Proceedings of the
47th IEEE Conference on Decision and Control
Cancun, Mexico, Dec. 9-11, 2008



# Adaptive Coordination of Decentralized Controllers using a Centralized Neural Network

Bong-Jun Yang,
Guided Systems Technologies, Inc.
1603 Zack Hinton Pkwy South, McDonough, GA 30253

Anthony J. Calise, James I. Craig, Kilsoo Kim
School of Aerospace Engineering, Georgia Institute of Technology
Atlanta, GA 30332

*Abstract*— **An adaptive approach that augments existing decentralized linear controllers is considered. By employing a neural network as a centralized element, the approach greatly broadens the class of system for which linear decentralized controllers can be designed. The stability proof naturally follows from the viewpoint that a set of decentralized controllers are a special class of multi-input multi-output controllers of an existing central method. The approach is illustrated using an inverted flexible pendulum in which a neural network coordinates an acceleration controller with a controller for an rigid inverted pendulum.**

## I. Introduction

Recent advances in the technology of sensors and actuators allow for implementation of distributed set of inexpensive sensors and actuators for large-scales systems. This poses a challenge on designing a controller because most conventional control methods become proportionally complicated by the dimension of the system to be controlled. Therefore it is not practical to design a single controller because the design of a concurrent controller processing a distributed set of sensors and actuators is a formidable task. Moreover, if the system to be controlled is uncertain, the design of a single controller for a high dimensional system becomes less feasible in most control systems. However, we note that a major obstacle associated with a concurrent controller in a large-scale system does not lie in setting up multiple communication channels among many subsystems in hardware, but in the lack of an appropriate information processing algorithm that is numerically efficient. In practice, recent advances in microprocessors and signal processing make it possible for a single system board to handle multiple channels of inputs and outputs with less power consumption compared to the past, but control design methodology for systems having distributed arrays of sensors and actuators has not kept up with this technology.

B.-J. Yang is a Research Scientist at Guided System Technologies, Inc. Email: jun.yang@guidedsys.com

A. J. Calise is a Professor at Georgia Tech. Email: anthony.calise@ae.gatech.edu

J. I. Craig is a Professor at Georgia Tech., Email:james.craig@ae.gatech.edu

K. Kim is a graduate research assistant at Georgia Tech., Email:gtg229q@mail.gatech.edu

In this paper, we propose a centralized neural network (NN) as a tool for providing a hierarchy in control design to those systems equipped with a practically manageable size of distributed sensors and actuators. The control architecture proposed in this paper is depicted in Figure 1. Compared to previous NN-based decentralized approaches [1]–[4], it is immediately clear that the main difference is the usage of the distributed outputs as an NN input, and this process does not drastically increase the complexity of neural processing because of its inherently parallel nature of data processing. It is also clear that the architecture in Figure 1 is centralized only in a sense that the NN processes all the measurements while the baseline controllers can remain decentralized. The NN in Figure 1 can be equivalently realized as a set of NNs in each subsystem [5].

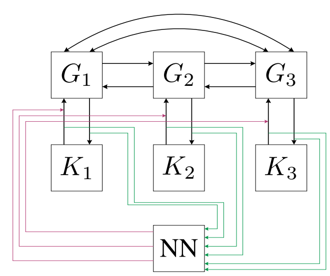



Fig. 1. Centralized NN control architecture

As a result of employing a centralized NN as a hierarchical controller for underlying decentralized controllers, we encompass a broader class of nonlinear systems compared to the classes in the literature [1]–[4]. Compared to the class in [4], implementing a centralized NN controller over a set of decentralized controllers allows for interconnections that cannot be approximated by a decentralized NN because of the requirement for observability. This further implies that the proposed architecture can accommodate a class of unmodeled dynamics which are prohibited in [4], in the same manner as in the centralized setting in [6].

The paper is organized as follows. In Section II we formulate a central, hierarchical NN control problem. Following the analysis of tracking error dynamics in Section III, we present the augmenting method of adaptive control design in

978-1-4244-3124-3/08/$25.00 ©2008 IEEE

SectionIV whose stability analysis is presented in Section V. In Section VI, we illustrate the proposed method in controlling a flexible inverted pendulum. Conclusions are given in Section VII. Throughout the manuscript, $\|\cdot\|$ means Euclidean norm for a vector and the induced 2-norm for a matrix unless otherwise mentioned. That is, $\|\boldsymbol{x}\| = \sqrt{\boldsymbol{x}^T\boldsymbol{x}}$ for $\boldsymbol{x} \in \mathbb{R}^n$, and $\|A\| = \sqrt{\lambda_{\max}(A^TA)}$ for $A \in \mathbb{R}^{m\times n}$. The Frobenius norm for a matrix is represented by the subscript $F$, i.e., $\|A\|_F = \sqrt{\mathrm{tr}(A^\top A)}$. The set $S|_k$ denotes a projection of the set $S$ to the $k$-dimensional Euclidian space $\mathbb{R}^k$. A vector is denoted by a bold symbol, and its $i$th element is represented by a plain symbol with a subscript $i$, i.e., $\boldsymbol{u} = [u_1, \ldots, u_i, \ldots]^\top$.

## II. Problem Formulation

Consider a system described by the following normal form

$$\begin{aligned}\dot{\xi}_{i_1} &= \xi_{i_2},\ \dot{\xi}_{i_2} = \xi_{i_3}, \ldots \\ \dot{\xi}_{i_{r_i}} &= \boldsymbol{a}_i^\top \boldsymbol{\xi}_i + \boldsymbol{p}_i^\top \boldsymbol{z}_i + h_i(\boldsymbol{x}, \boldsymbol{u}) \\ \dot{\boldsymbol{z}}_i &= F_{o_i}\boldsymbol{z}_i + G_{o_i}\boldsymbol{\xi}_i + \boldsymbol{g}_i(\boldsymbol{x}) \\ \dot{\boldsymbol{\eta}} &= \boldsymbol{f}_\eta(\boldsymbol{x}_p, \boldsymbol{\eta}) \\ y_i &= \xi_{i_1},\end{aligned} \tag{1}$$

where $\boldsymbol{x} = [(\boldsymbol{x}_p)^\top, \boldsymbol{\eta}]^\top \in \Omega_x \subset \mathbb{R}^n$, $\boldsymbol{x}_p = [\boldsymbol{x}_{p_1}, \ldots, \boldsymbol{x}_{p_m}]^\top \in \Omega_x|_{r+n_z}$, $\boldsymbol{x}_{p_i} = [\boldsymbol{\xi}_i^\top,\ \boldsymbol{z}_i^\top]^\top \in \Omega_x|_{r_i+n_{z_i}}$, which denotes the state of the $i$th subsystem, $\boldsymbol{\eta} \in \Omega_x|_{n-n_z-r}$ represents the state of unmodeled internal dynamics, i.e., $\boldsymbol{f}_\eta$ is unknown, smooth vector field, $\boldsymbol{u} \in \Omega_u \subset \mathbb{R}^m$ is the input, $y_i \in \Omega_x|_1 \in \mathbb{R}$, $i = 1, \ldots, m$ are the regulated output, $r_i$ represents the relative degrees of the output $y_i$, $r = r_1 + \ldots + r_m \le n$, $n_z = n_{z_1} + \ldots + n_{z_m}$. The sets $\Omega_x$ and $\Omega_u$ are open, and $(\mathbf{0}, \mathbf{0}) \in \Omega_x \times \Omega_u$. The function $h_i(\boldsymbol{x}, \boldsymbol{u})$ is a smooth partially known function ($h_i(\mathbf{0}, \mathbf{0}) = 0$), and $\boldsymbol{f}_\eta(\boldsymbol{x}_p, \boldsymbol{\eta})$ is a smooth partially known vector field ($\boldsymbol{f}_\eta(\mathbf{0}, \mathbf{0}) = \mathbf{0}$).

