On Design of a H_∞ Controller for a Flexible Link Robotic Manipulator

bidyadhar.subhidi@nitrkl.ac.in
B.Subudhi and A.S.Morris

***Abstract*- The paper presents a novel composite control scheme that is a superposition of a non-linear neural network controller (based on the slow dynamics) and a linear $H\infty$ controller (based on the fast dynamics) for a manipulator with flexible links and joints. The controller is robust in the face of uncertainties existing in both the slow and the fast dynamics which may be due to the inexactness in achieving the two-time-scale separation of the two subsystems (slow and fast) and unmodelled dynamics in the fast subsystem owing to model truncation. The performance of the proposed controller has been examined by numerical simulations.**

***Index Terms*- Flexible link, Singular Perturbation, $H\infty$ controller.**

I. INTRODUCTION

The singular perturbation (SP) technique is a basically useful in achieving reduced order modelling of complex systems since it allows the full-order complex dynamics to be divided into simpler subsystems consisting of the reduced order slow dynamics and the fast dynamics. Separate sub-controllers can then be designed for the slow and fast sub-systems and combined into a composite controller for the whole system which is also of lower order than a controller that would have been designed for the full order system.

SP-based controllers have previously been applied to manipulators where either link or joint flexibility is considered [1]. However, SP technique had not been applied for controllers that take account of flexibility in both links and joints together until the recently published paper by the authors [2], in which the slow subsystem comprises the non-flexible motion of the links and joints and the fast subsystem comprises the flexible modes of the links and joints. The frequencies of rigid modes are much less than that of the flexible modes of the manipulator thus enabling to choose joint angular positions of the links as the slow variables and the modes of vibration of the links and joints as the fast variables. In this implementation of SP, an integral manifold approach [1] was used to derive corrected slow and fast models. Then, an inverse dynamics (computed torque) method was used to control the slow dynamics and a linear quadratic regulator (LQR) algorithm was applied for the linearised fast dynamics, in a similar fashion to earlier work [1,2].

Whilst the integral manifold approach is useful for obtaining corrected slow and fast subsystems, solution of the manifold equations becomes very complicated when higher order perturbation terms are considered.Computational limitations mean that it is necessary to approximate the manifold expansion, which, together with unmodelled high-frequency modes, leads to model uncertainty that is reflected into the slow dynamics. Unfortunately, the inverse dynamics technique requires a perfect model so that the gains of the controller can be chosen to achieve a critically damped system response [1], and this perfect model condition is clearly not met. Likewise, the LQR technique becomes unsatisfactory when there is uncertainty in the fast subsystem. Thus, faced with uncertainly in both slow and fast sub-systems, better controllers that take proper account of this uncertainty are needed, as developed in this paper.

II. REDUED ORDER MODELLING

Consider a manipulator with n-flexible serial links and n-flexible actuated joints, with an inertial payload of mass m_p and inertia I_p [3]. Each flexible joint is modelled as a linear torsional spring that connects the rotor of the joint actuator to the link. α_i, θ_i are the ith rotor and link angular positions. I_{ri} is the ith rotor inertia, $u_i(t)$ is the input torque, N_i is the gear ratio for the ith rotor and k_{si} is the spring constant of the ith flexible joint $(\mathrm{FJ})_i$. Applying the Euler-Lagrange principle and assumed modes method, the dynamic equations are [2]:

$$\mathbf{J}\ddot{\alpha} - \mathbf{K}_s(\theta - \alpha) = \boldsymbol{u} \quad (1)$$

$$\mathbf{M}(\theta, q)\begin{Bmatrix}\ddot{\theta}\\ \ddot{q}\end{Bmatrix} + \begin{Bmatrix}f_1(\theta,\dot{\theta})\\ f_2(\theta,\dot{\theta})\end{Bmatrix} + \begin{Bmatrix}g_1(\theta,\dot{\theta},q,\dot{q})\\ g_2(\theta,\dot{\theta},q,\dot{q})\end{Bmatrix} + \begin{Bmatrix}0\\ \mathbf{D}\dot{q}\end{Bmatrix} + \begin{Bmatrix}\mathbf{K}_s(\theta-\alpha)\\ \mathbf{K}_w q\end{Bmatrix} = \begin{Bmatrix}0\\ 0\end{Bmatrix} \quad (2)$$

By using the SP technique [3], we can divide the dynamics of the manipulator into a slow subsystem:

$$\ddot{\bar{\theta}} = (\mathbf{M}_{11} + \mathbf{J})^{-1} \times \{-f_1(\bar{\theta}, \dot{\bar{\theta}}) + \bar{u}\} \quad (3)$$

and a fast subsystem: $\dot{x}_f = \mathbf{A}_f x_f + \mathbf{B}_f u_f$ (4)

where $(\mathbf{H}_{11} - \mathbf{H}_{12}\mathbf{H}_{22}^{-1}\mathbf{H}_{21}) = \mathbf{H}_{11} = \mathbf{M}_{11}^{-1}$

$$\mathbf{A}_f = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{I} \\ -\mathbf{H}_{22}\tilde{\mathbf{K}}_w & \mathbf{H}_{21}\tilde{\mathbf{K}}_s & \mathbf{0} \\ \mathbf{H}_{12}\tilde{\mathbf{K}}_w & (\mathbf{H}_{11} + \mathbf{J}^{-1})\tilde{\mathbf{K}}_s & \mathbf{0} \end{bmatrix} \qquad \mathbf{B}_f = \begin{bmatrix} \mathbf{0} \\ \mathbf{J}^{-1} \end{bmatrix};$$

$x_f = [z_1^T \quad z^T \quad z_2^T \quad z_4^T]^T$ [3] and $\mathbf{0}$ and $\mathbf{I}$ are zero and identity matrices.

