Control of a Flexible Manipulator Using a Sliding Mode Controller with Genetic Algorithm Tuned Manipulator Dimension

N.M. Kwok[1] and S. Kwong[2]

[1] Faculty of Engineering, University of Technology, Sydney, NSW, Australia
ngai.kwok@eng.uts.edu.au
[2] Department of Computer Science, City University of Hong Kong
Hong Kong SAR, P.R. China
cssamk@cityu.edu.hk

Abstract. The tip position control of a single-link flexible manipulator is considered in this paper. The cross-sectional dimension of the manipulator is tuned by genetic algorithms such that the first vibration mode dominates the higher order vibrations. A sliding mode controller of reduced dimension is employed to control the manipulator using the rigid and vibration measurements as feedback. Control effort is shared dynamically between the rigid mode tracking and vibration suppression. Performance and effectiveness of the proposed reduced dimension sliding mode controller in vibration suppression and against payload variations are demonstrated with simulations.

1 Introduction

Flexible manipulators are robotic manipulators made of light-weight materials, e.g. aluminium. Because of the reduction in weight, lower power consumption and faster movement can be realised. Other advantages include the ease of setting up and transportation, as well as reduced impact destruction should the manipulator system went faulty. Light-weight manipulators can be applied in general industrial processes, e.g. pick and place, other applications could include the space shuttle on-board robotic arm. However, due to its light-weight, vibrations are inherent in the flexible manipulator that hinders its wide application.

Earlier research on the control of flexible manipulators could be found in [1] where a state-space model of the manipulator was proposed and the linear quadratic gaussian approach was used. Other research work included the use of an H_∞ controller in [2], the manipulator considered allow movements in both the horizontal and vertical planes. The use of deterministic control was found in [3], where a sliding mode controller was applied. The control effort was switched between the vibration mode errors, however, in an ad-hoc weighting basis. Recently, in [4], a sliding mode controller with reduced controller dimension, using only the rigid mode and first vibration mode as feedback, was demonstrated. The sliding surfaces were made adaptive to enhance the controller performance.

E. Cantú-Paz et al. (Eds.): GECCO 2003, LNCS 2724, pp. 2191–2202, 2003.

However, a relatively large number of switching in controller gains had to be incorporated in the controller. Apart from these advanced controller designs, intelligent controllers were also reported in [5], where a model-free fuzzy controller was used to control the flexible manipulator. In [6], a flexible manipulator was controlled by a neural network controller, trained with genetic learning. Genetic tuning of a Lyapunov based controller was also reported in [7], where stability was guaranteed by the Lyapunov design. Moreover, the controller performance was enhanced by genetic algorithm tuning. The work in [8] also showed that a sliding mode controller retaining only the rigid mode and first vibration mode as feedback was feasible. The approach adopted was that the control on the tip position and vibrations were dynamically weighted in a fuzzy-like manner. Diverging from the focus on controller designs, structural dimensions of the flexible manipulator was considered in [9], where a tapered beam resulted and the vibration frequency was increased for faster manoeuvre.

In this work, we consider the tip angular positional control of a single-link flexible manipulator. The manipulator is constructed in the form of a narrow beam of rectangular cross-section. At the hub is driven by a dc motor and the tip carries a payload. We use a sliding mode controller for its robustness against model uncertainties and parameter variations, e.g. payload variations. An inspection on the manipulator mathematical model will reflect that the manipulator is of infinite dimension, and the design and implementation of an infinite dimension controller is challenging. Here, we follow the work of [4] and [8], that only the rigid mode and first vibration mode are used as feedback and the controller dimension is therefore reduced. When using only the first vibration mode as one of the feedback, we need to ensure that it dominates all other higher order vibrations. Motivated by the work in [9], we propose to adjust the cross sectional dimension of the manipulator. In addition to the above requirement, we also aim to increase the vibration frequency for reduced vibration magnitude, while rejecting a large cross section area that increases the weight of the manipulator. We also see from the manipulator model that vibrations are proportional to the slope of the mode shapes at the hub. Therefore, in order to reduce vibration as a whole, we penalise large mode shape slopes. The above arguments lead to the formulation of a multi-objective optimisation problem. From [10], we propose to use the genetic algorithm to search for the optimal manipulator cross sectional dimension. The resultant manipulator characteristics will facilitate the design and implementation of a sliding mode controller of reduced dimension.