*Assumption 1:* The Jacobian $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{u}}(\boldsymbol{x}, \boldsymbol{u})$ is nonsingular for every $(\boldsymbol{x}, \boldsymbol{u}) \in \Omega_x \times \Omega_u$.

*Assumption 2:* For the system $\dot{\boldsymbol{\eta}} = \boldsymbol{f}_\eta(\mathbf{0}, \boldsymbol{\eta})$, there exists a continuously differentiable function $V_\eta(\boldsymbol{\eta})$ satisfying

$$\begin{aligned} c_1\|\boldsymbol{\eta}\|^2 \le V_\eta(&\boldsymbol{\eta}) \le c_2\|\boldsymbol{\eta}\|^2 \\ \dot{V}_\eta &\le -c_3\|\boldsymbol{\eta}\|^2 \\ \left\|\frac{\partial V}{\partial \boldsymbol{\eta}}\right\| &\le c_4\|\boldsymbol{\eta}\|, \end{aligned} \tag{2}$$

with some positive constants $c_i$'s, $i = 1, \ldots, 4$. Furthermore, the vector field $\boldsymbol{f}_\eta$ is Lipschitz in its arguments.
Under this assumption, we have

$$\begin{aligned}\dot{V}_\eta &= \frac{\partial V_\eta}{\partial \boldsymbol{\eta}}\boldsymbol{f}_\eta(\boldsymbol{x}_p, \boldsymbol{\eta}) \\ &= \frac{\partial V_\eta}{\partial \boldsymbol{\eta}}\boldsymbol{f}_\eta(\mathbf{0}, \boldsymbol{\eta}) + \frac{\partial V_\eta}{\partial \boldsymbol{\eta}}[\boldsymbol{f}_\eta(\boldsymbol{x}_p, \boldsymbol{\eta}) - \boldsymbol{f}_\eta(\mathbf{0}, \boldsymbol{\eta})] \\ &\le -c_3\|\boldsymbol{\eta}\|^2 + c_4c_5\|\boldsymbol{\eta}\|\|\boldsymbol{x}_p\|,\end{aligned} \tag{3}$$

where $c_5$ is the Lipschitz constant. This implies that with $\boldsymbol{x}_p$ as an input, the dynamics $\dot{\boldsymbol{\eta}} = \boldsymbol{f}_\eta(\boldsymbol{x}_p, \boldsymbol{\eta})$ in (1) are input-to-state stable [7].

A baseline controller for the $i$th subsystem is assumed to be designed using only the local output $y_i$ and neglecting the effect of the other control signals. As a result, it is assumed that the baseline controller for the $i$th subsystem is designed based on a single estimate $\lambda_i$ for the term in $h_i(\boldsymbol{x}, \boldsymbol{u})$. This induces the following modeling error:

$$h_i(\boldsymbol{x}, \boldsymbol{u}) = \lambda_i u_i + \Delta_i(\boldsymbol{x}, \boldsymbol{u}), \tag{4}$$

where $\Delta_i(\boldsymbol{x}, \boldsymbol{u}) = h_i(\boldsymbol{x}, \boldsymbol{u}) - \lambda_i u_i$. The resulting controller is described by

$$\begin{aligned}\dot{\boldsymbol{x}}_{c_i} &= A_{c_i}\boldsymbol{x}_{c_i} + \boldsymbol{b}_{c_i}(y_{d_i} - y_i),\ \boldsymbol{x}_{c_i} \in \mathbb{R}^{n_{c_i}} \\ u_{c_i} &= \boldsymbol{c}_{c_i}{}^\top \boldsymbol{x}_{c_i} + d_{c_i}(y_{d_i} - y_i),\end{aligned} \tag{5}$$

which regulates a linear model

$$\begin{aligned}\dot{\boldsymbol{x}}_{p_i} &= A_i\boldsymbol{x}_{p_i} + \boldsymbol{b}_i\lambda_i u_i \\ y_i &= \boldsymbol{c}_i{}^\top \boldsymbol{x}_{p_i},\end{aligned} \tag{6}$$

where $A_i = \begin{bmatrix} A_{o_i} & P_{o_i} \\ F_{o_i} & G_{o_i}\end{bmatrix}$, $\boldsymbol{b}_i = \begin{bmatrix} \boldsymbol{b}_{o_i} \\ \mathbf{0}_{n_{z_i}\times 1}\end{bmatrix}$, $\boldsymbol{c}_i = \begin{bmatrix} \boldsymbol{c}_{o_i} \\ \mathbf{0}_{n_{z_i}\times 1}\end{bmatrix}$, $A_{o_i} = \begin{bmatrix} 0 & 1 & \ldots & 0 \\ 0 & 0 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ a_{i_1} & a_{i_2} & \ldots & a_{i_{r_i}}\end{bmatrix}_{r_i\times r_i}$, $P_{o_i} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \ldots & \boldsymbol{p}_i\end{bmatrix}^\top_{r_i\times n_{z_i}}$, $\boldsymbol{b}_{o_i} = \begin{bmatrix} 0 & 0 & \ldots & 1\end{bmatrix}^\top_{r_i\times 1}$, $\boldsymbol{c}_{o_i} = \begin{bmatrix} 1 & 0 & \ldots & 0\end{bmatrix}^\top_{r_i\times 1}$. The linear model in (6) regulated by the controller in (5) constitutes a reference model that represents the desired behavior of the $i$th subsystem:

$$\begin{aligned}\dot{\boldsymbol{x}}_{m_i} &= \bar{A}_i\boldsymbol{x}_{m_i} + \bar{\boldsymbol{b}}_{d_i}y_{d_i} \\ y_{m_i} &= \bar{\boldsymbol{c}}_i^\top \boldsymbol{x}_{m_i}\end{aligned} \tag{7}$$

where $\boldsymbol{x}_{m_i} = [(\boldsymbol{x}_{p_i})^{m\top}, \boldsymbol{x}_{c_i}^\top]^\top \in \Omega_{x_{m_i}} \subset \mathbb{R}^{r_i+n_{z_i}+n_{c_i}}$ in which the superscript $m$ is introduced to denote the reference model, and $\bar{A}_i = \begin{bmatrix} A_i - \boldsymbol{b}_i\lambda_i d_{c_i}\boldsymbol{c}_i^\top & \boldsymbol{b}_i\lambda_i\boldsymbol{c}_{c_i}^\top \\ -\boldsymbol{b}_{c_i}\boldsymbol{c}_i^\top & A_{c_i}\end{bmatrix}$, $\bar{\boldsymbol{b}}_{d_i} = \begin{bmatrix} \boldsymbol{b}_i\lambda_i d_{c_i} \\ \boldsymbol{b}_{c_i}\end{bmatrix}$, $\bar{\boldsymbol{c}}_i = \begin{bmatrix} \boldsymbol{c}_i \\ \mathbf{0}_{n_{c_i}\times 1}\end{bmatrix}$.

