza Miveh [b]
[a] Department of Applied Mechanics & Design, Faculty of mechanical engineering, Universiti Teknologi Malaysia, Johor, Malaysia
[b] Young Researchers Club, Saveh Branch, Islamic Azad University, Saveh, Iran

*** Corresponding author email address**: mogolzardian@gmail.com

Abstract

Position control of flexible manipulator system is normally accompanied with tip vibration that results in degradation of performance. This paper investigates an active control strategy by applying classical PID controller to suppress unwanted vibration of flexible manipulator in presence of disturbances. The parameters of PID controller are tuned by genetic algorithm (GA) and particle swarm optimization (PSO) in the intelligent (self-tuning) manner. The results of these two optimization methods are compared toward vibration control capability, moreover; modeling of flexible manipulator is conducted by applying system identification method in which autoregressive with exogenous input (ARX) model is intended as linear model. This research can be regarded as guidance for further elaborate research on implementing optimization method particularly integrated with PID controller for flexible manipulator system modeled by system identification approach.

Keywords: Flexible manipulator, System identification, Genetic algorithm (GA), Particle swarm optimization (PSO), PID controller

1. Introduction

Manipulator systems have been playing essential role in today's technology. They are attached with other automatic appendage to carry out accurate and heavy operation such as automotive industry, space robotic system (Zarafshan et al., 2013) biomedical engineering (Sekiguchi et al., 2001), even in the electronic application (Brouwer et al., 2010). Owing to drastic characteristic such as light weight, low energy consumption and satisfactory speed, flexible manipulator becomes eligible candidate than its rigid counterpart.

However, flexibility in terms of low stiffness has some undesirable effects namely, chaotic motion that takes system into instability state and time delay. Thus designing an active controller in order to cope with potential drawbacks and suppress such destructive vibration has been gaining importance in recent years. Basically, Control of flexible manipulator dealing with trajectory tracking and vibration control simultaneously which in the most of the cases two distinct control strategies are implemented to keep desired hub angle and cancel sever tip vibration. A challenging aspect of flexible manipulator control resemble to other control systems is to formulate an appropriate model of flexible manipulator before implementing any control scheme.

In the literature, several theories have been proposed to present a distinguished ways for modeling and suppressing the vibration of flexible manipulator. Qiu et al. (2012) proposed dynamic model of flexible manipulator using assumed mode method (AMM) and applying composite PD and direct adaptive fuzzy controller for control system. Developing input shaping controller in to cancel the vibration of single flexible link, in which modeling of system is done by finite element method (Zain and Tokhi, 2006). Cancelling the vibration in presence of nonlinearities such as motor friction (coulomb friction) is carried out by model predictive controller (MPC) (Abdolvand and Fatehi, 2012). Another solution described by Shawky et al. (2013) in which control of tip vibration and tracking control of flexible manipulator is performed by using the nonlinear state dependent Riccati Equation (SDRE) method, moreover; assumed mode method is intended in modeling part. In the recent years of research on flexible manipulator besides developing control strategy, modeling with the system identification draws the attraction of researcher considerably. Ramos and Feliu, (2008) investigated the online payload identification of (MIMO) single-link flexible robot base on captured signals from motor position and the coupling torque. The conventional PD controller is employed to control the tip vibration as well. As reported by Ziaei et al. (2009) the generalized orthonormal basis function (GOBF) as system identification approach for single flexible link is adapted for flexible link in contact with compliant and rigid environment. The robust force control as quantitative feedback theory is presented to control of link. By considering the payload change as uncertainty effect the natural frequency alters respectively so, fast online system identification is carried out to identify the changeable natural frequency in the frequency response function of model (Becedas et al., 2007). Recently, Mute et al. (2013) proposed the nonlinear autoregressive with exogenous input (NARX) model for single flexible link manipulator, and then linear PID controller is design to control the link as well.

Even though the efficiency of modeling and control of flexible manipulator system have been improved in recent years, most improvements have been achieved by presenting model with system identification method and developing optimal and adaptive controller. Based on approach indicated by Ahmed et al. (2010) the vibration control and trajectory control of flexible link is implemented by two separate control systems in which input shaping control is design to suppress the natural frequencies in the feedforward path and trajectory control is performed in the feedback path by PD type fuzzy controller. However; input shaping technique could not guarantee the vibration control performance in presence of any uncertainty, thus in order to cancel vibration the adaptive control would be one of the promising technique to deal with uncertainties. The purpose of this study is to propose modeling of flexible manipulator by using input and output data in the system identification approaches for modeling part and introducing the intelligent PID controller by performing intelligent optimization technique namely, genetic algorithm (GA) method and particle swarm optimization (PSO) to satisfy design specification such as overshoot and rise time.

The reminder of this paper is organized into 4 sections. In Section 2, modeling of flexible manipulator is discussed. Section 3 devoted to the optimization technique in which GA and PSO are well explained. Section 4 shows the implementing of GA optimization and PSO technique on PID controller and this work is concluded in Section 5.

