Robotica (2004) volume 22, pp. 419–438. © 2004 Cambridge University Press
DOI: 10.1017/S0263574703005204 Printed in the United Kingdom

# A methodology for fault diagnosis in robotic systems using neural networks

Arun T. Vemuri* and Marios M. Polycarpou†

(Received in Final Form: April 3, 2003)

## SUMMARY

Fault diagnosis plays an important role in the operation of modern robotic systems. A number of researchers have proposed fault diagnosis architectures for robotic manipulators using the model-based analytical redundancy approach. One of the key issues in the design of such fault diagnosis schemes is the effect of modeling uncertainties on their performance. This paper investigates the problem of fault diagnosis in rigid-link robotic manipulators with modeling uncertainties. A learning architecture with sigmoidal neural networks is used to monitor the robotic system for off-nominal behavior due to faults. The robustness, sensitivity, missed detection and stability properties of the fault diagnosis scheme are rigorously established. Simulation examples are presented to illustrate the ability of the neural network based robust fault diagnosis scheme to detect and accommodate faults in a two-link robotic manipulator.

KEYWORDS: Fault diagnosis; Neural networks; Robotic systems.

## 1. INTRODUCTION

Signal processing in robotic manipulators typically involves the analysis of robotic sensor signals to infer the current state of the robotic manipulator so that appropriate control action can be generated. With the advent of microprocessors and with recent advances in their computational power and memory, signal processing algorithms are increasingly being implemented in software instead of being implemented in hardware. One of the powerful applications of such digital signal processing techniques is the monitoring and diagnosis of robotic system faults.

Robotic systems are integral components of many complex engineering systems including manufacturing processes,[1] underwater autonomous vehicles and space-based systems.[2] Stricter operational and productivity requirements in such systems are resulting in robotic manipulators working near their design limits for much of the time. This may often lead to robotic system failures which are typically characterized by critical changes in the robotic system parameters or even by nonlinear changes in the inherent dynamics of the manipulator. Robotic system failures can potentially result not only in the loss of productivity but also in the unsafe operation of the manipulator. In general, modern control systems which are designed to handle small perturbations that may arise under "normal" operating conditions (in the "linear" regime) cannot accommodate abnormal behavior due to faults. Hence automated health monitoring of robotic systems and effective identification and/or accommodation of any faults play a crucial role in the operation of modern robotic systems and especially autonomous and intelligent robotic manipulators.

The design and analysis of Fault Diagnosis (FD) architectures for robot systems using the *model-based analytical redundancy* approach has received considerable attention.[2–4] In this approach, quantitative nominal models of the robotic system together with sensory measurements are used to provide estimates of measured and/or unmeasured variables. The deviations between estimated and measured signals provide a *residual* vector which can be utilized to detect and isolate system failures. In general, a fault is declared if a measure of the residual vector exceeds a certain *threshold* value. Note that this approach is in contrast to *hardware redundancy*-based approach[5] wherein additional physical instrumentation is used to provide the necessary redundancy.[6]

The appeal of model-based FD schemes lies in the fact that the redundancy required for detecting faults is created using powerful information processing techniques without the need of additional physical instrumentation in the system. However, the model-based FD approach relies on the key assumption that the mathematical characterization of the manipulator is known. In practice, this assumption is usually not valid since it is difficult to obtain the necessary modeling accuracy required for the construction of reliable analytical redundancy-based FD architectures. Unavoidable *modeling uncertainties*, which arise due to modeling errors, time variations, measurement noise, and external disturbances, deteriorate the performance of FD schemes by causing false alarms. This necessitates the development of FD algorithms which have the ability to detect manipulator failures in the presence of modeling uncertainties. Such algorithms are referred to as *robust fault diagnosis schemes*.

The construction of robust FD architectures for robotic manipulators has been investigated by a number of researchers. In reference [7], the authors use threshold adaptation based on fuzzy logic to improve robustness of state-space model-based FD architecture for robotic systems. Time varying, robotic system state-dependent thresholds are used in reference [8] to achieve robustness in parity relations-

* VLR Embedded, Inc., 411 E. Buckingham Rd. #636, Richardson, Texas 75081 (USA)
† Department of Electrical and Computer Engineering, University of Cincinnati, Cincinnati, Ohio 45221-0030 (USA).

based FD schemes for remote robots. The characteristics of the robotic joint accelerations under faulty conditions have been exploited in reference [9] to detect robotic faults. Terra and Mendes develop robotic fault detection technique based on the Luenberger observer in reference [10]. The authors in reference [11] use statistical techniques to implement fault detection schemes for robot manipulators. In reference [12], the authors develop a technique to detect robotic failures based on torque prediction error estimates. The reference paper [13] compares three neural network schemes for isolating robotic system faults-based on residuals that are generated using a neural network based model of the robotic system. Some researchers have also investigated methods to detect and identify robotic system sensor failures,[14,15] and actuator failures.[16]

Most of the robotic system FD schemes that have been proposed in the literature rely on one or more of the following three key assumptions: (i) the nominal model of the system is *linear*; (ii) the failures are modeled as external *additive* inputs (functions of time); (iii) the joint *accelerations* measurements are available. Although it is convenient from an analytical viewpoint to study the FD problem in a linear system framework, the dynamics of robotic systems are inherently nonlinear. Furthermore, most practical failures are nonlinear functions of the robotic system state and the input torque.

This paper presents a systematic *learning* methodology for robust fault diagnosis in rigid link robotic manipulators, which is based on a nonlinear nominal model of the manipulator and nonlinear deviation faults. The modeling uncertainties are assumed to be bounded by *a priori* known non-uniform bound while the faults are modeled as nonlinear functions of the measured variables. The principal idea behind this approach is to monitor the plant for any off-nominal system behavior (which could be either due to faults or uncertainties) utilizing a sigmoidal neural network. By using the knowledge of the bound on the uncertainty we develop a systematic procedure, based on adaptive and neural network techniques, for identifying the effects of system failures in the presence of modeling uncertainties. The neural network not only detects the occurrence of the fault but also provides a post-fault model of the robotic manipulator. This post-fault model can be effectively used to isolate and identify the fault and, if possible, for accommodation of the failure. It is noted that the approach presented in this paper does not require the measurements of joint accelerations; the method simply relies on the measurements of joint positions and velocities which are usually available for generating the control torques for the robotic manipulator.

The fault diagnosis scheme is rigorously analyzed for robustness, sensitivity, and stability. Specifically, the robustness analysis investigates the behavior of the neural network in the presence of modeling uncertainties prior to the occurrence of a fault. The sensitivity analysis examines the behavior of the neural network after the occurrence of a fault and characterizes the class of faults that can be detected by the robust nonlinear FD scheme. The sensitivity result is used to derive an example class of faults that will not be detected by the proposed technique and to characterize an upper bound for the failure detection time. Finally, the stability analysis examines the behavior of the neural network after the occurrence of a fault.

The application of the above concept to the detection and diagnosis of robotic system faults was first shown by the authors of this paper in reference [17], where we demonstrated the application of the on-line approximation-based technique to detect and diagnose abrupt robotic system failures under the assumption that the robotic system model is accurately known. The authors then demonstrated[18] the application of the non-linear learning technique to detect abrupt failures in robotic systems with modeling uncertainties. In this paper we extend the results developed in reference [18] to the case where the robotic fault is incipient or slowly-developing in contrast to being abrupt. Furthermore, this paper characterizes the sensitivity properties in addition to the robustness and stability properties that were characterized in reference [18]. This paper also derives an upper bound for failure detection time of the fault diagnosis architecture.

It is emphasized that the fault diagnosis scheme developed in this paper uses neural networks to obtain an approximation of the fault model. This is in contrast to the use of neural networks for robotic fault classification.[19] These fault classification techniques mainly concentrate on the use of neural networks to identify the robotic system failure, based on residuals that are generated using linear modeling techniques. The neural network-based technique presented in this paper instead generates a mathematical model of the fault dynamics which can be used to classify and accommodate the failure.

The organization of the paper is as follows: In Section 2, the robot dynamics and its control law are described, and the fault diagnosis problem is formulated. In Section 3, the neural network-based robust fault diagnosis scheme is described. The analytical properties of the robust FD algorithm are established in Section 4. Simulation examples illustrating the performance of the FD algorithm on a two-link robotic manipulator with modeling uncertainties are presented in Section 5. Section 6 has some concluding remarks.

## 2. PROBLEM FORMULATION

Consider an $n$ degree of freedom rigid link robotic manipulator whose motion is described by the differential equation

$$\ddot{q} = M^{-1}(q)[\tau - V_m(q,\dot{q})\dot{q} - G(q) - F(\dot{q}) - \tau_d(t)] + \beta(t-T)\psi(q,\dot{q},\tau), \quad (1)$$

where $q, \dot{q}, \ddot{q} \in \mathbb{R}^n$ are vectors of joint positions, velocities and accelerations respectively, $\tau \in \mathbb{R}^n$ is the input torque vector, $M(q) \in \mathbb{R}^{n\times n}$ is the inertia matrix (whose inverse exists),[20,21] $V_m(q,\dot{q}) \in \mathbb{R}^{n\times n}$ is a matrix containing the centripetal and Coriolis terms, $G(q) \in \mathbb{R}^n$ is the gravity vector, $F(\dot{q}) \in \mathbb{R}^n$ is a vector containing the unknown static and dynamic friction terms and various sources of uncertainties and the quantity $\tau_d(t) \in \mathbb{R}^n$ is a vector representing unknown additive bounded disturbances and noise. The terms $F(.)$ and $\tau_d(.)$ in (1) together represent the modeling

https://doi.org/10.1017/S0263574703005204 Published online by Cambridge University Press

uncertainties in the system. The map $\psi$: $\mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^n \mapsto \mathbb{R}^n$ is a vector that represents the fault in the robot manipulator, $\beta(t-T) \in \mathbb{R}$ represents the time profile of the fault, and $T$ is the unknown time of occurrence of the fault.

A fault in the robotic system changes the dynamics of the manipulator in an unpredictable way. An accurate description of fault conditions, most often, requires nonlinear modeling of faults, which is what is described by $\psi$ in (1). The nonlinear modeling capability is reflected in allowing the deviation $\psi$ due to faults to be a nonlinear function of the joint positions and velocities, and the input torque. This nonlinear modeling of the fault in the dynamic system allows the representation of not only additive faults (drifts in the robotic system) but also multiplicative faults (manipulator parameter changes) and faults which result in additional states (i.e. link breakage). For example, if the gravity vector $G$ changes to $\bar{G}$ due to a fault at time $T$, then this change in the gravity term can be represented by letting

$$\psi(q,\dot{q},\tau) = M^{-1}(q)[\bar{G}(q) - G(q)].$$

In other words, the formulation (1) models a fault in the robotic system as an unknown *change* in the dynamics of the manipulator, where the change is represented as a function of not time, but as a function of the system state and the system inputs. Note that this is in contrast to most traditional fault modeling approaches where the fault is represented as an additive or a multiplicative time-dependent quantity. We refer to FD schemes that are based on such nonlinearly modeled faults as *nonlinear fault diagnosis schemes*.

The price that one has to pay (in the design of fault diagnosis systems in the framework of (1)) for the potential to model a much larger class of faults, is the need to approximate unknown nonlinear functions. In other words, the detection and diagnosis of such nonlinearly modeled robotic faults requires the use of on-line nonlinear function approximation techniques to estimate the fault function $\psi(.)$. However, recent advances in both hardware implementation and software simulation tools have allowed the use of on-line approximators such as sigmoidal neural networks and wavelet networks for constructing and analyzing nonlinear models.[22,23] Powerful on-line estimation techniques, with rigorous theoretical basis and practical implementation techniques, have been developed to allow the construction of approximations of unknown functions. Hence, despite the complexity of the resultant fault diagnosis architecture, when compared to traditional fault detection systems, nonlinear fault diagnosis architectures are receiving more attention in the literature.

## 3. ROBUST NONLINEAR FAULT DIAGNOSIS ARCHITECTURE

In this section, we describe the robust nonlinear fault diagnosis architecture for detecting system faults in robotic manipulators described by (1). We begin by observing that in the absence of modeling uncertainties, any off-nominal behavior observed from input-output measurements (i.e., deviation of certain measured signals from expected values) can be attributed to a fault in the robotic system. Thus, the process of fault detection in the absence of modeling uncertainties can be achieved by approximating, on-line, the unknown function $\psi(q,\dot{q},\tau)$.[18,24,25] However, in the presence of modeling uncertainties, the difference in the dynamics could be either due to faults or due to modeling uncertainties. Hence, the problem of detecting faults in the framework of (1) not only involves the estimation of the nonlinear function $\psi(.)$ but also involves the problem of *distinguishing* the effects of a fault from the effects of modeling uncertainties on the nominal behavior of the system. Therefore, the key question is: how does one identify the effects of a fault in the presence of modeling uncertainties?

The importance of robustness in FD architectures was recognized early on and a variety of robust fault detection schemes have been proposed in the literature during the last twenty years. In most practical manipulators, both faults and modeling uncertainties are unknown *a priori*. However, in most practical situations the modeling uncertainties often exhibit certain known characteristics that distinguish them from faults. These known characteristics of the modeling uncertainties are exploited to construct appropriate robust fault diagnosis architectures.

Frank[26] describes a robust fault detection algorithm which, while providing an on-line estimate of the uncertainty level, deals exclusively with the detection of faults. In the context of fault diagnosis, in addition to detecting the occurrence of a fault, it would be useful to obtain an approximation of the fault function in the presence of modeling uncertainties, in order to identify, and possibly accommodate, the fault.

Robust fault diagnosis schemes are typically classified according to the description of the modeling uncertainty, which is either *structured* or *unstructured*. The uncertainty is said to be structured when the distribution matrix of the uncertainty is known; otherwise it is said to be unstructured. The decoupling of the effects of faults and modeling uncertainties in the case of structured uncertainties is achieved by appropriately assigning the eigenvalues and eigenvectors of the unknown input observers used for detecting faults.