III. COMPOSITE CONTROLLER

In the case of a manipulator with many flexible links and joints, the dynamic equations involve a set of highly non-linear and coupled partial differential equations, thus posing a serious control problem compared to a simple single flexible arm. A NN-based controller is likely to perform better than an inverse dynamics scheme in controlling the slow dynamics since it does not require either exact knowledge of the system dynamics or inverse dynamic model evaluation [4]. Furthermore, it guarantees boundedness in the tracking errors and control signals.

B.Subudhi, Dept. of Electrical Engineering, Indira Gandhi Institute of Technology, Sarang-759146, Orissa, India.
A.S.Morris, Dept. of Automatic Control & Systems Engg., Mappin Street, Sheffield S1 3JD, Univ. of Sheffield, UK

With regard to the fast subsystem, the $H\infty$ control strategy has previously been applied successfully to flexible manipulators assuming linear dynamics [5]. In the technique has been utilised the robust features of $H\infty$ optimal control to stabilize the fast subsystem in the presence of model uncertainty due to unmodelled high frequency modes, but because the standard $H\infty$ optimal control problem is solved for two Riccati equations i.e. one for the controller and the other for the observer, the order of the controller increases. This increases the computation time of the control task. Consequently, a state feedback $H\infty$ controller for the fast subsystem is proposed in this work where only one Riccati equation has to be solved, which is a special case of the standard $H\infty$ problem with static gains.

The structure of the new singular perturbation based neuro-H_∞ controller (SNHC) proposed is given in Fig.1. This is a composite controller with separate schemes for the slow and fast subsystems.

We start with design of the NN controller for slow subsystem as follows. The slow dynamics (3) can be re-written after pre-multiplying both sides by $(\mathbf{M}_{11}+\mathbf{J})$ to give

$$(\mathbf{M}_{11}+\mathbf{J})\ddot{\bar{\theta}} = -f_1(\bar{\theta},\dot{\bar{\theta}}) + \bar{u} \tag{5}$$

Incorporating a disturbance term P_d to account for the unmodelled dynamics due to the neglected high frequency modes and higher manifold terms, the slow subsystem can be rewritten as: $\mathbf{M}_s\ddot{\bar{\theta}} + \mathbf{C}_s(\bar{\theta},\dot{\bar{\theta}})\dot{\bar{\theta}} + P_d = \bar{u}$

(6)

where $\mathbf{M}_s = \mathbf{M}_{11}(\theta,0) + \mathbf{J}$ $\mathbf{C}_s(\bar{\theta},\dot{\bar{\theta}})\dot{\bar{\theta}} = f_1(\bar{\theta},\dot{\bar{\theta}})$.

Let $\theta_d(t) \in \Re^n$ be a desired trajectory, which is assumed to be at least twice differentiable. Then consider a trajectory tracking error defined as $e(t) = \theta_d(t) - \bar{\theta}(t)$

(7)

Therefore, the filtered tracking error becomes

$$e_f(t) = \dot{e}(t) + \Lambda e(t) \tag{8}$$

where Λ is a symmetric positive definite constant matrix. Using the filtered error from (8), the slow dynamics given in (6) can be rewritten as:

$$\mathbf{M}_s\dot{e}_f = -\mathbf{C}_s e_f - \bar{u} + f(x) + P_d \tag{9}$$

where $f(x)$ is the non-linear function (dynamics of the slow subsystem) given by

$$f(x) = \mathbf{M}_s(\ddot{\theta}_d + \Lambda\dot{e}) + \mathbf{C}_s(\dot{\theta}_d + \Lambda e) + P_d \tag{10}$$

with $x = \begin{bmatrix} e^T & \dot{e}^T & \theta_d^T & \dot{\theta}_d^T & \ddot{\theta}_d^T \end{bmatrix}^T$.

The unknown function $f(x)$ can be approximated by applying a three-layer NN such that [3]:

$$f(x) = \mathbf{W}^T a(\mathbf{V}^T x) + \Xi \tag{11}$$

where $a(x)$ is a sigmoidal activation function, $\mathbf{W}$ and $\mathbf{V}$ are respectively the ideal connection weights for the *input layer to hidden layer* and *hidden layer to output layer* and Ξ is the function approximation error.