This paper is arranged as follows. The manipulator model will be developed in Section 2. Section 3 will present the sliding mode controller design. The genetic tuning of manipulator dimensions will be treated in Section 4. Simulation results will be presented in Section 5 and a conclusion will be drawn in Section 6.

2 System Modelling

The flexible manipulator considered here is a uniform aluminium beam with rectangular cross section and moves in the horizontal plane. One end of the

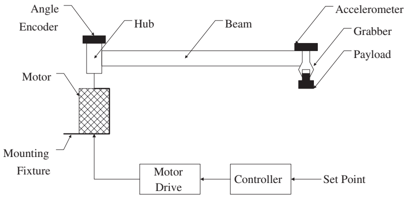

Fig. 1. System Configuration

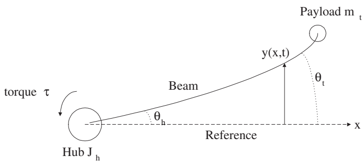

Fig. 2. Parameter Definition

beam is fixed to the hub consisting of a dc motor and associated mounting fixtures. At the other end, the tip, is mounted a gribber for pick and place operation. The gribber and the load together form the payload. Angular sensors, e.g. accelerometers and angular encoders, are mounted at the tip and the motor shaft respectively. The manipulator set-up is shown in Fig. 1. The definitions of the manipulator parameters are shown in Fig. 2. Using the Eular-Bernoulli beam theory, we assume that shear and rotary inertia are negligible and vibrations being small as compared to the length of the manipulator, the equation of motion of the manipulator can be given as [2]

$$EI\frac{\partial^4 y}{\partial x^4} + \rho\frac{\partial^2 y}{\partial t^2} = 0 \tag{1}$$

where E=Young's modulus, I=cross sectional moment of inertia, ρ=mass per unit length, y=displacement from reference
The boundary conditions are

$$y(0) = 0,\ EIy''(0) = \tau + J_h\ddot{\theta}_h,\ EIy'''(L) = m_t\ddot{y}(L),\ EIy''(L) = -J_t\ddot{y}'(L) \tag{2}$$

where τ=motor torque, J_h=hub inertia, θ_h=hub angle, m_t=payload mass, J_t=payload inertia, L=manipulator length, *prime* stands for differentiation against x, *dot* stands for differentiation against time t
Using separation of variables, put

$$y(x,t) = \phi(x)\varphi(t) \tag{3}$$

then we have

$$\ddot{\varphi} + \omega^2\varphi = 0,\ \phi'''' - \beta^4\phi = 0,\ \omega^2 = \beta^4\frac{EI}{\rho} \tag{4}$$

The general solution to Equ.4 is the mode shape given by

$$\phi(x) = Asin\beta x + Bcos\beta x + Csinh\beta x + Dcosh\beta x \tag{5}$$

where β is the solution of the characteristic equation formed from the boundary conditions
Equ.5 can be expressed with coefficient A only, giving

$$\phi(x) = A(sin\beta x + \gamma(cos\beta x - cosh\beta x) + \xi sinh\beta x) \tag{6}$$

where γ and ξ are functions of β and system parameters, and A is yet to be determined by normalisation
Using the assumed mode method, put

$$y(x,t) = \sum_{i=0}^{\infty} \phi_i(t)\varphi_i(t) = x\theta_h + \sum_{i=1}^{\infty} \phi_i(x)\varphi_i(t) \tag{7}$$