Let the decentralized control signal be augmented by the adaptive signal

$$u_i = u_{c_i} - u_{ad_i}. \tag{8}$$

Applying the linear controller in (5) to the system in (1) leads to

$$\begin{aligned}\dot{\boldsymbol{x}}_{t_i} &= \bar{A}_i\boldsymbol{x}_{t_i} + \bar{\boldsymbol{b}}_{d_i}y_{d_i} + \bar{\boldsymbol{b}}_i[-\lambda_i u_{ad_i} \\ &\quad + \Delta_i(\boldsymbol{x}, \boldsymbol{u})] + B_i^g\boldsymbol{g}_i(\boldsymbol{x}) \\ \dot{\boldsymbol{\eta}} &= \boldsymbol{f}_\eta(\boldsymbol{x}_p, \boldsymbol{\eta}) \\ y_i &= \bar{\boldsymbol{c}}_i^\top \boldsymbol{x}_{t_i},\ i = 1, \ldots, m\end{aligned} \tag{9}$$

where $\boldsymbol{x}_{t_i} = [\boldsymbol{x}_{p_i}^\top\ \boldsymbol{x}_{c_i}^\top]^\top$, and $\bar{\boldsymbol{b}}_i = \begin{bmatrix} \boldsymbol{b}_i^\top & \mathbf{0}_{1\times n_{c_i}}\end{bmatrix}^\top$, $B_i^g = \begin{bmatrix} \mathbf{0}_{n_{z_i}\times r_i} & I_{n_{z_i}} & \mathbf{0}_{n_{z_i}\times n_{c_i}}\end{bmatrix}^\top$.

The control objective is to design an adaptive law for $\boldsymbol{u}_{ad}$, which is the output of a centralized neural network that processes all the available outputs, such that each output $y_i(t)$ to track the desired output $y_{d_i}(t)$ while all the closed-loop signals are bounded. Our method employs a NN to cancel the effect of the modeling error $\Delta(\boldsymbol{x},\boldsymbol{u})$. Whereas the output of the $i$th system is decoupled from the other control signals $u_j$ $(j \neq i)$ in [4], the centralized architecture in Figure 1 allows for coupling in the input matrix, and therefore the class treated in this paper is broader than that in [4]. Compared to [6], the proposed approach takes the path in [8] and does not resort to the contraction mapping assumption.

## III. Tracking Error Dynamics

Defining the tracking error as

$$\boldsymbol{e}_i = \boldsymbol{x}_{m_i} - \boldsymbol{x}_{t_i}, \tag{10}$$

leads to the following error dynamics for the $i$th subsystem

$$\begin{aligned} \dot{\boldsymbol{e}}_i =& \bar{A}_i \boldsymbol{e}_i + \bar{\boldsymbol{b}}_i[\lambda_i u_{ad_i} - \Delta_i(\boldsymbol{x},\boldsymbol{u})] - B_i^g \boldsymbol{g}_i(\boldsymbol{x}) \\ \dot{\boldsymbol{\eta}} =& \boldsymbol{f}_\eta(\boldsymbol{x}),\ \ \boldsymbol{s}_i = \bar{C}_i \boldsymbol{e}_i, \end{aligned} \tag{11}$$

where $\boldsymbol{s}_i := [y_{m_i} - y_i,\ \boldsymbol{x}_{c_i}^{m\top} - \boldsymbol{x}_{c_i}^\top]^\top$ represents available measurements, and hence $\bar{C}_i = \begin{bmatrix} \bar{\boldsymbol{c}}_i^\top & \boldsymbol{0} \\ \boldsymbol{0} & I_{n_{c_i}} \end{bmatrix}$. Since $\bar{A}_i$ is Hurwitz by design, there exist a $P_i = P_i^\top > 0$ such that for an arbitrary $Q_i > 0$, $\bar{A}_i^\top P_i + P_i \bar{A}_i + Q_i = 0$. By introducing $\boldsymbol{e} = [\boldsymbol{e}_1^\top,\ldots,\boldsymbol{e}_m^\top]^\top \in \mathbb{R}^{r+n_z+n_c}$, Eq. (11) can be written as

$$\begin{aligned} \dot{\boldsymbol{e}} =& \bar{A}\boldsymbol{e} + \bar{B}[\Lambda \boldsymbol{u}_{ad} - \boldsymbol{\Delta}] - B^g \boldsymbol{g}(\boldsymbol{x}) \\ \dot{\boldsymbol{\eta}} =& \boldsymbol{f}_\eta(\boldsymbol{x}), \end{aligned} \tag{12}$$

where $\bar{A} = \mathrm{diag}\{\bar{A}_1, \bar{A}_2, \ldots, \bar{A}_m\} \in \mathbb{R}^{(r+n_z+n_c)\times(r+n_z+n_c)}$, $\bar{B} = \mathrm{diag}\{\bar{\boldsymbol{b}}_1, \bar{\boldsymbol{b}}_2, \ldots, \bar{\boldsymbol{b}}_m\} \in \mathbb{R}^{(r+n_z+n_c)\times m}$, $B^g = \mathrm{diag}\{B_1^g, B_2^g, \ldots, B_m^g\} \in \mathbb{R}^{(r+n_z+n_c)\times n_z}$, $\Lambda = \mathrm{diag}\{\lambda_1, \lambda_2, \ldots, \lambda_m\} \in \mathbb{R}^{m\times m}$

*Assumption 3:* The unmatched uncertainty $\boldsymbol{g}(\boldsymbol{x})$ is bounded as follows:

$$\|\boldsymbol{g}(\boldsymbol{x})\| \le \alpha_p \|\boldsymbol{x}_p\| + \alpha_\eta \|\boldsymbol{\eta}\|,\ \alpha_p, \alpha_\eta \ge 0.$$

Note that the system in (12) has the same form as that in [6, Eq. (21)] except the diagonalized system matrix $\bar{A}$ because the linear controllers are decentralized. To address the fact that the uncertainty $\boldsymbol{\Delta}(\boldsymbol{x},\boldsymbol{u})$ depends on the control signals, in [6] the uncertainty is assumed to be a contraction mapping with respect to the adaptive signal $\boldsymbol{u}_{ad}$. In this paper, we follow the path in [8] and note that

$$\Lambda \boldsymbol{u}_{ad} - \boldsymbol{\Delta} = -\boldsymbol{h}(\boldsymbol{x}, \boldsymbol{u}_c - \boldsymbol{u}_{ad}) + \Lambda \boldsymbol{u}_c. \tag{13}$$

