ROBUST NONLINEAR ANALYTIC REDUNDANCY FOR FAULT DETECTION AND ISOLATION OF ROBOTIC SYSTEMS

By

Bibhrajit Halder

Dissertation

Submitted to the Faculty of the

Graduate School of Vanderbilt University

in partial fulfillment of the requirements

for the degree of

DOCTOR OF PHILOSOPHY

in

Mechanical Engineering

December, 2006

Nashville, Tennessee

Approved:

Professor Nilanjan Sarkar

Professor Michael Goldfarb

Professor Akram Aldroubi

Professor George E. Cook

Professor Eric J. Barth

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

It is a long path to reach this point where I am writing acknowledgement for my dissertation. I did not walk alone to reach here. There are many without whom this walk would not be possible or have any meaning. I like to express my deep gratitude to my advisor Dr. Nilanjan Sarkar for his guidance, effort, and immense patience during the time of my work. He teaches me a lot both inside and outside the research world. Also, I gratefully acknowledge the ARO grant DAAD19-02-1-0160 and ONR grant N00014-03-1-0052 and N00014-06-1-0146 that partially supported this work.

I am grateful to have Dr. Eric Barth, Dr. Michael Goldfarb, Dr. George E. Cook, and Dr. Akram Aldroubi in my dissertation committee. I appreciate their helpful suggestion and comments, which makes this work better. I like to thank Dr. Kenneth D. Frampton and Dr. Xenofon D. Koutsoukos for their discussion and time.

I like to thank my friends at Robotics and Automations Systems Laboratory, Duygun Erol, Vishnu Mallapragrada, Naim Sidek, Changchun Liu, and Karla G. Conn for their help in my research experiments. I am thankful to all my friends at Vanderbilt whose presences make my life at Vanderbilt memorable.

Finally, I like to thank my family and wife for their support. Especially, I like to mention my brother Bijit Halder who constant encouragement is a gift in my life.

LIST OF FIGURES

Page

LIST OF TABLES

CHAPTER I

INTRODUCTION AND SUMMARY

Introduction

The demand for automation in modern society has been increasing steadily during the last few decades. Robotic systems have played an important role in automation that includes manufacturing, assembly, and biotechnology to name a few. In addition, there is a growing need for unmanned operation in different service and research sectors such as search and rescue, nuclear waste clean up, planetary exploration and others. Such complex applications increase the possibility of the system faults that are characterized by critical and unpredictable changes in the system dynamics. The consequences of the system faults can be extremely serious in terms of not only economic loss, but also environmental impact and even human lives. Therefore the ability to adapt to faults is important for the reliability and safety of the system. One way to address these needs is to design a fault tolerant control system (FTCS). Generally, the way to make a system fault tolerant consists of two steps:

(1) **Fault diagnosis**: The existence of a fault has to be detected and the fault needs to be isolated.

(2) **Fault accommodation**: The controller has to be able to adapt to the faulty situation so that the overall system continues to satisfy its goal.

There are significant research activities in the development of new methodologies for automated fault diagnosis and fault-tolerant control. However, unlike the fault diagnosis for linear systems, which has been investigated extensively in the literature, the fault diagnosis problem for nonlinear uncertain systems has received less attention.

The motivation for this dissertation stems from the above significant issues. We focus on the fault diagnosis of nonlinear uncertain systems. First, we investigate the problem of robust fault detection for a class of input affine nonlinear systems that include most robotic systems. In this research, a model-based fault detection method is used. A successful fault detection scheme should be robust to unavoidable modeling uncertainty, such as external disturbance and model-plant-mismatch (MPM), thus preventing any false alarm. We develop a new robust nonlinear fault detection methodology using nonlinear analytic redundancy technique. The detailed theoretical development along with the simulation results are presented in Manuscript 1. We investigate both the sensor and actuator faults and experimentally verified the robust nonlinear analytic redundancy (RNLAR) method on a PUMA 560 robotic manipulator. The experimental results are given in Manuscript 2. We further investigate the relationship between the order of redundancy and the robustness. We proposed a theorem in this regard and experimentally confirmed the claim in Manuscript 3. Finally, we investigate the problem of fault isolation, which is discussed is Manuscript 4. A Pioneer 3DX mobile robot is used to experimentally verify the fault detection and isolation mechanism.

Manuscript 1: A Robust Nonlinear Analytic Redundancy Framework for Actuator Fault Detection and its Application to Robotics

Background

Fault detection and isolation (FDI) techniques are broadly classified into two classes: model-free approaches and model-based approaches. Traditionally, model-free approaches use hardware redundancy method for FDI. The major problems with the hardware redundancy method are the extra cost and the additional space required to accommodate the equipment. Model-based fault detection, on the other hand, utilizes the mathematical model of the plant to generate residuals. Residuals are measures of discrepancy between the expected and the measured system behavior. A substantial research effort has been invested in model-based FDI during the last few decades. Given the success of model-based approach and the powerful mathematical tools it provides, we choose to concentrate on model-based methods. Model-based method gives better results for robotic systems where an approximate model is available.

Some important survey papers in the model-based fault detection methods are [2-4]. The fundamental concept of model-based fault detection is analytical redundancy (AR). The basic idea of AR is the comparison between the actual behavior of the monitored plant and the behavior of a mathematical plant. AR is an especially interesting and useful technique as it allows us to explicitly derive the maximum number of model-based linearly independent consistency tests for a system [1]. Another important feature of AR is that it guarantees that the test residuals generated by the techniques will test the entire space of "observable" faults [1]. Implementation methods of AR can be classified into two groups: 1) indirect implementation, based on diagnostic observers, and 2) direct implementation based on the parity relation technique. Conceptually, the direct implementation based on parity relation is more straightforward than the observer-based method.

The origin of parity relation based AR can be found in [1] for linear systems. The detail description of parity relation for linear systems is given in [5]. The standard AR fault detection technique is effectively limited to linear systems. It is worth noting that most robotic systems are modeled as nonlinear systems. The linear AR concept was later extended to nonlinear systems in [6]. The authors in [6] named this new technique nonlinear analytic redundancy (NLAR) for fault detection. Note that [6] did not consider fault detection in the presence of MPM and process disturbance. However, MPM and process disturbances almost always exist in practical systems. A model dependent fault detection scheme may not be useful under considerable MPM and process disturbances. Thus a robust fault detection method that does not require a perfect model will be valuable.

In the AR literature robustness issue is discussed mostly for linear systems. In [1] robust residual generation was considered for linear systems based on an optimization technique. Recently, in [7], [8] the authors extended the method presented in [1] to design the primary residual considering both the MPM and process disturbances in linear systems. As far as nonlinear systems are concerned, there is a lack of literature on parity relation based robust fault detection method.

In summary, the majority of robust FDI methods are applicable only for linear systems. But, most robotic systems are modeled as nonlinear systems to capture their complex dynamics. Therefore, there is a need for robust FDI method for nonlinear systems. We propose a new robust nonlinear analytic redundancy (RNLAR) fault detection technique. The proposed RNLAR fault detection technique accommodates both the MPM and process disturbances for multivariable dynamic systems. In this technique, we extend the robustness idea, used in [7] for linear systems, into the nonlinear domain.

Summary of Contribution

The main contribution of this part is the development of a rigorous method for deriving robust nonlinear analytic redundancy (RNLAR) test residual that can be applied to a wide range of nonlinear systems. RNLAR technique is applicable to systems described by input affine nonlinear ordinary differential equations. The RNLAR method extends the linear AR into nonlinear systems. It also extends the NLAR to include the MPM and uncertainty of the system. The effectiveness of the method is verified by simulation on a mobile robot. Manuscript 1 is based on the following papers:

- Halder. B and Sarkar. N, "*Robust Fault Detection Based on Nonlinear Analytic Redundancy Techniques With Application to Robotics,*" Proceedings of International Mechanical Engineering Congress and Exposition (IMECE), Orlando, Florida, November 5-11, 2005-81098.
- (Submitted) Halder. B and Sarkar. N, "*A Robust Nonlinear Analytic Redundancy Framework for Actuator Fault Detection and its Application to Robotics,*" Journal of Automatica.

Manuscript 2: Robust Fault Detection of Robotic Manipulator

Background

Robotic manipulation systems played an important role in automation industries that include manufacturing, assembly, biotechnology to name a few. However, notwithstanding their widespread applicability and use, robotic manipulators are known to fail under normal operations [9] due to various faults that include sensor and actuator faults, and component failure. Typical faults are caused by broken or bias sensor, wear in mechanical components, overheating, and locked or damaged actuator. The likelihood of developing fault increases both with the complexity and versatility of the manipulator mechanism and the uncertainty of application domains. Consequently, fault detection is important for the reliability and safety of robotic manipulators. A nonlinear fault detection method is needed for robotic manipulator.

Various nonlinear diagnostic observer designs are proposed and implemented on robotic manipulators to detect sensor and actuator faults [10-13]. Most of the works in fault detection consider either a sensor fault or an actuator fault. In [10], the authors proposed a method based on generalized momenta for actuator fault detection. However, the proposed method could not detect sensor faults and was not robust in the presence of disturbance, noise and model-plant-mismatch (MPM). In [14] the partial actuator fault was considered in detail but sensor fault detection method was not discussed. In papers [15-16] only sensor fault detection for robot manipulator was considered.

In summary, a fault detection method that includes both the sensor and actuator faults and considers the modeling uncertainty is still lacking in the literature but will be helpful for fault detection of the robotic manipulators. We implement our RNLAR method to detect sensor and actuator faults of the robotic manipulator.

Summary of Contribution

In this part of the research, the RNLAR method is extended to encompass the sensor fault detection. The RNLAR method is implemented on a PUMA 560 robotic manipulator. We present the experimental results under different sensor and actuator faults. Manuscript 2 is based on the following papers:

- (Accepted) Halder. B and Sarkar. N, "*Robust Fault Detection of Robotic Manipulator*," International Journal of Robotics Research.
- Halder. B and Sarkar. N, "*Robust Fault Detection of Robotic Systems: New Results and Experiments*," Proceedings of International Conference on Robotics and Automation (ICRA), Orlando, Florida, May 15-19, 2006, pp. 3795-3800.

Manuscript 3: Impact of the Order of Redundancy Relation in Robust Fault Detection of Robotic Systems

Background

Robust fault detection is important for safe and reliable robotic applications. The first step to successful fault detection is residual generation. Various model-based methods have been developed in the literature using the analytic redundancy (AR) method [1-3]. The AR method is suitable for robotic application where approximate model is available. To address the robustness issue, given in [1] the authors have proposed an optimization method to select a parity vector from the parity space. They described the order of redundancy relation as the 'memory span' of the redundancy relation. This work was later extended by various researchers in [17], [18]. Most recently in [7], [8] the authors designed optimal primary residual, which considered both the model-plant-mismatch (MPM) and process disturbances for linear systems. In a number of works [18], [19], it is pointed out that the selection of the order of the redundancy relation has an influence on the optimization performance. In fact, it is proved in [20] that increasing the order of redundancy relation leads to an increase in the dimension of the parity space, which in turn provides greater flexibility in residual generation as well as improves robustness. Note that the above-discussed conclusions regarding the increase in order of redundancy relation have been proven for linear systems. There are no equivalent results available in the literature for nonlinear systems. The objective of this work is to extend the above results for nonlinear systems.

Summary of Contribution

In this work, we have studied the relation between order of redundancy relation and robustness of the residual generation. The main contribution is to formulate and prove the theorem that increasing the order of redundancy relation improves the system robustness. The proposed theorem is an extension of the similar results obtained in linear systems [20]. Based on the theorem, an algorithm has been proposed to determine the optimal redundancy relation order. We have experimentally verified the claim on a PUMA 560 robotic arm. A comparative experimental study is presented to demonstrate the effect of robust residuals. Manuscript 3 is based on the following papers:

- Halder. B and Sarkar. N, "*Impact of the Order of Redundancy Relation in Robust Fault Detection of Robotic Systems*," Proceedings of Conference on Decision and Control (CDC), San Diego, California, December 13-15, 2006.
- (Submitted) Halder. B and Sarkar. N, "*Study the Order of Redundancy Relation for Nonlinear Systems*," Journal of Control Engineering Practice.

Manuscript 4: Robust Fault Detection and Isolation in Mobile Robot

Background

Fault detection and isolation are important problems in the development of reliable, robust mobile robots. Both the fault detection and isolation is needed for a successful fault diagnosis system. Residual generator for fault diagnosis needs to be designed to support the isolation of faults. To facilitate fault isolation, the residual set needs to have distinctive properties and unique characteristics of particular faults. There are two fundamental approaches to enhance the residual for fault isolation: structured residuals and directional residuals [5]. Structured residuals are so designed that each residual responds to a different subset of faults and is insensitive to the others.

Structure residual method is used in the literature for fault diagnosis of mobile robot [21-22] and other systems [7]. All the above methods does not account for the modeling error and uncertainty of the system. We designed the primary residual vectors (PRV) based on the robust nonlinear analytic method (RNLAR) in Manuscript 1. RNLAR method is further developed to generate robust structured residual vectors (SRV), which is fault-accentuated signal, for fault isolation in a mobile robot.

Summary of Contribution

A robust method for the detection and isolation of sensor and actuator faults is presented in this work. The main contribution of this paper is to extent the RNLAR method to design robust fault isolation method. The proposed robust nonlinear analytic redundancy method was experimentally verified on a Pioneer 3-DX mobile robot. The results show that both sensor and actuator fault detection and isolation are possible in the presence of model-plant-mismatch (MPM) and disturbances. Manuscript 4 is based on the following papers:

- (Accepted)Halder. B and Sarkar. N, "*Experimental Results of Fault Detection and Isolation in Mobile Robot*," International Journal of Automation and Computing.
- Halder. B and Sarkar. N, "*Robust fault detection and isolation in mobile robot*," Proceedings of International Federation of Automatic Control (IFAC), Beijing, China, August 30- September 01, 2006, pp. 1483-1488.

References

[1] E. Chow, and A. Willsky, "Analytic redundancy and the design of robust failure detection systems", IEEE trans. on Automatic Control, Vol. AC-29, No. 7. July 1984.

[2] P.M Frank, and X. Ding, "Survey of robust residual generation and evaluation method in observer-based fault detection system," J. Proc. Cont. Vol. 7. No. 6. pp 403-424, 1997.