Usually, in practical applications the structure of the uncertainty is not known or varies with time so that the uncertainty is unstructured. One of the popular approaches to deal with unstructured uncertainty is to assume an *appproximately* known structure for the uncertainty. The principle of disturbance decoupling[27] is then used to design robust residual generators[28,29] The problem of determining approximations to the structure of the uncertainty has been addressed by Patton and co-workers.[27,30]

Another approach for improving robustness of FD schemes in the presence of unstructured uncertainty is based on assuming that certain knowledge on the *size* of the unstructured uncertainty is given. The techniques which use this knowledge for designing robust FD schemes include the threshold selector based on robust control theory,[31] Markov theory,[32] adaptive threshold methodologies,[33,34] likelihood ratio techniques,[26,35] time-varying, state-dependent thresholds,[8] fuzzy logic based decision-making methods,[36] and a combination of signal processing and pattern recognition

https://doi.org/10.1017/S0263574703005204 Published online by Cambridge University Press

techniques.[37] Other approaches for robust fault detection in the presence of unstructured uncertainty include an improved residual generation based on the knowledge of the frequency characteristics of the unstructured uncertainty,[38] the design of a robust fault detection filter[39] and an improved residual evaluation process[40] for interval-type parametric uncertainty. Dixon et. al.[12] describe the design of a robust fault diagnosis technique for robotic system actuator failures. This technique assumes that the modeling uncertainties can be expressed as a nonlinear function with a linearly-parameterized structure and unknown linear parameters whose bounds are constant and known. For more thorough and detailed discussions on the design of robust fault detection and diagnosis scheme for dynamic systems, we refer the reader to the books by Gertler,[41] and Chen and Patton.[42]

In this paper, we develop a robust fault diagnosis algorithm for detecting faults in the presence of unstructured modeling uncertainties that satisfy the bounding assumption described below.

**Assumption 3.1.** *The modeling uncertainty is unstructured and is bounded by a known function; i.e.,*

$$|M^{-1}(q)[F(\dot{q})+\tau_d]| < \bar{\mu}(q,\dot{q},\tau), \quad \forall(q,\dot{q},\tau)\in\mathcal{D},$$

*where* $\mathcal{D}\subset\mathbb{R}^n\times\mathbb{R}^n\times\mathbb{R}^n$ *is some compact domain of interest and* $\bar{\mu}(q,\dot{q},\tau,t)$ *is a known, integrable, positive, and bounded function in* $\mathcal{D}$.

This assumption implies that this paper assumes that the "size" of the modeling uncertainties is known. It is noted that, in this paper, for a vector $\nu\in\mathbb{R}^n$, $n>1$, the notation $|\nu|$ is equivalent to $\|\nu\|_2$. The above assumption further implies that the state, input and time dependence of the modeling uncertainty size is also assumed to be known. By allowing the bound to be a function of the robotic system state, input and time, the formulation provides a framework for non-uniform bounds, thus enhancing fault sensitivity (i.e. the ability of the fault diagnosis architecture to detect robotic system faults). For example, in many practical applications the nominal model is obtained by small-signal linearization techniques around a nominal operating point or trajectory. In this case, the modeling uncertainty may represent the residual nonlinear terms, which are typically small close to the operating point, but can be large elsewhere. If non-uniform bounds on $\bar{\mu}(q,\dot{q},\tau,t)$ are not known, then the more conservative assumption of uniform constant bound can be used in the development of robust fault diagnosis architecture.[43] Note that under this conservative assumption, an estimate of the fault function is obtained provided that the ratio between the fault function magnitude and the modeling uncertainty level is sufficiently large.

The formulation (1) involves the time profile $\beta$ of the fault. This profile represents the "speed" of occurrence of the robotic failure; i.e, the time-profile describes how fast the robotic fault is developing. If the fault occurs instantaneously at the unknown time $T$, the fault is said to be an *abrupt* fault. If, however, the fault in the robotic manipulator develops slowly in time, starting at time $T$, the fault is said to be an *incipient* fault. In this paper, we assume that the fault time profile is modeled as described in the following assumption.[44]

**Assumption 3.2.** *The robotic fault occurs at some unknown time T vith the time profile of the failure given by*

$$\beta(t-T)=\begin{cases} 0 & \text{if } t<T \\ 1-\exp(-\rho(t-T)) & \text{if } t\geq T \end{cases}$$

*where* $p>0$ *denotes the unknown fault evolution rate.*

The time profile described in Assumption 3.2 can be used to represent both abrupt and incipient faults. A small value of $\rho$ represents an incipient fault,[44] while a very large value of $\rho$ characterizes abrupt failure.[45] This paper does not assume a specific speed of occurrence of the fault; i.e. the analysis of the nonlinear fault diagnosis architecture described in this paper assumes the generic time profile described by Assumption 3.2 without any assumption on the value of $\rho$. The simulations presented in this paper, however, illustrate the incipient fault cases only. The reader is referred to reference [18] for the simulation of abrupt failures.

The final assumption this paper makes in the development of the robust robot fault diagnosis architecture is regarding the stability of the robotic manipulator in the presence of faults. This assumption is described below.

**Assumption 3.3.** *The robotic system states and the torque command remain bounded in the compact domain of interest* $\mathcal{D}$ *after the occurrence of a fault; i.e.,* $(q,\dot{q},\tau)\in\mathcal{D}$.

Note that the above assumption does not deal with the stability of the overall fault diagnosis architecture. Rather, the assumption simply assumes that only the robotic link (i.e. the plant) remains stable after the occurrence of the fault. In some sense, this assumption implies that the robotic manipulator remains "functional". This is a reasonable assumption since this paper does not analytically address the robot controller reconfiguration problem (although simulations are used to illustrate the use of the fault approximation in control reconfiguration). It is noted that Assumption 3.3 will be used to establish, later in this paper, the stability properties of the overall system; i.e., the robotic manipulator plus the fault diagnosis architecture.

### *3.1. Nonlinear estimation model*

The goal of this paper is to design a fault diagnosis architecture for the robotic manipulator described by (1) under Assumptions 3.1–3.3 using the approximation properties of sigmoidal neural networks. The fault diagnosis architecture is developed in the remaining part of this section in the following way. First, a neural network-based nonlinear estimation model is described. This nonlinear estimation model, which uses the neural network in its structure, primarily provides an estimate of the robotic link velocities. Next, a learning algorithm, based on the nonlinear estimation model, for updating the parameters of the neural network is developed. The learning algorithm is designed so that the neural network approximates any off-nominal behavior due to faults in the presence of modeling uncertainties that satisfy Assumption 3.1.

https://doi.org/10.1017/S0263574703005204 Published online by Cambridge University Press

We now consider the nonlinear estimation model. Based on the system representation (1) and Assumption 3.2, we choose the following nonlinear estimated model:

$$\omega(t)=W_{\alpha}(s)[\alpha\dot{q}+M^{-1}(q)[\tau(q,\dot{q})-V_m(q,\dot{q})\dot{q}-G(q)] \\ +\hat{\psi}(q,\dot{q},\tau;\hat{\theta})], \quad (2)$$

where $\omega(0)=\dot{q}(0)$ and $W_{\alpha}$ is a first-order stable lag filter whose transfer function is given by

$$W_{\alpha}:=\frac{1}{s+\alpha}, \quad (3)$$

where $\alpha>0$ is the filter time constant, a design parameter,[*][†] The term $\hat{\psi}$ in (2) represents a three-layered sigmoidal neural network while $\hat{\theta}\in\mathbb{R}^p$ represents the adjustable weights of the network in vector form.

The input-output characteristics of a three-layered sigmoidal neural network are described by:

$$y=\hat{W}_{2\sigma}(\hat{W}_{1z}), \quad (4)$$

where $z=[1\ q^T\ \dot{q}^T\ \tau^T]^T$ is the input to the network, $y$ is the output of the network, $\hat{W}_1$ is the weight matrix from the input layer to the hidden layer and $\hat{W}_2$ is the weight matrix from the hidden layer to the output layer. If the number of neurons in the hidden layer is $L$, then the dimensions of $\hat{W}_1$ and $\hat{W}_2$ are $L\times(3n+1)$ and $n\times L$, respectively. The nonlinear operator $\sigma:\mathbb{R}^L\mapsto\mathbb{R}^L$ is the standard sigmoidal function such that the $i$th component $(i=1, 2, \ldots, L)$ of $\sigma$ is given by

$$(\sigma(x))_i=\frac{1-\exp(x_i)}{1+\exp(x_i)}.$$

For the convenience of analysis, we represent the input-output characteristics of the neural network described by (4) as

$$y=\hat{\psi}(q,\dot{q},\tau;\ \hat{\theta}),$$

where $\hat{\psi}(.)$ is a vector-valued function such that the $k$th output $(k=1, 2, \ldots, n)$ has the form

$$\hat{\psi}_k(q,\dot{q},\tau;\hat{\theta})=\sum_{j=1}^{L}\left[\hat{\theta}_{(k+3n)L+j}\,\sigma\left(\sum_{i=1}^{(3n+1)}\hat{\theta}_{(i-1)L+j}\,z_i\right)\right]. \quad (5)$$

The weight vector is given by

$$\hat{\theta}=\begin{bmatrix}\operatorname{vec}(W_1^T)\\ \operatorname{vec}(W_2)\end{bmatrix},$$

where vec($A$) operator stacks the columns of a matrix $A$ to form a vector. From the above representation, it can be inferred that $\hat{\theta}\in\mathbb{R}^p$, where $p=(4n+1)L$. The structure of the above neural network is shown in Figure 1.

Note that in this formulation, the estimation model (2) is a nonlinear observer-type scheme that is implemented in the form of stable filter. The filter output $\omega\in\mathbb{R}^n$ is the estimate of the velocity vector of the manipulator joints. The construction of an appropriate estimation model, able to follow changes in the input/output behavior of the physical system, is a crucial component in the development of the overall fault detection scheme. The output of the above nonlinear estimation model is used to update the weights of the neural network. The nonlinear estimation model (2) is not only easy to implement but, more importantly, has some desirable analytical properties, which are presented in the Analytical Properties Section.

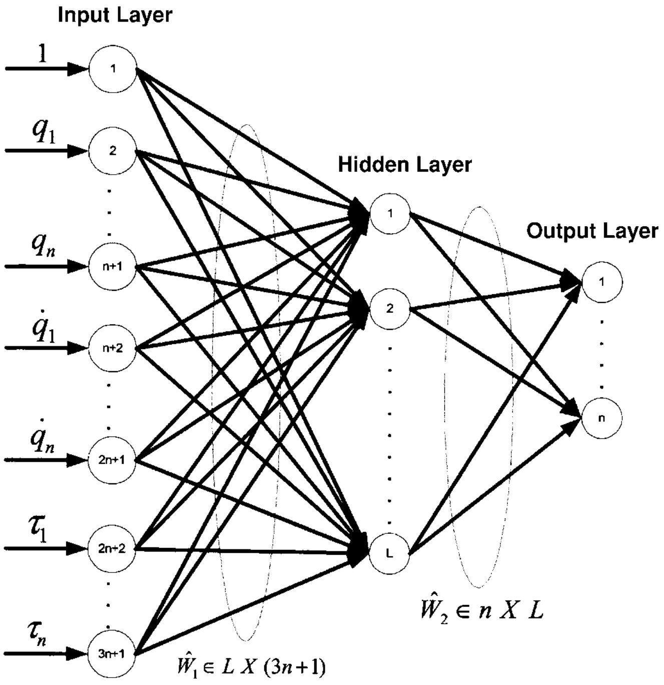

Fig. 1. Structure of the neural network used to learn the fault.

### 3.2. Learning algorithm

We now discuss the learning algorithm used to update the weights of the sigmoidal neural network so that the neural network provides an estimate of the unknown fault function in the presence of the modeling uncertainties described by Assumption 3.1.

The initial weight vector, $\hat{\theta}(0)=\hat{\theta}^0$, of the neural network is chosen such that

$$\hat{\psi}(q,\dot{q},\tau;\hat{\theta}^0)=0, \quad \forall(q,\dot{q},\tau)\in\mathcal{D}, \quad (6)$$

corresponding to the no-failure situation. Note that this can be achieved by simply setting the weights of the output layer to zero. Starting from these initial conditions, the main objective is to adjust (using input/output information) the weight vector $\hat{\theta}(t)$ at each time $t$ so that $\hat{\psi}(q,\dot{q},\tau;\hat{\theta})$ provides an approximation of the unknown function $\beta(t-T)\psi(q,\dot{q},\tau)$. Once this is achieved, the output of the neural network $\hat{\psi}$ can be used to diagnose and accommodate any system failures.

Based on system description (1) and the estimation model (2), we propose the following adaptive law for updating the weights of the neural network:

$$\dot{\hat{\theta}}=\mathcal{P}\{\Gamma Z^T D[e]\}, \quad \hat{\theta}(0)=\hat{\theta}^0, \quad (7)$$

[*] A ":" followed by the "=" sign implies that the quantity on the LHS of the ":=" sign is being defined as the quantity on the RHS.