The normalization coefficient A_i of the general solution is

$$A_i = \sqrt{\frac{J_{total}}{\int_0^L \rho\psi_i^2(x)dx + m_t\psi_i^2(L) + J_t\psi_i'^2(L) + J_h\psi_i'^2(0)}} \tag{8}$$

where ψ_i is the expression inside the bracket in Equ.6
After normalisation using the orthogonal property and keeping consistence with the rigid mode, we have

$$\ddot{\varphi}_i + 2\zeta\omega_i\dot{\varphi}_i + \omega^2\varphi_i = \frac{\tau}{J_{total}}\phi_i^{'}(0) \tag{9}$$

where ζ is the material damping of small value, $\phi_i^{'}0)$ is the slope of the ith vibration mode shape at the hub, J_{total} is the total inertia making up of the hub, beam and the tip
With further manipulations, a state space system equation can now be written as

$$\begin{bmatrix} \dot{\theta}_h \\ \ddot{\theta}_h \\ \dot{\varphi}_1 \\ \ddot{\varphi}_1 \\ \vdots \\ \dot{\varphi}_n \\ \ddot{\varphi}_n \end{bmatrix} = \begin{bmatrix} 0\;1 & . & . & . & . & . \\ 0\;0 & . & . & . & . & . \\ .\;\; . & 0 & 1 & . & . & . \\ .\;\; . & -2\zeta\omega_1 & -\omega_1^2 & . & . & . \\ .\;\; . & . & \ddots & . & . & . \\ .\;\; . & . & . & . & 0 & 1 \\ .\;\; . & . & . & . & -2\zeta\omega_n & -\omega_n^2 \end{bmatrix} \begin{bmatrix} \theta_h \\ \dot{\theta}_h \\ \varphi_1 \\ \dot{\varphi}_1 \\ \vdots \\ \varphi_n \\ \dot{\varphi}_n \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \\ \phi_1^{'}(0) \\ \vdots \\ 0 \\ \phi_n^{'}(0) \end{bmatrix} \frac{\tau}{J_{total}} \tag{10}$$

The motor dynamic equation is

$$J_h\ddot{\theta}_h = \tau = \frac{1}{R}k_t(V - k_b\dot{\theta}_h),\; V = \frac{1}{k_t}R\tau + k_b\dot{\theta}_h \tag{11}$$

where J_h is the hub inertia, τ is the torque generated by the motor, k_t is the torque constant, V is the voltage applied to the motor, k_b is the back-emf constant, R is the armature resistance
Let the controller generate the required driving torque τ, then we will use inverse dynamics to compute the motor voltage V. It is because the dc motor model has been well studied and its identification in real practice is not difficult. Finally, we will apply the derived control voltage to the motor to drive the flexible manipulator.

3 Controller Design

We will design the controller using only the rigid mode and first vibration mode as feedback signals. From the state space equation, Equ.10, we re-write the system differential equations

$$\ddot{\theta} = \frac{\tau_1}{J_{total}}, \ \ddot{\varphi} = -2\zeta\omega\dot{\varphi} - \omega^2\varphi + \frac{\tau_2}{J_{total}}\phi' \tag{12}$$

Note that we have dropped all subscripts for clarity. The control objective is to make

$$\theta \to \theta_d \qquad \text{and} \qquad \varphi \to 0 \qquad \text{for} \qquad t \to \infty$$

where θ_d is the desired hub angle
We then define the error variables as

$$\theta_e = \theta_d - \theta, \ \varphi_e = -\varphi \tag{13}$$

The system differential equations become

$$\ddot{\theta}_e = \ddot{\theta}_d - \frac{\tau_1}{J_{total}}, \ \ddot{\varphi}_e = -2\zeta\omega\dot{\varphi}_e - \omega^2\varphi_e - \frac{\tau_2}{J_{total}}\phi' \tag{14}$$