2. Modeling of Flexible Manipulator

Firstly, we started by introducing of flexible manipulator modelling definition in the classical manner and then novel modelling approach is explained by system identification method in which input and output data (signal) are used to create linear time invariant system. There are several choices in model selection but in this work linear model is presented.

2.1 *Flexible manipulator system*

Fig. 1 illustrates single flexible link that is manoeuvred by imposed torque at the one end, the governing equation of system is given by:

$$w(x,t) = x\theta(t) + \vartheta(x,t), \tag{1}$$

$$\begin{cases} EI\dfrac{\partial^4 w(x,t)}{\partial x^4} + \rho\dfrac{\partial^2 w(x,t)}{\partial t^2} = \tau(t) \\ w(0,t) = 0 \\ I_h\dfrac{\partial^3 w(0,t)}{\partial x \partial t^2} - EI\dfrac{\partial^2 w(0,t)}{\partial x^2} = \tau(t) \\ M_p\dfrac{\partial^2 w(L,t)}{\partial t^2} - EI\dfrac{\partial^3 w(L,t)}{\partial x^3} = 0 \\ EI\dfrac{\partial^2 w(L,t)}{\partial x^2} = 0, \;\; w(x,0) = 0, \;\; \dfrac{\partial^2 w(L,t)}{\partial x^2} = 0 \end{cases} \tag{2}$$

Where the $w(x,t)$ is deflection of the flexible link and $\tau(t)$ is the exerted torque.

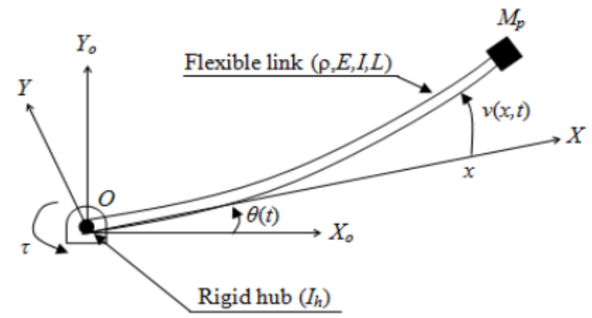


Fig. 1. Typical flexible manipulator diagram

Instead of solving the PDE equation researchers prefer use two well-known classical methods namely, finite element (FE) or assumed mode method (AMM) technique (Tokhi et al., 2001; Vakil et al., 2009). However, spillover effect due to not considering high vibration frequencies is unavoidable while model is estimated by these two methods. System identification method is another alternative method that is emerging during recent decade in which parameters of proposed model is estimated by using input and output signals (nille, 2001). As matter of fact, Uncertainties such as changeable mass and nonlinearities such as coulomb friction of DC motor are challenging part of flexible manipulator system modeling, however; the effect of coulomb friction and changeable mass is neglected in this study.

2.2 *System identification*

System identification due to considerable flexibility in implementation by applying real system input and output data has been gaining popularity. Input and output signals

An International Journal JSCDSS E-ISSN: 2289-8603

also can be obtained via finite element (FE) or finite difference (FD) simulation when accessibility to the real test rig is limited (Yatim and Darus, 2012; Tavakolpour et al., 2011; Gol Zardian and Ayob, 2015) .In this study autoregressive with exogenous input (ARX) model is proposed by Eq. (3) as linear model that correlates past input and past output data to find current output (response), the model structure is depicted in Fig. 2.

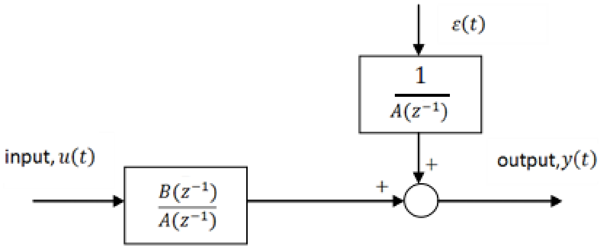


Fig. 2.. ARX model structure

$$(t) = \frac{B(z^{-1})}{A(z^{-1})}u(t) + \frac{1}{A(z^{-1})}\varepsilon(t) \tag{3}$$

Where, $A(z^{-1})$ and $B(z^{-1})$ are polynomials and u(t)is the input data and y(t) is the output data while $\epsilon(t)$ is the white noise. By neglecting the noise effect and rewriting the Eq. (3) we have:

$$y(t)A(z^{-1}) \ = B(z^{-1})u(t) \tag{4}$$

Where

$$A(z^{-1}) = 1 + a_1 z^{-1} + a_2 z^{-2} + \ \dots + \cdots + a_n z^{-n} \tag{5}$$

$$B(z^{-1}) = b_0 + b_1 z^{-1} + b_2 z^{-2} + \cdots + b_m z^{-m} \tag{6}$$

By substituting the Eq. (5) and Eq. (6) into Eq. (4) the model in Eq. (3) can be shown as:

$$y(t) = -\sum_{i=0}^{n} a_1 z^{-1} + a_2 z^{-2} + \cdots + a_n z^{-n} + \cdots \sum_{i=0}^{n} b_1 z^{-1} + b_2 z^{-2} + \cdots + b_m z^{-m} \quad a_0 = 1, b_0 = 0 \tag{7}$$