[3] J. J Gertler, "Analytic redundancy methods in fault detection and isolation – survey and synthesis", Preprints of IFAC Safeprocess conference, vol. 1, pp. 9-22. 1991.

[4] R. Isermann, "Process fault detection based on modeling and estimation methods- a survey," Automatica, 20, 387-404.

[5] J. J Gertler, "Fault Detection and Diagnosis in Engineering systems", 1 ed. Marcel Dekker , Inc.

[6] M. L. Leuschen, “Derivation and application of nonlinear analytic techniques with application on robotics”, PhD, dissertation, Rice University. Texas, 2001.

[7] Z. Han, W. Li, and S. L. Shah, “Fault detection and isolation in the presence of process uncertainties”, Control engineering practice, 13, 587-599, 2005.

[8] C. Kwan and R. Xu, “A note on simultaneous isolation of sensor and actuator faults,” IEEE transaction of control systems technology, vol. 1, no. 1, Jan 2004.

[9] J. F. Engelberg, “Three million hours of robot field experience,” The industrial robot, pp. 164-168, June 1974.

[10] A. De. Luca and R. Mattone, “Actuator failure detection isolation using generalized momenta,” proceedings of the 2003 IEEE, ICRA, 634639, 2003.

[11] A. De Luca and R. Mattone, “An adapt-and-detect actuator FDI scheme for robot manipulators,” Proc. IEEE, ICRA, New Orleans, LA, April 2004.

[12] G. Antonelli, F. Caccavale, and L. Villani, “Adaptive discrete-time fault diagnosis for a class of nonlinear systems: application to a mechanical manipulator,” Proc. IEEE, ISIC, Houston, Texas, October 5-8, 2003.

[13] M. L. McLntyre, W. E. Dixon, D. M. Dawson, and I. D. Walker, “Fault detection and identification for robot manipulators,” Proc. IEEE ICRA, New Orleans, LA, 2004.

[14] G. Liu, “Control of robotic manipulators with consideration of actuator performance degradation and failure,” Proc. IEEE ICRA, Seoul, Korea, May 21-26, 2001.

[15] L. Notash, “Kinematic solution for the effective implementation of parallel manipulators,” PhD Dissertation, Dept of Mech Engg, Univ of Victoria, June 1995.

[16] G. Paviglianiti, F. Caccavale, M. Mattei, and F. Pierri, “Sensor fault detection and isolation for robot manipulator,” Proc 13th Mediterranean conference on control and automation, Limassol, Cyprus, June 27-29, 2005.

[17] J. J Gertler, “Diagnosis parametric fault – from identification to parity relations”, Proc. American Control Conference, pp. 143-156.

[18] X. C. Lou, A. S. Willsky, and G. L. Verghese, “Optimally robust redundancy relation for failure detection in uncertain systems,” *Automatica*, vol. 22, pp. 333-344, 1986.

[19] J. Wuennenberg, “Observer-based fault detection in dynamic system,” Ph.D. dissertation, University Duisburg, 1990.

[20] X. Ding, L. Guo, and T. Jeinsch, “A characterization of parity space and its application to robust fault detection,” IEEE transaction on automatic control, VOL. 44, No. 2, February 1999.

[21] M. Hasimoto, H. Kawashima, T. Nakagami, and F. Oba, “Sensor fault detection and identification in dead-reckoning system of mobile robot: interacting multiple modal approach,” Int’l Conf. on Intelligent Robots and Systems, pp. 1321-1326, 2001.

[22] B. N. Umesh, “A fault diagnostic system for an unmanned autonomous mobile robot,” University of Cincinnati, 1997.

CHAPTER II: MANUSCRIPT 1

A ROBUST NONLINEAR ANALYTIC REDUNDANCY FRAMEWORK FOR ACTUATOR FAULT DETECTION AND ITS APPLICATION TO ROBOTICS

Bibhrajit Halder[1] Nilanjan Sarkar[2]

(Submitted to Journal of Automatica)

Abstract

A new approach to actuator fault detection in the presence of model uncertainty and disturbances, and its application to a wheeled mobile robot (WMR) are presented in this paper. Robust fault detection is important because of the universal existence of model uncertainties and process disturbances in most systems. This paper proposes a new approach, called *robust nonlinear analytic redundancy* (RNLAR) technique, to actuator fault detection for *input-affine nonlinear multivariable dynamic systems* in the presence of model-plant-mismatch (MPM) and process disturbance. Analytic redundancy, which is a basis for residual generation to detect fault, is primarily used in the linear domain. The

---

[1]Graduate Research Assistant, Department of Mechanical Engineering, Vanderbilt University, 2301 Vanderbilt Place, Nashville, TN-37235. Email: bibhrajit.halder@vanderbilt.edu Phone: 1-615-343-6472, Fax: 1-615-343-6687.

[2]Associate Professor, Department of Mechanical Engineering, Vanderbilt University, 2301 Vanderbilt Place, Nashville, TN-37235. Email: nilanjan.sarkar@vanderbilt.edu Phone: 1-615-343-7219, Fax: 1-615-343-6687.

proposed RNLAR can be used to design primary residual vectors (PRV) for nonlinear systems to detect actuator faults. The proposed methodology is applied to the actuator fault detection of a WMR and the simulation results are presented to demonstrate its effectiveness.

Keywords: Fault detection, analytical redundancy, robustness, nonlinear systems, mobile robots.

1. Introduction

Recent technological advances in hardware and control techniques have allowed us to design increasingly complex robots. However, it is unlikely that these complex robots could be immune to system faults. Faults may result in mission failures that are costly in mission critical enterprises such as planetary exploration, search and rescue, mine mapping, demining and nuclear waste cleanup. Therefore the ability to adapt to faults can be important for a robot in mission critical operations. One way to address these needs is to design a fault tolerant control system (FTCS). Generally, a FTCS consists of two major components: a fault detection and isolation (FDI) scheme, and a fault accommodation mechanism. In this work we focus on actuator fault detection for a class of input affine nonlinear systems that include robotic systems.

FDI techniques are broadly classified into two classes: model-free approaches and model-based approaches. In a model-free approach, the system model is constructed without the use of any knowledge obtained from physical laws [26] [28]. Recent model-free techniques include the use of neural networks [34], Bayesian belief network [25], and genetic programming [39] among others. Model-based fault detection, on the other hand, utilizes the mathematical model of the plant to generate residuals. Residuals are measures of discrepancy between expected and the measured system behavior. A substantial research effort has been invested in model-based FDI during the last few decades. Given the success of the model-based approach and the powerful mathematical tools it provides, we choose to concentrate on this method. Some important survey papers in this area are [6] [9] [13]. The fundamental concept of model-based fault detection is analytical redundancy (AR). The basic idea of AR is the comparison between the actual behavior of the monitored plant and the behavior of a mathematical plant. Implementation methods of AR can be classified into two groups: 1) indirect implementation, based on diagnostic observers, and 2) direct implementation based on the parity relation technique.

The origin of observer-based fault detection can be traced back to [1] [15]. A survey paper [7] gives the details about this method. In [24] the authors introduced a geometric approach to designing observers for linear systems. Later it was extended to nonlinear systems [3]. More details on the use of observer-based method can be found in [21] and the references therein. Robust fault detection for nonlinear systems is mostly based on nonlinear observer design approaches [32] [38] [40]. The authors in [38] proposed an existence criterion for an observer-based robust residual design approach that can accommodate disturbances in the system. This formulation, however, makes it more difficult to be satisfied under considerable model-plant-mismatch (MPM). Various methods are used to design the nonlinear observer to accommodate the MPM that include sliding mode [35], adaptive/learning [16] and neural network approaches [23].

There exists a rich literature in parity relation based residual generation. In [2] the fundamental formulation of a parity relation was presented for linear systems. Various researchers have combined linear AR with nonlinear systems [36] [41], by using the method of linearization of the nonlinear system. The AR concept was later extended in [19] [20] to nonlinear systems without linearization. They [19] introduced the idea of nonlinear analytic redundancy (NLAR) for fault detection.

In [19], the authors assumed the existence of a perfect system model for fault detection. However, MPM and process disturbances almost always exist in practical systems. A model dependent fault detection scheme may not be useful under considerable MPM and process disturbances. Thus a robust fault detection method that does not require a perfect model will be valuable. However, robustness issues have mostly been addressed for linear systems in the literature. In [2] a method is proposed for robust residual generation based on an optimization technique. Several other researchers applied the method given in [2], to minimize the effect of the disturbances or to minimize the effect of MPM [10]. Recently, in [12] both the MPM and process disturbances are considered. Also, in [18] a method is proposed for isolating sensor and actuator faults with least sensitivity to the MPM and process disturbances. All the above robust FDI methods are applicable to linear systems. As far as nonlinear systems are concerned, in [33] the authors proposed an analytical redundancy based robust fault detection method using a mathematical technique, called algebra of functions, to transfer the nonlinear model into a weakly nonlinear model as the main step in designing the residuals. This method provides satisfactory results but it assumes that modeling uncertainty can be specified in the form of unknown constant or slowly varying system parameters.

In this paper, we develop a new robust nonlinear analytic redundancy (RNLAR) fault detection technique. The proposed RNLAR fault detection technique accommodates both the MPM and process disturbances for multivariable dynamic systems to detect actuator faults. We extend the robustness idea, used in [12] for linear systems, into the nonlinear domain.

2. Problem Formulation

Consider a multivariable input-affine nonlinear dynamic system of the form:

$$\dot{x} = f(x) + \sum_{i=1}^{q} g_i(x) u_i + d(x,u), \quad y = Cx + o \tag{1}$$

where the state x is defined on an open subset U of $\Re^n$; $u = [u_1 \quad u_2 \quad \ldots \quad u_q]^T \in \Re^q$ is the input; $y \in \Re^m$ is the process output; C is the $m \times n$ output matrix; $d(x,u)$ represents an unmeasured deterministic process disturbance vector [11]; o represents a Gaussian-distributed white noise vector. The functions f, $g_1, \ldots, g_q$ are $\Re^n$ valued smooth mappings defined on the open set U, and define $g = [g_1 \;\; g_2 \;\; \ldots g_q] \in \Re^{n \times q}$.

In the presence of faults, the input can be represented by

$$u = u^g + u^f \tag{2}$$

where $u^g \in \Re^q$ represents the fault-free input vector and $u^f \in \Re^q$ represents the actuator fault vector. It is assumed that u^g is available for computation but u^f and o are not. The magnitude of the noise is assumed to be significantly smaller then the magnitude of faults. Under the nominal fault-free condition, u^f is a zero vector. However, when an actuator fault occurs in the system, u^f will become non-zero. A schematic diagram of the overall system is given in Figure 2-1.

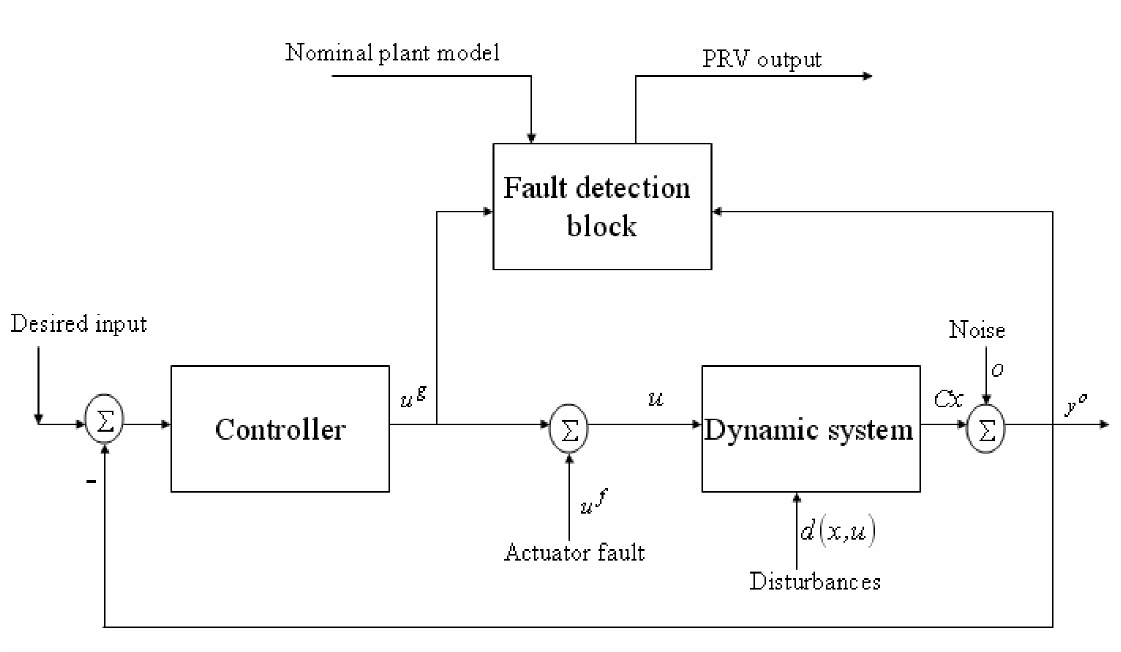


Figure 2-1. Schematic diagram of the FDI system

Model-plant-mismatch is represented by

$$f(x) = f^n(x) + f^u(x),\ \ g(x) = g^n(x) + g^u(x) \tag{3}$$

where $f^n(x), f^u(x), g^n(x)$, and $g^u(x)$ represent the nominal and uncertain part of the mappings *f* and *g*, respectively. Combining (1), (2) and (3), the overall system with faults is represented by

$$\dot{x} = f^n(x) + f^u(x) + \left(g^n(x) + g^u(x)\right)\left(u^g + u^f\right) + d(x,u),\ \ y = Cx + o \tag{4}$$

Simplifying (4) we get

$$\dot{x} = f^n(x) + g^n(x)u^g + e(x,u) + g^n(x)\,u^f,\ \ y^o = Cx + o \tag{5}$$

where $e(x,u) = f^u(x) + g^u(x)\left(u^g + u^f\right) + d(x,u)$. The vector $e(x,u)$ is called an error vector, which contains both the uncertainty of the model and the disturbances. The following assumptions will be used in this paper in order to design the robust actuator fault detection method:

Assumption 1: The fault-free system is asymptotically stable. This is a general assumption in the FDI literature [12] [20].

Assumption 2: The system in (1) is observable. This assumption is needed in order to guarantee the ability to find all the states from the system outputs and is also common in the literature [2] [19] . We should note that the observability assumption does not mean that we can (or we need to) find the fault-free states from faulty output measurements.