Following the sliding mode design methods, we define the sliding surface as

$$s_1 = c_1\theta_e + \dot{\theta}_e \tag{15}$$

where c_1 is the slope of the sliding surface
When sliding mode is attained, we have

$$\theta_e(t) = \theta_e(t_s)e^{-c_1 t} \qquad for \qquad t > t_s$$

where t_s is the time that sliding mode firstly occurred and $c_1 > 0$ then

$$\theta_e(t) \to 0 \qquad as \qquad t \to \infty$$

The sliding surface time derivative is

$$\dot{s}_1 = c_1\dot{\theta}_e + \ddot{\theta}_e = c_1\dot{\theta}_e + \ddot{\theta}_d - \frac{\tau_1}{J_{total}} \tag{16}$$

Put

$$\tau_1 = (c_1\dot{\theta}_e + \ddot{\theta}_d + j_1 sign(s_1))J_{total} \tag{17}$$

where $sign(x) = 1$ for $x > 0$ and $sign(x) = -1$ otherwise, and the sliding surface s_1 can be reached for $j_1 > 0$
Similarly, define the sliding surface s_2 as

$$s_2 = c_2\varphi_e + \dot{\varphi}_e \tag{18}$$

The time derivative is

$$\dot{s}_2 = c_2\dot{\varphi}_e + \ddot{\varphi}_e = (c_2 - 2\zeta\omega)\dot{\varphi}_e - \omega^2\varphi_e - \frac{\tau_2}{J_{total}}\phi' \tag{19}$$

Put

$$\tau_2 = ((c_2 - 2\zeta\omega)\dot{\varphi}_e - \omega^2\varphi_e + j_2 sign(s_2))\frac{J_{total}}{\phi'} \tag{20}$$

where sliding surfaces_2 can also be reached when $j_2 > 0$
Let the sliding mode controller output be

$$u_1 = \gamma_1 + k_1 sign(s_1),\ u_2 = \gamma_2 + k_2 sign(s_2) \tag{21}$$

where

$$\begin{array}{l} \gamma_1 = (c_1\dot{\theta}_e + \ddot{\theta}_d)J_{total},\ k_1 = j_1 J_{total} \\ \gamma_2 = ((c_2 - 2\zeta\omega)\dot{\varphi}_e - \omega^2\varphi_e)\frac{J_{total}}{\phi'},\ k_2 = j_2\frac{J_{total}}{\tau_2} \end{array} \tag{22}$$

Note that there is only one control torque from the motor but there are two control objectives; namely, $\theta \to \theta_d$ and $\varphi \to 0$. We have to assign weighting factors n_1 and n_2 applying to u_1 and u_2 respectively.

$$u = n_1u_1 + n_2u_2 = n_1\gamma_1 + n_1k_1 sign(s_1) + n_2\gamma_2 + n_2k_2 sign(s_2) \tag{23}$$

In order to ensure that sliding surfaces are reached, that is, to make $s_1 \to 0$, $s_2 \to 0$; we also have to assign the values of k_1 and k_2. In so doing, we define a Lyapunov function

$$V = \frac{1}{2}(s_1^2 + s_2^2) > 0 \qquad \text{for all} \qquad s_1 \neq 0, s_2 \neq 0 \tag{24}$$

The time derivative is required to be negative for reachability, that is

$$\dot{V} = s_1\dot{s}_1 + s_2\dot{s}_2 < 0 \tag{25}$$

When we apply the weighted and aggregated control, Equ.23, to the plant, Equ.25 becomes

$$\begin{array}{l} \dot{V} = s_1(n_2(\gamma_1 - \gamma_2) - n_1k_1 sign(s_1) - n_2k_2 sign(s_2)) + \\ \qquad s_2(n_1(\gamma_2 - \gamma_1) - n_1k_1 sign(s_1) - n_2k_2 sign(s_2)) \end{array} \tag{26}$$