2.2.1 Parameters estimation

After determining the ARX model next step in the system identification method is to define the un- known parameters, here the parameters are $[a_1, , \dots, a_n, b_1, \dots, b_m]$. In order to define the parameters recursive least square (RLS) is employed as statistical method by rewriting the Eq. (7) in form of:

$$y(t) = \varphi(t)\beta \tag{8}$$

Where $y(t)$ is actual output, $\varphi(t)$ is the regressor that includes the input and output data and β is the system parameters, in the other word:

$$\varphi(t) = [-y(t+1) \dots - y(t-n) \ \ u(t-1) \dots u(t-m)] \tag{9}$$

And

$$\beta = \begin{bmatrix} a_1 \\ \vdots \\ a_n \\ b_1 \\ \vdots \\ b_m \end{bmatrix}, \tag{10}$$

The main goal of any parameters estimation method is to minimize the cost function as:

$$J = \frac{1}{N}\sum_{K=1}^{N}(y(k) - \hat{y}(k))^2 \tag{11}$$

Where $y(k)$ is actual output and $\hat{y}(k)$ is estimated output. By considering the cost function and manipulating the Eq. (8) the estimated parameters are driven as:

$$\hat{\beta}\ (t+1) = \hat{\beta}(t) + p(t+1)\varphi(t+1)\epsilon(t+1) \tag{12}$$

Where $p(t+1)$arbitrary parameter that is defined as:

$$p(t+1) = p(t)\left[I - \frac{\varphi(t+1)\varphi^T(t+1)p(t)}{\gamma + \varphi^T(t+1)p(t)\ddot{o}(t+1)}\right] \tag{13}$$

And $\epsilon(t+1)$ is error in form of:

$$\epsilon(t+1) = y(t+1) - \varphi^T(t+1)\hat{\beta}(t) \tag{14}$$

As last the estimate output is found by:

An International Journal JSCDSS
E-ISSN: 2289-8603

$$\hat{y}(t) = \varphi(t)\hat{\beta} \quad (15)$$

As a best estimation, Eq. (15) should satisfy the cost function in Eq. (11).The structure of modeling loop with least square method is depicted in Fig. 3.

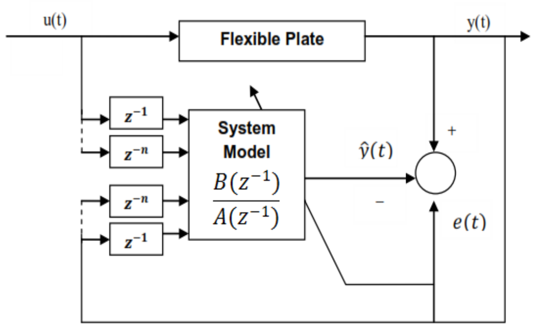


Fig. 3. Modeling loop structure

2.3 model estimation by using input and output data

In our research authors use the simulated input and output data of flexible manipulator by FD method that are developed in by Yatim and Darus (2012) in which input data is considered as bang-bang torque u(t) and output is signified as tip displacement y(t). Input and output signal are discrete signals that sampled with period of 0.02 ms into 13764 points that normalized between -1 and1 as well. Fig. 4 and Fig. 5 illustrate both signals.

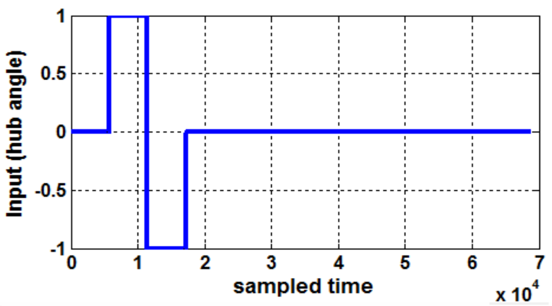


Fig 4. Bang-bang torque (input)

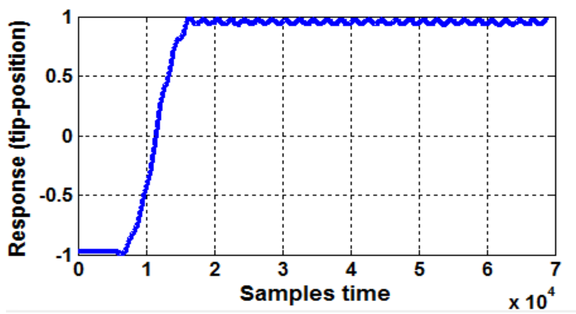


Fig. 5. Tip position (output)

Let to implement both input and output signal into ARX model and caste the linear time invariant system. The system identification method developed with two main parts in which estimation process uses half of data and remaining data are used for model validation. In our study 68843 data are implemented in estimation part and rests are used for validation part. After performing the system identification according to the formulation the model order 2 is found as best model order in terms of producing minimum mean square error (cost function) namely, 2.8546×10^{-8} , see Table 1.