Assumption 3: The modeling uncertainty, denoted by $f^u(x)$ and $g^u(x)$ in (3), which are unknown nonlinear vector functions of *x*, is bounded. We also assume that both the inputs and the disturbances are bounded, which is similar to the assumption made in [12]. Define fault-free error part, e^*, as $e^* = f^u(x) + g^u(x)u^g + d$. We assume that $\max\left\{\left\|f^u(x)\right\|, \ \left\|g^u(x)u^g\right\|, \ \left\|d(x)\right\|\right\} < \left\|F_o\left(x,u^g\right)\right\|$ where $F_o\left(x,u^g\right)$ is a known bounded function. Now, $\left\|e^*\right\| \le \left\|f^u(x)\right\| + \left\|g^u(x)u^g\right\| + \left\|d\right\|$. Thus we can say e^* is bounded, e.g., $\left\|e^*\right\| \le L$, where $\| \ \|$ stands for the L_2 norm and $L = 3\left\|F\left(x,u^g\right)\right\|$.

Thus the problem we seek to solve in this paper becomes: design robust residuals for actuator faults for the nonlinear systems given by (5). By robust we mean the residual will need to be sensitive to the faults but insensitive to the MPM and disturbances of the system, i.e., insensitive to error as much as possible.

3. Background Information

Consider the nonlinear system (1) without disturbance and noise:

$$\dot{x} = f(x) + \sum_{i=1}^{q} g_i(x) u_i; \quad y = Cx \tag{6}$$

We briefly describe the basic steps of the NLAR technique as given in [19] to motivate the design of a robust nonlinear analytic redundancy technique. For detailed information on the NLAR technique, please refer to [19]. Two vectors O_Δ and $O_{\Delta DD}$ were defined as follows:

$$O_\Delta = \begin{bmatrix} Cx \\ \sum_{j=0}^{q} L(j) \\ \sum_{l=0}^{q}\sum_{j=0}^{q} L(j,l) \\ \sum_{m=0}^{q}\sum_{l=0}^{q}\sum_{j=0}^{q} L(j,l,m) \\ \cdot \\ \cdot \\ \cdot \end{bmatrix} \tag{7}$$

where q is the number of inputs and

$L(j,l,m,\ldots) = \left(u^g{}_j u^g{}_l u^g{}_m \ldots\right) L_{\ldots k(m)k(l)k(j)} Cx.$

$u^g{}_0 = 1,\ k(j) = \begin{Bmatrix} f\ ,\ j = 0 \\ g_j\ ,\ j \neq 0 \end{Bmatrix}$

The Lie derivative [14] [17] is defined as $L_f h = \sum_{i=1}^{n} \frac{\partial h}{\partial x_i} f_i$ and the repeated Lie derivative in (7) is written in the following ways: $L_i(L_j(L_k h)) = L_i L_j L_k h = L_{ijk} h$.

$$\text{O}_{\Delta DD} = \begin{bmatrix} y \\ \dot{y} \\ \ddot{y} - \sum \dot{u} L_g \\ . \\ . \end{bmatrix} \tag{8}$$

where $L_g = L_g Cx = gC$

The general NLAR method works in the following manner:

1. Calculate O_Δ from (7).
2. From O_Δ calculate the left null matrix, Ω^Γ, such that $\Omega^\Gamma \text{O}_\Delta = 0$.
3. Next calculate $\text{O}_{\Delta DD}$ using (8).
4. Finally, apply the NLAR equation: $\Omega^\Gamma \text{O}_{\Delta DD} = PRV$, where PRV is the primary residual vector of the NLAR.

It is worth noting that the following important issues are not addressed in the method mentioned above.

- It is not clear how to calculate Ω^Γ given O_Δ. This question is not a trivial one. It was originally discussed for linear systems in [2] and later in [12] and [18]. Computationally, the PRV is given by $\Omega^\Gamma \text{O}_{\Delta DD}$, and as a consequence, the design of PRV is equivalent to finding a Ω^T. Clearly, there are number of choices for Ω^Γ for a given O_Δ. We will show that the multiple choices for Ω^Γ can be utilized to design the robust PRV. We will address this issue in detail in the subsequent sections.
- The effect of model uncertainty and the process disturbances are not taken into account while designing the *PRV*.

4. Robust Fault Detection Method

The two important mathematical structures for the NLAR technique are O_{Δ} and $\mathrm{O}_{\Delta DD}$. O_{Δ} and $\mathrm{O}_{\Delta DD}$ are defined based on (6) that does not include the MPM and the process disturbances. The design of RNLAR technique, on the other hand, is based on (5). In order to effectively analyze and account for the MPM and disturbance terms we develop new mathematical structures that are analogous to O_{Δ} and $\mathrm{O}_{\Delta DD}$ but are more appropriate for the RNLAR technique. In addition, we define an error matrix, G_s.

We describe the theoretical development of the RNLAR technique here. Some practical implementation issues will be discussed later along with the simulation results. Starting with the output y from (5), take the derivative of y for s times and stack them together in (9), where s is the order of the redundancy relation as defined in [2]. s describes the 'memory span' of the redundancy relation.

$$\begin{bmatrix} y \\ \dot{y} \\ \ddot{y} \\ . \\ . \\ y^{s} \end{bmatrix} = \begin{bmatrix} Cx + o \\ C\dot{x} + \dot{o} \\ C\ddot{x} + \ddot{o}. \\ . \\ . \\ Cx^{s} + o^{s} \end{bmatrix} \tag{9}$$

$$= \begin{bmatrix} Cx + o \\ C\left(f^{n}(x) + g^{n}(x)\, u^{g} + e(x,u) + g^{n}(x)\, u^{f} \right) + \dot{o} \\ C\dfrac{d}{dt}\left(f^{n}(x) + g^{n}(x)\, u^{g} + e(x,u) + g^{n}(x)\, u^{f} \right) + \ddot{o} \\ . \\ . \\ . \end{bmatrix} \tag{10}$$

Define the stacked output vectors, $y_s = [y \ \dot{y} \ \ddot{y} \ \dddot{y} \ ...]^T \in R^{m(s+1)}$. Similarly, we define the input stack vector, u_s, the error stack vector, e_s, actuator fault stack vector, u_s^f, and the noise stacked vector, o_s as follows:

$u_s = [u^g \ \dot{u}^g \ \ddot{u}^g \ ...]^T \in R^{qs}$, $e_s = [e \ \dot{e} \ \ddot{e} \ ...]^T \in R^{ns}$, $o_s = [o \ \dot{o} \ \ddot{o} \ ...]^T \in R^{m(s+1)}$

$u_s^f = [u^f \ \dot{u}^f \ \ddot{u}^f \ ...]^T \in R^{qs}$

The right-hand side of (10) can be grouped into three major components: collection of the error terms, collection of the input terms, and collection of the states. This leads to the following compact form:

$$y_s = H_s u_s + \Gamma_s + G_s e_s + H_s u_s^f + o_s \tag{11}$$

where,

$$\Gamma_s = \begin{bmatrix} Cx \\ L_f Cx \\ L_{ff} Cx + \sum_{l=0}^{q} \sum_{j=0}^{q} L(j,l) \\ L_{fff} Cx + \sum_{k=0}^{q} \sum_{l=0}^{q} \sum_{j=0}^{q} L(j,l,k) \\ \\ ..... \end{bmatrix}_{m(s+1)\times 1} \qquad G_s = \begin{bmatrix} 0 & 0 & 0 \\ C & 0 & 0 \\ C\left(\sum_{i=0}^{l} u_i \partial k_i / \partial x\right) & C & 0 \\ \Lambda & C\left(\sum_{i=0}^{l} u_i \partial k_i / \partial x\right) & C \end{bmatrix}_{m(s+1)\times ns}$$

$$H_s = \begin{bmatrix} 0 & 0 & 0 \\ Cg & 0 & 0 \\ C\left[(\partial f/\partial x) g + (\partial g/\partial x) f\right] & Cg & 0 \\ \Lambda_h & C\left[(\partial f/\partial x) g + (\partial g/\partial x) f\right] & Cg \end{bmatrix}_{m(s+1)\times qs}$$

The terms Λ and Λ_h contain higher order derivatives of the vector functions f and g_i. The term Γ_s replaces O_Δ in (7). We define new group formation, O_{NDD} for the RNLAR technique as follows:

$$O_{NDD} = y_s - H_s u_s \tag{12}$$

Using (11) and (12), we get

$$O_{NDD} = \Gamma_s + G_s e_s + H_s u_s^f + o \tag{13}$$

Equations (11) and (13) will be used to derive the residuals for actuator faults. Note that in (12) y_s and u_s are outputs and inputs of the system described in (5) (i.e., which represents the actual plant). In (13) H_s, Γ_s and G_s are computed from the nominal system given in (6) (i.e., the mathematical model of the plant).

4.1 Robust Actuator Fault Detection

Our objective is to design a residual vector that is less sensitive to the error vector and most sensitive to the actuator fault. The ideal outcome would be to design a residual vector that is only sensitive to the actuator fault and completely insensitive to the error vector. Let us investigate whether we can achieve the ideal outcome. We rearrange (13) to obtain

$$O_{NDD} = \Omega_s E_s + H_s u_s^f + o_s \tag{14}$$

where $\Omega_s = [\Gamma_s \quad G_s]$ and $E_s = \begin{bmatrix} 1 \\ e_s \end{bmatrix}$

Select a transformation matrix, c_s, from the space C_s defined by $C_s = \{c_s : c_s^T \Omega_s \equiv [0]\}$. Pre-multiplying both sides of (14) with c_s^T results:

$$R_a = c_s^T O_{NDD} = c_s^T H_s u_s^f + o_s \tag{15}$$

R_a is called the primary residual vector (PRV) for actuator fault. It appears that R_a is completely insensitive to the error vector. But note that, for a full rank *C* matrix c_s can only have its first *m* columns to be nonzero and the rest of the elements to be zero due to the block-triangular structure of the matrix G_s since $c_s^T G_s = 0$. Also note that the first *m* rows of H_s matrix are zero. Hence $c_s^T H_s \equiv [0]$. Substituting this in (15) gives

$$R_a = c_s^T O_{NDD} = c_s^T o_s \tag{16}$$

Hence both the error vector and the fault contributing term $c_s^T H_s u_s^f$ are annihilated at the same time. This implies that the actuator residual is insensitive to not only the error vector but also to the fault. Therefore, no actuator fault can be detected if the error vector is completely removed when the outputs are non-redundant (i.e., *C* is a full row rank matrix).

4.2 Generating PRV for Actuator Fault

Faced with the above problem, it can be concluded from (16) that complete elimination of the effect of the error vector from the PRV is not possible when $c_s^T \Omega_s \equiv [0] \Rightarrow c_s^T H_s = [0]$. This result is consistent with actuator fault detection results obtained for linear systems. In this case, we present a design methodology for the PRV that makes it insensitive to the error vector but sensitive to the actuator faults as much as possible. Select a transformation vector, w_s , from the parity space w_s defined by $W_s = \{w_s : w_s^T \Gamma_s \equiv [0]\}$. Pre-multiplying both sides of (13) with w_s^T results:

$$R_a = w_s^T (y_s - H_s u_s) = w_s^T \left(G_s e_s + H_s u_s^f + o_s\right) \tag{17}$$

It can be observed from (17) that the actuator residual, R_a, is sensitive to both the actuator faults and the uncertainty of the system. It is desirable that R_a should be highly sensitive to the actuator faults and mostly insensitive to the error terms in order to be able to detect actuator fault in the presence of error term. The above desired property can be translated mathematically into the statement, $\left\| w_s^T G_s \right\|$ is less than $\left\| w_s^T H_s \right\|$, where the coefficient of the error vector is $w_s^T G_s$ and the coefficient of the fault vector is $w_s^T H_s$. Both G_s and H_s are system dependent matrices. However, w_s can be chosen independently from the parity space to satisfy the above requirement. Hence the problem becomes, select a transformation vector w_s for the parity space in such a way that $\left\| w_s^T G_s \right\|$ is less than $\left\| w_s^T H_s \right\|$. In the literature this problem is discussed for linear systems. Both [2] and [12] frame this problem as a linear optimization problem and use the linearity property to determine w_s. For a nonlinear system, which is the case here, this translates into solving a nonlinear optimization problem where the functional structure of w_s is unknown. In other words, we do not know the functional form of each element of w_s (e.g., whether they are polynomial, exponential etc.) and we cannot realistically guess them without any other knowledge. This makes the nonlinear optimization problem very difficult to solve. In order to overcome this problem, we propose a novel method for designing W_a for nonlinear systems.

4.3 Design Methodology

Given the states $x \in \Re^n$ and inputs $u \in \Re^q$, consider an open set $U_r \in \Re^{n+q}$ such that the states and the inputs are restricted on U_r, i.e., $x_e = (x,u) \in U_r$. We define the following performance function, $J_s(x,u) = \dfrac{{w_s}^T G_s {G_s}^T w_s}{{w_s}^T H_s {H_s}^T w_s}$.

We formulate the robust problem as follows: Find a w_s from the parity space such that $J \leq K(x_e) \quad \forall x_e \in U_r$ for some predefined $0 < K(x_e) < 1$. The choice of $K(x_e)$ will determine the sensitivity of actuator residual to the actuator fault and insensitivity to the error term. A small value of K will guarantee the sensitivity requirement of w_s. Here we omit the subscripts from w_s and other terms for notational simplicity. Define $S_G = GG^T$, $S_H = HH^T$ and $R = S_G - KS_H$. Now using the newly defined notation, the above problem becomes:

Given Γ, G, and H, produce a vector function $w \in R^{m(s+1)}$ such that the following conditions are satisfied:

$$\begin{aligned} &1.\ w^T \Gamma \equiv [0],\ w^T H \neq [0] \\ &2.\ w^T R w \leq 0\ \forall\ x_e \in U_r \end{aligned} \tag{18}$$

We propose the following theorem in order to solve the above problem.

Theorem:

Part (i): Let $\mu^-(R)$ be the number of distinct, non-positive, eigenvalues of *R*. If, $\mu^-(R) \geq 2$ then $\exists w$ that satisfies both Conditions 1 and 2 in (18). Also if λ_i are the non-positive eigenvalues of *R* and v_i are the corresponding eigenvectors, for $i \in \{1,2...,n\}$, then

$w(x_e)=\sum_{i=1}^{\mu^-}\alpha_i(x_e)V_i(x_e)$ satisfies Condition 2. For $i \geq 2$, we can always choose $\alpha_i(x_e)$ such that Condition 1 satisfies.