The NN estimate of $f(x)$ is:

$$\hat{f}(x) = \hat{\mathbf{W}}^T a(\hat{\mathbf{V}}^T x) \tag{12}$$

where $\hat{\mathbf{V}}$ and $\hat{\mathbf{W}}$ are the actual NN weights. Now, define a control input vector for the slow dynamics based on the function approximation as:

$$\bar{u}(t) = \hat{f}(x) + \mathbf{K_D} e_f - u_r(t) \tag{13}$$

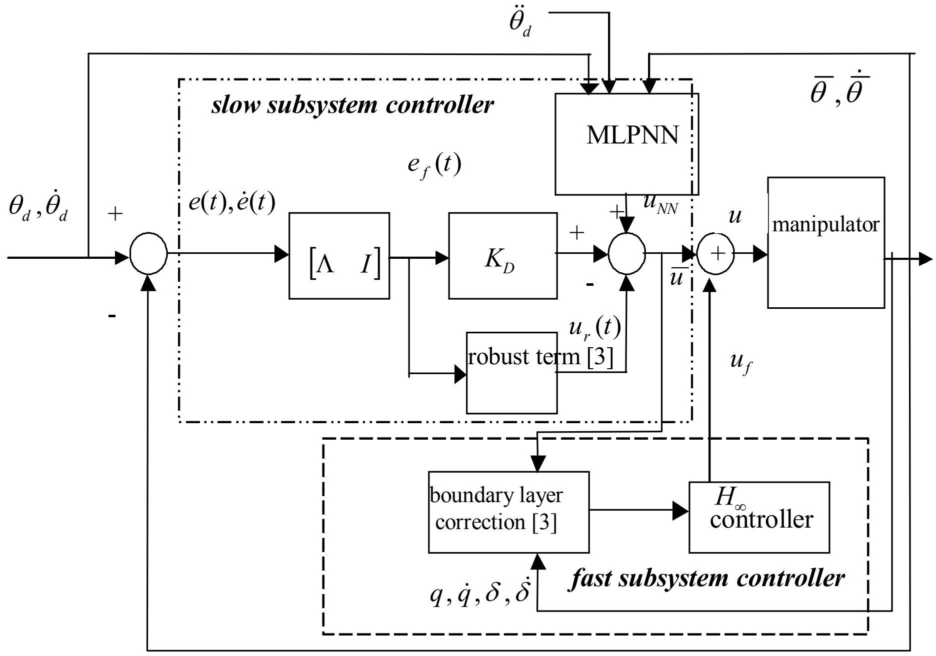

Fig.1 Structure of the neuro-H_∞ controller

where $\mathbf{K_D}$ is a positive gain matrix and $\boldsymbol{u}_r(t)$ provides robustness in the face of higher-order terms in the Taylor series. Substituting for $\hat{f}$ from (12) in (13) gives

$$\overline{\boldsymbol{u}} = \hat{\mathbf{W}}^{\mathrm{T}}\boldsymbol{a}(\hat{\mathbf{V}}^{\mathrm{T}}\boldsymbol{x}) + \mathbf{K_D}\boldsymbol{e}_f - \boldsymbol{u}_r(t) \qquad (14)$$

Substituting (14) in (9), the inner slow control system becomes

$$\mathbf{M}_s\dot{\boldsymbol{e}}_f(\boldsymbol{t}) = -(\mathbf{K_D} + \mathbf{C}_s)\boldsymbol{e}_f(\boldsymbol{t}) + \mathbf{W}^{\mathrm{T}}\boldsymbol{a}(\hat{\mathbf{V}}\boldsymbol{x}) - \hat{\boldsymbol{W}}^T\boldsymbol{a}(\hat{\mathbf{V}}\boldsymbol{x}) + \Xi + \boldsymbol{P}_d + \boldsymbol{u}_r(\boldsymbol{t}) \qquad (15)$$

Some further manipulation yields:

$$\mathbf{M}_s\dot{\boldsymbol{e}}_f(t) = -(\mathbf{K_D} + \mathbf{C}_s)\boldsymbol{e}_f(t) + \tilde{\mathbf{W}}^{\mathrm{T}}\hat{\boldsymbol{a}} + \hat{\mathbf{W}}^{\mathrm{T}}\tilde{\boldsymbol{a}} + \tilde{\mathbf{W}}^{\mathrm{T}}\tilde{\boldsymbol{a}} + \boldsymbol{w}(t) + \boldsymbol{u}_r(t) \qquad (16)$$

where: $\tilde{\mathbf{W}} = \mathbf{W} - \hat{\mathbf{W}}$, $\hat{\boldsymbol{a}}$ is the hidden layer output error given by [3]: $\tilde{\boldsymbol{a}} = \boldsymbol{a} - \hat{\boldsymbol{a}} = \boldsymbol{a}(\mathbf{V}^{\mathrm{T}}\boldsymbol{x}) - \boldsymbol{a}(\hat{\mathbf{V}}^{\mathrm{T}}\boldsymbol{x})$ and $\boldsymbol{w}(t) = \tilde{\mathbf{W}}^{\mathrm{T}}\hat{\boldsymbol{a}}'\tilde{\mathbf{V}}^{\mathrm{T}}\boldsymbol{x} + \mathbf{W}^{\mathrm{T}}O(\tilde{\mathbf{V}}^{\mathrm{T}}\boldsymbol{x})^2 + (\Xi + \boldsymbol{P}_d)$ which is bounded by

$$\boldsymbol{w}(t) \le \mathrm{c}_0 + \mathrm{c}_1\|\tilde{\mathbf{Z}}\| + \mathrm{c}_2\|\boldsymbol{x}\|\|\mathbf{Z}\| \qquad (17)$$