Since $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{u}_{ad}}(\boldsymbol{x}, \boldsymbol{u}_c - \boldsymbol{u}_{ad}) = -\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{u}}(\boldsymbol{x},\boldsymbol{u})$ is nonsingular by Assumption 1, applying the implicit function theorem as in [9] guarantees that there exists a smooth function $\boldsymbol{u}_{ad_*} = \boldsymbol{u}_{ad_*}(\boldsymbol{x},\boldsymbol{x}_c)$ such that

$$-\boldsymbol{h}(\boldsymbol{x}, \boldsymbol{u}_c(\boldsymbol{x}_c) - \boldsymbol{u}_{ad_*}) + \Lambda \boldsymbol{u}_c, = \boldsymbol{0} \tag{14}$$

for every $(\boldsymbol{x},\boldsymbol{x}_c) \in \Omega_x \times \Omega_{x_c}$. With the definition of $\boldsymbol{u}_{ad_*}$, Eq. (13) can be expressed as

$$\Lambda \boldsymbol{u}_{ad} - \boldsymbol{\Delta} = -\boldsymbol{h}(\boldsymbol{x}, \boldsymbol{u}_c - \boldsymbol{u}_{ad}) + \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{u}_c - \boldsymbol{u}_{ad_*}). \tag{15}$$

Unlike a single-input single-out system in [10], the mean value theorem in general does not hold in a multivariable function. Therefore, we follow the steps in [8] and define a mapping $f(\boldsymbol{x}_m, \boldsymbol{x}, \boldsymbol{u}) := \boldsymbol{e}^\top P \bar{B} \boldsymbol{h}(\boldsymbol{x},\boldsymbol{u})$, where $P := \mathrm{diag}\{P_1,\ldots,P_m\} \in \mathbb{R}^{(r+n_z)\times(r+n_z)}$. Then, the mapping becomes a scalar mapping $f : \Omega_{x_m} \times \Omega_x \times \Omega_u \to \mathbb{R}$, and we have

$$\begin{aligned} &\boldsymbol{e}^\top P \bar{B}(\Lambda \boldsymbol{u}_{ad} - \boldsymbol{\Delta}) \\ &= -f(\boldsymbol{x}_m, \boldsymbol{x}, \boldsymbol{u}_c - \boldsymbol{u}_{ad}) + f(\boldsymbol{x}_m, \boldsymbol{x}, \boldsymbol{u}_c - \boldsymbol{u}_{ad_*}) \\ &= -\left[\frac{\partial f}{\partial \boldsymbol{x}_m},\ \frac{\partial f}{\partial \boldsymbol{x}},\ \frac{\partial f}{\partial \boldsymbol{u}}\right]\Big|_{(\boldsymbol{x}_m,\boldsymbol{x},\bar{\boldsymbol{u}})} \begin{bmatrix} \boldsymbol{0}_{r+n_z+n_c} \\ \boldsymbol{0}_n \\ -\boldsymbol{u}_{ad} + \boldsymbol{u}_{ad_*} \end{bmatrix} \\ &= \boldsymbol{e}^\top P \bar{B} H(\bar{\boldsymbol{u}})[\boldsymbol{u}_{ad} - \boldsymbol{u}_{ad_*}] \end{aligned} \tag{16}$$

where $H(\bar{\boldsymbol{u}}) := \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{u}}\Big|_{(\boldsymbol{x},\bar{\boldsymbol{u}})}$ that is nonsingular by Assumption 1, and $\bar{\boldsymbol{u}} = \boldsymbol{u}_c - \theta \boldsymbol{u}_{ad} - (1-\theta)\boldsymbol{u}_{ad_*}$ for a constant $\theta \in [0,1]$. The above expression implies that as far as a Lyapunov candidate function of the form $\frac{1}{2}\boldsymbol{e}^\top P \boldsymbol{e}$ ( $P > 0$) is considered, the error dynamics in (12) can be treated as if the mean value theorem holds, i.e., $\dot{\boldsymbol{e}} = \bar{A}\boldsymbol{e} + \bar{B}H(\bar{\boldsymbol{u}})[\boldsymbol{u}_{ad} - \boldsymbol{u}_{ad_*}] - B^g \boldsymbol{g}(\boldsymbol{x})$ because both error dynamics lead to the same product term in a stability analysis. As in [8], we introduce the following assumption regarding $H(\bar{\boldsymbol{u}})$, which is fundamental in most nonlinear robust control methods [11] and further explained in Remark 1.

*Assumption 4:* The matrix $H(\bar{\boldsymbol{u}})$ can be decomposed as $H(\bar{\boldsymbol{u}}) = \hat{H}(I + \Delta H(\boldsymbol{x},\boldsymbol{u}))$ with a known nonsingular $\hat{H}$ and $0 \le \|\Delta H(\boldsymbol{x},\boldsymbol{u})\| \le b_\Delta < 1$ on $\Omega_x \times \Omega_u$.

## IV. Adaptive Control

A single hidden layer NN (SHLNN) is used to approximate $\boldsymbol{u}_{ad_*}(\boldsymbol{x},\boldsymbol{x}_c)$ in (14) because it is a universal approximator [12]. Since the arguments of $\boldsymbol{u}_{ad_*}$ include the unavailable states $\boldsymbol{x}$, we recall the main result in [13] that establishes universal approximation for an unknown function of the states and control in an *observable* system using sampled values of its input/output. Following [13], [14], for a given $\epsilon^* > 0$, $\boldsymbol{u}_{ad_*}$ is parametrized on the compact set $\mathcal{C}_{\bar{x}} := \Omega_x \times \Omega_{x_c}$ by

$$\boldsymbol{u}_{ad_*}(\boldsymbol{x},\boldsymbol{x}_c) = M^\top \boldsymbol{\sigma}(N^\top \boldsymbol{\mu}) + \varepsilon(\boldsymbol{\mu}),\ \ \|\varepsilon(\boldsymbol{\mu})\| \le \epsilon^*, \tag{17}$$

where $\varepsilon(\boldsymbol{\mu})$ is the NN reconstruction error and $\boldsymbol{\mu}$ is the network input vector

$$\begin{aligned} \boldsymbol{\mu}(t) &= [\ 1\quad \bar{\boldsymbol{u}}_d^\top(t)\quad \bar{\boldsymbol{y}}_d^\top(t)\quad \boldsymbol{x}_c(t)^\top\ ]^\top,\quad \|\boldsymbol{\mu}\| \le \mu^* \\ \bar{\boldsymbol{u}}_d^\top(t) &= [\boldsymbol{u}(t), \boldsymbol{u}(t-d), \ldots, \boldsymbol{u}(t-(n_1-r-1)d)]^\top \\ \bar{\boldsymbol{y}}_d^\top(t) &= [\boldsymbol{y}(t), \boldsymbol{y}(t-d), \ldots, \boldsymbol{y}(t-(n_1-1)d)]^\top, \end{aligned} \tag{18}$$

where $r$ is the relative degree, and $n_1$ is selected greater than or equal to the observability index of the system output $\boldsymbol{y}$ that guarantees a diffeormorphism between $\boldsymbol{x}$ and $\boldsymbol{Y} := [\boldsymbol{y}, \dot{\boldsymbol{y}}(t), \ldots, \boldsymbol{y}^{(n_1-1)}(t)]^\top$. The constant $d > 0$ is a time delay, and $\boldsymbol{\sigma}$ is a vector of squashing functions $\boldsymbol{\sigma}(\cdot)$, its $i^{th}$ element being defined like $\left[\boldsymbol{\sigma}(N^\top \boldsymbol{\mu})\right]_i = \boldsymbol{\sigma}\left[(N^\top \boldsymbol{\mu})_i\right]$.