Part (ii): When $\mu^-(R)=1$ then there exists w such that $w=\alpha V$, only if $w^T\Gamma \equiv [0]$ where V is the eigenvector corresponding to the non-positive eigenvalue of R.

Part (iii): If $\mu^-(R)=0$, i.e., all the eigenvalues of R are positive, then there is no such w that satisfies both Conditions 1 and 2. The proof of the above theorem is given in the Appendix I.

Here, we give a few simple examples of both linear and nonlinear systems to illustrate the design method.

Example 1. (Linear system) Let us consider the following matrices:

$$\Gamma = \begin{bmatrix}5 & 3 & 1 & 2\end{bmatrix}^T \qquad G = \begin{bmatrix}0 & 0 & 0\\ 1 & 0 & 0\\ 15 & -1 & 0\\ 23 & -5 & 1\end{bmatrix} \; H = \begin{bmatrix}0 & 0 & 0 & 0 & 0\\ 2 & -5 & 0 & 0 & 0\\ 34 & -15 & -5 & 0 & 0\\ 41 & -30 & 2 & 5 & 0\end{bmatrix} \tag{19}$$

and let $K=.05$

The matrix R is easy to calculate and so are the eigenvalues and the corresponding eigenvectors of R

$$R = GG^T - KHH^T = \begin{bmatrix}0 & 0 & 0 & 0\\ 0 & -0.4500 & 7.8500 & 11.4000\\ 0 & 7.8500 & 155.70 & 258.30\\ 0 & 11.40 & 258.30 & 424.50\end{bmatrix}$$

$\lambda_1 = -1.7351,\ V_1 = \begin{bmatrix}0 & -.639 & .665 & -.386\end{bmatrix}^T,\ \lambda_2 = -0.1168,$
$V_2 = \begin{bmatrix}0 & -.7688 & .537 & -.347\end{bmatrix}^T$
$\lambda_3 = 0,\ V_3 = \begin{bmatrix}1 & 0 & 0 & 0\end{bmatrix}^T,\ \lambda_4 = 581.602,\ V_4 = \begin{bmatrix}0 & .023 & .5187 & -.8546\end{bmatrix}^T$

Here $\mu^{-}(R) = 3 \geq 2$, hence $\exists w$ that satisfies the sensitivity condition, which can be represented by $w = \alpha_1 V_1 + \alpha_2 V_2 + \alpha_3 V_3$. Pick $\alpha_1 = 1, \alpha_2 = 0.9419$ and $\alpha_3 = 0$ to satisfy the condition $w^T \Gamma = 0$. To check that this choice of α_i's also satisfies the inequality of Condition 2, we calculate the performance function *J*.

$$J = \frac{w^T GG^T w}{w^T HH^T w} = \frac{7.8008}{192.7911} = 0.0405 < 0.05 = K$$

Once w is determined then the residual can be calculated using the relation $R_a = w^T (y_s - Hu_s)$. Next we give a nonlinear example.

Example 2. (Nonlinear system) Consider the following matrices:

$$\Gamma = [5x_1 \quad 8x_2]^T \qquad G = \begin{bmatrix} 0 & 0.1 & 0 \\ .2x_1 & 0 & 0 \end{bmatrix} \quad H = \begin{bmatrix} 1 & 0 \\ 0 & x_1 \end{bmatrix} \tag{20}$$

and let $K = .2$

Here the eigenvalues and the corresponding eigenvectors of *R* are given as follows:

$$R = GG^T - KHH^T = \begin{bmatrix} -0.19 & 0 \\ 0 & -(4/25)x_1^2 \end{bmatrix}$$

$$\lambda_1 = -0.19,\ V_1 = [1 \quad 0]^T$$

$$\lambda_2 = -(4/25)x_1^2,\ V_2 = [0 \quad 1]^T$$

We chose $w = \alpha_1(x)V_1 + \alpha_2(x)V_2$ where $\alpha_1 = x_2$ and $\alpha_2 = -(5/8)x_1$. To check that this choice of $\alpha_i s$ also satisfy the inequality condition, we calculate the performance function *J*.

$$J = \frac{w^T GG^T w}{w^T HH^T w} = \frac{.01x_2^2 + (1/64)x_1^4}{x_2^2 + (25/64)x_1^4} \leq 0.2 = K$$

The above two simple examples demonstrate the design methodology. Next, we make some comments on the choice of *K*. The choice of K will determine the sensitivity of PRV to the actuator fault and insensitivity to the error term. The desired value of K is $0 < K < 1$. Any value of $K > 1$ will amplify the error term. A small value of K, i.e., $K < 1$, guarantees a robust PRV. The following question arises naturally. What is the minimum possible value of *K?* The optimization problem solved in [12] indirectly answers this question for linear systems. For nonlinear systems, the general approach of our method is as follows:

Step 1. Choose a small value for *K*.

Step 2. Find the nature of eigenvalues of *R* based on the choice of *K*.

Step 3. If there are more then two distinct non-positive eigenvalues, then calculate *W*.

Step 4. If the above condition does not satisfy, then increase the value of *K* and go to Step 2.

Below, we give a simple example to demonstrate the effect of *K* on the eigenvalue of *R*. Consider the following linear matrices

$$G = \begin{bmatrix} 1 & 0 & 1 \\ 6 & 1 & -1 \\ 3 & 2 & -2 \end{bmatrix} \qquad H = \begin{bmatrix} 2 & 3 \\ 7 & 3 \\ 0 & 1 \end{bmatrix} \tag{21}$$

We calculate the value of *R* by keeping *K* as variable

$$R = GG^T - KHH^T = \begin{bmatrix} 2-13K & 5-23K & 1-3K \\ 5-23K & 38-58K & 22-3K \\ 1-3K & 22-3K & 17-K \end{bmatrix}$$

Figures 2-2, 2-3, and 2-4 show the nature of the three eigenvalues for $0 < K < 1$. As observed from the figures that for $K = 0.1$ there is only one non-positive eigenvalue. As we increase the value of K to $K = 0.2$, the number of non-positive eigenvalues increases to two and we can design the robust PRV.

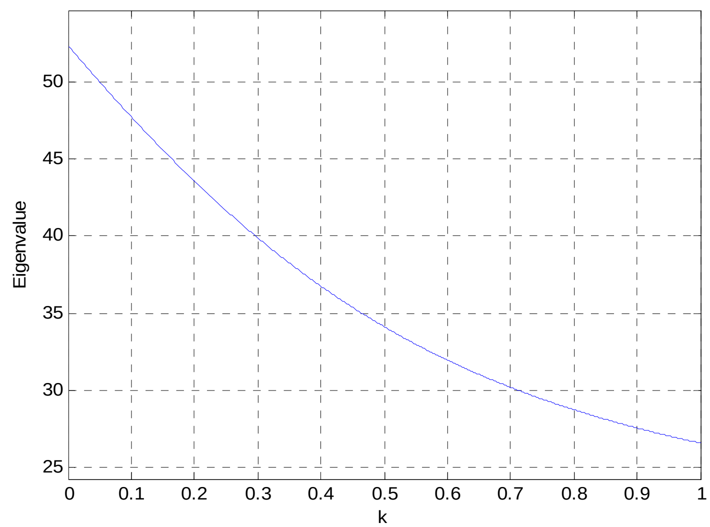


Figure 2-2. Nature of first eigenvalue

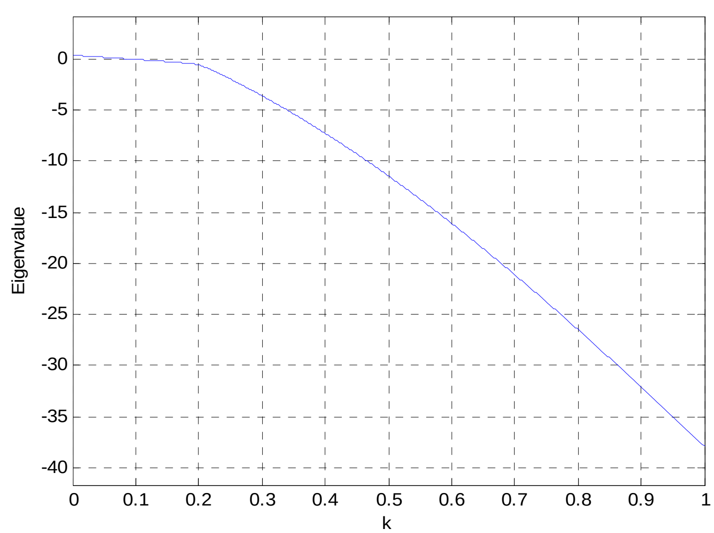


Figure 2-3. Nature of second eigenvalue

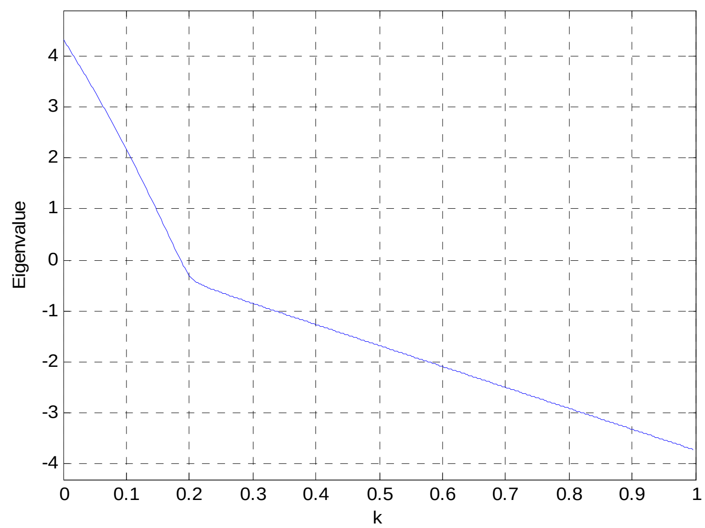


Figure 2-4. Nature of third eigenvalue

5. Simulation Results

We performed computer simulation on a wheeled mobile robot (WMR) to support our theoretical results. A Simulink model of WMR was used to examine the effect of actuator faults using the RNLAR residuals. Before presenting the results, a dynamic model of the WMR is presented.

The WMR is subject to both holonomic and nonholonomic constraints. A detailed discussion on modeling of WMR can be found in the literature [31]. As a result, only the relevant equations are briefly mentioned here. It is assumed that the WMR is driven by two differential wheels (the front passive caster is omitted). The relevant parameters for the WMR are presented in Table I in Appendix II. The nominal equations of the WMR, without MPM and the disturbance, can be written as:

$$M(q_x)\ddot{q}_x + V(q_x, \dot{q}_x) = E(q_x)u^g ,\ A(q_x)\dot{q}_x = 0 \qquad (22)$$

$A(q_x) = \begin{bmatrix} -\sin\phi & \cos\phi & -cd & cd \\ -\cos\phi & -\sin\phi & r/2 & r/2 \end{bmatrix}, E(q_x) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}^T$ and

$q_x = [x_c \quad y_c \quad \theta_r \quad \theta_l]^T$ where (x_c, y_c) is the center of mass of mobile robot, ϕ is the heading angle measured from the x-axis, θ_r, θ_l are angular positions of the two driving wheels as shown in Fig. 2-5. $u^g = [\tau_r^g \quad \tau_l^g]^T$ are the given torques applied to the two wheels. $M(q_x) \in R^{4\times4}$ is the symmetric, positive definite inertia matrix and $V(q_x, \dot{q}_x)$ is the vector of centrifugal and Coriolis forces. The elements of $M(q_x)$ and $V(q_x, \dot{q}_x)$ are given in Appendix II.

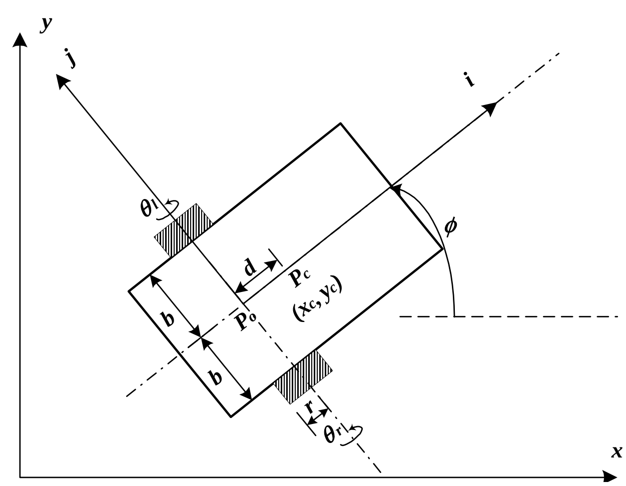


Figure 2-5. Schematic diagram of the mobile robot

Equation (22) can be represented into the state space as follows:

$$\dot{x} = f^{n}(x) + g^{n}(x)u^{g} \tag{23}$$

where $x = \begin{bmatrix} x_c & y_c & \theta_r & \theta_l & \dot{\theta}_r & \dot{\theta}_l \end{bmatrix}^T$ is the state vector. The details of each term of (23) are given in Appendix II. Equation (23) represents the nominal model of the WMR. We introduce model-plant-mismatch in the system by varying the mass of mobile robot by 25%. This changes the actual value of matrix M and V as follows:

$$M^{a} = M + \Delta M; \quad V^{a} = V + \Delta V \tag{24}$$

The detail of ΔM and ΔV are given in Appendix II. We also introduce a friction, F, and disturbance terms, d, in the actual system. The WMR in the presence of MPM and disturbances can be represented as:

$$M^{a}(q_x)\ddot{q}_x + V^{a}(q_x, \dot{q}_x) + F + d = E(q_x)u \tag{25}$$

Equation (25) can be represented into the state space as follows:

$$\begin{aligned} &\dot{x} = f(x) + g(x)u \\ &\Rightarrow \dot{x} = f^{n}(x) + g^{n}(x)u^{g} + e(x,u) + g^{n}(x)\, u^{f} \end{aligned} \tag{26}$$

where $e(x,u) = f^{u}(x) + g^{u}(x)(u)$. The details of each term of (26) are given in Appendix II. The following output equation is used:

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} x = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} x \tag{27}$$

We make the following remarks to explain the simulation results:

1) The nominal WMR model does not include the friction term. However, in the Simulation a coefficient of Coulomb friction of 0.1 and a coefficient of viscous friction of 0.001 are used as MPM.