[†] The use of both time domain and transfer function notation is common in the adaptive control literature. For example, see Ioannou and Sun.[46]

https://doi.org/10.1017/S0263574703005204 Published online by Cambridge University Press

where $e := \dot{q} - \omega$ is the error between the measured velocity vector and its estimate, $\Gamma = \Gamma^T \in \mathbb{R}^{p \times p}$ is a positive definite *learning rate* matrix, $Z \in \mathbb{R}^{n \times p}$ is the gradient of the neural network with respect to its adjustable weights; i.e.

$$Z := \frac{\partial \hat{\psi}(q, \dot{q}, \tau, \hat{\theta})}{\partial \hat{\theta}},$$

and $D[.]$ is the *dead-zone* operator, defined as

$$D[e(t)] := \begin{cases} \mathbf{0}_n & \{\text{if } |e(t)| < \mu(t)\} \\ e & \{\text{otherwise}\}, \end{cases}$$

where $\mathbf{0}_n$ is an $n$-dimensional vector of zeros and $\mu(t)$ is the output of the stable filter $W_\alpha$ when subject to the input signal $\bar{\mu}(q, \dot{q}, \tau, t)$; i.e.

$$\mu(t) = W_\alpha(s)[\bar{\mu}(q(t), \dot{q}(t), \tau(t), t)], \quad \mu(0) = \mathbf{0}_n. \qquad (8)$$

The *projection operator* $\mathcal{P}$ (which restricts the parameter estimate vector $\hat{\theta}$ to the compact, convex region $\mathcal{M}_{\hat{\theta}}$) is used to avoid *parameter drift*, a phenomenon that may occur with standard adaptive laws in the presence of modeling uncertainties.[46] If $\mathcal{M}_{\hat{\theta}}$ is chosen to be a hypersphere of radius $M$, then the above adaptive law can be expressed as

$$\dot{\hat{\theta}} = \Gamma Z^T D[e] - \chi^* \Gamma \frac{\hat{\theta}\hat{\theta}^T}{\hat{\theta}^T \Gamma \hat{\theta}} \Gamma Z^T D[e], \quad \hat{\theta}(0) = \hat{\theta}^0, \qquad (9)$$

where $\chi^*$ denotes the indicator function given by:

$$\chi^* := \begin{cases} 0 & \text{if } \{|\hat{\theta}| < M\} \text{ or } \{|\hat{\theta}| = M\} \text{ and } \hat{\theta}^T \Gamma Z^T e \leq 0\} \\ 1 & \text{if } \{|\hat{\theta}| = M\} \text{ and } \hat{\theta}^T \Gamma Z^T e > 0\}. \end{cases}$$

In summary, the nonlinear estimation model (2) and the weight update algorithm (7) together describe the robust nonlinear fault diagnosis architecture for the robotic manipulator described by (1). The nonlinear approximator $\hat{\psi}$ is used not only for detecting a fault in the robotic manipulator but also for identifying the fault function. Note that a fault in the robotic manipulator is declared when the output of the neural network becomes non-zero, or equivalently, when the error $e(t)$ exceeds the threshold $\mu(t)$. A block diagram representation of the nonlinear fault diagnosis architecture described in this paper is shown in Figure 2.

Intuitively, the above architecture achieves robustness against modeling uncertainties by using a non-uniform threshold for the velocity estimation error; i.e. if the velocity estimation error is below a designed threshold, the robotic system is deemed to have no fault. On the contrary, if the velocity estimation error is greater than the designed threshold, the neural network weights begin to update, thus driving the neural network outputs to non-zero values. This indicates a fault in the robotic manipulator.

We make a key observation regarding the fault diagnosis architecture described by (2) and (7). The above architecture used two operators: (i) the dead-zone operator, and (ii) the projection operator. The dead-zone term in the above architecture prevents the adaptation of the neural network weights when the estimation error is within a (small) bound, thus enhancing robustness in the fault detection scheme. The projection operator, on the other hand, is required in order to guarantee stability of the overall fault detection scheme in the presence of network approximation errors. It is noted that although a bound $\mu$ on the modeling uncertainty is required to be known for selecting the size of the dead-zone, the learning algorithm requires no knowledge of a bound on the *network approximation error* (defined as the "inability" of the neural network to provide an accurate approximation of the fault function $\psi$). As expected, however, the actual value of the network approximation error does affect the performance of the fault diagnosis scheme after the occurrence of the fault (as shown in the Analytical Properties Section).

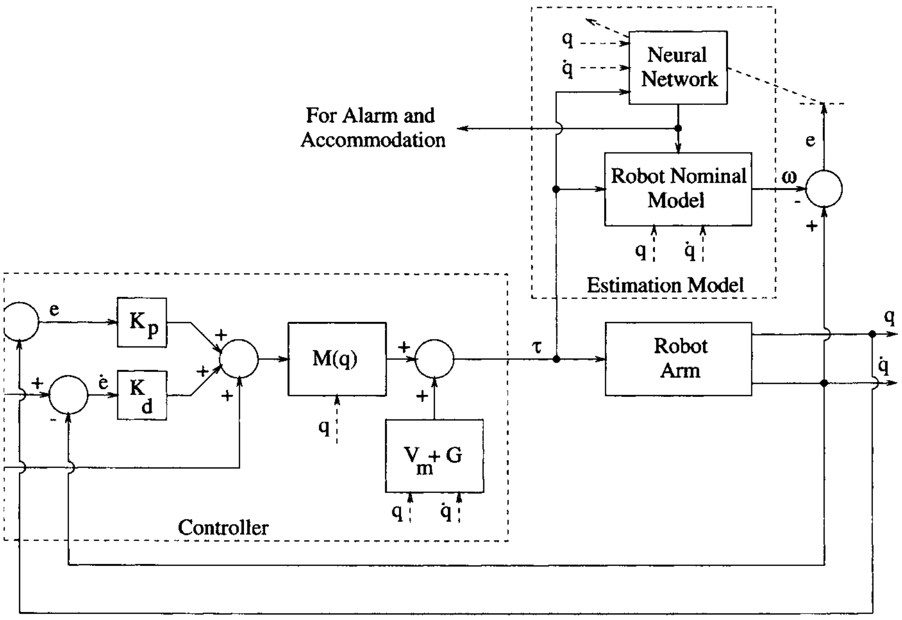

Fig. 2. Block diagram representation of the fault diagnosis architecture.

https://doi.org/10.1017/S0263574703005204 Published online by Cambridge University Press

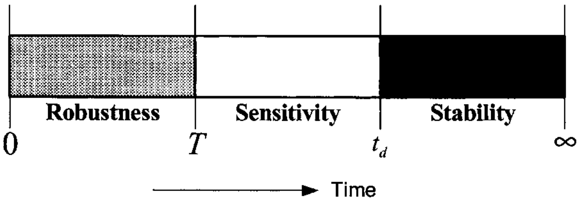

Fig. 3. Analytical properties of interest during different time-zones.

**Remark 3.4.** *Because of the dead-zone, the adaptive law for for $\hat{\theta}(t)$ is a discontinuous function of time. This may cause problems of existence and uniqueness of solutions as well as computational problems at the switching surface.*[47] *One way to avoid these problems is to use a so-called "continuous dead-zone", which has similar stability properties as the discontinuous dead-zone.*[46]

## 4. ANALYTICAL PROPERTIES

In this section we examine the robustness, sensitivity and stability properties of the robust nonlinear FD scheme. The robustness analysis investigates the behavior of the neural network in the presence of modeling uncertainties prior to the occurrence of a fault; i.e. $t<T$. The sensitivity analysis examines the behavior of the neural network after the occurrence of a fault and characterizes the class of faults that can be detected by the robust nonlinear FD scheme. The missed detection result characterizes a class of fault that cannot be detected by the fault diagnosis scheme. Finally, the stability analysis examines the behavior of the neural network after the neural network adaptation begins, i.e. after the detection of the fault. Figure 3 shows the relevant analytical properties that are of interest as the robotic system evolves in time. In the figure, $T$ represents the time at which the fault occurs, while $t_d$ represents the time at which the robotic system fault is detected (i.e, the time at which the sigmoidal neural network output first becomes non-zero). The detection $t_d$ is characterized later in this section.

### 4.1. Robustness property

The robustness property of the fault diagnosis architecture presented in this paper is given by the following result:

**Theorem 4.1 (Robustness).** *The nonlinear fault detection scheme described by (2), (7) guarantees*

$$\hat{\psi}(q(t), \dot{q}(t), \tau(t); \hat{\theta}(t)) = 0,$$

*for $t<T$ prior to the occurrence of a fault.*

*Proof*: The error dynamics in the time period $t<T$ prior to the occurrence of the fault are described by

$$\begin{aligned} e(t) &= \dot{q}(t) - \omega(t) \\ &= W_\alpha(s)[-M^{-1}(q)[F(\dot{q})+\tau_d] - \hat{\psi}(q,\dot{q},\tau;\hat{\theta})] \qquad (10) \end{aligned}$$

Recalling from (6) that the initial parameter vector $\hat{\theta}^0$ is chosen such that $\hat{\psi}(q, \dot{q}, \tau; \hat{\theta}^0) = 0$, the question we need to examine is whether the parameter estimates would start adapting, or equivalently, whether $|e(t)| \geq \mu(t)$, prior to the occurrence of a fault.

Suppose (for the sake of contradiction) that there exists a time $t_e$ (where $0<t_e<T$) such that

$$|e(t_e)| = \mu(t_e), \qquad (11)$$

where $t_e$ denotes the first time instant that $e(t)$ reaches the dead-zone bound.

Using (9) and the continuity of $e(t)$, the parameter vector $\hat{\theta}(t)$ has not been adapted in the interval $[0, t_e)$. Furthermore, by the continuity of $\hat{\theta}(t)$ we have $\hat{\theta}(t_e) = \hat{\theta}^0$; in other words, $\hat{\psi}(q,\dot{q},\tau,\hat{\theta}(t_e)) = 0$. Hence, from (10), the velocity estimation error for $t \in [0, t_e]$ is given by

$$e(t) = -\int_0^t \exp(-\alpha(t-s)) M^{-1}(q(s))[F(\dot{q}(s)) + \tau_d(s)]\, ds.$$

Therefore, the magnitude of the error vector satisfies

$$\begin{aligned} |e(t)| &< \int_0^t \exp(-\alpha(t-s)) \bar{\mu}(q(s),\dot{q}(s),\tau(s),s)\, ds \\ &= \mu(t). \end{aligned}$$

Now, since $|e(t_e)| < \mu(t_e)$, this is clearly a contradiction of (11), which implies that for $t \in [0, T)$ the estimation error $e(t)$ remains within the dead-zone and the output of the neural network remains zero. □

The above theorem states that if there are no faults in the robotic system (i.e., $\psi \equiv 0$), then the output of the neural network will remain zero (i.e., $\hat{\psi} = 0$) even in the presence of modeling uncertainties. Equivalently, if the neural network output is non-zero then it is guaranteed that a fault has occurred. (We note that a failure can also be declared whenever $|e| \geq \mu$). Therefore, the neural network is robust with respect to modeling uncertainties that satisfy Assumption 3.1.

### 4.2. Sensitivity property

The robustness (or "insensitivity") of the fault diagnosis scheme to modeling uncertainties is directly related to its sensitivity to robotic system faults. The following theorem characterizes the class of faults that can be detected by the nonlinear fault diagnosis scheme.

**Theorem 4.2 (Sensitivity).** *Let $\eta(t)$, $t \geq T$, be defined as the output of the first order lag filter (3) when the filter is subject to the fault function $\beta\psi$ as the input; i.e., $\eta(t)$ is a low-pass filtered version of the fault function and is given by*

$$\begin{aligned} \eta(t) &= W_\alpha(s)[(1-\exp(-\rho(t-T)))\psi(q(t),\dot{q}(t),\tau(t))],\ \eta(T) \\ &= \mathbf{0}_n. \qquad (12) \end{aligned}$$

*If*

$$|\eta(t_d)| > 2\mu(t_d), \qquad (13)$$

*for some $t_d > T$, then $e(t_d) \geq \mu(t_d)$.*

*Proof:* The error dynamics after the occurrence of the fault and prior to the adaptation of the network parameters is given by

$$\begin{aligned} e(t) &= \dot{q}(t) - \omega(t) \\ &= W_\alpha(s)[-M^{-1}(q)[F(\dot{q}) \\ &\quad + \tau d] + (1-\exp(-\rho(t-T)))\psi(q,\dot{q},\tau)]. \end{aligned}$$

Therefore, the velocity estimation error during the time interval $T \leq t < t_d$ is given by

$$\begin{aligned} e(t) = {} & \exp(-\alpha(t-T))e(T) \\ & - \int_T^t \exp(-\alpha(t-s))M^{-1}(q(s))[F(\dot{q}(s)) + \tau_d(s)]\, ds \\ & + \int_T^t \exp(-\alpha(t-s))(1-\exp(-\rho(s-T)))\psi(q(s), \\ & \dot{q}(s), \tau(s))\, ds. \end{aligned}$$

Bringing the first two terms of the right hand side to the left and using the triangle inequality, we obtain

$$\begin{aligned} |e(t)| \geq {} & -|\exp(-\alpha(t-T))e(T)| \\ & - \left| \int_T^t \exp(-\alpha(t-s))M^{-1}(q(s))[F(\dot{q}(s)) + \tau_d(s)]\, ds \right| \\ & + \left| \int_T^t \exp(-\alpha(t-s))(1-\exp(-\rho(s-T))) \psi(q(s),\dot{q}(s),\tau(s))\, ds \right| \\ \geq {} & -\exp(-\alpha(t-T))\mu(T) \\ & - \int_T^t \exp(-\alpha(t-s))\bar{\mu}(q(s),\dot{q}(s),\tau(s),s)\, ds \\ & + \left| \int_T^t \exp(-\alpha(t-s))(1-\exp(-\rho(s-T))) \psi(q(s),\dot{q}(s),\tau(s))\, ds \right| \\ = {} & -\mu(t) + |\eta(t)|. \end{aligned}$$

Therefore, if there exists $t_d > T$, such that (13) is satisfied, then $|e(t_d)| \geq \mu(t_d)$, which implies that adaptation of the network will begin and the failure will be detected. □

Figure 4 illustrates the sensitivity result. The figure shows example time evolutions of $\mu, |\eta|$, and the velocity error $e$ for a single-link robotic manipulator under the worst-case detectability conditions. The figure shows that at the time of occurrence of the fault, the sign of the estimation error $e$ is negative while its magnitude is close to $-\mu$. Furthermore, the fault is such that the velocity estimation error tends towards a positive value. Hence, for the fault to be detected by the architecture proposed in this paper, the error vector has to "travel" from almost $-\mu$ in value to $\mu$; in other words, the fault magnitude should be such that the magnitude of the fault is twice the size of the modeling uncertainties which is what is stated by the sensitivity result. Furthermore, the detection time under these worst-case detectability conditions is equal to the time at which the condition $|\eta(t)| > 2\mu(t)$ is satisfied.

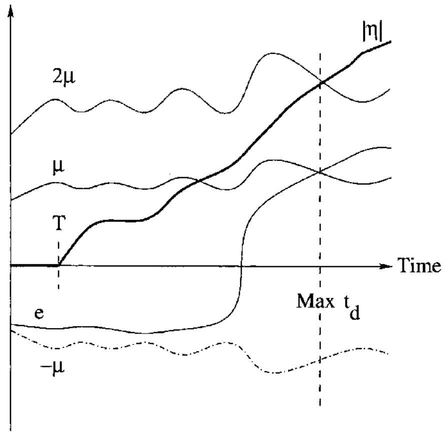

Fig. 4. Illustration of the sensitivity of the fault diagnosis architecture under the worst-case detectability conditions.