*Assumption 5:* On the compact set $\mathcal{C}_{\bar{x}}$, the ideal NN weights $M, N$ are bounded, i.e., $\|M\|_F \le M^*$ and $\|N\|_F \le N^*$.

The adaptive signal $\boldsymbol{u}_{ad}$ is designed as

$$\boldsymbol{u}_{ad} = \widehat{M}(t)^\top \boldsymbol{\sigma}(\widehat{N}(t)^\top \boldsymbol{\mu}), \tag{19}$$

where $\widehat{M}(t), \widehat{N}(t)$ are the estimates for $M, N$ in (17) and adapted on-line.

For an adaptation law for $\widehat{M}(t), \widehat{N}(t)$, we follow the path in [14] and introduce an error observer. For this, we note that when $\boldsymbol{u}_{ad} = \boldsymbol{u}_{ad_*}$, $\Lambda \boldsymbol{u}_{ad} = \boldsymbol{\Delta}$ by (15), and hence $\lambda_i u_{ad_i} - \Delta_i(\boldsymbol{x}, \boldsymbol{u}) = 0$ for $i = 1, \ldots, m$. Therefore, we design the following linear observer for $\boldsymbol{e}_i$ [14]: $\dot{\hat{\boldsymbol{e}}}_i = \bar{A}_i \hat{\boldsymbol{e}}_i + K_i(\boldsymbol{s}_i - \hat{\boldsymbol{s}}_i)$, where $K_i$ is selected such that $\tilde{A}_i := \bar{A}_i - K_i \bar{C}_i$ is Hurwitz. Let $\tilde{\boldsymbol{e}}_i = \hat{\boldsymbol{e}}_i - \boldsymbol{e}_i$. The the observation error dynamics are described by $\dot{\tilde{\boldsymbol{e}}}_i = \tilde{A}_i \tilde{\boldsymbol{e}}_i - \bar{\boldsymbol{b}}_i[\lambda_i u_{ad_i} - \Delta_i(\boldsymbol{x}, \boldsymbol{u})] + B^g_i \boldsymbol{g}_i(\boldsymbol{x})$. Since $\tilde{A}_i$ is Hurwitz, for a $\tilde{Q}_i > 0$, there exists a $\tilde{P} > 0$ such that $\tilde{A}_i^\top \tilde{P}_i + \tilde{P}_i \tilde{A}_i^\top + \tilde{Q}_i = 0$. With the notation $\tilde{\boldsymbol{e}} = [\tilde{\boldsymbol{e}}_1^\top, \ldots, \tilde{\boldsymbol{e}}_m^\top]^\top$, the overall observation error dynamics are written as

$$\dot{\tilde{\boldsymbol{e}}} = \tilde{A}\tilde{\boldsymbol{e}} - \bar{B}[\Lambda \boldsymbol{u}_{ad} - \boldsymbol{\Delta}] + B^g \boldsymbol{g}(\boldsymbol{x}). \tag{20}$$

The NN weights $\widehat{M}(t),\ \widehat{N}(t)$ are updated according to the following adaptation laws [14]

$$\begin{aligned} \dot{\widehat{M}} &= -\Gamma_M[(\hat{\boldsymbol{\sigma}} - \hat{\boldsymbol{\sigma}}'\widehat{N}^\top \boldsymbol{\mu})\boldsymbol{r} + k\widehat{M}], \\ \dot{\widehat{N}} &= -\Gamma_N[\boldsymbol{\mu}\boldsymbol{r}\widehat{M}^\top \hat{\boldsymbol{\sigma}}' + k\widehat{N}], \end{aligned} \tag{21}$$

in which $\Gamma_M,\ \Gamma_N > 0$ are positive definite adaptation gain matrices, $k > 0$ is a $\sigma-$modification constant, $\hat{\boldsymbol{\sigma}} \triangleq \boldsymbol{\sigma}(\widehat{N}^\top \boldsymbol{\eta})$, $\hat{\sigma}'$ is the Jacobian computed at the estimates: $\hat{\sigma}' = \sigma'(\widehat{N}^\top \boldsymbol{\eta})$, and the training signal is given by

$$\boldsymbol{r} = \boldsymbol{e}^\top P \bar{B} \hat{H} \tag{22}$$

*Remark 1:* In a completely decentralized setting, $\boldsymbol{r} = [r_1, \ldots, r_m]^\top$ where $r_i = \hat{\boldsymbol{e}}_i^\top P_i \boldsymbol{b}_i \lambda_i$. Therefore, compared to the decentralized approach, the centralized neural controller is beneficial in two aspects: 1) the approach effectively accommodates couplings in the control input matrix in regulating all the outputs 2) the observability is enhanced by processing all the outputs. As a price,the control input matrix is required to meet Assumption 4. In other words, addressing the coupling effects require an estimate for the couplings in the input channels. This estimate should be made such that the nominal control action should not be overcome by uncertain modeling error ($\|\Delta H(\boldsymbol{x}, \boldsymbol{u})\| \leq b_\Delta < 1$), which has been a fundamental requirement in nonlinear robust control literature [7], [11]. In implementation aspects, NNs have an inherent parallel structure, and numerical complexity does not increase significantly compared to the decentralized NN architecture. As a matter of fact, a single centralized NN can be decomposed as multiple NNs that processes the same input [5].

*Remark 2:* Our analysis shows that when the control system, including a NN, is fully decentralized, the decentralized control system is valid if $\|\Delta H\| = \|H(\bar{\boldsymbol{u}}) - \Lambda\| < 1$, and each output is observable with respect to the entire system. That is, if the coupling effect among control signals is not dominant, the validity of the decentralized controllers is still guaranteed with an each observable output.