2) The mass of the WMR in nominal model is 7Kg. We alter the mass value up to 25% to introduce another MPM.

3) To calculate the residuals as in (17) we run the nominal and actual model in parallel. We use the nominal model as in (23) to calculate the terms, G_s , H_s , and Γ_s while y_s and u_s comes from model (26) with the friction term.

4) We have used a band-limited white noise block in Simulink to add noise in the simulation. The specific values that we have used in the noise block for simulation were: noise power= 0.008, sample time= 0.1 and speed=23341, which corresponded to a mean of 0.019 and variance of 0.0918.

5) In the simulation we have used numerical differentiation using the derivative block available in Matlab. The derivative block in Matlab uses forward differentiation technique for numerical differentiation. We should mention that numerical differentiation of noisy sensor signal is well-known to be ill-posed in the sense that a small noise in measurement data can induce a large error in the approximate derivatives [37]. In the simulation we obtained reasonable results using standard numerical differentiation. However, in other instances when the standard numerical differentiation is not sufficient, one can use various low-pass-filters and regularization methods such as Savitzky-Golay smoothing filters [29] to reduce the effect of noise in differentiation.

6) Faults are considered detected if the magnitudes of the residuals cross some pre-determined threshold value. We design the threshold value as twice the absolute maximum value achieved in a fault-free run with the same parameters to demonstrate the effectiveness of the proposed RNLAR technique.

7) Generating the RNLAR residuals for WMR has been automated using the Matlab symbolic toolbox and Mathematica.

5.1 Actuator Fault Detection Results

We present actuator fault detection results when the WMR is tracking i) a straight-line trajectory with desired velocity of 2cm/sec along the x-direction and ii) a circular trajectory of radius 25 cm and angular velocity pi/30 radian/sec. The desired x-axis for the straight-line trajectory is given in Fig. 2-6. The x-axis and y-axis for the circular task trajectories are given in Figs. 2-7 and 2-8. These two sets of trajectories were chosen because it was shown by Dubin [5] that a WMR can reach any arbitrary final position and orientation starting from any arbitrary initial position and orientation using trajectories that are composed of only straight-line and circular segments. In [22] various types of actuator faults have been discussed that are relevant for a WMR operation. We choose two common actuator faults among them to demonstrate the proposed fault detection methodology. First, we consider a partial actuator fault where one actuator generates only a part of the desired torque. This type of fault represents degradation in the actuator system (e.g., friction due to jamming, problems in transmission etc.). The second actuator fault that we consider is a constant torque output. This may occur due to constant polarization of the actuator, called actuator bias.

In order to demonstrate the robustness of the proposed RNLAR technique, we compare fault residuals generated from our proposed technique with that of the NLAR technique as presented in [20] for the same fault conditions. We followed the procedure presented in [24] to design NLAR residuals for this comparison. First, we present results when there is no MPM and disturbances. We consider a partial fault in the right actuator for the straight-line trajectory. An 80% partial fault is introduced to the right actuator in the simulation at *t=7s*. Under no MPM and disturbance condition we run the two nominal models in parallel and calculate the PRV using NLAR and RNLAR methods. The residual result using a NLAR and RNLAR test under no MPM and disturbances is presented in Fig. 2-9 (a) and (b) respectively. As expected, both NLAR and RNLAR detect the fault. Next, we introduced the MPM, disturbance and noise as discussed earlier. The fault detection result with the NLAR residual is presented in Fig. 2-10(a) and that with the RNLAR residual is presented in Fig. 2-10(b). The absolute maximum value of the NLAR signal in a fault-free run, which was obtained separately, was 246.05. Thus the magnitude of the threshold value for NLAR residual was chosen as 492.1. It can be seen that before the fault occurred, the maximum value of the residual stayed within ± 246.05. Now observe that, in 2-10(a) the absolute maximum value of NLAR signal is 473.56, which is less than the threshold value, 492.1. Hence, we can conclude that the fault is not detected for the given threshold. On the other hand, the absolute maximum value of RNLAR signal in a fault-free run was 4.02 and in faulty run was 77.44. Magnitude of the threshold value is 8.04. Hence, the fault is detected clearly and rapidly (i.e., almost instantaneously) in the RNLAR test. It can be seen that the RNLAR residual is significantly more sensitive to the partial actuator fault when there exist both the MPM and disturbance in the system.

We conducted another straight-line trajectory simulation with the bias actuator fault. A constant right wheel torque $\tau_r = 0.14$ was introduced to the right actuator at *t=7s*. The residual test results are presented in Fig. 2-11 (a) for the NLAR technique, and in Fig. 2-

11 (b) for the RNLAR technique for the bias actuator fault. In this case, the absolute maximum value of the NLAR signal in a fault-free run was 246.005 and in faulty run were 349.7. The absolute maximum values for the RNLAR signal were 7.712 and 526.4 in a fault-free run and in a faulty run, respectively. From these values we can conclude that with the chosen threshold the NLAR residual cannot detect the faults while RNLAR residual detects the fault clearly and quickly (i.e., almost instantaneously). We conclude that the RNLAR residual is more sensitive to the bias actuator fault detection in the presence of MPM and disturbance.

Next, we conducted the circular trajectory simulation for both partial and bias faults in the left actuator. First, a 75% partial fault was introduced to the left actuator in the simulation at *t*=5*s*. The fault detection result with the NLAR and RNLAR residuals are presented in Fig. 2-12 (a) and (b), respectively. Here the absolute maximum value of the NLAR signal in a fault-free run was 91.60. The maximum value of the NLAR residual in a faulty run was 170.35, which was less than the threshold value, 183.2. Hence the fault was not detected. For the RNLAR residual, the absolute maximum value in a fault-free run was 23.58 and with fault were 1221.62. Hence, the fault was detected clearly and rapidly (i.e., almost instantaneously) in the RNLAR test.

Finally, simulation with a bias left actuator fault was conducted for the circular trajectory. A constant left wheel torque $\tau_r = 0.13$ was introduced to the left actuator at *t*=5*s*. With the bias actuator fault, the residual test results are presented in Fig. 2-13 (a) for the NLAR test and in Fig. 2-13 (b) for the RNLAR test. In this case, the absolute maximum values of the NLAR signal were 49.212 and 86.48 in a fault-free run and in the faulty run, respectively. On the other hand, the absolute maximum values for RNLAR signal were 1.435 and 30.31 in a fault-free and the fault run, respectively. Thus we can conclude that with the chosen threshold the NLAR residual can not detect the fault while the RNLAR residual detects the fault clearly and quickly.

It is clear from the above set of results that the presented RNLAR technique is useful in detecting actuator faults in the presence of MPM and disturbance. The fault detection using RNLAR technique is clear and fast.

6. Conclusion

A robust methodology for detecting the actuator faults in multivariable input-affine nonlinear dynamic systems has been proposed in this paper. The presented robust nonlinear analytic redundancy (RNLAR) technique is an extension to the robustness idea used in the linear domain into the nonlinear domain. It also extends the current state-of-the-art of nonlinear analytic redundancy (NLAR) techniques used for fault detection of nonlinear systems. For actuator faults, it is shown that PRV cannot be made perfectly insensitive to the MPM and disturbances. We proposed a new design methodology that produces PRV, which are significantly more sensitive to the actuator fault then they are to MPM and disturbances. We applied the RNLAR technique to the fault detection of a wheeled mobile robot in computer simulation. A comparative study was presented between the NLAR and the RNLAR techniques. It was shown that RNLAR residuals perform significantly better under MPM and disturbances.

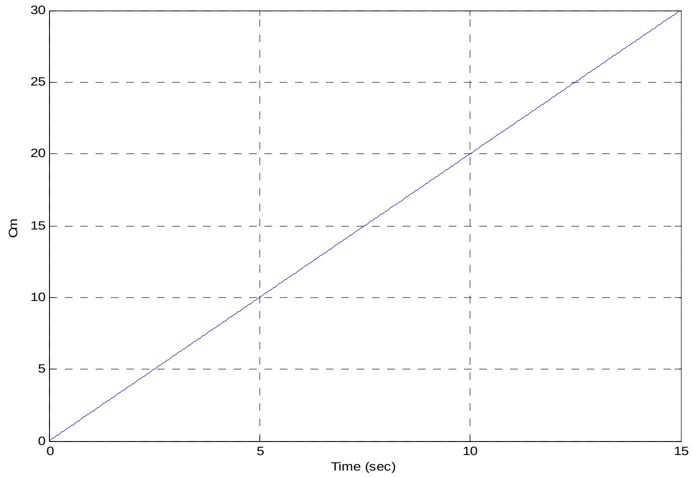


Figure 2-6. Desired X-axis for straight line trajectory

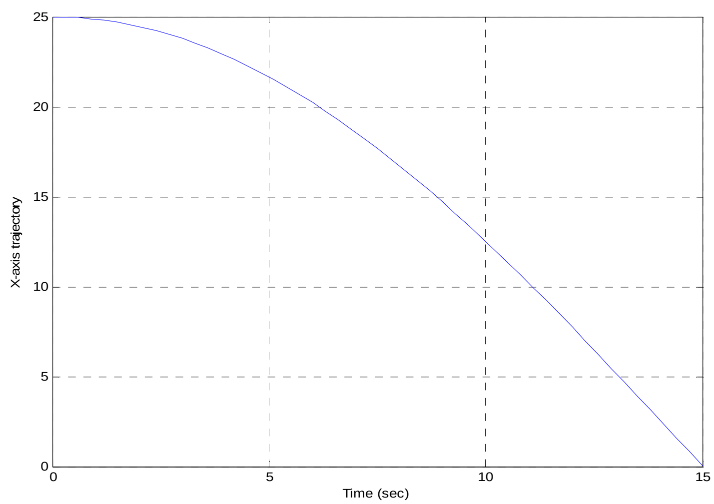


Figure 2-7. Desired X-axis for circular trajectory

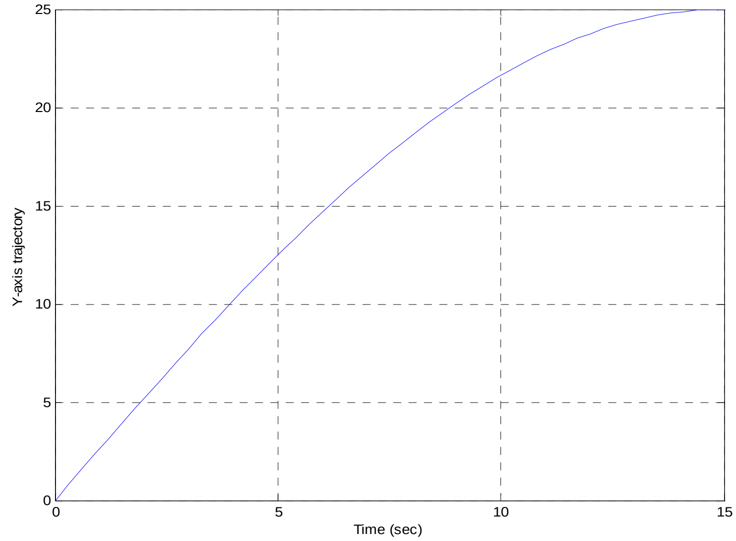
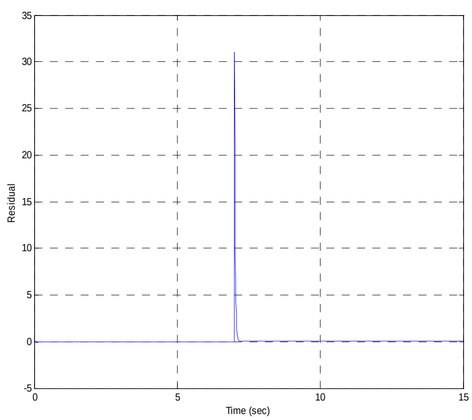
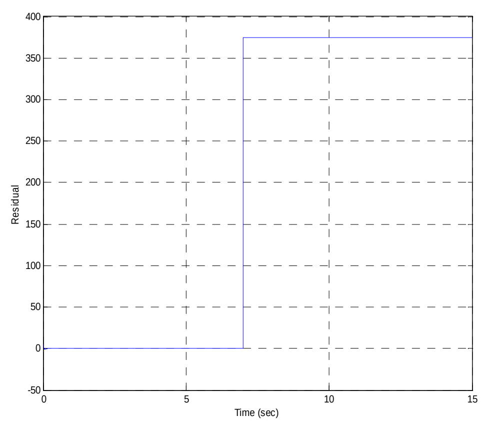
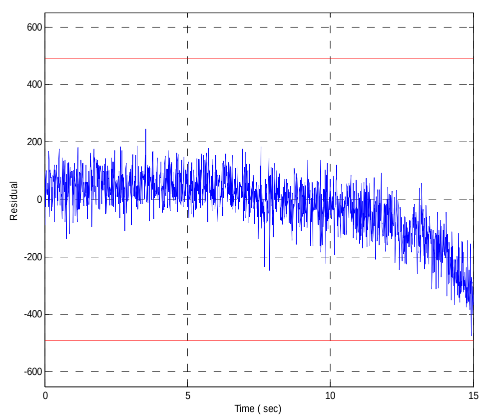
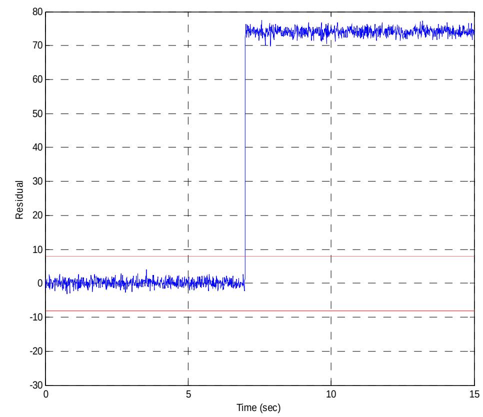
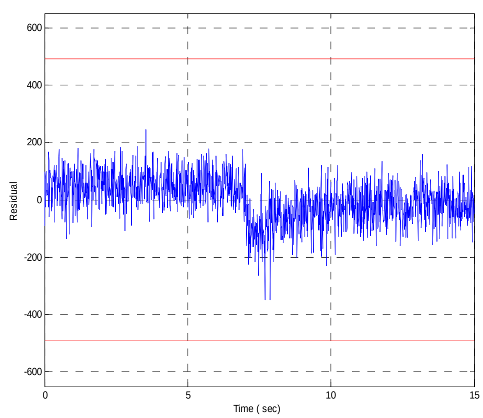
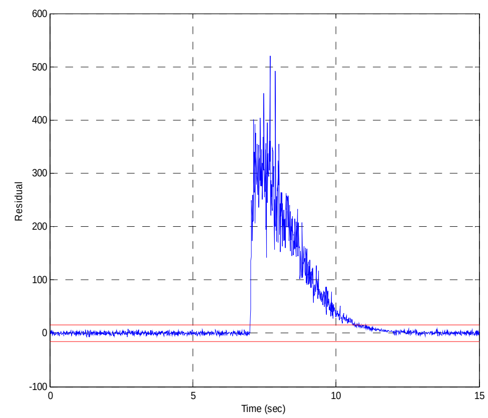
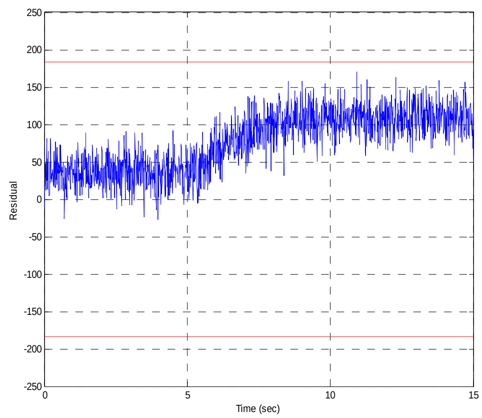
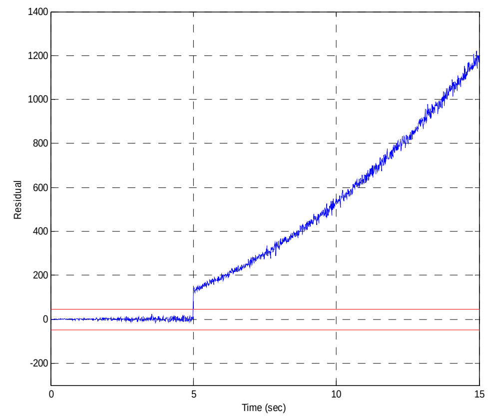
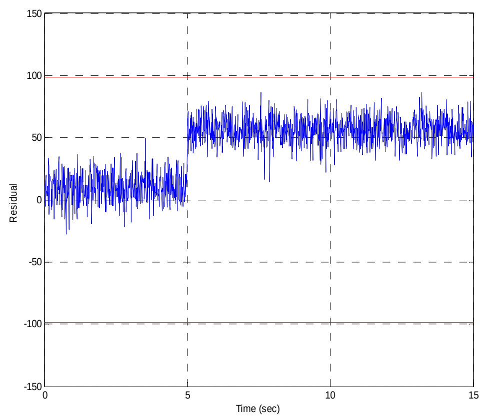
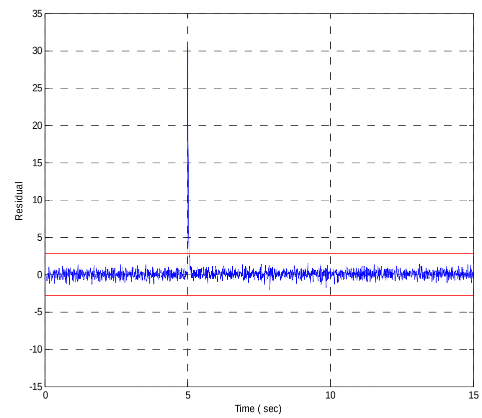