The class of detectable faults characterized by the above sensitivity analysis is obtained under the worst-case detectability conditions. In other words, the inequality (13) is a sufficient condition for activating adaptation of the neural network for any modeling uncertainty satisfying Assumption 3.1. Under "best-case" detectability conditions, the class of faults that can be detected by the fault diagnosis scheme is significantly larger. For example, let the robotic system modeling uncertainties be such that $-(M^{-1}(q)[F(\dot{q}) + \tau_d]) = (1-\varepsilon)\bar{\mu}(q,\dot{q},\tau,t), \forall t > 0$, where $\varepsilon \in \mathbb{R}^+$ is a "sufficiently small" positive number. Then it can be easily shown that

$$e(T) = (1-\varepsilon)\mu(T).$$

That is, the network weights will not adapt indicating that the fault has not occurred.

Now, if the failure occurs at $T$ and if the failure is such that $\eta(t)$ given by equation (12) satisfies

$$\eta(t) = \varepsilon\mu(t).$$

Then the robotic manipulator velocity estimation error for $t > T$ satisfies

$$\begin{aligned} e(t) & = \dot{q}(t) - \omega(t) \\ & = W_\alpha(s)[-M^{-1}(q)[F(\dot{q}) + \tau_d] \\ & \quad + (1-\exp(-\rho(t-T)))\psi(q,\dot{q},\tau)] \\ & = W_\alpha(s)[(1-\varepsilon)\bar{\mu}(q,\dot{q},\tau,t)] + \varepsilon\mu(t) \\ & = \mu(t) \end{aligned}$$

Thus a larger class of faults (than those described by 13) is detectable by the robust fault diagnosis scheme presented in this paper. In other words, the sensitivity result given by Theorem 4.2 is only a sufficient condition, and is not a necessary condition.

**Remark 4.3.** *The class of detectable faults is characterized by the "size" of the filtered version of the fault function. This is a direct consequence of the assumed knowledge of the modeling uncertainties and the chosen structure of the network learning algorithm. In other words, since this paper assumes that the size of the modeling uncertainties is known, the sensitivity result characterizes the types of robotic faults detectable by the proposed architecture in*

https://doi.org/10.1017/S0263574703005204 Published online by Cambridge University Press

*terms of the size of the robotic system fault. Furthermore, since the learning algorithm (7) uses the filtered version of the bound on the modeling uncertainties, the sensitivity result also characterizes the class of detectable faults in terms of the size of the filtered version of the fault. This comparison and the role of design filter $W_\alpha(s)$ in the fault diagnosis scheme is illustrated in Figure 5. Note that the figure illustrates the sensitivity result under the worst-case detectability conditions; i.e. if the fault function satisfies the inequality (13), then the fault is guaranteed to be detected and the neural network weights are guaranteed to begin adaptation. It is noted, however, that the the fault will be detected as soon as the condition $e(t) \geq \mu(t)$, which may occur even before the inequality (13) is satisfied.*

The sensitivity result, Theorem 4.2 characterizes the faults that are guaranteed to be detected by the architecture presented in this paper. At the same time, it is important to understand there are some faults that will not be detected by the architecture proposed in the paper. An example class of faults that are guaranteed not to be detected is now discussed below.

Before beginning the discussion, we introduce the following definitions for the purpose of brevity. Let

$$\begin{matrix}\hat{\eta}_1(t)\\ \cdot\\ \hat{\eta}_n(t)\end{matrix} := (1-e^{-\rho(t-T)})\psi(q(t),\dot{q}(t),\tau(t)),$$

$$\begin{matrix}\hat{\mu}_1(t)\\ \cdot\\ \hat{\mu}_n(t)\end{matrix} := -M^{-1}(q(t))[F(\dot{q}(t))+\tau_d(t)]. \tag{14}$$

Now consider the robotic system modeling uncertainties and the robotic fault function such that the following conditions are satisfied:

(a) The fault function is such that:

(i) $\text{sgn}\{\hat{\eta}_i(t)\} = -\text{sgn}\{\hat{\mu}_i(t)\}, \quad i=1,\ldots,n;$
$\forall t>T;$

(ii) $(1-\varepsilon)|\hat{\mu}_i(t)| < |\hat{\eta}_i(t)| < |\hat{\mu}_i(t)|, \quad i=1,\ldots,n;$
$\forall t>T;$

where $|x|$ is the absolute value of $x \in \mathbb{R}$ and $\varepsilon \in \mathbb{R}^+$ with $0<\varepsilon<1$.

(b) The modeling uncertainties are such that:

(iii) $e_i(T)=0, \quad i=1,\ldots,n;$

i.e. the state estimation vector at the time of occurrence of the fault is 0.

Under the above conditions, the fault will not be detected; i.e.

$$e(t) < \mu(t), \quad \forall t>T. \tag{15}$$

The above conclusion can be inferred in the following manner. The error dynamics $e_i(t)$ where $i=1,\ldots,n$ after the occurrence of the fault and prior to the adaptation of the network parameters is given by

$$\begin{aligned} e_i(t) &= \dot{q}_i(t) - \omega_i(t)\\ &= W_\alpha(s)[\hat{\eta}_i(t)+\hat{\mu}_i(t)]. \end{aligned}$$

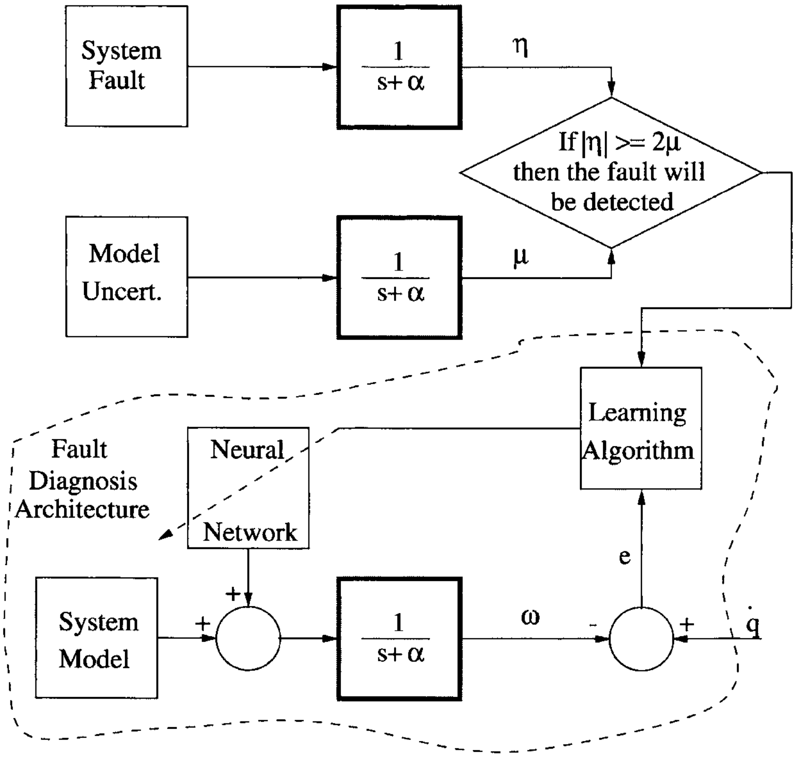

Fig. 5. Illustration of the sensitivity property under the worst-case detectability conditions. The sensitivity property of the fault diagnosis architecture is characterized by the design filter $W_\alpha(s)=1/(s+\alpha)$. The figure also shows the role of $W_\alpha$ in the overall fault diagnosis architecture.

https://doi.org/10.1017/S0263574703005204 Published online by Cambridge University Press

Using the conditions (i)–(iii), the velocity estimation error $e_i(t)$, where $i=1, \ldots, n$, during the time interval $t>T$ is given by

$$e_i(t)=\int_T^t e^{-\alpha(t-s)}\hat{\mu}_i(s)\,ds+\int_T^t e^{-\alpha(t-s)}\hat{\eta}_i(s)\,ds$$

$$=\int_T^t e^{-\alpha(t-s)}\operatorname{sgn}\{\hat{\mu}_i(s)\}(|\hat{\mu}_i(s)|-|\hat{\eta}_i(s)|)\,ds$$

$$=\int_T^t e^{-\alpha(t-s)}\operatorname{sgn}\{\hat{\mu}_i(s)\}(|\hat{\mu}_i(s)|-|\hat{\eta}_i(s)|)\,ds$$

Because of condition (ii) stated in the theorem, the quantity $|\hat{\mu}_i(s)|-|\hat{\eta}_i(s)|$ satisfies

$$0<|\hat{\mu}_i(s)|-|\hat{\eta}_i(s)|<|\hat{\mu}_i(s)|-(1-\varepsilon)\hat{\mu}_i(s)|<\varepsilon|\hat{\mu}_i(s)|.$$

Using the above inequality and the triangle law, the absolute value of each of the robotic link joint velocities satisfies

$$|e_i(t)|=\left|\int_T^t e^{-\alpha(t-s)}\operatorname{sgn}\{\hat{\mu}_i(s)\}(|\hat{\mu}_i(s)|-|\hat{\eta}_i(s)|)\,ds\right|$$

$$<\int_T^t e^{-\alpha(t-s)}\varepsilon\hat{\mu}_i(s)\,ds. \tag{16}$$

Recalling that $e_i(T)$ is given by

$$e_i(T)=\int_0^{\mathrm{T}} \mathrm{e}^{-\alpha(t-s)}\hat{\mu}_i(s)\,ds,$$

and since $e_i(T)=0$, therefore the inequality (16) for $i=1, \ldots, n$ can be written as

$$|e_i(t)|<\int_0^{\mathrm{T}} \mathrm{e}^{-\alpha(t-s)}\varepsilon\hat{\mu}_i(s)\,ds+\int_{\mathrm{T}}^{\mathrm{t}} \mathrm{e}^{-\alpha(t-s)}\varepsilon\hat{\mu}_i(s)\,ds$$

$$\varepsilon\int_0^t e^{-\alpha(t-s)}\hat{\mu}_i(s)\,ds.$$

Therefore the state estimation error vector norm satisfies the inequality

$$|e(t)|<\varepsilon\mu(t).$$

In other words, the fault will not be detected.

In summary, the above result characterizes an example class of faults that will not be detected by the fault diagnosis architecture under certain modeling uncertainties conditions. Note once again, that the class of faults not detectable by the fault diagnosis scheme is characterized by the size of the fault. Figure 6 illustrates an example fault (in the case of a single link manipulator) that will not be detected by the fault diagnosis architecture.

An important performance parameter in fault detection is the length of the time interval between the occurrence of a fault and the detection of the fault, which is usually referred to as the *detection time*. The fault sensitivity Theorem 4.2 not only characterizes the class of detectable faults but it also provides a measure of the detection time. Specifically, the smallest $t_d$ for which the inequality (13) holds is equal to the detection time under the worst-case detectability conditions as shown in Figure 4. Hence, $t_d$ represents the maximum detection time over all allowable scenarios of modeling uncertainties. The following result shows that under additional conditions on the fault function and the modeling uncertainties, we can obtain an upper bound for the detection time.

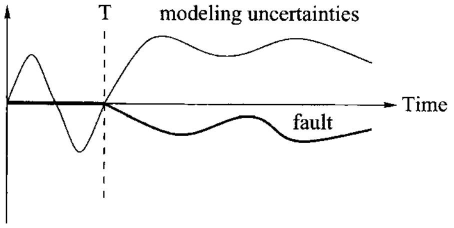

Fig. 6. Example illustration of a fault that will not be detected by the fault diagnosis architecture proposed in this paper.

**Corollary 4.4 (Detection Time).** *If*

(i) *the modeling uncertainties are such that its non-uniform bound, $\bar{\mu}$, satisfies $\bar{\mu}(q(t), \dot{q}(t), \tau(t))<K$ for some $K\in\mathbb{R}^+$; i.e. the size of the modeling uncertainties is less than K everywhere in $\mathcal{D}$*;

(ii) *the fault is such that $|\psi(q(t),\dot{q}(t),\tau(t))|>2K$; i.e. the size of the fault is at least twice the maximum size of the modeling uncertainties*;

(iii) *the sign of each component of $\psi$ does not change for all $t\in[T, t_d]$; i.e. the fault functions remain either positive or negative until the fault is detected*;

(iv) *$\alpha>\rho$; i.e., the filter time constant is faster than the fault evolution rate*;

*then the detection time $t_d$ satisfies the following inequality*:

$$t_d<\frac{1}{\alpha-\rho}\ln\left\{\frac{\rho e^{\alpha T}+(\alpha-\rho)}{\alpha e^{\rho T}}\right\}. \tag{17}$$

*Proof:* Based on the Theorem 4.2, the fault is guaranteed to be detected if the sensitivity condition (13) is satisfied for a given pair of modeling uncertainties and fault functions. This condition is rewritten as:

$$\left|\int_T^t e^{-\alpha(t-s)}(1-e^{-\rho(s-T)})\psi(q(s),\dot{q}(s),\tau(s))\,ds\right|$$

$$>2\int_0^t e^{-\alpha(t-s)}\bar{\mu}(q(s),\dot{q}(s),\tau(s),s)\,ds. \tag{18}$$

The corollary can be proved if we can show that the above inequality is satisfied under the conditions (i)–(iv) stated in the corollary and the detection time (17). We begin the proof of the result by simplifying, under conditions (i)–(iv), the left hand side and the right hand side of the above inequality

https://doi.org/10.1017/S0263574703005204 Published online by Cambridge University Press

separately and by substituting the detection in the simplified equations.