## V. Stability Analysis

Define $\widetilde{M} \triangleq \widehat{M} - M$, $\widetilde{N} \triangleq \widehat{N} - N$, $\widetilde{Z} \triangleq \begin{bmatrix} \widetilde{M} & 0 \\ 0 & \widetilde{N} \end{bmatrix}$, where $M, N$ are ideal weights defined in (17). The term $\boldsymbol{u}_{ad} - \boldsymbol{u}_{ad_*}$ allows for the following upper bound [14] $\|\boldsymbol{u}_{ad} - \boldsymbol{u}_{ad_*}\| \leq \delta_1 \left\|\widetilde{Z}\right\|_F + \delta_2, \quad \delta_1, \delta_2 > 0$. The NN approximation error $\boldsymbol{u}_{ad} - \boldsymbol{u}_{ad_*}$ can, using Taylor series expansion, be described as follows $\boldsymbol{u}_{ad} - \boldsymbol{u}_{ad_*} = \widetilde{M}(\hat{\boldsymbol{\sigma}} - \hat{\boldsymbol{\sigma}}'\widehat{N}^\top \boldsymbol{\mu}) + \widehat{M}^\top \hat{\boldsymbol{\sigma}}'\widetilde{N}^\top \boldsymbol{\mu} + \boldsymbol{\omega} - \boldsymbol{\varepsilon}$, where $\boldsymbol{\omega} = \widetilde{M}^\top \hat{\boldsymbol{\sigma}}' N^\top \boldsymbol{\mu} - M^\top \mathcal{O}((\widetilde{N}^\top \boldsymbol{\mu})^2)$ with $\mathcal{O}((\widetilde{N}^\top \boldsymbol{\mu})^2)$ as higher order terms. Using the bound for $\boldsymbol{\mu}$ in (18), the term $\boldsymbol{\omega} - \boldsymbol{\varepsilon}$ can be bounded [14] $\|\boldsymbol{\omega} - \boldsymbol{\varepsilon}\| \leq \delta_3 \left\|\widetilde{Z}\right\|_F + \delta_4, \quad \delta_3, \delta_4 > 0$.

For a stability analysis, consider the following Lyapunov candidate function

$$\begin{aligned} V(\boldsymbol{e}, \tilde{\boldsymbol{e}}, \widetilde{M}, \widetilde{N}) &= \boldsymbol{e}^\top P \boldsymbol{e} + \tilde{\boldsymbol{e}}^\top \tilde{P} \tilde{\boldsymbol{e}} \\ &+ \operatorname{tr}\left(\widetilde{M}^\top \Gamma_M^{-1} \widetilde{M}\right) + \operatorname{tr}\left(\widetilde{N}^\top \Gamma_N^{-1} \widetilde{N}\right) + V_\eta(\boldsymbol{\eta}). \end{aligned} \tag{23}$$

The time derivative of $V$ along (12) and (20) becomes

$$\begin{aligned} \dot{V} &= 2\boldsymbol{e}^\top P \left[\bar{A}\boldsymbol{e} + \bar{B}[\Lambda \boldsymbol{u}_{ad} - \boldsymbol{\Delta}] - B^g \boldsymbol{g}(\boldsymbol{x})\right] \\ &+ 2\tilde{\boldsymbol{e}}^\top \tilde{P}\left[\tilde{A}\tilde{\boldsymbol{e}} - \bar{B}[\Lambda \boldsymbol{u}_{ad} - \boldsymbol{\Delta}] + B^g \boldsymbol{g}(\boldsymbol{x})\right] \\ &+ \operatorname{tr}\left(\widetilde{M}^T \Gamma_M^{-1} \dot{\widehat{M}}\right) + \operatorname{tr}\left(\widetilde{N}^T \Gamma_N^{-1} \dot{\widehat{N}}\right) + \dot{V}_\eta. \end{aligned} \tag{24}$$

Applying the mean value theorem as in (16) leads to

$$\begin{aligned} \dot{V} &= -\boldsymbol{e}^\top Q \boldsymbol{e} + 2\boldsymbol{e}^\top P \bar{B} H(\bar{\boldsymbol{u}})[\boldsymbol{u}_{ad} - \boldsymbol{u}_{ad_*}] \\ &- \tilde{\boldsymbol{e}}^\top \tilde{Q} \tilde{\boldsymbol{e}} - 2\tilde{\boldsymbol{e}}^\top \tilde{P} \bar{B} H(\bar{\boldsymbol{u}})[\boldsymbol{u}_{ad} - \boldsymbol{u}_{ad_*}] + \ldots \end{aligned} \tag{25}$$

Using Assumption 4, the product term $2\boldsymbol{e}^\top P \bar{B} H(\bar{\boldsymbol{u}})[\boldsymbol{u}_{ad} - \boldsymbol{u}_{ad_*}]$ can be arranged as

$$\begin{aligned} &2\boldsymbol{e}^\top P \bar{B} H(\bar{\boldsymbol{u}})[\boldsymbol{u}_{ad} - \boldsymbol{u}_{ad_*}] \\ &= 2(\hat{\boldsymbol{e}} - \tilde{\boldsymbol{e}})^\top P \bar{B} H(\bar{\boldsymbol{u}})[\boldsymbol{u}_{ad} - \boldsymbol{u}_{ad_*}] \\ &= 2\hat{\boldsymbol{e}}^\top P \bar{B} \hat{H}(I + \Delta H)[\boldsymbol{u}_{ad} - \boldsymbol{u}_{ad_*}] \\ &- 2\tilde{\boldsymbol{e}}^\top P \bar{B} H(\bar{\boldsymbol{u}})[\boldsymbol{u}_{ad} - \boldsymbol{u}_{ad_*}] \\ &= 2\hat{\boldsymbol{e}}^\top P \bar{B} \hat{H}[\widetilde{M}(\hat{\boldsymbol{\sigma}} - \hat{\boldsymbol{\sigma}}'\widehat{N}^\top \boldsymbol{\mu}) + \widehat{M}^\top \hat{\boldsymbol{\sigma}}'\widetilde{N}^\top \boldsymbol{\mu} \\ &+ \boldsymbol{\omega} - \boldsymbol{\varepsilon}] + 2\hat{\boldsymbol{e}}^\top P \bar{B} \hat{H} \Delta H[\boldsymbol{u}_{ad} - \boldsymbol{u}_{ad_*}] \\ &- 2\tilde{\boldsymbol{e}}^\top P \bar{B} H(\bar{\boldsymbol{u}})[\boldsymbol{u}_{ad} - \boldsymbol{u}_{ad_*}]. \end{aligned} \tag{26}$$

Note that Eq. (25) is the same as that in [14] except the presence of $H(\bar{\boldsymbol{u}})$ and $V_\eta$. Since $\|\Delta H\| \leq b_\Delta < 1$, and hence $\|H(\bar{\boldsymbol{u}})\| \leq \left\|\hat{H}\right\|(1 + b_\Delta)$, the product terms associated with $H(\bar{\boldsymbol{u}})$ and $\Delta H$ can be treated using upper bounds similar to those in [14]. For example, $|\tilde{\boldsymbol{e}}^\top P \bar{B} H(\bar{\boldsymbol{u}})[\boldsymbol{u}_{ad} - \boldsymbol{u}_{ad_*}]| \leq \left\|P\bar{B}\right\|(1 + b_\Delta)\|\boldsymbol{e}\|[\delta_1 \left\|\widetilde{Z}\right\|_F + \delta_2] \leq \gamma_1 \|\boldsymbol{e}\| \left\|\widetilde{Z}\right\|_F + \gamma_2 \|\boldsymbol{e}\|$, where $\gamma_1 = \delta_1 \left\|P\bar{B}\right\|(1 + b_\Delta)$, and $\gamma_2 = \delta_2 \left\|P\bar{B}\right\|(1 + b_\Delta)$. This means that the stability directly follows from the results in [14] and [6]. Therefore, we state the main result of the paper without a proof.