Figure 2-8. Desired Y-axis for circular trajectory

NLAR residual	RNLAR residual
 Figure 2-9(a). Under no MPM and disturbances	 Figure 2-9(b). Under no MPM and disturbances

NLAR residual	RNLAR residual
 Figure 2-10(a). Under partial fault in the right actuator	 Figure 2-10(b). Under partial fault in the right actuator
 Figure 2-11(a). Under bias right actuator fault	 Figure 2-11(b). Under bias right actuator fault

NLAR residual	RNLAR residual
 Figure 2-12(a). Under partial fault in the left actuator	 Figure 2-12(b). Under partial fault in the left actuator
 Figure 2-13(a). Under bias left actuator fault	 Figure 2-13(b). Under bias left actuator fault

Appendix I

Proof:

Part (i): The eigenvalues of $R, \lambda_i(x_e)$, is a function of x_e. We say $\lambda_i(x_e)$ is positive if $\lambda_i(x_e) > 0 \quad \forall x_e \in U_r$, and negative if $\lambda_i(x_e) < 0 \quad \forall x_e \in U_r$.

Let us consider the case where $\mu^-(R) \geq 2$. First we prove that $w = \sum_{i=1}^{\mu^-} \alpha_i V_i$ satisfies Condition 2. Without loss of generality consider $i = 2$, let λ_1 and λ_2 be the non-positive eigenvalues and V_1 and V_2 be the corresponding eigenvectors. Then $w(x_e) = \alpha_1(x_e)V_1(x_e) + \alpha_2(x_e)V_2(x_e)$, where $\alpha_1(x_e)$ and $\alpha_2(x_e)$ are the chosen coefficients.

Observe that *R* is a symmetric matrix. To see this,

$$S_G^T = \left(GG^T\right)^T = \left(G^T\right)^T G^T = GG^T = S_G$$

This implies S_G is symmetric. For similar reason S_H is also symmetric. *R* is the linear combination of S_G and S_H, hence *R* is also symmetric.

Now,

$$
\begin{aligned}
w^T Rw &= \left(\alpha_1 V_1^T + \alpha_2 V_2^T\right) R \left(\alpha_1 V_1 + \alpha_2 V_2\right) \\
&= \left(\alpha_1 V_1^T + \alpha_2 V_2^T\right)\left(\alpha_1 RV_1 + \alpha_2 RV_2\right) \\
&= \left(\alpha_1 V_1^T + \alpha_2 V_2^T\right)\left(\alpha_1 \lambda_1 IV_1 + \alpha_2 \lambda_2 IV_2\right) \\
&= \alpha_1^2 \lambda_1 V_1^T V_1 + \alpha_2^2 \lambda_2 V_2^T V_2 + \alpha_1 \alpha_2 \left(\lambda_2 V_1^T V_2 + \lambda_1 V_2^T V_1\right)
\end{aligned}
$$

where I is the identity matrix and V_1^T and V_2^T represent the transpose of V_1 and V_2, respectively. Since *R* is a symmetric matrix, it implies $V_2^T V_1 = 0$ and $V_1^T V_2 = 0$. Hence

$w^T Rw = \lambda_1 \alpha_1^2 V_1^T V_1 + \lambda_2 \alpha_2^2 V_2^T V_2$ But $\alpha_1^2 V_1^T V_1$ and $\alpha_2^2 V_2^T V_2$ are always positive, hence $w^T Rw \le 0$ for $\lambda_1, \lambda_2 \le 0$.

This proves the first part of Part (*i*).

The second part of the claim, i.e., for $i \ge 2$ we can always choose $\alpha_i(x_e)$ such that the Condition 1 is satisfied. This is obvious because there is only one constraint and more than one variable. This completes the proof of first claim.

Part (iii): Lets assume that $\exists$ nonzero w that satisfies Conditions 1 and 2 when $\mu^-(R) = 0$. More specifically w satisfies $w^T Rw \le 0 \forall x$. R is a symmetric matrix with all positive eigenvalues. That implies R is a positive definite matrix, which then means $X^T RX > 0 \ \forall X \ne 0$, where X is an arbitrary nonzero vector. This is a contradiction. This completes the proof.

Part (ii): This is a direct consequence of the other two claims. This completes the proof of the theorem.

APPENDIX II:

Part A:

Table 2-1. Parameters for WMR

P_o	the intersection of the axis of symmetry with the driving wheel axis
P_c	the center of mass of the platform with coordinates (x_c, y_c)
x, y	the world coordinate system
ϕ	the heading angle measured from the x-axis
i, j	the local coordinate system fixed with the WMR with (0 ,0) at P_o
d	the distance between P_o and P_c
b	the distance between either driving wheel and the axis of symmetry

r	radius of each driving wheel
c	$r/2b$
M_c	the mass of the WMR
J_c	the rotation inertia of the WMR about a vertical axis through P_c
m_w	mass of each wheel
I_w	inertia of each wheel
θ_r, θ_l	angular positions of the two driving wheels, respectively

Part B: Description of matrixes *M* and *V* and the error term

$$M(q_x) = \begin{bmatrix} M_r & 0 & -M_c cd\sin\phi & M_c cd\sin\phi \\ 0 & M_r & M_c cd\cos\phi & -M_c cd\cos\phi \\ -M_c cd\sin\phi & M_c cd\cos\phi & Jc^2 + I_w & -Jc^2 \\ M_c cd\sin\phi & -M_c cd\cos\phi & -Jc^2 & Jc^2 + I_w \end{bmatrix}$$

where

$$M_r = M_c + 2m_w$$

$$J = J_c + 2m_w b^2$$

$$V(q_x, \dot{q}_x) = \begin{bmatrix} -M_c d\phi^2 \cos\phi \\ -M_c d\phi^2 \sin\phi \\ 0 \\ 0 \end{bmatrix}$$

$$f^n(x) = \begin{bmatrix} S\dot{\Theta} \\ -\left(S^T MS\right)^{-1}\left(S^T MS\dot{\Theta} + S^T V\right) \end{bmatrix}, \dot{\Theta} = \begin{bmatrix}\dot{\theta}_r & \dot{\theta}_l\end{bmatrix}^T$$

$$S=\begin{bmatrix} cb\cos\phi-cd\sin\phi & cb\cos\phi+cd\sin\phi \\ cb\sin\phi+cd\cos\phi & cb\sin\phi-cd\cos\phi \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \text{ and } \quad g^n(x)=\begin{bmatrix} 0 \\ (S^T MS)^{-1} \end{bmatrix}$$

Actual mass of the mobile robot, $M_c^a = M_c + \Delta M_c$. Substituting this in the overall mass of the robot we get $M_r^a = M_c^a + 2m_w$

$$M^a(q)=\begin{bmatrix} M_r^a & 0 & -M_c^a cd\sin\phi & M_c^a cd\sin\phi \\ 0 & M_r^a & M_c^a cd\cos\phi & -M_c^a cd\cos\phi \\ -M_c^a cd\sin\phi & M_c^a cd\cos\phi & Jc^2+I_w & -Jc^2 \\ M_c^a cd\sin\phi & -M_c^a cd\cos\phi & -Jc^2 & Jc^2+I_w \end{bmatrix} \text{ and } \Delta M = M^a - M$$

$$V^a(q_x,\dot{q}_x)=\begin{bmatrix} -M_c^a d\phi^2\cos\phi \\ -M_c^a d\phi^2\sin\phi \\ 0 \\ 0 \end{bmatrix} \text{and } \Delta V = V^a - V$$

$$f(x)=\begin{bmatrix} S\dot{\Theta} \\ -\left(S^T M^a S\right)^{-1}\left(S^T M^a S\dot{\Theta} + S^T V^a + S^T F + S^T d\right) \end{bmatrix}, \text{ and}$$

$$f^u(x) = f(x) - f^n(x)$$

$$g(x)=\begin{bmatrix} 0 \\ (S^T M^a S)^{-1} \end{bmatrix} \text{ and } g^u(x) = g(x) - g^n(x)$$

References

[1] R. V. Bread, “Failure Accommodation in Linear Systems through Self-reorganization.” PhD, dissertation, Massachusetts Inst. Technology., Cambridge, MA, 1971.

[2] E. Chow, and A. Willsky, “Analytic redundancy and the design of robust failure detection systems”, IEEE trans. on Automatic Control, Vol. AC-29, No. 7. July 1984.

[3] C. DePersis and A. Isidori, “A Geometric Approach to Nonlinear Detection and Isolation”, IEEE trans on Automatic Control, vol. 46, No. 6, June 2001.

[4] X. Ding, L. Guo, and T. Jeinsch, “A characterization of parity space and its application to robust fault detection,” IEEE transaction on automatic control, VOL. 44, No. 2, February 1999.

[5] L. E. Dubins, “On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents,” American Journal of mathematics, 79: 497-516, 1957.

[6] P.M Frank, and X. Ding, “Survey of robust residual generation and evaluation method in observer-based fault detection system,” J. Proc. Cont. Vol. 7. No. 6. pp 403-424, 1997.

[7] P.M Frank and X. Ding, “Frequency domain approach to optimally robust residual generation and evaluation for model-based fault diagnosis”, Automatica, 30, 789-804, 1994.

[8] P. M. Frank and J. Wunnenberg,, “Robust fault detection using unknown input observer schemes”, in fault diagnosis in dynamic systems: theory and application, R. Patton, P. M. Frank and R. Clark, Eds: Prentice hall, 1989.

[9] J. Gertler, “Analytic redundancy methods in fault detection and isolation – survey and synthesis”, Preprints of IFAC Safeprocess conference, vol. 1, pp. 9-22. 1991.

[10] J. Gertler, and M Kunwer, “Optimal residual decoupling for robust fault diagnosis”, International Journal of Control, 61, 395-421, 1995.

[11] F. Gustafsson and S. Graebe, “Closed-loop performance monitoring in the presence of systems changes and disturbances,” Automatica, 34, 1311-1326, 1998.

[12] Z. Han, W. Li, and S. L. Shah, “Fault detection and isolation in the presence of process uncertainties”, Control engineering practice, 13, 587-599, 2005.

[13] R. Isermann, “Process fault detection based on modeling and estimation methods- a survey,” Automatica, 20, 387-404.

[14] A. Isidori. Nonlinear control systems. Springer-Verlag, London, UK, 1995.

[15] H. L. Jones, “Failure Detection in Linear Systems,” PhD, dissertation, Massachusetts Inst. Technol., Cambridge, MA, 1973.

[16] M. Junzheng and W. Shoukun, “Robust fault detection using iterative learning observer for nonlinear systems,” Proc. 5^{th} world conf. intelligent control and automation, June 15-19, China, 2004.

[17] H. K. Khalil. Nonlinear Systems. Second edition, 1996, pp.148.