Table 1
Mean square error versus model order.

Model order	Mean square error
2	2.8546×10^{-8}
4	3.8546×10^{-8}
6	4.0289×10^{-7}
8	6.1127×10^{-7}

The discrete transfer function of flexible manipulator can be presented by:

$$G(z) = \frac{y(z)}{u(z)} = \cdots$$
$$= \frac{-2.0733\times 10^{-6}z^{-1} - 2.091\times 10^{-6}z^{-2}}{1 - 0.5z^{-1} - 0.5z^{-2}} \quad (16)$$

In Eq. (16) u(z) is input signal, y(z) is output signal, $b_1 = -2.0733\times 10^{-6}$, $b_2 = -2.091\times 10^{-6}$, $a_1 = -0.5$ and $a_2 = -0.5$. The graphical illustration of model estimation by using RLS method against real data is shown in Fig. 6, on the other hand, absolute error between actual and estimated output is depicted in Fig. 7.

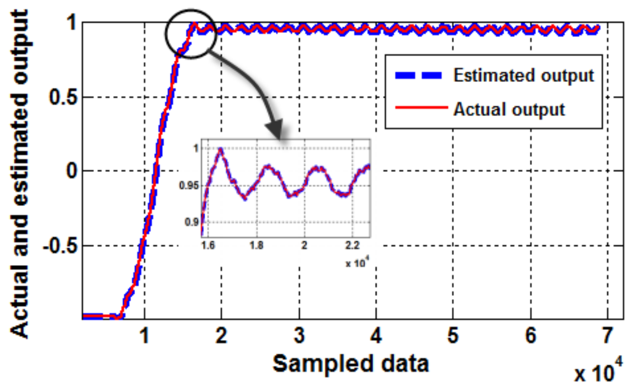


Fig. 6. Estimated output (blue dash line) and actual output (red line)

An International Journal JSCDSS E-ISSN: 2289-8603

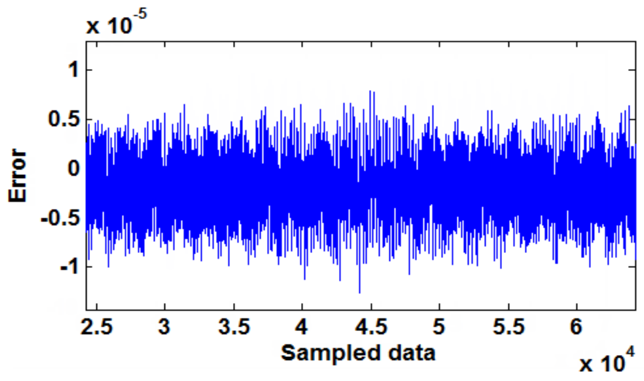


Fig. 7. Absolute error between estimated and actual output

In order to validate proposed ARX model there are some alternative methods. In this study frequency response function (FRF) is carried out to prove the model accuracy by showing all produced frequencies excited by bang-bang torque, see Fig. 8.

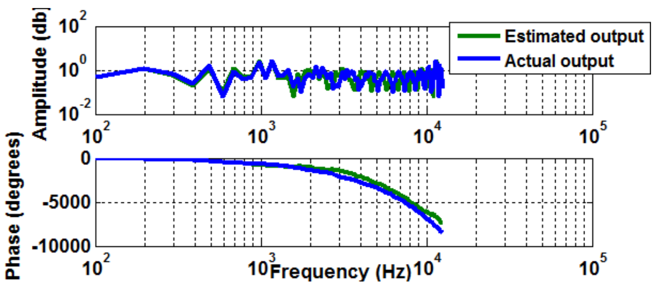


Fig. 8. FRF graph, excited frequencies belong to actual signal (blue line) and excited frequencies belong to estimated signal (green line)

3. Optimization Technique

In this section the concept of two well-known optimization methods namely genetic algorithm (GA) and particle swarm optimization (PSO) are discussed in detail, later an intelligent controller would be designed base on this optimization technique.

3.1 Genetic algorithm (GA)

Genetic algorithm is versatile optimization method in which best solution is found by selecting an each individual of big population results in satisfying objective function either by minimizing (cost function) or maximizing (fitness function) the special value. Artificial GA mimics the evolution of living being in the nature encompassing selection, cross over, mutation as three main activities. Resembling to nature in the artificial GA chromosome possess the ability to transfer information over the whole process or evolution.