Consider the left hand side of the inequality (18). Using conditions (ii), (iii) and (iv) stated in the corollary, this can be written as

$$LHS=\int_T^t e^{-\alpha(t-s)}(1-e^{-\rho(s-T)})|\,\psi(q(s),\dot{q}(s),\tau(s))\,|\,ds$$

$$>2K\left[\int_{\mathrm{T}}^{\mathrm{t}} \mathrm{e}^{-\alpha(t-s)}\,ds-\int_T^t e^{-\alpha t+\rho T+(\alpha-\rho)s}\,ds\right]$$

$$=2K\left[\frac{1-e^{-\alpha(t-T)}}{\alpha}-\frac{e^{-\rho(t-T)}-e^{-\alpha(t-T)}}{\alpha-\rho}\right]$$

$$=\frac{2K}{\alpha}\left[1+\frac{\rho e^{-\alpha(t-T)}-\alpha e^{-\rho(t-T)}}{(\alpha-\rho)}\right].$$

Consider the right hand side of the inequality (18). Using condition (i) stated in the corollary, this can be written as

$$RHS=2\int_0^t e^{-\alpha(t-s)}\bar{\mu}(q(s),\dot{q}(s),\tau(s),s)\;ds$$

$$<2K\left[\int_0^t e^{-\alpha(t-s)}\,ds\right]$$

$$=2K\left[\frac{1-e^{-\alpha t}}{\alpha}\right]$$

$$=\frac{2K}{\alpha}\,[1-e^{-\alpha t}].$$

Based on the above simplifications of the left hand side and the right hand side of the inequality, the proof of this corollary will conclude if we show that the following inequality is true:

$$\frac{\alpha e^{-\rho(t-T)}-\rho e^{-\alpha(t-T)}}{(\alpha-\rho)}<e^{-\alpha t}.$$

Or equivalently, using the condition (iii) stated in the corollary, the proof will conclude if we show that

$$\alpha e^{-\rho(t-T)}-\rho e^{-\alpha(t-T)}<(\alpha-\rho)e^{-\alpha t}.$$

After using simple algebraic equations, proving the above inequality is the same as proving that

$$e^{(\alpha-\rho)t}<\frac{\rho e^{\alpha T}+(\alpha-\rho)}{\alpha e^{\rho T}}.$$

Now consider the left hand side of the above inequality. Using the detection time bound given by (17) and the condition (iv) stated in the corollary, the left hand side of the above inequality at the time instant $t_d$ (i.e. when the fault is detected) satisfies

$$e^{(\alpha-\rho)t_d}<e^{\;(\alpha-\rho)\frac{1}{\alpha-\rho}\ln\;\frac{\rho e^{\alpha T}+(\alpha-\rho)}{\alpha e^{\rho T}}}=\frac{\rho e^{\alpha T}+(\alpha-\rho)}{\alpha e^{\rho T}}.$$

Therefore, if $t_d$ satisfies the inequality (17), then the inequality (18) is satisfied and hence using the sensitivity result, the fault would be detected by the time $t=t_d$. □

It is noted that the sensitivity and the detection time properties of the fault diagnosis architecture presented in this paper depend on the time constant of the low pass filter. The influence of this filter time constant on these properties of the fault diagnosis architecture has been analyzed in reference [8] for a generic dynamic system in the presence of incipient faults.

### 4.3. *Stability property*

We now examine the stability properties of the robust FD algorithm. The stability analysis of the robust FD algorithm deals with the boundedness of all the signals in the fault detection system and characterizes $L_2$ bounds of the velocity estimation error.

**Theorem 4.5 (Stability)** *In presence of faults, the robust nonlinear FD scheme described by* (2) *and* (9) *has the following properties*:

(a) *$e(t)$ and $\tilde{\theta}(t)$ are uniformly bounded*;
(b) *there exists a non-negative constant $\lambda$ that such for any finite $t_f$*,

$$\int_0^{t_f}|\,e(t)\,|^2\,dt\leq\lambda+\frac{4}{\alpha^2}\int_0^{t_f}(\mu^2+|\,\nu(t)\,|^2)\;dt. \qquad (19)$$

*Proof*: The stability properties of the error dynamics (24) and the adaptive law (9) are analyzed using Lyapunov's Direct Method [46]. Using (1) and (2) we obtain the following error dynamics:

$$\begin{aligned} e(t)&=\dot{q}(t)-\omega(t)\\ &=W_\alpha(s)[-M^{-1}(q)[F(\dot{q})+\tau_d]\\ &\quad+(1-\exp(-\rho(t-T)))\psi(q,\dot{q},\tau)-\hat{\psi}(q,\dot{q},\tau;\hat{\theta})]. \end{aligned}$$

The above dynamics can be written as

$$\dot{e}=-\alpha e-M^{-1}(q)[F(\dot{q})+\tau_d] \qquad (20)$$

$$+(1-\exp(-\rho(t-T)))\hat{\psi}(q,\dot{q},\tau;\hat{\theta}^*)-\hat{\psi}(q,\dot{q},\tau;\hat{\theta})+\bar{\nu}(t), \qquad (21)$$

where $\bar{\nu}$ denotes the *network approximation error*[49,50] and is defined as

$$\begin{aligned}\bar{\nu}(t):=&(1-\exp(-\rho(t-T)))[\psi(q(t),\dot{q}(t),\tau(t))\\ &-\hat{\psi}(q(t),\dot{q}(t),\tau(t);\hat{\theta}^*)].\end{aligned}$$

The network approximation error $\bar{\nu}$ is a critical quantity, representing the minimum possible deviation between the unknown function $\psi$ and the neural network $\hat{\psi}$. Ideally we

https://doi.org/10.1017/S0263574703005204 Published online by Cambridge University Press

would like to have $\bar{\nu}(t)=0$; in other words, we wish to be able to approximate the off-nominal behavior of the system due to faults *exactly*. Unfortunately, this is not always possible, and hence the network approximation error, $\bar{\nu}(t)$, needs to be considered. Since $\psi$ is unknown, the approximation error $\bar{\nu}(t)$ cannot be calculated *a priori*. Several factors including the number of hidden layers in the neural network and the number of weights (denoted by $p$) influence the value of $\bar{\nu}(t)$. Universal approximator results for sigmoidal neural networks indicate that if $p$ is sufficiently large then $\bar{\nu}(t)$ can be made arbitrarily small on a compact region (assuming $\psi$ is continuous).[51]

The vector $\hat{\theta}^*$ (an "artificial" quantity required only for analytical purposes) is the value of $\hat{\theta}$ that minimizes the $L_\infty$-norm between the fault function, $\psi(q,\dot{q},\tau)$, and the neural network, $\hat{\psi}(q,\dot{q},\tau;\hat{\theta})$, over all $(q,\dot{q},\tau)$ in the compact learning domain $\mathcal{D}$, subject to the restriction that $\hat{\theta}^*$ belongs to a compact, convex region $\mathcal{M}_{\hat{\theta}}\subset\mathbb{R}^p$; i.e.

$$\hat{\theta}^*:=\arg\min_{\hat{\theta}\in\mathcal{M}_\theta}\left\{\sup_{(q,\dot{q},\tau)\in\mathcal{D}}\left|\psi(q,\dot{q},\tau)-\hat{\psi}(q,\dot{q},\tau;\hat{\theta})\right|\right\}.$$

In addition to the factors described above, the choice of $\mathcal{M}_{\hat{\theta}}$ influences the approximation error $\bar{\nu}(t)$ and the weight vector $\hat{\theta}^*$.[25]

By considering the Taylor series expansion of $\hat{\psi}(q,\dot{q},\tau;\hat{\theta})$ with respect to $\hat{\theta}$ around $(q,\dot{q},\tau,\hat{\theta}^*)$, the error dynamics (21) can be written as

$$\begin{aligned} e= & -\alpha e-M^{-1}(q)[F(\dot{q})+\tau_d] \\ & -\exp(-\rho(t-T))\hat{\psi}(q,\dot{q},\tau;\hat{\theta}^*)-Z\tilde{\theta} \\ & -\hat{\psi}_0(q,\dot{q},\tau;\hat{\theta})+\bar{\nu}(t), \end{aligned}\tag{22}$$

where $\tilde{\theta}:=\hat{\theta}-\hat{\theta}^*$ is the weight estimation error, and $\hat{\psi}_0(q,\dot{q},\tau;\hat{\theta})$ represents the higher-order terms (with respect to $\hat{\theta}$) of the Taylor series expansion. The higher-order term encapsulates the nonlinear parameterization structure of the sigmoidal neural network. It can be shown that $\hat{\psi}_0$ satisfies[52]

$$|\hat{\psi}_0(q,\dot{q},\tau;\hat{\theta})|\leq\delta(q,\dot{q},\tau;\hat{\theta})|\tilde{\theta}|,\quad\forall(q,\dot{q},\tau)\in\mathcal{D},\quad\hat{\theta}\in\mathcal{M}_\theta$$

with

$$\delta(q,\dot{q},\tau;\hat{\theta}):=\sup_{\theta\in S_\theta}\left\{\left|\frac{\partial\hat{\psi}(q,\dot{q},\tau;\theta)}{\partial\hat{\theta}}-\frac{\partial\hat{\psi}(q,\dot{q},\tau;\hat{\theta})}{\partial\hat{\theta}}\right|\right\},$$

where $S_{\hat{\theta}}:=\{\theta\in\mathbb{R}^p:\lambda\hat{\theta}+(1-\lambda)\hat{\theta}^*,\ 0\leq\lambda\leq1\}$. Note that for each $(q,\dot{q})\in\mathcal{D}$ we have $\lim_{\hat{\theta}\to\hat{\theta}^*}\hat{\psi}_0(q,\dot{q},\tau;\hat{\theta})=0$; i.e. as the weight estimates of the neural network, $\hat{\theta}(t)$ converges to the optimal weight, $\hat{\theta}^*$, the effect of the higher-order terms decreases.[52]

Now if we define $\nu:=\bar{\nu}-\hat{\psi}_0$, then the error equation (22) can be written as

$$\begin{aligned} \dot{e}= & -\alpha e-M^{-1}(q)[F(\dot{q})+\tau_d] \\ & -\exp(-\rho(t-T))\hat{\psi}(q,\dot{q},\tau;\hat{\theta}^*)-Z\tilde{\theta}+\nu(t) \end{aligned}\tag{23}$$

$$\dot{\phi}=-\rho\phi,\tag{24}$$

with $\phi(T)=-1$.

Consider the Lyapunov function candidate

$$V(e,\tilde{\theta})=\frac{1}{2}e^Te+\frac{1}{2}\tilde{\theta}^T\Gamma^{-1}\tilde{\theta}+\frac{c^2}{2\rho\alpha}\phi^2,$$

where $c:=\sup_{t\geq T}\hat{\psi}(q(t),\dot{q}(t),\tau(t);\hat{\theta}^*)$ is a constant which exists because of the smoothness assumption on $f$. This quantity is required for analytical purposes and is not required for the implementation of the proposed fault diagnosis algorithm.

The time derivative of $V$ along the solution of (24) and (9) is given by

$$\begin{aligned} \dot{V}= & -\alpha|e|^2-(M^{-1}(q)[F(\dot{q})+\tau_d])^Te+\phi(\hat{\psi}(q,\dot{q},\tau;\hat{\theta}^*))^Te \\ & -\chi^{*\,\tilde{\theta}^T\hat{\theta}}\frac{\hat{\theta}^T\Gamma Z^Te}{\hat{\theta}^T\Gamma\hat{\theta}}+\nu^Te-c^2\phi^2. \end{aligned}\tag{25}$$

We next show that

$$\chi^{*\,\tilde{\theta}^T\hat{\theta}}\frac{\hat{\theta}^T\Gamma Z^Te}{\hat{\theta}^T\Gamma\hat{\theta}}\geq0.\tag{26}$$

Suppose that $|\hat{\theta}|=M$ and $\hat{\theta}^T\Gamma Z^Te>0$; if these conditions are not satisfied then $\chi^*=0$ and the inequality holds trivially. By completing the squares, the term $\tilde{\theta}^T\hat{\theta}$ is expressed as

$$\begin{aligned} \tilde{\theta}^T\hat{\theta} & =\tilde{\theta}^T(\tilde{\theta}+\hat{\theta}^*) \\ & =\frac{1}{2}(\tilde{\theta}^T\tilde{\theta}+(\tilde{\theta}^T\tilde{\theta}+2\tilde{\theta}^T\hat{\theta}^*+\hat{\theta}^{*\,T}\hat{\theta}^*)-\hat{\theta}^{*\,T}\hat{\theta}^*) \\ & \geq\frac{1}{2}(M^2-|\hat{\theta}^*|^2). \end{aligned}$$

Since by definition $|\tilde{\theta}^*|<M$, we obtain that $\tilde{\theta}^T\hat{\theta}\geq0$ which proves the inequality (26).

Using this inequality, equation (25) can be rewritten as

$$\begin{aligned} \dot{V} & \leq-\alpha|e|^2-(M^{-1}(q)[F(\dot{q})+\tau_d])^Te+c\phi|e|+\nu^Te-\frac{c^2\phi^2}{\alpha} \\ & =-\frac{\alpha}{4}|e|^2 \\ & \quad-\left(\frac{\alpha|e|^2}{4}+(M^{-1}(q)[F(\dot{q})+\tau_d])^Te+\frac{|M^{-1}(q)[F(\dot{q})+\tau_d]|^2}{\alpha}\right) \\ & \quad+\frac{|M^{-1}(q)[F(\dot{q})+\tau_d]|^2}{\alpha}-\left(\frac{\alpha|e|^2}{4}-\nu^Te+\frac{|\nu|^2}{\alpha}\right)+\frac{|\nu|^2}{\alpha} \\ & \quad-\left(\frac{\alpha|e|^2}{4}-c\phi|e|+\frac{1}{\alpha}c^2\phi^2\right) \\ & \leq-\frac{\alpha|e|^2}{4}+\frac{1}{\alpha}\bar{\mu}(q,\dot{q},\tau,t)^2+\frac{1}{\alpha}\;|\nu|^2. \end{aligned}\tag{27}$$

By combining the above inequality which guarantees that $\dot{V} \le 0$ for

$$|e| \ge \frac{2\sqrt{\bar{\mu}^2 + |\nu|^2)}}{\alpha},$$

and the fact that $|\hat{\theta}| \le M$, we establish that $e, \tilde{\theta} \in L_\infty$. Furthermore, by integrating (27) from $t=0$ to $t=t_f$, we obtain

$$\int_0^t |e(t)|^2\, dt \le \frac{4}{\alpha}[V(0) - V(t_f)]$$

$$+\frac{4}{\alpha^2}\int_0^{t_f} \bar{\mu}^2\, dt + \frac{4}{\alpha^2}\int_0^{t_f} \nu(t)|^2\, dt$$

$$\le \lambda + \frac{4}{\alpha^2}\int_0^{t_f} (\bar{\mu}^2 + |\nu(t)|^2)\, dt,$$

where

$$\lambda = \frac{4}{\alpha}\left\{\sup_{t_f \ge 0}\, [V(0) - V(t_f)]\right\}. \qquad \square$$

The above theorem guarantees the uniform boundedness of the error signal and weights in the neural network based FD scheme. Furthermore, (19) implies that the extended $L_2$-norm of the estimation error vector is, at most, of the same order as the extended $L_2$-norm of the modeling uncertainties (represented by $\bar{\mu}$) and the network approximation error (described by $\nu$). Thus this theorem quantifies the overall performance of the learning scheme and the effect of the design parameter $\alpha$ on the performance.