[18] C. Kwan and R. Xu, "A note on simultaneous isolation of sensor and actuator faults," IEEE transaction of control systems technology, vol. 1, no. 1, Jan 2004.

[19] M. L. Leuschen, I. D. Walker, Joseph R. Cavallaro, "Fault residual generation via nonlinear analytic redundancy", IEEE trans. On control systems technology, vol. 13. No. 3, May, 2005.

[20] M. L. Leuschen, "Derivation and application of nonlinear analytic techniques with application on robotics", PhD, dissertation, Rice University. Texas, 2001.

[21] T. F. Lootsma, "Observer-based fault detection and isolation for nonlinear system", PhD, dissertation, Aalborg University., Denmark, 2001.

[22] A. D. Luca and R. Mattone, "Actuator failure detection isolation using generalized momenta," proceedings of the 2003 IEEE, ICRA, 634639, 2003.

[23] T. Marcu and L. Mirea, "Robust detection and isolation of process faults using neural networks," IEEE control systems, pp: 72-79, 1997.

[24] M. A. Massoumnia, G.C. Varghese and A. S. Willsky, (1989) "Failure Detection and Identification. IEEE Trans. Automat. Control. 34(3), 316-321

[25] N. Mehranbod, M. Soroush and C. Panjapornpon, "A method of sensor fault detection and identification ", Journal of Process Control, Volume 15, Issue 3, April 2005, Pages 321-339 G.

[26] H. Noura, D. Sauter, F. Hamelin, and T. Didier, "Fault tolerant control in dynamic systems: Application to a winding machine," IEEE Control Syst. Mag., vol. 20, pp. 33-49, 2000.

[27] S. Okina, K. Kawabata, T. Fuji, Y. kunii, H. Asama and I. Endo, " Self-diagnosis Systems of an Autonomous Mobile Robot Using Sensory Information" J. of Robotics and Mechatronics, 12, pp.72-77, 2000.

[28] R.J. Patton, J. Chen, and C.J. Lopez-Toribio, " Fuzzy observer for nonlinear dynamic systems fault diagonosis," in Proc 37th IEEE Conf. on Decision and control, 1998, vol. 1, pp. 84-89.

[29] W. Press et al. "Numerical recipes in C," Cambridge University press, 1997.

[30] S.I. Raumeliotis and G. S. Sukhatme, "Fault detection and identification in a mobile robot using multiple model estimation" , Robotics and automation, vol. 3, 2223-2228, 1998.

[31] N. Sarkar, X. Yun, and V. Kumar, “Control of Mechanical Systems with Rolling Constraints: Application to Dynamic Control of Mobile Robots”, International Journal of Robotics Research (MIT press), vol. 13, pp. 55-69, 1994.

[32] D. N. Shields and S. Du, “Fault detection observers for continuous non-linear polynomial systems of general degree,” Int journal of control, vol. 76, No. 5, pp. 437-452, 2003.

[33] A. Shumsky, “Robust analytical redundancy relations for fault diagnosis in nonlinear systems,” Asian journal of control, vol. 4, No. 2, pp. 159-170, June 2002.

[34] E. N. Skoundrianos and S. G. Tzafestas, “Fault diagnosis on the wheels of a mobile robot using local model neural networks”, IEEE Robotics and Automation Magazine, pp. 83-90, 2004.

[35] R. Sreedhar, B. Fernandez, and G. Y. Masada, “Robust fault detection in nonlinear systems using sliding mode observer,” second IEEE conference on control applications, September 13-16, Vancouver, B.C, 1993.

[36] M. Staroswiecki, J. P. Cassar, and G. Comtet-Varga, “Analytic redundancy relations for state affine systems.” In proceedings of the fourth European control conferences, Brussels, Belgium, 1997.

[37] T. Wei, Y.C. Hon, and Y. B. Wang, “Reconstruction of numerical derivatives from scattered noisy data,” Institute of Physics publishing, pp. 657-672, 2005.

[38] J. Wunnenburg, “Observer based fault diagnosis in dynamic systems,” VDI-Fortschrittsberichte, 8-222, 1990.

[39] L. Zhang, L. B. Jack and A. K. Nandi, “Fault detection using genetic programming ”, Mechanical Systems and Signal Processing, Volume 19, Issue 2, March 2005, Pages 271-289

[40] X. Zhang, “Fault diagnosis and fault-tolerant control in nonlinear systems,” PhD thesis, University of Cincinnati, May 2002.

[41] A.N. Zhirabok and O.V. Prebragenskaya, “Instrument fault detection in nonlinear dynamic systems.” In proceedings of the 1993 IEEE international conferences on systems, Man , and cybernetics, pp. 114-119, Touquet, France, oct 1993.

CHAPTER III: MANUSCRIPT 2

ROBUST FAULT DETECTION OF ROBOTIC MANIPULATOR

Bibhrajit Halder[1] Nilanjan Sarkar[2]

(Accepted in International Journal of Robotics Research)

Abstract

In this paper, a new robust fault detection technique for robotic manipulators is developed. The new approach called *robust nonlinear analytic redundancy* (RNLAR) technique detects both the sensor and actuator faults in robotic manipulator. The proposed RNLAR technique can compensate for the effects of model-plant-mismatch (MPM) and process disturbance. The RNLAR can be used to design primary residual vectors (PRV) for nonlinear robotic systems to detect sensor and actuator faults. A nonlinear PRV design method to detect faults is proposed where the PRVs are highly sensitive to the faults and less sensitive to MPM and process disturbance. Experimental

---

[1]Graduate Research Assistant, Department of Mechanical Engineering, Vanderbilt University, 2301 Vanderbilt Place, Nashville, TN-37235. Email: bibhrajit.halder@vanderbilt.edu Phone: 1-615-343-6472, Fax: 1-615-343-6687.

[2]Associate Professor, Department of Mechanical Engineering, Vanderbilt University, 2301 Vanderbilt Place, Nashville, TN-37235. Email: nilanjan.sarkar@vanderbilt.edu Phone: 1-615-343-7219, Fax: 1-615-343-6687.

results on a PUMA 560 are presented to justify the effectiveness of the RNLAR scheme.

Keywords: Fault detection, analytical redundancy, robustness, nonlinear system

1. Introduction

The demand for automation in modern society has been increasing steadily during the last few decades. Robotic manipulation systems played an important role in automation industries that include manufacturing, assembly, biotechnology to name a few. In addition, there is a growing need for unmanned operation in different service and research sectors such as search and rescue, nuclear waste clean up, planetary exploration and others where robotic manipulators play an equally important role. However, notwithstanding their widespread applicability and use, robotic manipulators are known to fail under normal operations [1] due to various faults that include sensor and actuator faults, and component failure. Typical faults are caused by broken or bias sensor, wear in mechanical components, overheating, and locked or damaged actuator. The likelihood of developing fault increases both with the complexity and versatility of the manipulator mechanism (e.g., the more the number of components, the more the possibility of developing faults) and the uncertainty of application domains (e.g., operating in hazardous unstructured situations). Consequently, the reliability and safety of robotic manipulators have received significant interest in recent years. One way to address these needs is to design a fault tolerant control system (FTCS) for robotic manipulators. Generally, a FTCS consists of two major components: fault detection and isolation (FDI) scheme, and a fault accommodation mechanism. In this work, we focus on the development of a new fault detection technique for robotic manipulators that can be effective in the presence of modeling uncertainties and disturbances.

There are significant research works on robot fault detection in the literature. Fault detection techniques are broadly classified into two classes: model-free [2-7] approaches and model-based approaches [8-18]. Model-free approaches to manipulator fault detection include neural networks and fuzzy logic to generate the residuals. Residuals are measures of discrepancy between the expected and the measured system behaviors. Model-based fault detection techniques, on the other hand, utilize mathematical models of the plant to generate residuals. Given the previous research [27] on the modeling of robotic manipulators as well as the success of model-based approach, we choose to concentrate on designing a new fault detection mechanism using model-based techniques. Some important survey papers in the model-based fault detection method are [8-10]. The fundamental concept of model-based fault detection is analytical redundancy (AR). The basic idea of AR is the comparison between the actual behavior of the monitored plant and the behavior of a mathematical plant. Implementation methods of AR can be classified into two groups: 1) indirect implementation, based on diagnostic observers, and 2) direct implementation based on parity relation technique [11]. In [38] fundamental equivalence between parity relation and diagnostic observer based method was presented. In this work, we present a new fault detection mechanism that is based on parity relation technique. As a result, we only mention a few major works on fault detection based on diagnostic observers and concentrate primarily on the relevant literature on fault detection using parity relation technique.

Various nonlinear diagnostic observer designs are proposed and implemented on robotic manipulators to detect sensor and actuator faults [12-16]. Most of the works in fault detection consider either a sensor fault or an actuator fault. In [12], the authors proposed a method based on generalized momenta for actuator fault detection. However, the proposed method could not detect sensor faults and was not robust in the presence of disturbance, noise and model-plant-mismatch (MPM). The authors of [12] later presented [13] an adaptive scheme to encompass the uncertain robot dynamics. A discrete-time diagnostic observer was designed in [14], where MPM, disturbance and noise were included in the system. They experimentally tested the proposed adaptive method on an industrial manipulator. For more observer design methods for FDI please refer to [15-16] and the reference therein. In [17] an observer-based fault detection approach was demonstrated experimentally for total actuator failure (i.e., the actuator was considered to be completely damaged). In [18] the partial actuator fault was considered in detail but sensor fault detection method was not discussed. In papers [19-20] only sensor fault detection for robot manipulator was considered.

Conceptually, the direct implementation based on parity relation is more straightforward than the observer based approach [11]. But the literature on parity relation based fault detection of robot manipulator is not rich. This is mainly due to lack of theoretical work on parity relation for nonlinear systems. Most research results on parity based fault detection techniques are for linear systems. The origin of parity relation that was based on analytic redundancy (AR) can be found in [28] for linear systems. The detail description of parity relation for linear systems is given in [25]. However, since a robotic manipulator is a highly nonlinear system, the above-mentioned results cannot be directly applied here.

Various researchers have combined linear AR with nonlinear systems [39] [40], by using the method of linearization of the nonlinear system. The AR based parity relation was later extended to nonlinear systems in [21-22]. This work assumed the existence of a perfect system model for fault detection and did not consider the presence of disturbance. We argue that model-plant-mismatch (MPM) and disturbances exist in most practical systems and therefore require explicit analysis of these issues in the context of parity relation based approach to fault detection of robotic manipulators.

In the AR literature robustness issue is discussed only for linear systems. In [28] robust residual generation was considered for linear system based on an optimization technique. Later, the authors in [29] extended the method given in [28], where the effect of the disturbances was minimized through the use of an unknown input observer. In papers [29-31] the robust residuals were designed but they all considered only MPM in residual generation. Recently, in [32-33] the authors extended the method presented in [28] to design the primary residual, which considered both the MPM and process disturbances in linear systems. As far as nonlinear systems are concerned, there is a lack of literature on parity relation based robust fault detection method. One notable work in this context is [33], where a mathematical technique, called algebra of functions is used for robust fault detection. However, it assumes that modeling uncertainty can be specified in the form of unknown constant or slowly varying system parameters.

It is worth mentioning at this point that there exists a body of work on the fault tolerance of robotic manipulators that seek to determine fault tolerance measures (mostly using kinematic redundancies of the systems) with the assumption of existence of fault detection method [24-26]. The success of such an approach depends on robust and reliable fault detection techniques [23]. We believe that our proposed technique will complement this body of literature.

The new robust nonlinear analytic redundancy (RNLAR) method accommodates MPM and process disturbances. The RNLAR method can detect both the sensor and actuator fault. We extend the robustness idea, given in [32] for linear systems, into the nonlinear domain. The RNLAR scheme is experimentally tested on a PUMA 560 robotic manipulator.

2. Dynamic Model of Robot Manipulator

We used a Unimate PUMA 560, as shown in Figure 3-1, for experiment. PUMA is well-characterized industrial manipulator that has been utilized in numerous industrial and robotic research applications. PUMA is a three degree-of-freedom harmonic-drive manipulator with a three degree-of-freedom wrist attached at its endpoint.

Figure 3-1. Unimate PUMA 560 robot manipulator

Armstrong et al. [27] derived an explicit dynamic model of the PUMA 560 arm and measured the parameters necessary to implement model-based control. We mention the equation of motion that is expressed in generalized coordinates of the PUMA arm, θ_1, θ_2, and θ_3, where

θ_1 : the angle of rotation of the Link 1 about the vertical axis

θ_2 : the angle measured from horizontal to Link 2

θ_3 : the angle measured from Link 2 to Link 3

They are represented in vector form by:

$$\theta = \begin{bmatrix}\theta_1 & \theta_2 & \theta_3\end{bmatrix}^T \tag{1}$$

In the absence of joint friction, the equation of motion for the robot manipulator is:

$$M(\theta)\ddot{\theta} + N(\theta)\left[\theta\,\dot{\theta}\right] + P(\theta)\left[\dot{\theta}^2\right] + G(\theta) = T \tag{2}$$

where $M(\theta)$ represents the inertia matrix, $N(\theta)$ is the matrix of Coriolis torques, $P(\theta)$ is the matrix of centrifugal torques, $G(\theta)$ is the vector of gravity torques, $\left[\theta\,\dot{\theta}\right]$ are notation for the vector of velocity products, $\left[\dot{\theta}^2\right]$ are vectors of squared velocities and T is the generalized joint force torques. The details of each term and the numeric parameters for the components of the model of the PUMA arm are given in [27].

Equation (2) can be expressed in state space form as follows:

$$\dot{x} = f(x) + \sum_{i=1}^{3} g_i(x) u_i, \qquad y = Cx = \begin{bmatrix}\theta_1 & \theta_2 & \theta_3\end{bmatrix}^T \tag{3}$$

where $x = \begin{bmatrix}\theta_1 & \theta_2 & \theta_3 & \dot{\theta}_1 & \dot{\theta}_2 & \dot{\theta}_3\end{bmatrix}^T$,

$$f(x) = \begin{bmatrix} \dot{\theta}_1 & \dot{\theta}_2 & \dot{\theta}_3 & -M^{-1}\left(N\left[\theta\,\dot{\theta}\right] + P\left[\dot{\theta}^2\right]\right) - G \end{bmatrix}^T,\ g(x) = \begin{bmatrix} g_1 & g_2 & g_3 \end{bmatrix} = \begin{bmatrix} 0_{3\times 3} \\ M^{-1} \end{bmatrix}_{6\times 3}$$

$u = \begin{bmatrix} u_1 & u_2 & u_3 \end{bmatrix}^T \in \Re^3 = T$, C is a 3×6 output matrix and $y \in \Re^3$ is the fault-free actual process output. It is worth mentioning that (3) represents the nominal model of a PUMA 560.