3.1.1 Selection

Selection is the repetitive action of genetic algorithm method that forms the initial population and also adapts the new superior population. In our research initial population is made randomly by choosing the real number, however; if we proceed to select the new population roulette wheel method is used to select chromosomes. In this method each chromosome possess own probability (P_i) that defines as:

$$P_i = \frac{c_{max}-c_i}{\sum_{j=1}^{npop}(c_{max}-c_j)} \tag{17}$$

Where, $npop$ is number of chromosome in population, c_{max} is maximum cost function, c_i is current chromosome cost function and c_j is cost function of remaining chromosomes. After characterizing the eligibility of each chromosome the roulette wheel selection tries to select members according to the high probability criterion. Finally by making cumulative vector as sum of probabilities as:

$$v = [p_1, p_{1+}\, p_2, p_{1+}\, p_{2+}\, p_3, ..., \; p_{1+\cdots+}\, p_n] \tag{18}$$

Or in the new form:

$$v(i) = \sum_{j=1}^{i} p_j \tag{19}$$

Where n is number of chromosome. Then by searching for indices belong to vector v an appropriate element can be found:

$$i = min\{j | r \le v_j\} \qquad 0 < r < 1 \tag{20}$$

Where i is an indice that in the each iteration helps to call chromosome of old population to build the new eligible population. As matter of fact, selecting is fundamental task that should be done before and after crossover and mutation activities.

3.1.2 Crossover phenomenon

The main purpose of crossover is to produce a new chromosome (population member) with sharing the partial characteristic of parents, in other word; manifestation of two new offsprings is result of mixing two mature chromosomes as parents with together. Graphically Fig. 9 shows this definition.

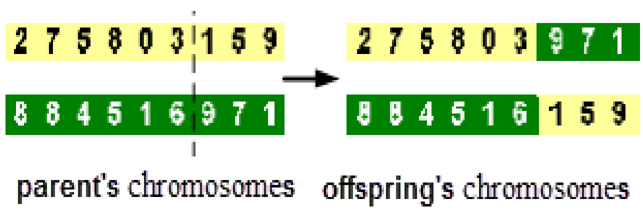


Fig. 9. Crossover

An International Journal JSCDSS E-ISSN: 2289-8603

Crossover can be done in single, double points and uniform crossover. In this study uniform crossover is conducted for real numbers. Mathematically two parents as $x1$ and $x2$ do a crossover to make two offsprings as $y1$ and $y2$ in form of:

$$y1 = \propto * x1 + (1-\propto) * x2;$$
$$y2 = \propto * x2 + (1-\propto) * x1; \quad ,0 < \propto < 1 \qquad (21)$$

3.1.3 Mutation

Occurring mutation phenomenon in nature is to create new population member with best adaptively towards the surrounding environment or nature. However; in the artificial environment mutation happens to satisfy the objective function without doing any crossover activity, in other word; one parent mutated into superior version. In our artificial programming the mutation is implemented to change one part of the chromosome y in form of:

$$y(i) = x(i) + \gamma * \beta \qquad (22)$$

Where γ is weighted value that is produced by:

$$\gamma = 0.1 * (varmax - varmin) \qquad (23)$$

In Eq. (23), $varmax$ and $varmin$ are the maximum and minimum value of variable that each chromosomes can possess respectively. And β is nominal value that selected randomly with respect to the size (length) of each chromosome. Finally, in Eq. (22) the i th element of chromosome is placed by new value.

3.2 Particle Swarm Optimization (PSO)

PSO mimics the social movement of some species such as fishes and birds in the nature in which populations follow intelligent manner. The artificial version of such natural movement can be inspired for optimization purposes. In contrast to GA the optimization process in PSO is done by directly intelligent interactive among the population members. In other word, in the each iteration of program current cost function of members (particles) evaluated and compared with its previous (local) and global value in order to make new eligible population. In this method position and speed of each particle in the each iteration evaluated for finding new particle position that minimize cost function.

3.2.1 PSO formulation

The new position of particle in population is express as:

$$x_i^{new} = x_i^{old} + v_i^{new} \qquad (24)$$

$$v_i^{new} = c_1 r_1 (x_i^{local} - x_i^{old}) + c_2 r_2 (x^{global} \ldots - x_i^{old}) + w v_i^{old} \qquad (25)$$

Where v_i is the velocity of particle also known as direction of particle, $c_1 + c_2 = 4$, w is the inertial coefficient and r_1 are the random number between 0 and 1. The graphical presentation of particle position in 2D is illustrated in Fig. 10.

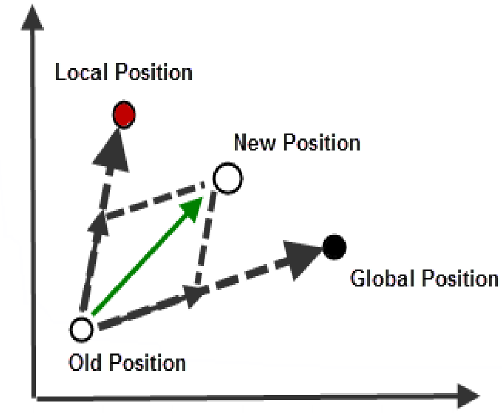


Fig. 10. Position of one particle

4. PID Controller

The PID controller as main control scheme is indented to regulate the proposed plant (discrete system) with respect to stability and some other design specification such as rise time, overshoot and settling time. The conventional discrete PID controller is expressed as:

$$c(z) = K_p + K_i \frac{T_s}{Z-1} + K_d \frac{N}{1 + N(\frac{T_s}{Z-1})} \qquad (24)$$

Where T_s is sampling time, N is filter coefficient the location of poles in the derivative filter, K_p, K_i and K_d are the proportional, integral and derivative coefficient of PID controller respectively.