From Equ.26, we see that there are two sliding surfaces s_1 and s_2 being inter-related to each other. Their combination will affect the Lyapunov function derivative. It was shown in [8] that the effect will be reduced if the values of k_1 and k_2 are given by
for $s_1 > 0$ and $s_2 > 0$

$$k_1 = max(\gamma_2 - \gamma_1, 0) + \epsilon_1,\ k_2 = max(\gamma_1 - \gamma_2, 0) + \epsilon_2 \tag{27}$$

for $s_1 < 0$ and $s_2 < 0$

$$k_1 = max(\gamma_1 - \gamma_2, 0) + \epsilon_1,\ k_2 = max(\gamma_2 - \gamma_1, 0) + \epsilon_2 \tag{28}$$

for $s_1 > 0$ and $s_2 < 0$

$$k_1 = max(\frac{n_2(\gamma_1 - \gamma_2)}{n_1}, 0) + \epsilon_1,\ k_2 = 0 \tag{29}$$

for $s_1 < 0$ and $s_2 > 0$

$$k_1 = max(\frac{n_2(\gamma_2 - \gamma_1)}{n_1}, 0) + \epsilon_1,\ k_2 = 0 \tag{30}$$

where ϵ_1 and ϵ_2 are small positive constants to compensate for model uncertainties and parameter variations
Regarding the sharing of control effort between regulation and vibration suppression, we state our control strategy in the following rule.

A larger control effect is to be applied for hub angle regulation when the hub angle error is large; when the hub angle is near the set point (small error), apply larger control effort to suppress the vibration mode.

We also observe from Equ.29 and Equ.30; that k_1 may go unbounded if $n_1 \to 0$. Therefore, we put

$$n_1 > n_2 \qquad \text{with} \qquad n_1 + n_2 = 1 \tag{31}$$

We also assign n_1 according to the control strategy in the rule stated above. Put

$$n_1 = \frac{1}{2}\left(1 + \frac{1}{1 + exp(-a(\left|\frac{\theta_e}{\theta_d}\right| - b))}\right),\ n_2 = 1 - n_1 \tag{32}$$

where θ_e is the hub angle error, θ_d is the desired set point, a determines the slope of weight crossover, b determines the point where crossover in control weighting is to occur

4 Manipulator Dimension Tuning by Genetic Algorithm

The reduced dimension sliding mode controller developed in Section 3 depends critically on the fact that the first vibration mode magnitude dominates the higher order vibrations. When the tip displacement is given by Equ.7, what we can do to make y_1 dominates y_i, $i > 1$, is to make ϕ_1 and φ_1 to dominate the corresponding higher order variables. Referring to Equ.6, we see that the value of ϕ_i depends on the value of A_i and β_i. Also from Equ.10, the vibrations will be excited according to the gain $\phi_i^{'}(0)$. We also see that in normalising the variable A_i in Equ.8, it is a function of the variable β_i. To sum up, the variable β_i plays a critical role in satisfying our requirements in making the first vibration mode dominate.

From a practical point of view, the hub parameters and the tip parameters are relatively fixed by the motor torque required and the task assigned. The length of the manipulator is also fixed by the workspace. A relatively free parameter, which bears an effect on β_i is the cross sectional area of the manipulator. Adjusting the height and width of the manipulator, we change the value of mass per unit length, cross sectional moment of inertia as well as the beam inertia. The effect of the manipulator dimension on β_i and in turn on ϕ_i is very non-linear. From the equation of motion in Equ.1 and boundary conditions in Equ.2, the determination of β_i in closed form analytical solution is very involved. Therefore, we turn to the use of genetic algorithms to search for the manipulator dimension that best satisfies our requirements.

Genetic algorithms are stochastic search methods based on the theory of evolution, the survival of the fittest, and the theory of schema [10]. When implementing the genetic algorithm, variables to be searched are coded in binary strings. For two such strings, parents, the binary bit patterns are exchanged in some bit position selected randomly. The resultant pair of strings, off-springs, become members of the population in the next generation. The exchange of binary bits, crossover, is conducted according to a crossover probability p_c. The parents with higher fitness have a higher chance for crossover. The above process exploits the search space but diversity should be explored. This is made possible by applying mutation to the off-springs. Mutation flips one of the bit in the string according to the probability p_m.