*Theorem 1:* Suppose that the system satisfies Assumptions 1-5. The feedback control law in (8) with the adaptive signal in (19), together with the NN weights updated by (21), guarantees that the signals $\boldsymbol{e}, \tilde{\boldsymbol{e}}, \widetilde{M}, \widetilde{N}$, and $\boldsymbol{\eta}$ in the closed-loop system are ultimately bounded.

*Remark 3:* The rationale in this section is to view the decentralized linear controllers in (5) as a subclass of centralized controllers in [6]. Therefore, the stability proof is the same except that a SHLNN is used while a radial basis function NN is used in [6]. A main feature that distinguishes the control design from that in [6] is the coupling terms in the input matrix that are treated by (26).

*Remark 4:* The decentralized nature of the proposed architecture allows for straightforward addition of an additional control system when a new set of sensors and actuators are deployed; A local controller for a new pair is designed and is coordinated by a NN that processes all the input and outputs. The prerequisite for this procedure is to obtain a new $\hat{H}$ that includes additional elements arising from added sensor/actuator pairs.

## VI. Simulations

The notations used to derive the equation of motion in this section are independent from the other section, which should be clear in the context.

Consider a flexible inverted pendulum in Figure 2. The inverted pendulum has been often used to study tasks associated with balancing such as stabilization for rocket thrusters [15]. Our motivation is to study the proposed architecture in suppressing vibrations caused by flexibilities in a slender launch vehicle and providing attitude stabilization during its ascent phase. In Figure 2, the tip mass is mounted to account for an additional payload carried by such slender launch vehicles. The flexible pendulum in Figure 2 consists

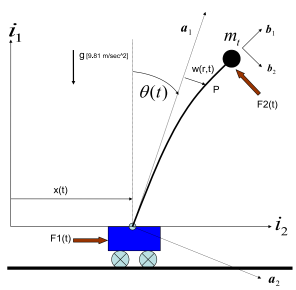



Fig. 2. A flexible inverted pendulum on a cart

of a motor driven cart, which is equipped with two encoders that measure the position of the cart ($x$) and the angle of the pendulum ($\theta$), which is free to swing at the side of the cart. Due to limited spaces, we do not include a complete system model, however, a similar model with detailed system properties can be found in [16].

The available measurements are :

$$y_1 = x, \; y_2 = \theta,$$
$$y_3 = \ddot{x}\cos(\theta+\beta) + (L\ddot{\theta}+\ddot{w})\cos\beta + (L\dot{\theta}+\dot{w})\dot{\theta}\sin\theta,$$

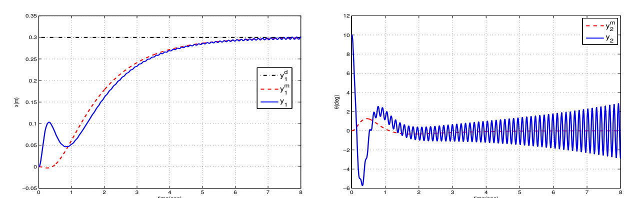
where $L$ is the length of the pendulum, $\beta(t) := \frac{\partial w}{\partial r}(L,t)$, and therefore $y_3$ represents the tip acceleration in the direction of $\boldsymbol{b}_2$ in Figure 2. The control forces include the external force applied to the cart $\boldsymbol{F}_1(t)$ and the external force $\boldsymbol{F}_2(t)$, which mimics a gas-jet thruster located at the payload in case of lanuch vehicles. They are realized by the control voltages according to the following relation:

$$\begin{aligned} \boldsymbol{F}_1 &= F_1\boldsymbol{i}_2 = (a_1u_1 - a_2\dot{x})\boldsymbol{i}_2, \\ \boldsymbol{F}_2 &= -F_2\boldsymbol{b}_2 = -k_a u_2 \boldsymbol{b}_2, \end{aligned} \tag{27}$$

where $a_1 = 1.72$, $a_2 = 7.68$ are due to the gear actuation mechanism [17], and $k_a = 1$ is a force gain.

The rationale for the decentralized controllers is as follows: design a controller for $u_1$ in the same manner as in controlling a rigid pendulum on the cart while $u_2$ is independently used to suppress vibrations due to flexibility. An immediate issue in this pursuit is that blindly suppressing the accelerations at the tip mass also suppresses accelerations due to the rigid body motion. Therefore, a reference model, composed of a linear pendulum model regulated by the controller for $u_1$, is utilized in the design of a controller for $u_2$. That is, the nominal controller for $u_2$ is designed only to suppress the deviation of the acceleration from that of the reference model, which is the nominal closed-loop system.

*Linear controller for $u_1$:* The nominal model for the system in Figure 2 is obtained by neglecting elasticity of the beam and then linearizing the pendulum equation with respect to the vertical equilibrium. Since the nominal model is the linearized model with respect to the vertical position, the nominal control design is carried out in the same manner as was done in [17]. A linear quadratic Gaussian (LQG) controller is designed for output tracking. Figure 3 compares

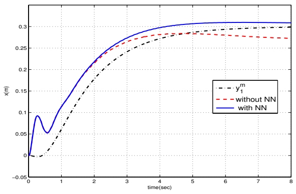


(a) the cart position $y_1(= x)$ (b) the pendulum angle at the root $y_2 = (\theta)$

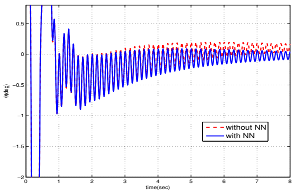
Fig. 3. Times responses of the reference model and the closed loop system

the responses of the reference model to the closed-loop responses when the LQG controller is applied to the system . Initially the pendulum is tilted with $10^\circ$ while all the other states are set to zero. While the reference model shows desired behavior for $(x, \theta)$, the system regulated by the LQG controller goes unstable. The instability is caused by the flexibility in the beam. The LQG controller fails to stabilize the inverted pendulum even for an arbitrarily small initial tilt of the pendulum. Without the flexibility, it can be proven that a domain of attraction exists, and the closed-loop linear system is exponentially stable.

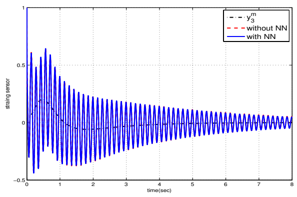
*Linear controller for $u_2$:* For acceleration feedback, an independent control law for $u_2$ is designed. Since the goal of the acceleration feedback is to suppress vibrations due to the flexibility of the pendulum, the desired profile for the acceleration, $y_{d_3}(t)$, is obtained by the reference model consisting of the plant model regulated by the LQG controller in Section VI, and hence $y_{d_3}(t) = y_{m_3}(t) = \ddot{x}_m(t) + L\ddot{\theta}_m(t)$. An issue related to acceleration feedback is that it has relative




(a) the cart position $y_1(=x)$




(b) the pendulum angle at the root $y_2 = (\theta)$




(c) the tip acceleration $y_3$

Fig. 4. Time responses with and without NN augmentation for the decentralized controllers

degree zero while most approaches, including the proposed adaptive control in this paper, assume the relative degree greater than zero. Considering that acceleration signals are usually filtered before use for feedback, we introduce a first-order filter for $y_3$: $y_{3_f} = L(s)y_3$, where $L(s) = \frac{1}{s/\omega_a+1}$, $\omega_a = 6(Hz) = 12\pi(rad/sec)$, and treat $y_{3_f}$ as the regulated output. The resulting controller is designed as $u_{c_2} = -K_a L(s)(y_{d_3} - y_3)$, where $K_a = 1$.