c_0, c_1 and c_2 denote positive constants. The tuning algorithm for the weights of the NN used to give slow control action is an unsupervised back propagation through time scheme with zero initial weights and no off-line learning phase. Control action is performed by the PD loop to keep the system stable until the NN begins to learn. The weights are tuned on-line in real-time as the system tracks the desired trajectory. The tracking performance improves as the NN learns $f(x)$. $\boldsymbol{u}_r$ is chosen as

$$\boldsymbol{u}_r = -\mathbf{K}_z(\|\hat{\mathbf{Z}}\|_F + \mathbf{Z}_M)\boldsymbol{e}_f(\boldsymbol{t}) \qquad (18)$$

where $\mathbf{K}_z > c_2$. Consider the weights for the NN to be tuned on-line using the following adaptation algorithm as

$$\left.\begin{aligned} \dot{\hat{\mathbf{W}}}^{\mathrm{T}} &= \mathbf{G}_1\hat{\boldsymbol{a}}\boldsymbol{e}_f^T(t) - \mathbf{G}_1\hat{\boldsymbol{a}}'\mathbf{V}^{\mathrm{T}}\boldsymbol{x}\boldsymbol{e}_f^T(t) - k_d\mathbf{G}_1\|\boldsymbol{e}_f(t)\|\hat{\mathbf{W}} \\ \dot{\hat{\mathbf{V}}}^{\mathrm{T}} &= \mathbf{G}_2\boldsymbol{x}(\hat{\boldsymbol{a}}'\hat{\mathbf{W}}\boldsymbol{e}_f(t))^T - k_d\mathbf{G}_2\|\boldsymbol{e}_f(t)\|\hat{\mathbf{V}} \end{aligned}\right\} \qquad (19)$$

where G_1 and G_2 are constant positive diagonal matrices and $k_d > 0$ is a design parameter. Inputs to the NN consist of derivative position error, whereas the back-propagation law uses the error between the desired NN output and the actual NN output. It may be noted that $\{\dot{\mathbf{M}}_s - 2\mathbf{C}_s\}$ is skew-symmetric as in the case of rigid manipulators. Therefore, by using this symmetric property $\{\dot{\mathbf{M}}_s - 2\mathbf{C}_s\}$, it can be shown [4] that the stability of the resulting slow NN controller with the tuning rules (19) is guaranteed. We then explain the design of the $\boldsymbol{H}_\infty$ controller for fast subsystem in the following paragraphs. Referring [2], the singularly perturbed model equations can be defined as:

$$\left.\begin{aligned} \mathbf{A}_1(x_1,x_2,\varepsilon^2 y_1,\varepsilon y_2) &= -\mathbf{H}_{11}(x_1,\varepsilon^2 y_1)[f_1+g_1] - \mathbf{H}_{12}(x_1,\varepsilon^2 y_1)[f_2+g_2] \\ \mathbf{A}_2(x_1,x_2,\varepsilon^2 y_1,\varepsilon y_2) &= \mathbf{H}_{21}(x_1,\varepsilon^2 y_1)[f_1+g_1] - \mathbf{H}_{22}(x_1,\varepsilon^2 y_1)[f_2+g_2] \\ \mathbf{A}_3(x_1,\varepsilon^2 y_1) &= \mathbf{H}_{11}(x_1,\varepsilon^2 y_1)\tilde{\mathbf{K}}_s \\ \mathbf{A}_4(x_1,\varepsilon^2 y_1) &= \mathbf{H}_{21}(\mathrm{x}_1,\varepsilon^2\mathrm{y}_1)\tilde{\mathbf{K}}_s \\ \mathbf{A}_5(x_1,\varepsilon^2 y_1) &= \mathbf{H}_{12}(\mathrm{x}_1,\varepsilon^2\mathrm{y}_1)\tilde{\mathbf{K}}_w \\ \mathbf{A}_6(x_1,\varepsilon^2 y_1) &= \mathbf{H}_{22}(\mathrm{x}_1,\varepsilon^2\mathrm{y})\tilde{\mathbf{K}}_w \\ \mathbf{A}_7(x_1,\varepsilon^2 y_1) &= (\mathbf{H}_{11}(\mathrm{x}_1,\varepsilon^2\mathrm{y}) + \mathbf{J}^{-1})\tilde{\mathbf{K}}_w \end{aligned}\right\} \qquad (20)$$

Using (20) and incorporating the disturbance effects due to neglecting the higher order manifold terms, the augmented fast subsystem can be rewritten as

$$\dot{\boldsymbol{x}}_f = \mathbf{A}_f\boldsymbol{x}_f + \mathbf{B}_w\boldsymbol{w}_f + \mathbf{B}_f\boldsymbol{u}_f \qquad (21)$$

where the subsystem matrices are given by

$$\mathbf{A}_f = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{I} \\ -\mathbf{A}_6 & \mathbf{A}_4 & \mathbf{0} \\ \mathbf{A}_5 & \mathbf{A}_7 & \mathbf{0} \end{bmatrix}, \mathbf{B}_f = \begin{bmatrix} \boldsymbol{0} \\ \mathbf{J}^{-1} \end{bmatrix}; \quad \mathbf{B}_w = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{I} \\ \boldsymbol{\Delta}_1 & \boldsymbol{\Delta}_2 & \mathbf{0} \\ \boldsymbol{\Delta}_3 & \boldsymbol{\Delta}_4 & \mathbf{0} \end{bmatrix}; \text{ and}$$

the states are: $\boldsymbol{x}_f = [z_1^T \; z_3^T \; z_2^T \; z_4^T]^T$, the disturbance vector is $\boldsymbol{w}_f = [\boldsymbol{w}_1 \; \boldsymbol{w}_2 \; \cdot \; \boldsymbol{w}_n]$; $\Delta_1, \Delta_2, \Delta_3$ and Δ_4 denote the neglected higher manifold expansion contributions. Fig.2 gives the structure of the $\boldsymbol{H}_\infty$ - based fast subsystem controller, in which weighting functions $\mathbf{W}_t(s)$, $\mathbf{W}_s(s)$ are selected such that the output is immune to disturbances in the low frequency range and high frequency robustness is guaranteed [5].