For every string in the population, fitness is evaluated and then the process repeats in the next generation. Termination criteria may be selected from tracking the improvement over the average fitness of a generation or according to the count of number of generations being processed.

Variations from the standard genetic algorithms are adopted here in tuning the manipulator dimensions. We will apply elitism, such that the string with the highest fitness value is stored irrespective of the processing of generations. It is because the last generation before termination of the algorithm may or may not contain the best string. We also adjust the crossover and mutation probability dynamically according to the improvement on the average fitness of a generation. If found improving, then p_c and p_m are reduced by a scale factor. While the average fitness is not improving, we increase p_c and p_m to gain more chances in finding a string of higher fitness. Our algorithm terminates when the average fitness is improving for several consecutive generations. The genetic algorithm becomes that described below.

```
initialise the population randomly
evaluate fitness and average
while not end of algorithm
  select parents by fitness
```

```
  apply crossover and mutation
  evaluate fitness and average
  if improving then decrease pc and pm
  else increase pc and pm
  store the fittest string
end on fitness improved or generations count
```

The fitness function is defined as

$$fitness = f_r + f_1 + f_2 - f_3 - f_4 - f_5 + f_6 \tag{33}$$

where f_r =reference datum keeping fitness> 0, $f_1 = 1$ when $\phi_i(L)$ are in descending order $f_1 = 0$ otherwise, $f_2 = \sum(\phi_i(L) - \phi_{i+1}(L))$ reward greater dominance of lower order vibrations, $f_3 = \sum |i \times \phi_i(L)|$ penalise weighted magnitude of vibrations at tip, $f_4 = \sum \phi_i^{'}(0)$ penalise mode shape slopes at the hub, $f_5 = H_b W_b$ penalise for large cross-sectional beam area, $f_6 = \sum \omega_i$ reward higher vibration frequencies

5 Simulation

Simulations will be described in this section as well as the presentation of simulation results. Cases for comparisons include responses from open-loop and from the use of the proposed sliding mode controller. We will also present cases for initially un-tuned and the resulting tuned manipulator responses. The parameters used in the simulations are given below in Table 1.

Table 1. Simulation Parameters

beam initial height/width/length	0.04m/0.002m/0.8m
hub inertia	$1.06^{-4} kgm^2$
payload mass/inertia	0.38kg/$1.78^{-4} kgm^2$
no. of population/generation	10/20
probability p_c/p_m (initial)	0.9/0.02
scaling factor for p_c and p_m	1.1
termination on improvement	3 generations

5.1 Response with Un-tuned Dimensions

Using the initial manipulator parameters, an open-loop simulation was conducted for subsequent performance comparisons. The motor was fed with a voltage pulse, by some trial and error that brings the response to 0.2rad as the set-point for close-loop control. The result is shown in Fig. 3. The close-loop response using the sliding mode controller is shown in Fig. 4, where initial un-tuned dimensions were used. Comparing Figs. 3 and 4, we see that vibrations are effectively suppressed in the steady state with the close-loop control. However, vibration during the transient period is still observed.

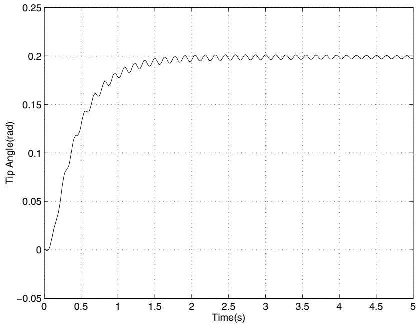

Fig. 3. Open-loop response, dimensions not tuned

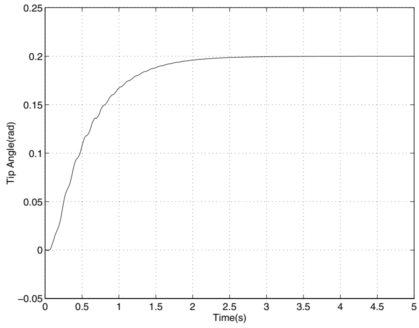

Fig. 4. Close-loop response, dimensions not tuned

5.2 Genetic Algorithm Tuning

Tuning of the manipulator height and width was conducted according to the genetic algorithm described in Section 4. It is noted that the algorithm terminated before reaching the maximum number of generations with a relatively small number of populations. This observation shows the modification on the standard genetic algorithms with dynamic adjustment of the crossover and mutation probabilities were effective. Figures 5 and 6 shows the value of fitness function and the average fitness over the generations. It is also observed from the plots that the maximum fitness function though not increasing explicitly, the occurrences of low fitness were decreasing.