**Remark 4.6.** *The learning algorithm only uses the knowledge of the size of the modeling uncertainties and does not require an* a priori *known bound on the network reconstruction error that results due to inaccurate modeling of the fault function $\psi$. Note that the projection operator in the weight update law deals with the network approximation error while the dead-zone operator is used to achieve robustness in the fault diagnosis architecture.*

## 5. SIMULATION EXAMPLES

This section illustrates the robust nonlinear fault diagnosis scheme for robotic manipulators, presented in this paper, via simulations. The robot system used for simulation purposes is a two rigid link robotic manipulator. The lengths of the two links are assumed to be $l_1$ and $l_2$ with point masses $m_1$ and $m_2$ at the distal ends of the two links respectively. The robotic link, with its parameters, used for simulations purposes is shown in Figure 7.

The dynamic equations of the robot manipulator are given in [20] while link parameters of the manipulator are given in Table 1.[53]

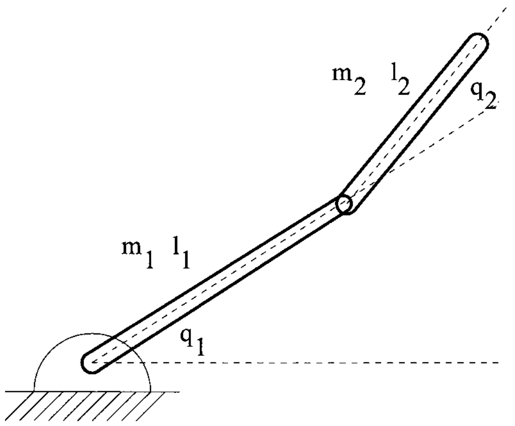

Fig. 7. Two link robotic manipulator.

Table I. Manipulator parameters.

| $j=1, 2$ | Link 1 | Link 2 |
|---|---|---|
| $l_j(m)$ | 1.0 | 1.0 |
| $m_j(Kg)$ | 0.5 | 2.0 |

We assume that the robotic system has the following modeling uncertainties:[53]

$$F = \begin{bmatrix} 0.5\ \mathrm{sgn}(\dot{q}_1) + 2\dot{q}_1 \\ 0.5\ \sin(\dot{q}_2) + 2\dot{q}_2 \end{bmatrix},$$

The control objective of the robotic system used in the simulations is to follow a desired trajectory. A number of techniques are available in the literature for deriving position control laws for robotic manipulators in the presence of modeling uncertainties and in the absence of faults (i.e. $\psi \equiv 0$).[20,54,55], Without any loss of generality, in this paper we use the computed-torque method to obtain a trajectory-tracking controller for the robotic manipulator described by (1). The controller derived using this method relies on the position and velocity measurements of each link, and the nominal model given by

$$\tau = M(q)\ddot{q} + V_m(q,\dot{q})\dot{q} + G(q) \qquad (28)$$

The structure of the computed torque controller is described by

$$\tau = M(q)(\ddot{q}_d + K_d\dot{\epsilon} + K_p\epsilon) + V_m(q,\dot{q})\dot{q} + G(q), \qquad (29)$$

where $q_d$ is the desired trajectory, $\epsilon := q_d - q$ is the tracking error, $K_d$ is an $n \times n$ diagonal matrix of damping gains, $K_p$ is an $n \times n$ diagonal matrix of position gains. If the robot dynamics are known exactly, then these matrices can be chosen so that the control law leads to an exponentially convergent tracking error.[20,21]

The controller gains in the control law (29) are chosen as follows: $K_p = 100I_{2\times 2}$, $\mathrm{K_d} = 20\mathrm{I}_{2\times 2}$, where $I_{n\times n}$ represents an identity matrix of dimension $n \times n$. The position and velocity of each joint is assumed to be available for measurement in order to be able to implement this controller.

The domain of interest is:

$$\mathcal{D} = \{[q_1\ q_2\ \dot{q}_1\ \dot{q}_2]^T : |q_1| \le 0.19,\ |q_2| \le 0.23,\ |\dot{q}_1| \le 0.36,$$
$$|\dot{q}_2| \le 0.63\}.$$

https://doi.org/10.1017/S0263574703005204 Published online by Cambridge University Press

The units of $q$ and $\dot{q}$ are radians and radians/sec respectively. The $L_2$-norm of the modeling uncertainty in $\mathcal{D}$, using the modeling uncertainties structure described above and the robotic manipulator parameters, is given by

$$\frac{1}{|4\cos^2(q_2)-5|}\{((1+\cos(q_2))(\sin(\dot{q}_2)+4\dot{q}_2+\sin(30t)) \\ -(\dot{q}_1/|\dot{q}_1|+4\dot{q}_1+\sin(30t)))^2 \\ +((1+\cos(q_2))(\dot{q}_1/|\dot{q}_1|+4\dot{q}_1+\sin(30t)) \\ -(4.5+4\cos(q_2))(0.5\sin(\dot{q}_2)+2\dot{q}_2+0.5\sin(30t)))^2\}^{1/2}.$$

The above expression shows that the size of the assumed modeling uncertainties depends on the system state. Since the modeling uncertainties are unknown (and hence the exact size of the uncertainties is unknown), this paper assumes that the size of the uncertainties has been determined empirically. Furthermore, this empirically determined non-uniform bound on modeling uncertainties in $\mathcal{D}$, for the purpose of simulations, is assumed to be

$$\bar{\mu}=4.4(|2\dot{q}_1-6\dot{q}_2|+1.0).$$

We use a three-layer sigmoidal neural network with 35 neurons in the hidden layer and two neurons in the output layer (corresponding to the velocity estimate of each of the two joints) for detecting system failures. The inputs to the neural network are the link position and velocity measurements, and the input torques $\tau$. The initial parameter vector of the network is chosen such that the output of the neural network is zero in $\mathcal{D}$ (corresponding to the no-failure situation). This is achieved by setting the output weights of the neural network to zero. We set the learning rate as $\Gamma=2.50I_{2\times 2}$ and the constant $\alpha=10$. The size of the hypersphere for the projection algorithm is selected as $M=10$ in this experiment.

The above manipulator is subject to two types of faults. The fault models and the behavior of the robotic manipulator with respect to these faults, and the performance of the fault diagnosis architecture are now discussed.

### 5.1. *Fault 1: Change in mass of link 1*

In the first simulation, we consider a fault which starts at $t=10s$. The fault is assumed to result in the change of mass of link 1 exponentially (in other words, the mass of link 1 is slowly falling). The mathematical form of this change in mass is assumed to be

$$m_1(t)=0.125-0.375e^{-0.5(t-10)},\ t>10.$$

Observe that this change in mass causes nonlinear changes in the terms $M(q)$, $V_m(q,\dot{q})$ and $G(q)$ of the robotic system (1) which results in a nonlinear change in the dynamics describing the robotic manipulator. Moreover, this nonlinear change depends on the current positions and velocities of the links.

Figure 8 shows the joint angle of each link, when the there is above fault. It can be seen that the the robotic system tracks the desired trajectories before the occurrence of the fault with a small tracking error, which results due to the presence of modeling uncertainties. The figure also shows that the fault causes considerable tracking error in the system for $t>10s$.

Based on the Figure 8, it can be inferred that both the modeling uncertainties and the fault in the system cause the robotic manipulator to deviate from the desired trajectory. The key however is that deviations caused by modeling uncertainties are "tolerable" (with the controller offering necessary robustness) while the deviations caused by faults are not "acceptable" from both the systems perspective (deviation into not-desired domain) and the controller perspective (due to the inability of the controller to maintain robustness). Hence, even though the modeling uncertainties and the fault cause deviations in the robotic system trajectory, it is desirable to distinguish the effects of faults

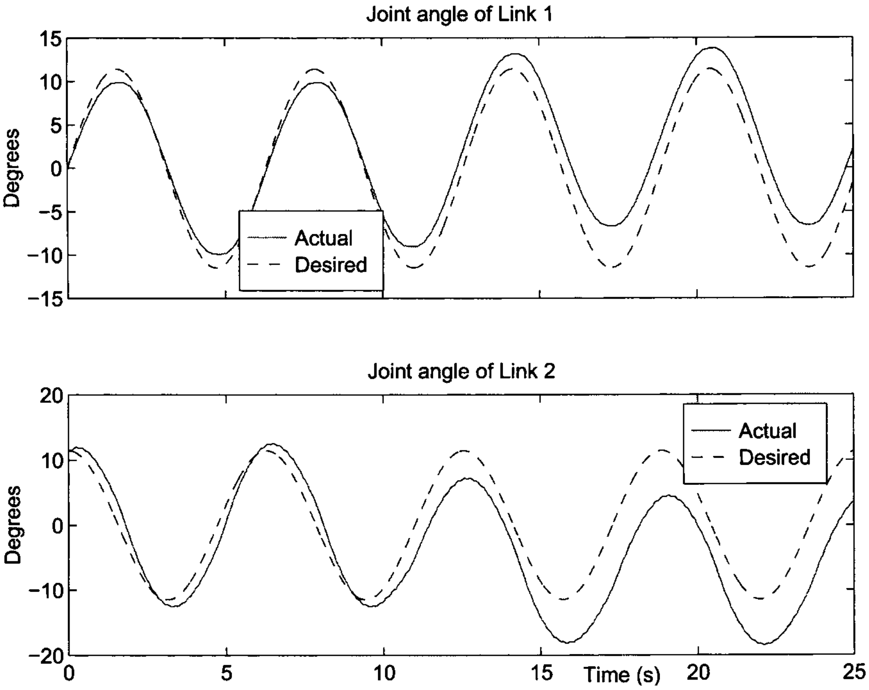

Fig. 8. Joint angles of the manipulator with fault that results in gradual change in mass of link 1.

https://doi.org/10.1017/S0263574703005204 Published online by Cambridge University Press

and the effects of modeling uncertainties; i.e. it is important to identify the deviations caused by the fault for $t > 10s$ in the above figure.

Figure 9 shows the performance of the fault diagnosis architecture in the presence of this fault. Specifically, the figure shows the time histories of the fault functions and the neural network outputs (i.e., the estimates of the fault functions). It can be seen from the figure that the outputs of the neural networks remain zero prior to the fault (i.e. for $t < 10s$), demonstrating the robustness properties of the robust fault diagnosis scheme presented in this paper. Furthermore, the neural network outputs jump to non-zero values soon after the occurrence of the fault. Thus the neural network outputs can be effectively used for the detection of system faults in this simulation example.

The learning capability of the fault diagnosis architecture is also illustrated in Figure 9. This figure shows that the neural network outputs provide a good approximation of the fault functions in each link. The presence of modeling uncertainties and the network approximation error causes the observed mismatch between the fault function and the neural network outputs.

Figure 10 shows the low-pass filtered signals of the size of the modeling uncertainty and the fault function. The low-pass filter used to obtain these signals is the same as the one used in obtaining the estimates of the robotic manipulator joint velocities. From the figure, it can be inferred that the size of the fault function becomes non-zero at $t = 10s$. The size of the fault function exceeds the size of the modeling uncertainties at about $t = 10.6s$. Comparing Figure 10 with Figure 9, it can be seen that the neural network output becomes non-zero approximately at $t = 10.6s$ which is time at which the size of the filtered function exceeds the filtered modeling uncertainty size.

In summary, the fault diagnosis scheme proposed in this paper is able not only to detect the occurrence of a fault in the presence of modeling uncertainties, but also provides a model of the fault (described by $\hat{\psi}(q,\dot{q},\tau;\hat{\theta})$) in the robotic manipulator.

In the above simulation study we have considered a fault that causes a 75% gradual change in the mass of link 1 in order to illustrate the ability of the accommodation scheme to handle such large faults. Although such large faults are relatively easily to detect, the proposed fault diagnosis scheme can detect smaller faults as long as the ratio between the fault function magnitude and the size of the modeling uncertainties is sufficiently large. Further simulation study indicated that for the uncertainty level considered here, the fault diagnosis scheme is able to detect change of mass greater than or equal to 10%. As the analysis in Section 4 shows, the performance of the fault diagnosis scheme can be enhanced by improving the accuracy of the nominal model.

### 5.2. *Fault 2: Complex nonlinear function*

In the second simulation, we consider a fault that occurs due to a tangle of complex factors; this fault is assumed to manifest itself as a nonlinear change (in the robotic system dynamics) described by

$$\psi(q,\dot{q},\tau) =$$

$$(1-\exp(-0.5(t-10)))\begin{bmatrix} 7.5q_1^2+10q_1^2\dot{q}_2^2+7q_2+17] \\ 50q_1q_2+12.5 \end{bmatrix}. \quad (30)$$

This fault is also assumed to occur at $t = 10s$.

Figure 11 shows the joint angle of each link, when a fault described by (30) occurs. It can be once again seen that the tracking of the robotic system deteriorates considerably after the occurrence of the fault (i.e. for $t > 10s$).