4.1 PID tuning by GA

The chromosome C should satisfy the cost function in form of:

$$J = mse(error) \qquad (25)$$

Where mse is the mean square error of closed loop, see Fig. 11. In the each iteration of GA program cost function J is evaluated by new chromosome C. the GA simulation is terminated after 100 iterations. The best value of cost function is recorded 0.015 at the iteration 0.015, see Fig.12.

An International Journal JSCDSS E-ISSN: 2289-8603

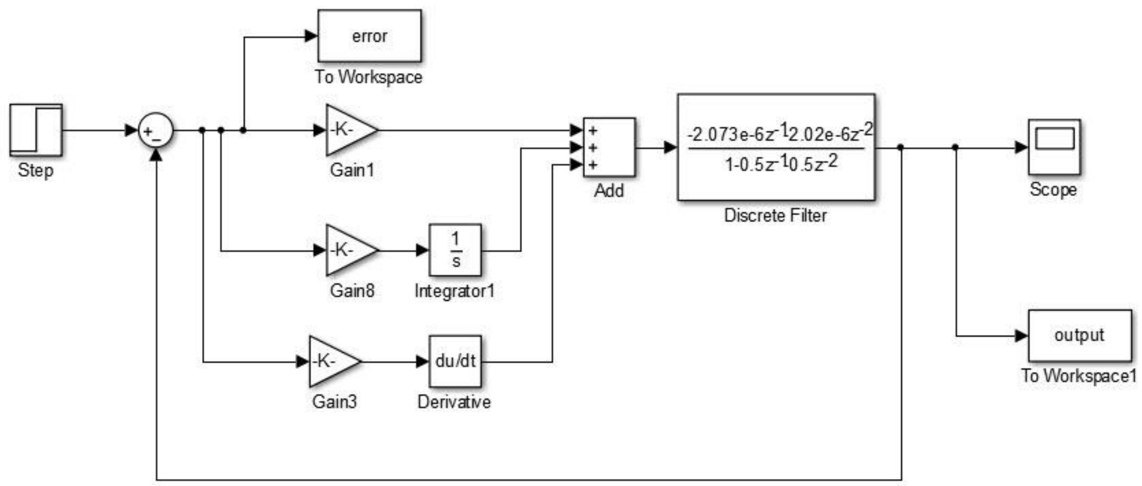
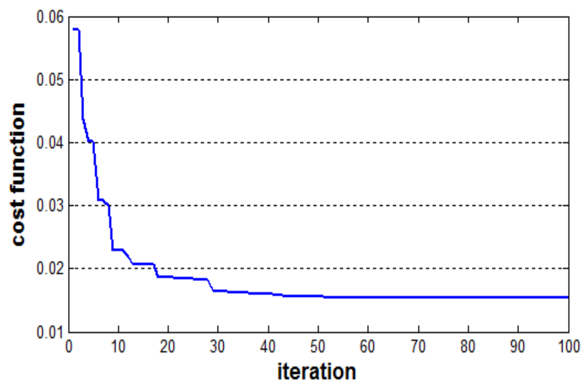


Fig. 11. Closed loop control system



The step response of the closed loop control system after 100 repetitions is achieved, see Fig.13. Moreover; in order to ensure the robustness of the closed loop system the repeating sequence signal is exerted as disturbance after controller, Fig. 14 manifests the disturbance signal. At the end of the simulation as it is shown in Fig. 13 one response signal with rise time 2.5 second and overshoot 13.6 % and another response signal with rise time 3.5 second and overshoot 1% are plotted with violet and green color respectively, however; according to the design specification compromising between these two response should be taken into account.

Fig. 12. Cost function versus iteration

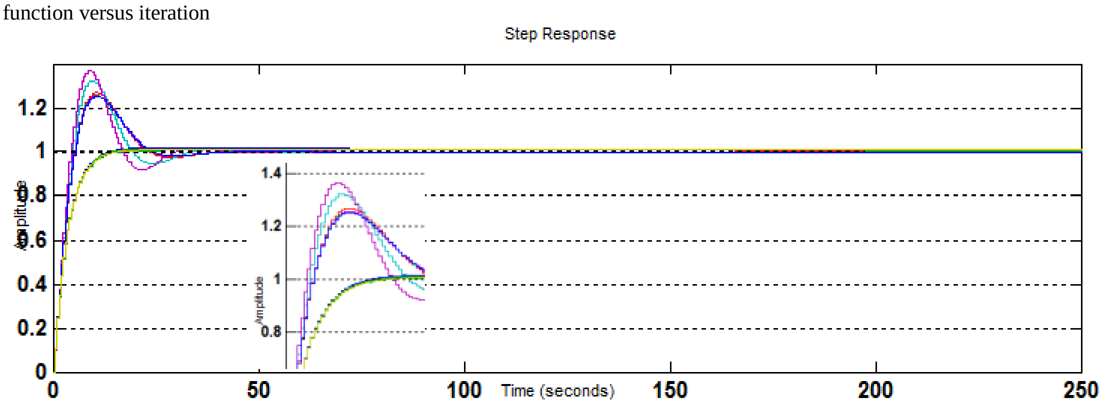


Fig. 13. Step response of closed loop control system tuned by GA, overshoot with 13.6% (violet line), overshoot with 1% (green line)