Results of mode shapes at tip, $\phi_i(L)$, mode shape slopes at the hub, $\phi_i^{'}$, and the vibration frequencies, ω_i, are tabulated in Table 2 for comparison between the initially un-tuned and the resulting tuned values. From the results, we see that the mode shape at the tip is arranged in descending order as required. The vibration frequency shows an increase from the tuned dimensions. An overall

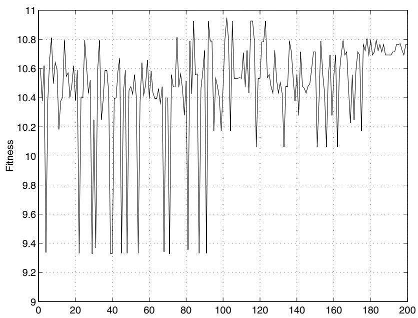

Fig. 5. Fitness Function

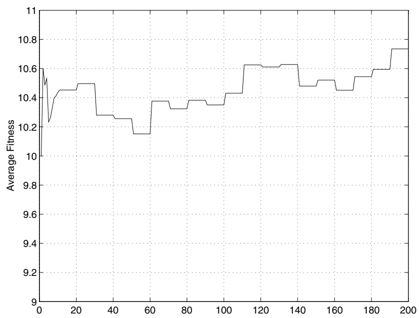

Fig. 6. Average fitness function

Table 2. Comparison of design variables resulted from initially un-tuned and tuned manipulator dimension

Results from un-tuned dimensions

mode	$.\phi_i(L)$	$.\phi_i'$	$.\omega_i$
1	-0.228	6.983	45.985
2	0.118	17.242	164.929
3	0.012	23.389	320.977

Results from tuned dimensions

mode	$.\phi_i(L)$	$.\phi_i'$	$.\omega_i$
1	-0.211	7.421	68.366
2	0.102	18.074	243.026
3	0.000	24.191	465.226

optimal manipulator dimension results irrespective of the slight increase in mode shape slopes.

5.3 Responses with Tuned Dimensions

The tip responses with the use of tuned dimensions are shown in Figs. 7 and 8 below. The tuning of manipulator dimension using genetic algorithms is effective