Figure 12 shows the plot of the neural network output and the fault function. This figure also shows that the neural

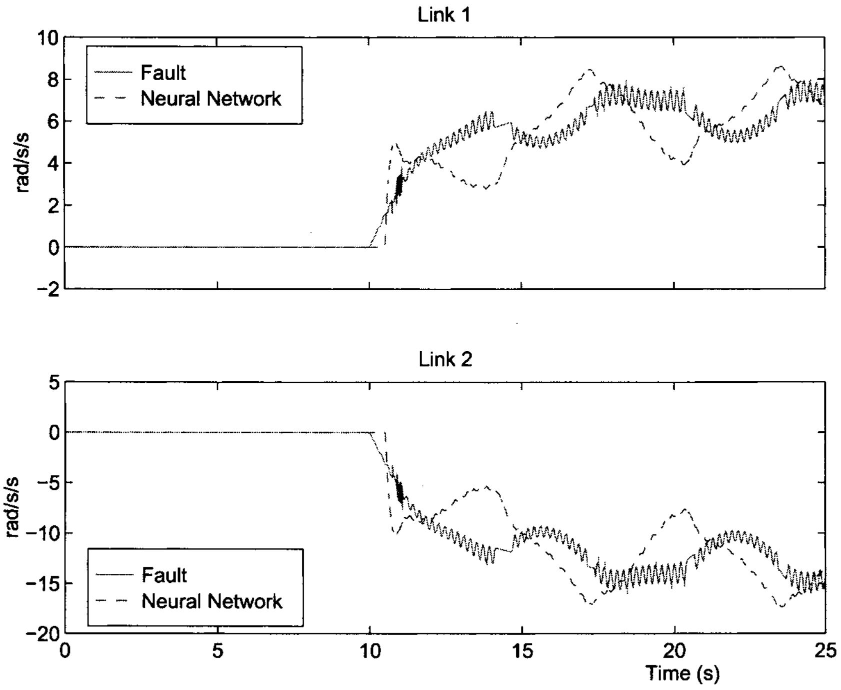

Fig. 9. Time histories of the fault functions and the neural network output when the fault results in gradual change in mass of link 1.

https://doi.org/10.1017/S0263574703005204 Published online by Cambridge University Press

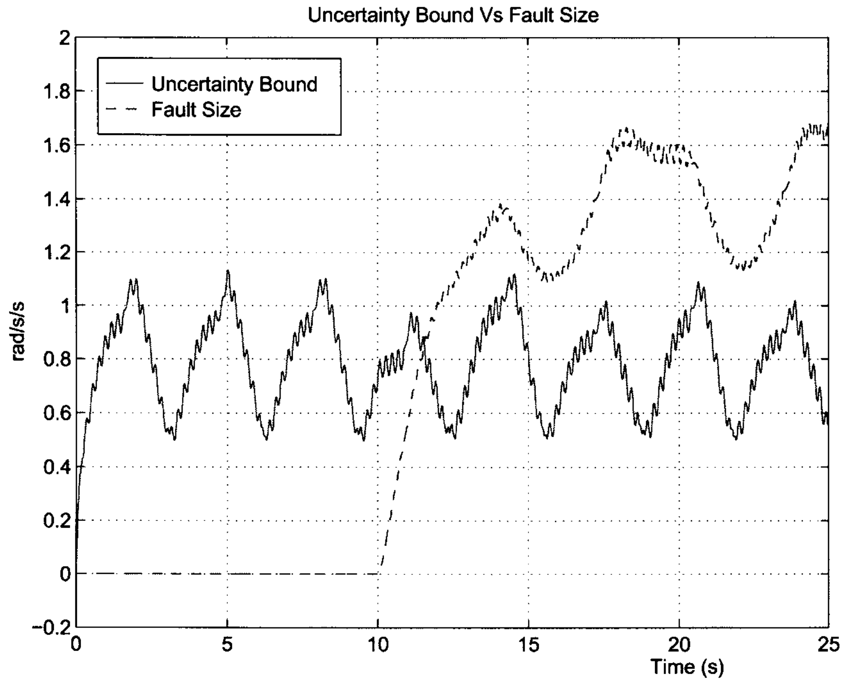

Fig. 10. Time histories of the filtered signals of the size of the modeling uncertainties and the size of the fault function when the fault results in gradual change in mass of link 1.

network output remains prior to $t=10s$ and become non-zero shortly after the fault occurs. From this figure, it can also be inferred that the neural network provides a good approximation of the fault function.

Figure 13 shows the low-pass filtered signals of the size of the modeling uncertainty and the fault function. From the figure, it can be inferred that the size of the fault function becomes non-zero at $t=10s$ and the size exceeds the size of the modeling uncertainties at about $t=11.4s$. Comparing Figure 13 with Figure 12, it can be seen that the neural network output becomes non-zero approximately at $t=11.0s$.

### *5.3. Example application of the the Neural Network Model: Fault accommodation*

The fault model provided by the neural network can be used for identifying the failure mode by comparing it with any known failure modes. *Signatures* of such known failure modes of the relevant robotic manipulator can be stored in a post-failure model bank. Pattern recognition techniques

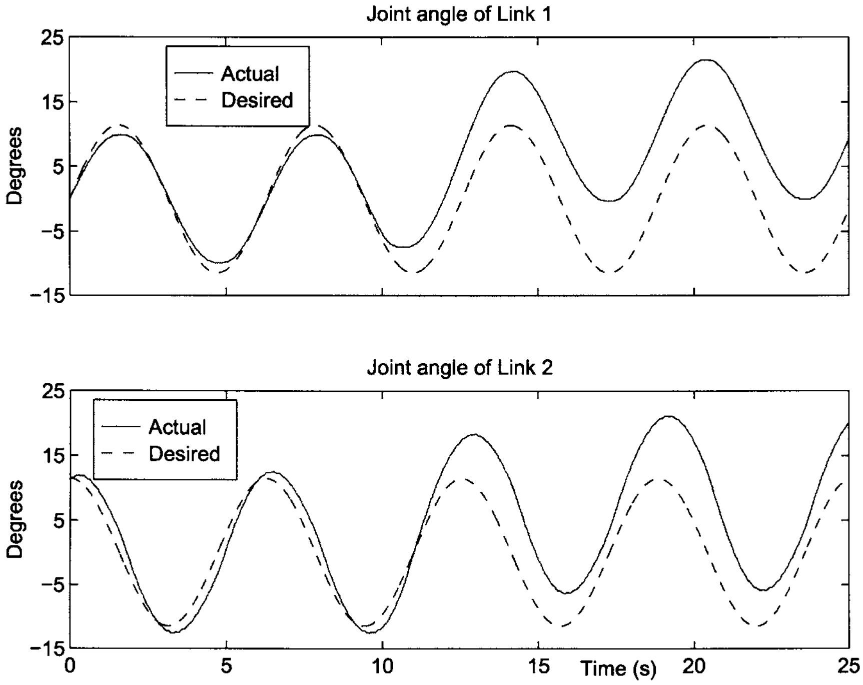

Fig. 11. Joint angles of the manipulator when the fault occurs due to a tangle of complex factors.

https://doi.org/10.1017/S0263574703005204 Published online by Cambridge University Press

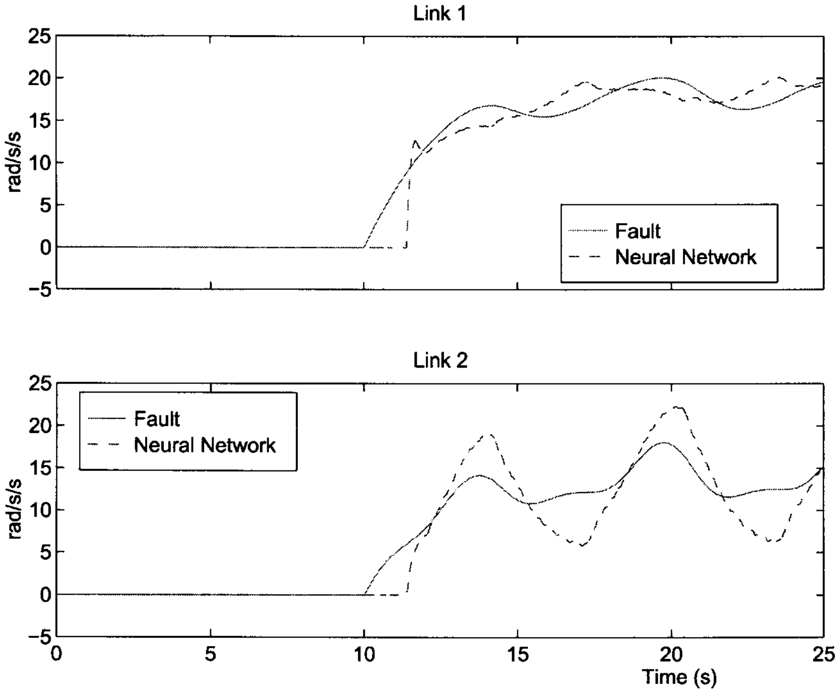

Fig. 12. Time histories of the fault functions and the neural network output when the fault occurs due to a complex tangle of factors.

and associative memories provide effective means for comparison purposes. It is important that the post-failure model bank include (and be updated periodically) the signatures of recurring and well-understood failures. In many cases, however, unexpected failure situations are encountered. This may occur as a result of minimal knowledge of possible manipulator faults, unexpected breakdowns, or even as a result of errors in assembling the post-failure model bank. The construction of a post-failure model bank and its effective use for fault identification and fault isolation based on the post-fault models provided by the proposed fault diagnosis methodology needs further investigation.

The post-fault model provided by the sigmoidal neural network can also be used for failure accommodation purposes.[56] One of the nonlinear control tools available for controller reconfiguration purpose is feedback linearization.[57] The main idea behind feedback linearization is to transform the nonlinear system into a system with linear dynamics through a change of coordinates and nonlinear feedback. If feedback linearization is achievable (see reference [57] for conditions under which a system is

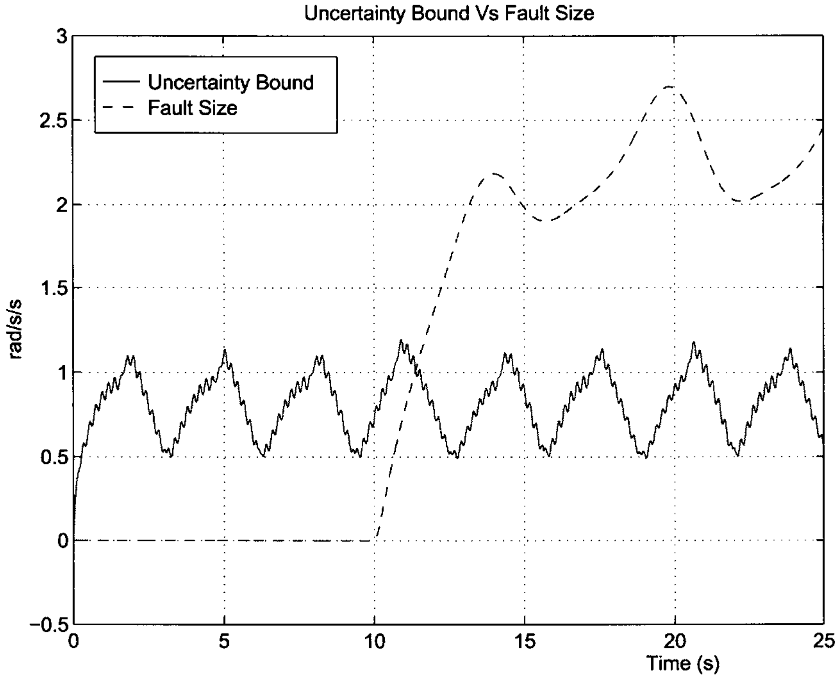

Fig. 13. Time histories of the filtered signals of the size of the modeling uncertainties and the size of the fault function when the fault occurs due to a complex tangle of factors.

https://doi.org/10.1017/S0263574703005204 Published online by Cambridge University Press

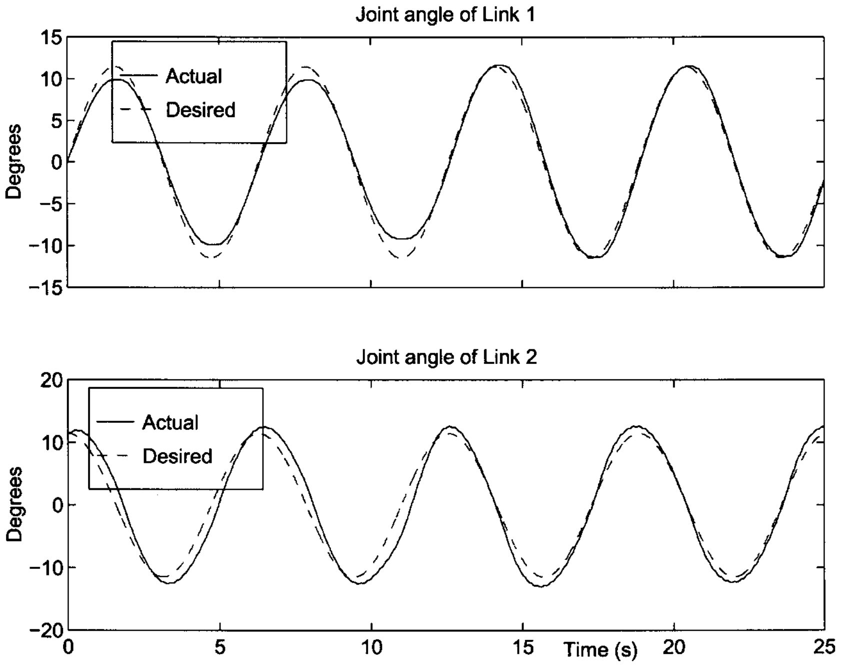

Fig. 14. Joint angles plots illustrating accommodation when the fault results in a gradual change in the mass of link 1.