*Adaptive control augmentation:* The input matrix $\hat{H}$ in Assumption 4 is obtained by considering the linearized dynamics that include the low-pass filter $L(s)$ for the acceleration. An error observer for $\boldsymbol{e}_1 = [e_1^r, x_{c_1}^{m\top} - x_{c_1}{}^\top]^\top$ is designed as the following reduced observer $\dot{\hat{\boldsymbol{e}}}_1^r = A_1\hat{\boldsymbol{e}}_1^r + \boldsymbol{b}_1[u_{c_1}(y_{m_1}) - u_{c_1}(y_1)] + L_1(\boldsymbol{s}_1 - \hat{\boldsymbol{s}}_1)$, where $\boldsymbol{s}_1 = [y_{m_1} - y_1, y_{m_2} - y_2]^\top = C_1^r \boldsymbol{e}_1^r$ because $x_{c_1}^m - x_{c_1}$ is available. The signal $u_{c_1}(y_{m_1})$ is the linear control signal in the reference model and $u_{c_1}(y_1)$ is the output of the LQG controller. Therefore, the error estimate $\hat{\boldsymbol{e}}_1$ is constructed as: $\hat{\boldsymbol{e}}_1 = \begin{bmatrix} (\hat{\boldsymbol{e}}_1^r)^\top & (\boldsymbol{x}_{c_1}^m - \boldsymbol{x}_{c_1})^\top \end{bmatrix}^\top$, and the resulting overall error estimate becomes $\hat{\boldsymbol{e}} = [\hat{\boldsymbol{e}}_1^\top, x_{c_2}^\top]^\top$.

A SHLNN consisting of 10 neurons in the hidden layer is implemented to achieve coordination between $(u_1, y_1, y_2)$ and $(u_2, y_3)$. The input for the NN is composed of 2 delayed values of $y_1$,$y_2$, and $y_3$, as well as the inputs $\boldsymbol{u}$. The time delay is set as $d = 0.002$ sec. The tuning parameters for two networks are set as: $\Gamma_M = 0.1I, \Gamma_N = 0.5I, k_1 = 0.02$ where $I$ is the identity matrix with compatible dimension.

Figure 4 compares output responses of the closed-loop systems with and without the NN. While the accelerations levels are kept at the same level, the NN provides improved responses in the position of the cart and the angle of the pendulum. This implies that the NN helps to coordinate the rigid motion with suppressing action of the tip actuator while independent design for acceleration suppression leads to large deviation in the rigid body motion.

## VII. Summary

We propose a central neural network based adaptive algorithm that coordinates a distributed pair of sensors and actuators in controlling an uncertain complex system. Compared to the previous approaches, the approach allows for control couplings among the input/output pair which is utilized for updating a centralized neural network. The stability proof naturally follows from that of the existing multi-input multi-output approach by taking a viewpoint that decentralized controllers are a subclass of linear controllers. We illustrate the proposed method in control of a flexible inverted pendulum mounted on a cart in which acceleration feedback is added to a linear quadratic Gaussian regulator designed for a rigid pendulum and a cart system.

## Acknowledgments

This research was in part supported by the NASA Marshall Space Flight Center, under grant number NAG8-1912.

## References


[1] J.T. Spooner and K.M. Passino. Decentralized adaptive control of nonlinear systems using radial basis neural networks. *IEEE Transactions on Automatic Control*, 44(11):2050–2057, 1999.

[2] S.N. Huang, K.K. Tan, and T.H. Lee. Stable decentralized adaptive control design of robot manipulators using neural network approximations. *Advanced Robotics*, 17(4):369–383, 2003.

[3] S.N. Huang, K.K. Tan, and T.H. Lee. Decentralized control design for large-scale systems with strong interconnections using neural networks. 48(5):805–810, 2003.

[4] N. Hovakimyan, E. Lavretsky, B.-J. Yang, and A.J. Calise. Coordinated decentralized adaptive output feedback for control of interconnected systems. 16(1):185–194, 2005.

[5] F.L. Lewis, S. Jagannathan, and A. Yeşildirek. *Neural Network Control of Robot Manipulators and Nonlinear Systems*. Taylor & Francis, 1999.

[6] N. Hovakimyan, B.-J. Yang, and A. J. Calise. Adaptive output feedback control methodology applicable to non-minimum phase nonlinear systems. *Automatica*, 42(4):513–522, April 2006.

[7] H.K. Khalil. *Nonlinear Systems*. Prentice-Hall, Upper Saddle River, NJ, 1996.

[8] B.-J. Yang and A. J. Calise. Adaptive control of a class of multivariable nonaffine systems. In *Proceedings of Conference on Decision and Control*, pages 4809–4814, New Orleans, LA, 2007.

[9] S.S. Ge, C.C. Hang, T.H. Lee, and T. Zhang. *Stable Adaptive Neural Network Control*. Kluwer Academic Publishers, Boston, 2002.

[10] B.-J. Yang and Anthony J. Calise. Adaptive control of a class of nonaffine systems using neural networks. *IEEE Transactions on Neural Networks*, 18(4):1149–1159, 2007.

[11] Z. Qu. *Robust control of nonlinear uncertain systems*. John Wiley & Sons, 1998.

[12] N. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2:359–366, 1989.

[13] E. Lavretsky, N. Hovakimyan, and A.J. Calise. Upper bounds for approximation of continuous-time dynamics using delayed outputs and feedforward neural networks. *IEEE Transactions on Automatic Control*, 48(9):1606–1610, 2003.

[14] N. Hovakimyan, F. Nardi, N. Kim, and A.J. Calise. Adaptive output feedback control of uncertain systems using single hidden layer neural networks. *IEEE Transactions on Neural Networks*, 13(6), 2002.

[15] C. W. Anderson. Learning to control an inverted pendulum using neural networks. *IEEE Control Systems Magazine*, 9(3):31–37, 1989.

[16] B. J. Yang, A. J. Calise, J. I. Craig, and K. Kim. Centralized adaptive control of a complex flexible system. In *AIAA Guidance, Navgation, and Control Conference*, Hilton Head, SC, 2007. AIAA-2007-6686.

[17] B.-J. Yang, N. Hovakimyan, A.J. Calise, and J.I. Craig. Experimental Validation of an Augmenting Approach to Adaptive Control of Uncertain Nonlinear Systems. In *Proceedings of AIAA guidance,navigation and control conference*, Austin, TX, 2003. AIAA-2003-5715. (Submitted to IEEE Transactions on Control System Technology, 2006).