An International Journal JSCDSS
E-ISSN: 2289-8603

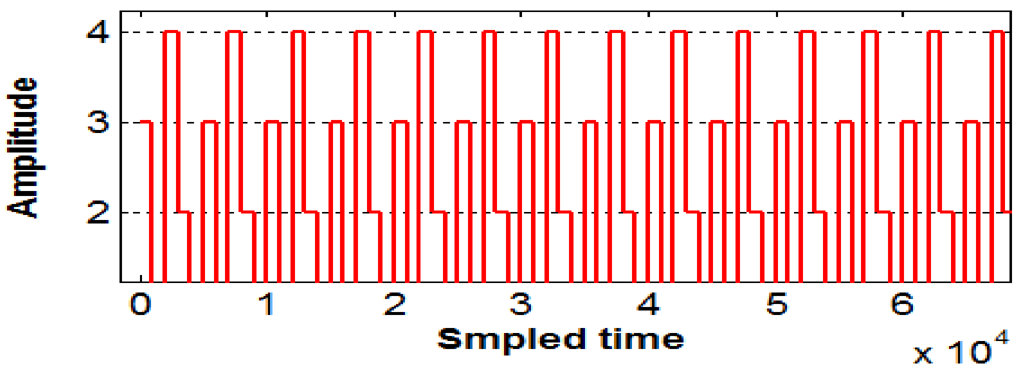


Fig. 14. Repeating sequence disturbance signal over operating domain

4.1 PID tuning by PSO

By observing the same scenario with GA three coefficients of PID controller is defined as one particle. Each particle attempts to minimize the cost function (mean square error of closed loop) by locating in the best position in population. At the end of simulation three eligible particles that satisfy the cost function (mean square error) are selected as PID controller coefficients, however; for getting an enough understanding of PSO tuning process all 100 response in the each iteration is depicted in Fig. 15. Moreover; cost function versus iteration is shown in Fig. 12. As it shown in Fig. 16 the PSO tries to improve the rise time of step response with negligible changes in overshoot percentage. The 2.5 second is reported as smallest rise time with 12% overshoot for the last iteration.

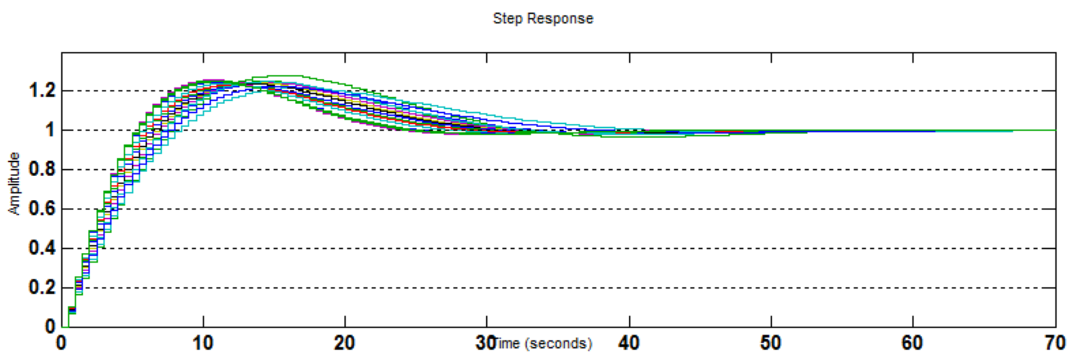


Fig.15. Step response of closed loop control system tuned by PSO

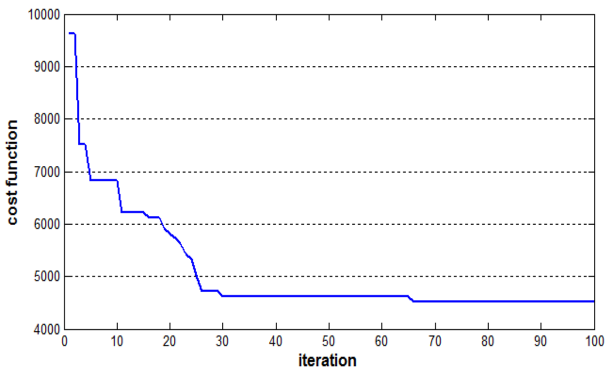


Fig. 16. Cost function versus iteration

5. Discussion and Conclusion

Summing up the results, it can be concluded that the dynamic model of single flexible manipulator is achieved by applying straightforward modeling method namely, system identification method in which autoregressive with exogenous input (ARX) model is adapted as model structure. Then the intelligent control system is design with incorporation of conventional PID controller and optimization technique such as genetic algorithm (GA) and particle smarm optimization (PSO).