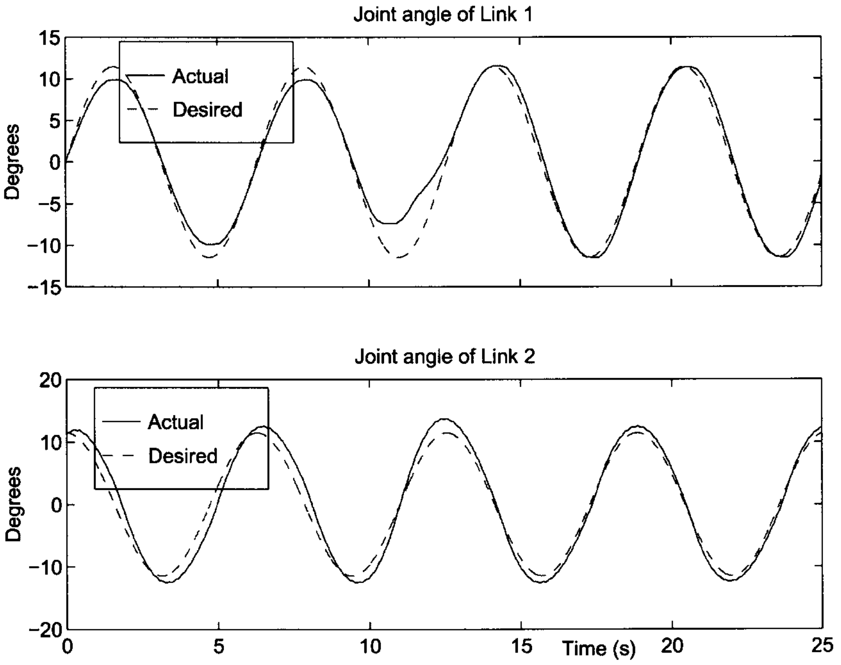

Fig. 15. Joint angles plots illustrating accommodation when the fault occurs due to a tangle of complex factors.

feedback linearizable), then it is possible to obtain, first, cancellation of the nonlinear functions and, second, desired closed-loop performance through the application of powerful linear control design methodologies. We employ this technique in this paper to reconfigure the control law (29).

Based on the post-fault robotic system model given by

$$\tau = M(q)\ddot{q} + V_m(q,\dot{q})\dot{q} + G(q) - M(q)\hat{\psi}(q,\dot{q},\tau;\hat{\theta}),$$

and using the feedback linearization technique, the control law (29) can be reconfigured to

$$\tau_\tau = \tau - M(q)\hat{\psi}(q,\dot{q},\tau;\hat{\theta})), \tag{31}$$

where $\tau_r$ is the reconfigured control law.

Figure 14 shows the trajectories of the robotic system when the reconfigured control law is used. By comparing this figure with Figure 8 it can be concluded that the trajectory tracking error is considerably reduced by using the reconfigured control law (31). The accommodation of the fault (30) is illustrated in Figure 15. By once again comparing this figure with Figure 11 it can seen that tracking of the desired trajectories is considerably improved. Thus the proposed fault diagnosis scheme helps in providing a post-failure model that can be used for accommodating system failures via control reconfiguration.

https://doi.org/10.1017/S0263574703005204 Published online by Cambridge University Press

## 6. CONCLUSIONS

A neural network-based learning methodology with guaranteed robustness, sensitivity, stability properties is described for the detection of faults in robotic systems with modeling uncertainties. The nonlinear modeling approach is one of the key features of the algorithm proposed in this paper. This feature enables the development of a post-fault robotic system model which can be used to isolate as well as accommodate robotic system faults. Simulation examples are used to illustrate the algorithm in the detection of faults in a two-link robotic manipulator.

## References

1. B. Fieyermuth, "An approach to model-based fault diagnosis of industrial robots", *Proceedings of the 1991 IEEE Conference on Robotics and Automation* (Sacramento, CA) (1991) pp. 1350–1356.
2. M.L. Visinsky, J.R. Cavallaro and I.D. Walker, "Expert system framework for fault detection and fault tolerance in robotics", *Computers and Electrical Engineering* **20**, 421–435 (1994).
3. A. Pouliezos, S. Tzafestas and G. Stavrakakis, "Fault detection and location in MRAC robotic systems", **In:** *System Fault Diagnostics, Reliability and Related Knowledge-Based Approaches* (S. Tzafestas, M. Singh and G. Schmidt, eds.) (Reidel, 1986) pp. 203–217.
4. J. Wunnenberg and P.M. Frank, "Dynamic model-based incipient fault detection concept for robots", *Proceedings of the IFAC 11th Triennial World Congress* (Tallinn, Estonia) (1990) pp. 61–66.
5. F. Zanaty, "Consistency checking techniques for the space shuttle remote manipulator design", *SPAR Journal of Engineering and Technology* **2**, No. 1, 40–49(1993).
6. M. Hashimoto and R. Paul, "Integration of manipulator joint sensor information for robust robot control", *Proceedings of the 26th Conference on Decision and Control* (Los Angeles, CA) (1987) pp. 603–604.
7. H. Schneider and P.M. Frank, "Observer-based supervision and fault detection in robots using nonlinear and fuzzy logic residual evaluation", *IEEE Transactions on Control Systems Technology* **4**, 274–282 (1996).
8. M.L. Visinsky, J.R. Cavallaro and I.D. Walker, "A dynamic fault tolerance framework for remote robots", *IEEE Trans. on Robotics and Automation* **11**, 477–490 (1995).
9. F. Caccavale and I.D. Walker, "Observer-based fault detection for robot manipulators", *Proceedings of the 1997 IEEE Conference on Robotics and Automation* (Albuquerque, NM) (1997) pp. 2881–2887.
10. M.H. Terra and R.A. Mendes, "Fault detection and isolation applied in a manipulator robot via luenberger observers", *Proc. IFAC Symposium on Fault Detection, Supervision and Safety for Processes (SAFEPROCESS)* (Budapest, Hungary) (June, 2000), pp. 524–529.
11. F. Kerestecioglu and B.S. Nalbantoglu, "Fault detection in robot manipulators using statistical tests", **In:** *Recent Advances in Mechatronics* (O. Kaynak, M. Ang and S. Tosunoglu, eds.) (Singapore: Springer, 1999) pp. 481–492.
12. W.E. Dixon, I.D. Walker, D.M. Dawson and J.P. Hartranft, "Fault detection for robotic manipulators with parametric uncertainties", *Proceedings of the 1999 IEEE Conference on Robotics and Automation* (San Francisco, CA) (1999) pp. 3628–3634.
13. M.H. Terra and R. Tinos, "Fault detection and isolation in robotic manipulators: a comparison among three architectures for residual analysis", *Journal of Robotic Systems* **18**, No. 3, 357–374 (2001).
14. M. Soika, "Grid-based fault detection and calibration of sensors on mobile robots", *Proceedings of the 1997 IEEE Conference on Robotics and Automation* (Albuquerque, NM) (1997) pp. 2589–2594.
15. L. Notash, "Joint sensor fault detection for fault tolerant parallel manipulators", *Journal of Robotic Systems* **17**, No. 3, 149–1557 (2000).
16. J.-H. Shin and J.-J. Lee, "Fault detection and robust fault recovery control for robotic manipulators with actuator failures", *Proceedings of the 1999 IEEE Conference on Robotics and Automation* (Detroit, MI) (1999) pp. 861–866.
17. A.T. Vemuri, M.M. Polycarpou and S.A. Diakourtis, "Neural network-based fault detection and accommodation in robotic manipulators", *IEEE Iransactions on Robotics and Automation* **48**, 342–348 (1999).
18. A.T. Vemuri and M.M. Polycarpou, "Neural network-based robust nonlinear fault diagnosis in robotic systems", *IEEE Transactions on Neural Networks* **8**, 1410–1420 (1997).
19. R. Tinos and M.H. Terra, "Fault detection and isolation in robotic manipulators and the radial basis function network trained by the kohonen's self organizing map", *Proceedings of the Vth Brazilian Symposium on Neural Networks* (Belo Horizonte, MG, Brazil) (1998) pp. 603–604.
20. F.L. Lewis, C.T. Abdallah and D.M. Dawson, *Control of Robot Manipulators* (McMillan, 1993).
21. M.W. Spong and M. Vidyasagar, *Robot Dynamics and Control* (Wiley, 1989).
22. S. Haykin, *Neural Networks: A Comprehensive Foundation* (McMillan, 1994).
23. D.A. White and D.A. Sofget (eds), *Handbook of Intelligent Control: Neural, Fuzzy, and Adaptive Approaches* (Van Nostrand and Reinhold, 1993).
24. J.A. Farrell, T. Berger and B.D. Appleby, "Using learning techniques to accommodate unanticipated faults", *IEEE Control Systems* **13**, 40–49 (1993).
25. M.M. Polycarpou and A.T. Vemuri, "Learning methodologies for failure detection and accommodation", *IEEE Control Systems Magazine* **15**, 16–24 (1995).
26. P.M. Frank, "On-line fault detection in uncertain nonlinear systems using diagnostic observers: a survey", *Int. J. System Science* **25**, 2129–2154 (1994).
27. R.J. Patton, "Robust model-based fault diagnosis: the state of the art", *Proc. IFAC Symposium on Fault Detection, Supervision and Safety for Processes (SAFEPROCESS)* (Espoo, Finland) (June, 1994) pp. 1–24.
28. J.J. Gertler and M.M. Kunwer, "Optimal residual decoupling for robust fault diagnosis", *Proc. of Int. Conf. on Fault Diagnosis: TOOLDIAG'93* (Toulouse, France) (April, 1993) pp. 5–7.
29. R.J. Patton and J. Chen, "Robust fault detection of jet engine sensor systems by using eigenstructure assignment", *J. Guidance, Contr., Dynamics* **15**, 1491–1496 (1992).
30. R.J. Patton and J. Chen, "Optimal unknown input distribution matrix selection in robust fault diagnosis", *Automatica* **29**, 837–841 (1993).
31. A. Emami-Naeini, M.M. Akhter and S.M. Rock, "Effect of model uncertainty on failure detection: the threshold selector", *IEEE Transactions on Automatic Control* **33**, 1106–1115 (1988).
32. B.A. Walker, "Fault detection threshold determination using Markov theory", **In:** *Fault Diagnosis in Dynamic Systems, Theory and Application* (R.J. Patton, P.M. Frank and R.N. Clark, eds) (Englewood Cliffs, NJ: Prentice Hall, 1989) pp. 447–508.
33. H. Schneider and P.M. Frank, "Fuzzy logic-based threshold adaption for fault detection in robots", *Proceedings of the Third IEEE Conference on Control Applications* (Scotland, UK) (1994) pp. 1127–1132.
34. R.J. Patton, P.M. Frank and R.N. Clark, *Fault Diagnosis in Dynamic Systems: Theory and Application* (Englewood Cliffs, NJ: Prentice Hall, 1989).

https://doi.org/10.1017/S0263574703005204 Published online by Cambridge University Press

35. P.M. Frank, "Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy – a survey and some new results", *Automatica* **26**, 459–474 (1990).
36. P.M. Frank and N. Kiupel, "Fuzzy supervision and application to lean production", *Int. J. System Science* **24**, 1935–1944 (1993).
37. J. Jiang and F. Jia, "A robust fault diagnosis scheme based on signal modal estimation", *Int. J. Control* **62**, 461–475 (1995).
38. H.J. Manquez and C.P. Diduch, "Sensitivity of failure detection using generalized observers", *Automatica* **28**, 837–840 (1992).
39. P.M. Olin and G. Rizzoni, "Design of robust detection filters", *Proc. American Control Conf.* (June, 1991) pp. 1522–1527.
40. J.J. Gertler and D. Singer, "A new structural framework for parity equation-based failure detection and isolation", *Automatica* **26**, 381–388 (1990).
41. J.J. Gertler, *Fault Detection and Diagnosis in Engineering Systems* (Marcel Dekker, 1998).
42. J. Chen and R.J. Patton, *Robust Model-Based Fault Diagnosis for Dynamic Systems* (Kluwer Academic Publishers, 1999).
43. A.T. Vemuri and M.M. Polycarpou, "Robust fault diagnosis in input-output systems", *Int. J. Control* **68**, No. 2, 343–360 (1997).
44. M.A. Demetriou and M.M. Polycarpou, "Incipient fault diagnosis of dynamical systems using on-line approximators", *IEEE Transactions on Automatic Control* **48**, 1612–1617 (1998).
45. A.T. Vemuri and M.M. Polycarpou, "On-line approximation-based methods for robust fault detection", *Proceedings of the 13th IFAC World Congress* (San Francisco, CA) (July, l996) **Vol. K**, pp. 319–324.
46. P.A. Ioannou and J. Sun, *Stable and Robust Adaptive Control* (Prentice-Hall, 1995).
47. M.M. Polycarpou and P.A. Ioannou, "On the existence and uniqueness of solutions in adaptive control systems", *IEEE Transactions on Automatic Control* **38**, 474–479 (1993).
48. M.M. Polycarpou and A. Trunov, "Learning approach to nonlinear fault diagnosis: Detectability analysis", *IEEE Transactions on Automatic Control* **45**, 806–812 (2000).
49. M.M. Polycarpou and P.A. Ioannou, "Modeling, identification and stable adaptive control of continuous-time nonlinear dynamical systems using neural networks", *Proceedings of the 1992 American Control Conference* (1992) pp. 36–40.
50. R.M. Sanner and J.-J.E. Slotine, "Gaussian networks for direct adaptive control", *IEEE Transactions on Neural Networks*, 837–863 (1992).
51. K. Hornik, M. Stinchcombe and H. White, "Multilayer feedforward networks are universal approximators", *Neural Networks* **2**, 359–366 (1989).
52. J.D. DePree and C.W. Swartz, *Introduction to Real Analysis* (Wiley, 1988).
53. Z. Qu, D.M. Dawson and J.H. Dorsey, "Exponentially stable trajectory following of robotic manipulators under a class of adaptive controls", *Automatica* **28**, 579–586 (1992).
54. D.M. Dawson, M.M. Bridges and Z. Qu, *Nonlinear Control of Robotic Systems for Environmental Waste and Restoration* (Prentice Hall, 1995).
55. T.D. Sanger, "Neural network learning control of robot manipulators using gradually increasing task difficulty", *IEEE Transactions on Robotics and Automation* **10**, 323–333 (1994).
56. M.M. Polycarpou, "Fault accommodation for a class of multivariable nonlinear dynamical systems using a learning approach", *IEEE Transactions on Automatic Control* **46**, 736–742 (2001).
57. A. Isidori, *Nonlinear Control Systems* (Springer-Verlag, 2nd ed., 1989).

https://doi.org/10.1017/S0263574703005204 Published online by Cambridge University Press