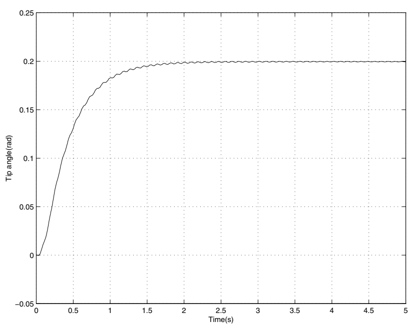

Fig. 7. Open-loop response, dimensions tuned

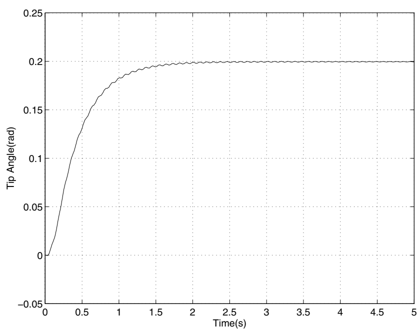

Fig. 8. Close-loop response, dimensions tuned

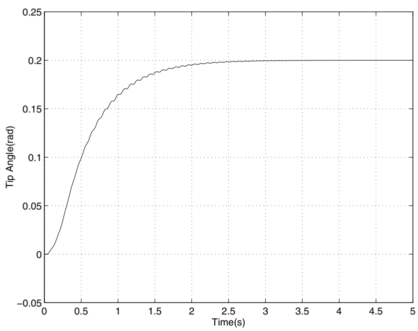

Fig. 9. Close-loop response, dimensions tuned, payload increased by 50%

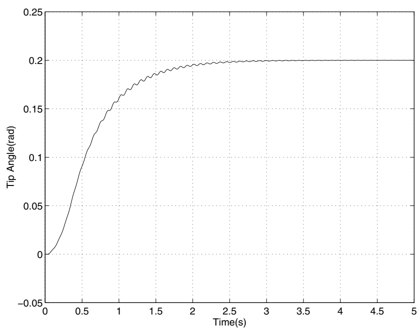

Fig. 10. Close-loop response, dimensions tuned, payload increased by 100%

in reducing the vibration magnitudes even in the open-loop response. Figures 9 and 10 show the close-loop response when the payload is increased by 50% and 100% respectively. No significant degrade in response is observed.

6 Conclusion

A sliding mode controller was used to control the position of a single-link flexible manipulator. Only the rigid mode and first vibration mode were used as feedback signals that reduced the controller dimension. The manipulator dimensions were tuned using a genetic algorithm routine, with the application of elitism and adaptive crossover and mutation probabilities, had made the first vibration mode dominating the higher order vibrations and reduced the complexity in the implementation of the controller. Simulation results had demonstrated that the controller was effective in suppressing vibrations and achieving set-point regulation.

Acknowledgement. This work is supported by the City University Strategic Grant 7001416 in part.

References

1. R.H.Cannon, E.Schmitx: Initial Experiments on the End-point Control of a Flexible One-link Robot. Intl. J. of Robotics Research, Vol.3, No.3 (1984) 62–75
2. R.P.Sutton, G.D.Halikas, A.R.Plumer, D.A.Wilson: Modelling and H_∞ Control of a Single-link Flexible Manipulator. Proc. Instn. Mech. Engrs., Vol.213, Pt.1 (1999) 85–104
3. A.R.Ingole, B.Bandyopadhyay, R.Gorez: Variable Structure Control Application for Flexible Manipulators. Proc. 3^{rd} IEEE Conf. Control Applications, Glasgow, UK (1994) 1311–1316
4. F.H.K.Fung, C.K.M.Lee: Variable Structure Tracking Control of a Single-link Flexible Arm Using Time-varying Sliding Surface. J. Robotic Sys., Vol.16, No.12 (1999) 715–726
5. J.X.Lee, G.Vukovich, J.Z.Sasiadek: Fuzzy Control of a Flexible Link Manipulator. Proc. American Control Conf., Baltimore, USA (1994) 568–574
6. S.Jain, P.Y.Peng, A.Tzes, F.Khorrami: Neural Network Designs With Genetic Learning for Control of a Single Link Flexible Manipulator. Proc. American Control Conf., Baltimore, USA (1994) 2570–2574
7. S.S.Ge, T.H.Lee, G.Zhu: Genetic Algorithm Tuning of Lyapunov-based Controllers: An Application to a Single-link Flexible Robot System. IEEE Trans. Ind. Elecn, Vol.43, No.5 (1996) 567–573
8. C.K.Lee, N.M.Kwok: Control of a Flexible Manipulator Using a Sliding Mode Controller With a Fuzzy-like Weighting Factor. Proc. 2001 IEEE Intl. Symposium on Ind. Elecn., Pusan, Korea (2001) 52–57
9. F.Y.Wang, J.L.Russell: Optimum Shape Construction of Flexible Manipulators With Total Weight Constraint. IEEE Trans. Sys. Man and Cybernetics, Vol.24, No.4 (1995) 605–614
10. K.F.Man, K.S.Tang, S.Kwong: Genetic Algorithms: Concepts and Applications. IEEE Trans. Ind. Elecn., Vol.43, No.5 (1996) 519–534