$$\mathbf{G}_{fl}(\mathbf{s}) = \mathbf{G}_{fn}(\mathbf{s})(\mathbf{I} + \mathbf{G}_m(\mathbf{s})) \qquad (22)$$

where $\mathbf{G}_{fl}(s)$, $\mathbf{G}_{fn}(s)$ and $\mathbf{G}_m(s)$ are the transfer functions of the perturbed system, reduced order system and the multiplicative uncertainty respectively. Robust stability will be achieved if the following norm inequality holds

$$\|\mathbf{G}_m(\mathbf{s})\mathbf{T}(\mathbf{s})\|_\infty < 1 \qquad (23)$$

where $\mathbf{T(s)}$ is the transfer function from the disturbance $\boldsymbol{w}_f$ to the control input $\boldsymbol{u}_f$.

Typically, $\mathbf{W}_t(s)$ and

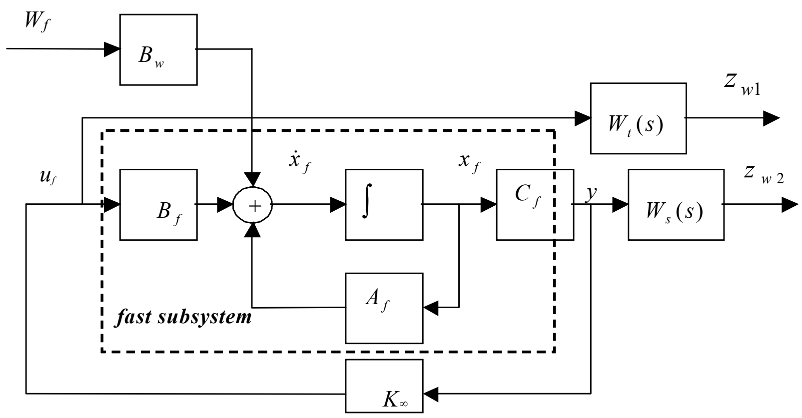

Fig.2 Structure of the fast controller

Due to difficulty in representing the uncertainty exactly, $\mathbf{W_t}(s)$ is selected to cover the upper bound of the uncertainty in the entire frequency range, i.e.

$$\bar{\sigma}(\mathbf{G_m}(j\omega)) \le \mathbf{W_t}(j\omega) \quad \forall\omega \tag{24}$$

where $\bar{\sigma}(\cdot)$ is the singular value, ω is the frequency. Therefore, (23) can be rewritten as

$$\|\mathbf{W_t(s)T(s)}\|_\infty < 1 \tag{25}$$

To improve the system performance such that the effects of the disturbances on the output are reduced, the controller must satisfy the following criteria

$$\|\mathbf{W_s(s)S(s)}\|_\infty < 1 \tag{26}$$

where $\mathbf{S}(s)$ is the transfer function between $\boldsymbol{w_f}$ and $\boldsymbol{y}$. The specifications expressed in (23) to (26) are achieved by designing a controller $\mathbf{K_\infty(s)}$ that satisfies the following mixed sensitivity criteria:

$$\left\|\begin{matrix}\mathbf{W_t(s)T(s)}\\ \mathbf{W_s(s)S(s)}\end{matrix}\right\|_\infty < 1 \tag{27}$$

If these weighting transfer functions are written in state-space form as: $\mathbf{W_t(s) = C_{wt}(sI - A_{wt})^{-1}}$ and $\mathbf{W_s(s) = C_{ws}(sI - A_{ws})^{-1}}$, then, referring to Fig.2, the augmented fast subsystem can be written as

$$\left.\begin{aligned}\dot{\boldsymbol{x}}_g &= \mathbf{A_g}\boldsymbol{x}_g + \mathbf{B_{g1}}\boldsymbol{w}_f + \mathbf{B_{g2}}\boldsymbol{u}_f\\ z &= \mathbf{C_{g1}}\boldsymbol{x}_g + \mathbf{D_{g12}}\boldsymbol{u}_f\\ \boldsymbol{y}_g &= \mathbf{C_{g2}}\boldsymbol{x}_g\end{aligned}\right\} \tag{28}$$

where the augmented state vector is: $\boldsymbol{x}_g = [\boldsymbol{x}_f \quad \boldsymbol{x}_{wt} \quad \boldsymbol{x}_{ws}]^T$, and the states of the augmented system are:

$$\mathbf{A_g} = \begin{bmatrix}\mathbf{A_f} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{A_{wt}} & \mathbf{0}\\ \mathbf{D_{ws}C_f} & \mathbf{0} & \mathbf{A_{ws}}\end{bmatrix}; \quad \mathbf{B_{g1}} = \begin{bmatrix}\mathbf{B_w}\\ \mathbf{0}\\ \mathbf{0}\end{bmatrix}; \quad \mathbf{B_{g2}} = \begin{bmatrix}\mathbf{B_f}\\ \mathbf{B_{wt}}\\ \mathbf{0}\end{bmatrix};$$

$$\mathbf{C_{g1}} = \begin{bmatrix}\mathbf{0} & \mathbf{C_{wt}} & \mathbf{0}\\ \mathbf{D_{ws}C_f} & \mathbf{0} & \mathbf{C_{ws}}\end{bmatrix}; \quad \mathbf{D_{g12}} = \begin{bmatrix}\mathbf{D_{wt}}\\ \mathbf{0}\end{bmatrix}.$$

The objective of the state feedback H_∞ controller is to find a constant gain matrix, $\mathbf{K_\infty}$ such that the state feedback control law $\boldsymbol{u}_f = \mathbf{K_\infty}\boldsymbol{x}_g$ (29) stabilizes the augmented uncertain linear system given in (28) and that $\|\mathbf{T_{zw}}\|_\infty \le 1$, where $\|\mathbf{T_{zw}(s)}\|_\infty$ is the closed loop transfer function matrix between $\boldsymbol{w}_f$ and $\boldsymbol{z}_w$ and the closed-loop fast subsystem matrix $(\mathbf{A_g - B_{g2}K_\infty})$ is stable. Therefore, to find a state feedback controller such that $\|\mathbf{T_{zw}(s)}\|_\infty \le 1$, it is necessary to find a positive definite matrix $\mathbf{P_\infty}$ that satisfies

$$\begin{aligned}&\mathbf{(A_g + B_{g2}K_\infty)^T P_\infty + P_\infty(A_g + B_{g2}K_\infty) + (C_{g1} + D_{g12}K_\infty)^T}\\ &\mathbf{(C_{g1} + D_{g12}K_\infty) + P_\infty B_{g1}B_{g1}^T P_\infty = 0}\end{aligned} \tag{30}$$

Defining $\mathbf{\Pi = D_{12}^T D_{12}}$ and $\mathbf{\Sigma = D_{12}^T C_{g1} + B_{g2}P_\infty}$ and substituting these in (30):

$$\begin{aligned}&\mathbf{(A_g^T P_\infty + P_\infty A_g + C_{g1}{}^T C_{g1} - \Sigma^{-1}\Pi\Sigma + P_\infty B_{g1}B_{g1}^T P_\infty}\\ &\mathbf{+ (K_\infty + \Sigma^{-1}\Pi)^T \Pi(K_\infty + \Sigma\Pi) = 0}\end{aligned} \tag{31}$$

Now assuming $\mathbf{K_\infty = -\Pi^{-1}\Sigma}$ in (31):

$$\begin{aligned}&\mathbf{(A_g^T P_\infty + P_\infty A_g + C_{g1}{}^T C_{g1} - \Sigma^{-1}\Pi\Sigma}\\ &\mathbf{+ P_\infty B_{g1}B_{g1}^T P_\infty = 0}\end{aligned} \tag{32}$$

Then the state feedback controller is given by

$$\boldsymbol{u}_f = \mathbf{K_\infty}\boldsymbol{x}_g \quad \text{where } \mathbf{K_\infty = -B_{g2}^T P_\infty} \tag{33}$$

Let $\boldsymbol{x}_m$ be the vector of available states, which can be written as a linear combination of the augmented state variables $\boldsymbol{x}_g$ as $x_m = \mathbf{L}x_g$ (34)

where $\mathbf{L}$ is a constant matrix. Let the control input in terms of $\boldsymbol{x}_m$ be expressed as $\boldsymbol{u}_f = \mathbf{K_m}\boldsymbol{x}_m$ (35)

Substituting x_m from (34) in (35) gives $\boldsymbol{u}_f = \mathbf{K_m}\boldsymbol{L}\boldsymbol{x}_g$ (36)

But the control based on the full state feedback is

$$\boldsymbol{u}_f = \mathbf{K_\infty}\boldsymbol{x}_g \tag{37}$$

Therefore, using (35,37), the gains $\mathbf{K_m}$ can be computed from the full state gain matrix by minimising the matrix norm $\|\mathbf{K_\infty - K_m L})^{-1}\|_\infty$ to give a mean square solution given by

$$\mathbf{K_m = K_\infty L^T(LL^T)^{-1}} \tag{38}$$

VI. RESULTS AND DISCUSSION

Simulations were performed to compare the performance of the new SNHC algorithm with the alternative singular-perturbation-based inverse dynamics and linear quadratic regulator composite controller (SCLC) reported in [3] when applied to a manipulator with two flexible-links and two flexible-joints having the parameters as specified in [3]. The manipulator was commanded to follow a desired trajectory given by $\boldsymbol{\theta_d}(t) = \boldsymbol{\theta_0}(t) + (6\frac{t^5}{t_d^5} - 15\frac{t^4}{t_d^4} + 10\frac{t^3}{t_d^3})(\boldsymbol{\theta_f} - \boldsymbol{\theta_0})$; where $\boldsymbol{\theta_d}(t) = [\theta_{d1}(t) \quad \theta_{d2}(t)]^T$ are the desired link trajectories, $\boldsymbol{\theta_0} = [0 \quad 0]^T$ are the initial link positions, $\boldsymbol{\theta_f} = \begin{bmatrix}\frac{\pi}{2} & \frac{\pi}{6}\end{bmatrix}^T$ are the final positions, t_d is the time taken along the trajectory to reach the final position which is taken as 4 seconds.