As it is discussed in the previous sections, GA optimization shows the variety of response than PSO in the same number of iteration. PSO just improves the rise time in the consecutive iteration without improving (decreasing) the overshoot percentage; on the other hand, decreasing the rise time with increasing the overshoot percentage is seen in GA. However; in the flexible manipulator system low overshoot value with fast response is intended as design specification. So by compromising between these two methods GA shows the excellent ability to estimate the value of PID controller coefficients corresponding to the design standard of flexible manipulator system. By using two versatile optimization methods this study can be regarded as basic yardstick for incorporating the optimization technique with conventional PID controller in designing controller on flexible manipulator system. The next stage of our research will be experimental confirmation of our theory.

References

Abdolvand, M., & Fatehi, M. H. (2012). Model-Base Predictive Control for Vibration Suppression of a Flexible Manipulator. UKACC Internatinal Conference on Control. cardfii, uk, 3–5.

Ahmed, M.A., Nasir, A.N., Ismail, R., & Ramli, M.S. (2010). Control Strategy for Active Vibration Suppression of Flexible Robot Manipulator, Proceedings of IEEE Internatinal Conference on Information and Automation, Harbin, Chaina, 741–746.

Becedas, J., Trapero, J.R., Sira-Ramorez, H., & Feliu-Battle, V.(2007). Fast Identification Method to Control a Flexible Manipulator with Parameter Uncertainties. IEEE internatinal Conference on Robotics and Automation, Roma, Italy, 10-14

Brouwer, D., Jongc, B., & Soemers, H.M.(2010). Design and Modelling of a Six DOFsMEMS-Based Precision Manipulator. Precision Engineering, 18(2),307-319.

Gol Zardian, M., & Ayob, A. (2015). Intelligent modelling and active vibration control of flexible manipulator system. Journal of Vibroengineering, 17(4), 1879-1891.

Mute, D., Ghosh, S., &Subudhi, B. (2013). Iterative Learning Control of a Single-Link Flexible Manipulator Based on an Identified Adaptive NARX Model. Annual IEEE India Conference (INDICON), 1(1).

Nille ,O (2001). Nonlinear System Identification: From Classical Approaches to Neural Networks and Fuzzy Models. Springer Science & Business Media.

Qiu, Z., Shi, M., Wang, B., & Xie, Z. (2012). Genetic Algorithm Based Active Vibration Control for a Moving Flexible Smart Beam Driven by a Pneumatic Rod Cylinder. Journal of Sound and Vibration, 331(10), 2233-2256.

Ramos, F., & Feliu,V. (2008). New online payload identification for flexible robots, aplication to adaptive control, Juornal of Sound and Vibration. 315(1-2), 34-57.

Sekiguchi, Y., Kobayashi, Y., Tomono, Y., Watanabe, H., Toyoda, K., Hashizume, M., & Fujie, M. (2010). Development of a Tool Manipulator Driven by a Flexible Shaft for Single Port Endoscopic Surgery. Proceedings of the 3rd IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics, The University of Tokyo, Tokyo, Japan, September 26-29,120–125.

Shawky, A., Zydek, D., Yehia Z., & Ordys, A. (2013). Modelling and Nonlinear Control of a Flexible-Link Manipulator. Applied Mathematical Modelling, 37(23),9591-9602.

Tavakolpour, A., Maliah, M., & Darus, I.Z.M.(2011). Modeling and Simulation of a Novel Active Vibration Control System for Flexible Structures, Wseas Transactions On Systems And Control, 6(5).

Tokhi, M.O., Mohamed, Z., & Shaheed, M.H. (2001). Dynamic Characteristic of Flexible Manipulator System, Robatica, 19, No, 571-580.

Vakil, M., Fotouhi,R., & Nikiforuk, P.N.(2009). Causal End-Effector Inversion of a Flexible Link Manipulator. Mechatronics, 19(7), 1197-1210.

Yatim, H.M., & Darus, I.Z.M. (2012). Intelligent Parametric Identification of Flexible Manipulator System . IEEE Conference On Control, System And Industrial Information(ICCSII), Bandung, Indonesia, 23-26

Zain, M., Tokhi, M.O., & Mohamed, B. (2006). Hybrid Learning Control Schemes with Input Shaping of a Flexible Manipulator System, Mechateronics, 16, 209-219.

Zarafshan, P., & Moosavian, S. A. A. (2013). Dynamics modelling and Hybrid Suppression Control of space robots performing cooperative object manipulation. Communications in Nonlinear Science and Numerical Simulation, 18(10), 2807-2824.

Ziaei, K., &Wang, D. (2009). QFT-Based Design of Force and Contact Transition Controllers for a Flexible Link Manipulator. Control Engineering Practice, 17(3), 329-344.

.

An International Journal JSCDSS
E-ISSN: 2289-8603