For the slow system NN controller, 10 input neurons were used corresponding to $\boldsymbol{x} = [\boldsymbol{e}_{2\times1} \quad \dot{\boldsymbol{e}}_{2\times1} \quad \boldsymbol{\theta}_{d\,2\times1} \quad \dot{\boldsymbol{\theta}}_{d\,2\times1} \quad \ddot{\boldsymbol{\theta}}_{d\,2\times1}]^T$. The output layer consisted of 2 nodes for two control signals, and 10 hidden nodes were chosen. The controller parameters were set as $\mathbf{K_D} = diag(2.5, 2.5)$, $\mathbf{K_z} = diag(15.0, 15.0)$, $\mathbf{\Lambda} = diag(1.5, 1.5)$ and $\mathbf{Z_M} = 50.0$, these values having been chosen by trial and error to give small tracking errors. The weight tuning algorithm (40) was implemented by using a trapezoidal integration method with a step size of 1ms and with $\mathbf{G_1} = diag(12.0, 12.0)$, $k_d = 0.000001$ and $\mathbf{G_2} = diag(120, 120)$. For the fast system controller, the disturbance matrix elements Δ_i, $i = 1,2...4$ in $\mathbf{B_w}$ in (21) were set at 1% of the nominal values of the $\mathbf{A_f}$ matrix elements. In designing the fast controller, the reduced order model used comprised of one flexible link mode and one joint flexible mode for each link.

The neglected higher modes were considered to be an output multiplicative uncertainty whose transfer function was found using (22). The weighting matrix $\mathbf{W_t(s)}$ corresponding to two control inputs and referring to the singular plots was chosen as $\mathbf{W_t(s)} = diag(w_t(s), w_t(s))$, where $w_t(s) = \frac{25s^2}{(s+10)^2}$. Similarly, $\mathbf{W_s(s)}$ was selected as $\mathbf{W_s(s)} = diag(w_s(s), w_s(s), w_s(s), w_s(s))$, with $w_s(s) = \frac{5}{(s+0.01)}$. The full state feedback gains and the observer-based controller gains were determined with regard to these weighing functions. The performances of the SNHC and SCLC are compared in figures 4 to 8. Fig.4 shows that, although the initial tracking errors are bigger in the case of SNHC, these become more damped and decay faster after a small time, whereas significant errors persist in the case of SCLC. It is clear from the joint deflection trajectories shown in Fig 5 that SNHC yields smaller joint deflections and supresses them more quickly. The damping characteristics in Fig.6 shows that the first flexible mode for both links are less excited with SNHC, leading to smaller tip deflections (Fig.7). The other modes of vibration have been also effectively controlled by SNHC (which are not shown due to space limitations). Fig. 8 shows the control torque profiles generated by the two control schemes and it can be seen that, for both joints, the control torque magnitudes required are less for SNHC compared to SCLC.

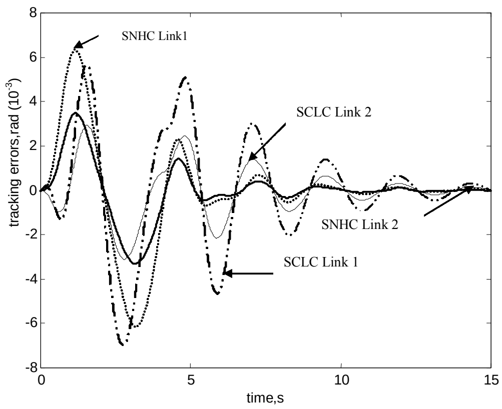

Fig. 4 Comparison of tracking performances

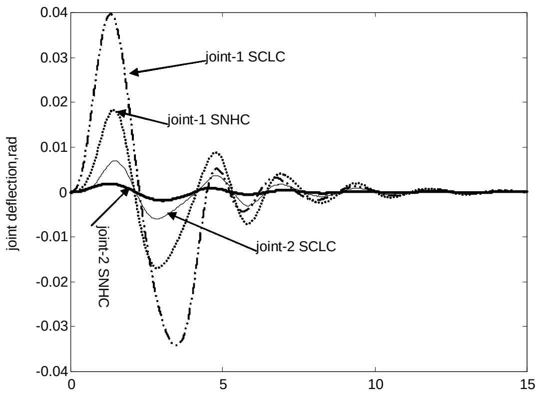

Fig.5 Comparison of joint deflections

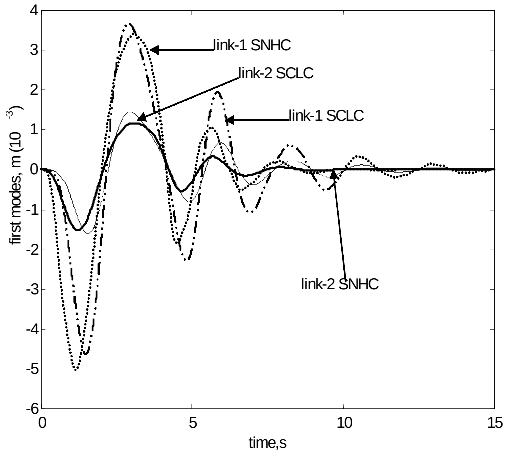

Fig.6 Comparison of first modal vibration

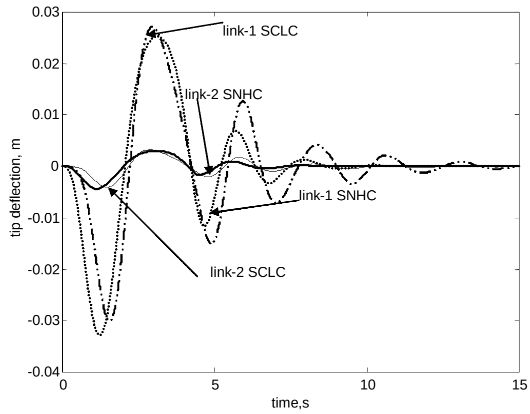

Fig.7 Comparison of tip deflection trajectories

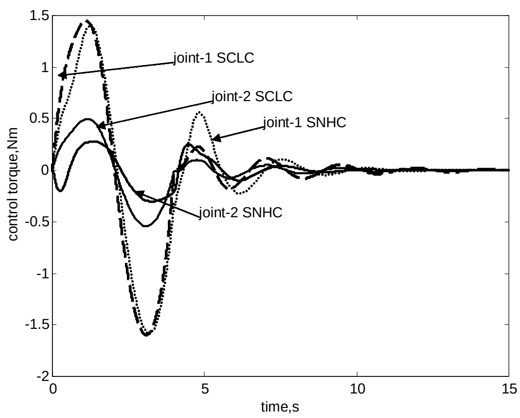

Fig.8 Comparison of control torque profiles

VII. CONCLUSIONS

The paper has described the development of a novel neuro-H∝ robust composite control scheme for a manipulator with flexible links and joints based on a two-time-scale singular perturbation model. This composite controller consists of a neural controller for the slow subsystem and a H_∞ controller for the fast subsystem. A neural controller has been employed for the control of the non-linear slow dynamics to overcome the model uncertainty, which may be due to the difficulty in achieving an exact time-scale separation. However, the fast dynamics being linear the uncertainty in model can be considered by the help of a robust controller. The By exploiting the singular perturbation technique, the full-order dynamics of the manipulator of dimension $[(2n+m)\times(2n+m)]$ has been reduced into two separate subsystems i.e. a slow subsystem of dimension $[(n)\times(n)]$ and a fast subsystem of dimension $[(n+m)\times(n+m)]$. Thus, this reduced- order modelling has facilitated in design of two reduced order controllers for the composite control scheme i.e. the composite control is generated by superposition of the slow and fast subsystem controllers. It is quite obvious that the complexity of the composite controller presented here is much less than any full-order controller. The new neuro-control scheme has been shown to perform better than an alternative inverse dynamics/LQR controller proposed earlier that was also based on a singular perturbation model. Improvement has been demonstrated both in trajectory tracking accuracy and also in the efficiency with which links and joint vibrations are suppressed. The stability of the resulting two-time-scale neuro- is ensured as both the slow and the fast subsystem controllers are stable, leading to the composite control also being stable. Also, by using a static $H\infty$ controller, the controller implementation is smooth and fast. Furthermore, the overall computational burden is greatly reduced by exploiting the two-time-scale separation of the complex dynamics of the flexible link and joint manipulator, as the product terms involving $(\boldsymbol{\theta},\dot{\boldsymbol{\theta}},\boldsymbol{q},\dot{\boldsymbol{q}})$ do not appear in either the slow or the fast control schemes. It has been observed from that the simulation of the complete model dynamics with this controller coded with 'C ' language takes 10 seconds for 500 iterations in an Intel Pentium-4 Computer with 1.0 GHZ frequency. Thus, the scheme is suitable for real-time control of such a manipulator.

REFERENCES

1. M.Moallem, K.Khorasani and R.Patel "An Integral Manifold Approach for Tip-Position Tracking of Flexible Multi-Link Manipulators" *IEEE Trans. Robotics and Autom,* vol.13, no.6, pp.823-865, 1997.
2. B.Subudhi and A.S.Morris, Dynamic modelling, simulation and control of a manipulator with flexible links and joints, *Robotics and Autonomous Systems,* vol.41, no.4, pp.257-270, 2002.
3. B.Subudhi and A.S.Morris "On the Singular Perturbation Approach to Control of a Multi-Link Manipulator with Flexible Links and Joints" *Proc. IMechE, J. Syst. Control Eng.*, vol.215, no.16, pp. 587-598, 2001.
4. F.L.Lewis, K.Liu and A.Yesildirek "Neural Net Robot Controller with Guaranteed Tracking Performance" *IEEE Trans. Neural Networks,* vol.6, no.3, pp.703-715, 1995.
5. R.N.Banavar and P.Dominic "An LQG/H_∞ Controller for a Flexible Manipulator" *IEEE Trans. Control Syst. Technol,* vol.3, no.4, pp.409-416, 1995.