In the presence of faults, the actuator input and the sensor output can be represented as:

$$u = u^g + u^f,\ \ y^o = Cx + y^f + o \tag{4}$$

where $u^g \in \Re^3$ represents the fault-free input vector, $u^f \in \Re^3$ represents the actuator fault vector, $y^o \in \Re^3$ represents the observed output vector, $y^f \in \Re^3$ represents the sensor fault vector and o represents a Gaussian-distributed white noise vector. It is assumed that u^g and y^o are available for computation but u^f, y^f and o are not. The magnitude of the noise is assumed to be significantly smaller then the magnitude of faults. Under the nominal fault-free condition, u^f and y^f are zero vectors. However, when either a sensor and/or an actuator fault occur in the system, u^f and y^f will become non-zero.

Model-plant-mismatch is represented by

$$f(x) = f^n(x) + f^u(x),\ \ g(x) = g^n(x) + g^u(x) \tag{5}$$

where $f^n(x), f^u(x), g^n(x)$, and $g^u(x)$ represent the nominal and uncertain part of the mappings f and g, respectively. Combining (3), (4), (5), and an unmeasured deterministic process disturbance vector, $d(x,u)$, the overall system with faults is represented by

$$\dot{x} = f^n(x) + f^u(x) + \left(g^n(x) + g^u(x)\right)\left(u^g + u^f\right) + d(x,u),\ \ y^o = Cx + y^f + o \tag{6}$$

Simplifying (6) we get

$$\dot{x} = f^n(x) + g^n(x)u^g + e(x,u) + g^n(x)\, u^f, \;\; y^o = Cx + y^f + o \tag{7}$$

where $e(x,u) = f^u(x) + g^u(x)\left(u^g + u^f\right) + d(x,u)$.

For a general system, we represent number of states by n, number of inputs by q, and number of outputs by m. The vector $e(x,u)$, called an error vector, contains both the uncertainty of the model and the disturbances.

The following assumptions will be used in this paper in order to design the robust fault detection method:

Assumption 1: The fault-free robotic manipulator is asymptotically stable. This is a general assumption in the literature [22] [32].

Assumption 2: System (3) is observable. This assumption is needed in order to guarantee the ability to find all the states from the system outputs. We should note that observability assumption does not mean that we can (or we need to) find the fault-free states from faulty output measurements.

Assumption 3: The modeling uncertainties denoted by $f^u(x)$ and $g^u(x)$ in (4), which are unknown nonlinear vector functions of *x*, are bounded. We also assume that both the inputs and the disturbances are bounded. Define fault-free error part, e^*, as $e^* = f^u(x) + g^u(x)u^g + d$. We assume that $\max\left\{\left\| f^u(x)\right\|, \;\; \left\|g^u(x)u^g\right\|, \;\; \left\|d(x)\right\|\right\} < \left\|F_o\left(x,u^g\right)\right\|$ where $F_o\left(x,u^g\right)$ is a known bounded function. Now, $\left\|e^*\right\| \le \left\|f^u(x)\right\| + \left\|g^u(x)u^g\right\| + \left\|d\right\|$ Thus we can say e^* is bounded, e.g., $\left\|e^*\right\| \le L$, where $\|\;\|$ stands for the L_2 norm and $L = 3\left\|F\left(x,u^g\right)\right\|$.

Thus the problem we seek to solve becomes: design robust residuals for sensor and/or actuator faults for the robotic manipulator given by (6). By robust we mean the residual will need to be sensitive to the faults but insensitive to the MPM and disturbances of the system, i.e., insensitive to $e(x,u)$ as much as possible.

3. Robust Nonlinear Analytic Redundancy

The analytic redundancy (AR) method for linear systems is given in [28]. The major issue in the use of analytic redundancy technique is how to deal with the presence of MPM and process disturbances, and their effect on the robustness of the resulting fault detection algorithm. In this work, we address the above issue for nonlinear systems. In order to present our mathematical framework, we first define several key matrices that will be needed in the subsequent development. We define the following matrices: a state matrix, Γ_s, an error matrix, G_s, and an input matrix, H_s

$$\Gamma_s = \begin{bmatrix} Cx \\ L_f Cx \\ L_{ff} Cx + \sum_{l=0}^{q} \sum_{j=0}^{q} L(j,l) \\ L_{fff} Cx + \sum_{k=0}^{q} \sum_{l=0}^{q} \sum_{j=0}^{q} L(j,l,k) \\ \\ \cdots\cdots \end{bmatrix}_{m(s+1)\times 1} \tag{8}$$

$$G_s = \begin{bmatrix} 0 & 0 & 0 \\ C & 0 & 0 \\ C\left(\sum_{i=0}^{l} u_i^g \partial k_i / \partial x\right) & C & 0 \\ \Lambda & C\left(\sum_{i=0}^{l} u_i^g \partial k_i / \partial x\right) & C \end{bmatrix}_{m(s+1)\times ns} \tag{9}$$

$$H_s = \begin{bmatrix} 0 & 0 & 0 \\ Cg^n & 0 & 0 \\ C\left[\left(\partial f^n / \partial x\right) g^n + \left(\partial g^n / \partial x\right) f^n\right] & Cg^n & 0 \\ \Lambda_h & C\left[\left(\partial f^n / \partial x\right) g^n + \left(\partial g^n / \partial x\right) f^n\right] & Cg^n \end{bmatrix}_{m(s+1)\times qs} \tag{10}$$

where

$L(j,l,m,...) = \left(u_j u_l u_m ...\right) L..._{k(m)k(l)k(j)} Cx.$

$u_0 = 1,\ k(j) = \begin{Bmatrix} f\ ,\ j = 0 \\ g_j\ ,\ j \neq 0 \end{Bmatrix}$

The Lie derivative is defined as $L_f h = \sum_{i=1}^{n} \frac{\partial h}{\partial x_i} f_i$ and the repeated Lie derivative [34] [35] is written in the following ways: $L_i(L_j(L_k h)) = L_i L_j L_k h = L_{ijk} h$. s is the order of the redundancy relation as defined in [28]. s describes the 'memory span' of the redundancy relation. The terms Λ and Λ_h contain higher order derivatives of the vector functions f^n and g_i^n.

Next, we define a new group formation, O_{NDD}, which is based on the sensor reading and given control inputs. We start with the output y^o as given in (7). We take the derivative of y^o for s times and stack them together in (11),

$$\begin{bmatrix} y^o \\ \dot{y}^o \\ \ddot{y}^o \\ . \\ . \\ y^{so} \end{bmatrix} = \begin{bmatrix} Cx + y^f + o \\ C\dot{x} + \dot{y}^f + \dot{o} \\ C\ddot{x} + \ddot{y}^f + \ddot{o}. \\ . \\ . \\ Cx^s + y^{sf} + o^s \end{bmatrix} \tag{11}$$

$$= \begin{bmatrix} Cx + y^f + o \\ C\left(f^n(x) + g^n(x)u^g + e(x,u) + g^n(x)\, u^f \right) + \dot{y}^f + \dot{o} \\ C\dfrac{d}{dt}\left(f^n(x) + g^n(x)u^g + e(x,u) + g^n(x)\, u^f \right) + \ddot{y}^f + \ddot{o} \\ . \\ . \end{bmatrix} \tag{12}$$

Define the stacked vectors, $y_s = \left[y^o \;\; \dot{y}^o \;\; \ddot{y}^o \;\; \dddot{y}^o \;\; \ldots \right]^T \in R^{m(s+1)}$, Similarly, we define the input stack vector, u_s, the error stack vector, e_s, actuator fault stack vector, u_s^f, sensor fault stack vector y_s^f, and the noise stacked vector, o_s as follows:

$$u_s = \left[u^g \;\; \dot{u}^g \;\; \ddot{u}^g \;\; \ldots \right]^T \in R^{qs}; \;\; e_s = \left[e \;\; \dot{e} \;\; \ddot{e} \;\; \ldots \right]^T \in R^{ns}$$
$$o_s = \left[o \;\; \dot{o} \;\; \ddot{o} \;\; \ldots \right]^T \in R^{m(s+1)}; \;\; y_s^f = \left[y^f \;\; \dot{y}^f \;\; \ddot{y}^f \;\; \ldots \right]^T \in R^{m(s+1)}$$

$$u_s^f = \left[u^f \;\; \dot{u}^f \;\; \ddot{u}^f \;\; \ldots \right]^T \in R^{qs}$$

Using the definitions of Γ_s, G_s, and H_s we rearrange (12) as follows:

$$y_s = H_s u_s + \Gamma_s + y_s^f + G_s e_s + H_s u_s^f + o_s \tag{13}$$

We define O_{NDD} for RNLAR as follows:

$$\mathrm{O}_{NDD} = y_s - H_s u_s \tag{14}$$

Equation (14) together with the definition of O_{NDD} implies

$$\mathrm{O}_{NDD} = \Gamma_s + y_s^f + G_s e_s + H_s u_s^f + o_s \tag{15}$$

Equations (13) and (15) will be used to derive the residuals for sensor and actuator faults. Note that in (14) y_s and u_s are outputs and inputs of the actual system. In (15) H_s, Γ_s and G_s are computed from the nominal system (i.e., the mathematical model of the plant).

4. Robust Fault Detection Method

In this section, we discuss the design procedure for robust fault detection for both actuator and sensor fault. The detail of robust actuator fault detection residual method was discussed in Manuscript I. The robust sensor fault detection residual method is discussed here.

4.1 Robust Sensor Fault Detection Method

We consider the sensor fault, hence the stacked actuator fault vector, $u_s{}^f$, is assumed to be zero. This simplifies (15) into

$$\mathrm{O}_{NDD} = \Gamma_s + y_s^f + G_s e_s + o_s \tag{16}$$

We would like to design the residual such that the sensor residual is completely insensitive to the error vector. To achieve this, we rearrange (16) to obtain

$$\mathrm{O}_{NDD} = y_s - H_s u_s = [\Gamma_x \, G_s] \begin{bmatrix} 1 \\ e_s \end{bmatrix} + y_s^f + o_s = \Omega_s \mathrm{E}_s + y_s^f + o_s \tag{17}$$

where $\Omega_s = [\Gamma_x \;\; G_s]$ and $E_s = \begin{bmatrix} 1 \\ e_s \end{bmatrix}$

We select a transformation matrix, w_s, which is located in the left null space of Ω_s, i.e., $w_s^T \Omega_s \equiv [0]$. Pre-multiplying both sides of (17) with w_s results:

$$R_s = w_s^T \mathrm{O}_{NDD} = w_s^T y_s^f + w_s^T o_s \tag{18}$$

It appears that R_s is completely insensitive to the error vector. But note that, for a full rank C matrix w_s can only have its first m columns to be nonzero and the rest of the elements to be zero due to the block-triangular structure of the matrix G_s since $w_s^T G_s = 0$. This implies that redundant sensors are needed to detect the sensor faults. Therefore, no sensor fault can be detected if the error vector is completely removed when the outputs are non-redundant (i.e., C is a full row rank matrix).

Faced with the above problem, we present a design methodology for the PRV that makes it insensitive to the error vector but sensitive to the sensor faults as much as possible. Select a transformation matrix, W_r that is located in the left null space of Γ_s, i.e., $W_r^T \Gamma_s \equiv [0]$. Pre-multiplying both sides of (16) with W_r^T result:

$$R_r = W_r^T \mathrm{O}_{NDD} = W_r^T y_s^f + W_r^T G_s e_s + W_r^T o_s \tag{19}$$

It can be observed that both the sensor fault and the error vector affect the PRV. It is desirable that R_r should be highly sensitive to the sensor faults and mostly insensitive to the error terms. The above desired property can be translated mathematically into the following statement: $\left\| W_r^T G_s \right\|$ is less than $\left\| W_r^T \right\|$, where the coefficient of the error vector is $W_r^T G_s$ and the coefficient of the fault vector is W_r^T. A similar problem arises during the design of PRV for actuator faults. In the literature this problem was discussed for linear systems. In [32] they frame this problem as a linear optimization problem and use the linearity property. For a nonlinear system, which is the case here, this translates into solving a nonlinear optimization problem where the functional structure of W_r is unknown. In other words, we do not know the functional form of each element of W_r (e.g., whether they are polynomial, exponential etc.) and we cannot realistically guess them without any other knowledge. This makes the nonlinear optimization problem very difficult to solve. In order to overcome this problem, we propose a novel method for designing W_r for nonlinear systems. Given the states $x \in \Re^n$ and inputs $u \in \Re^q$, consider an open set $U_r \in \Re^{n+q}$ such that the states and the inputs are restricted on U_r, i.e., $x_e = (x, u) \in U_r$. Here we omit the subscripts from W and other terms for notational simplicity. Define two performance functions,

$J = \dfrac{W^T GG^T W}{W^T HH^T W}$ for actuator fault residual, R_a and

$J = \dfrac{W^T GG^T W}{W^T II^T W}$ for sensor fault residual, R_r, where I is the identity matrix. We formulate the robust problem for actuator performance index in Manuscript I. The same procedure will be applicable to sensor performance index where H will be substituted by I.

5. Experimental Results

We use the first three joints of a Unimation PUMA 560 to verify the presented fault detection algorithm. We replaced the microcontroller board of the PUMA to develop an open architecture system. This allows us to implement the controllers that are essential for this experiment. In addition, we interfaced the robot with Matlab and Real-time Workshop to allow fast and easy system development. The joint angles of the robot are measured using encoders. The encoder readings are acquired with a sample time of 0.001seconds from a Measurement Computing PCI-QUAD04 card. The torque output to the robot is given with a Measurement Computing PCIM-DDA06/16 card with the same sample time. The encoder outputs are used for calculating the residuals in the experiment.

5.1 Experimental Set-up

We designed experiments to detect both actuator and sensor faults. In these experiments the PUMA was asked to track a circular trajectory in the *x*-*y* plane. The *x*-axis and *y*-axis for the circular task trajectories are given in Figures 3-2 and 3-3.

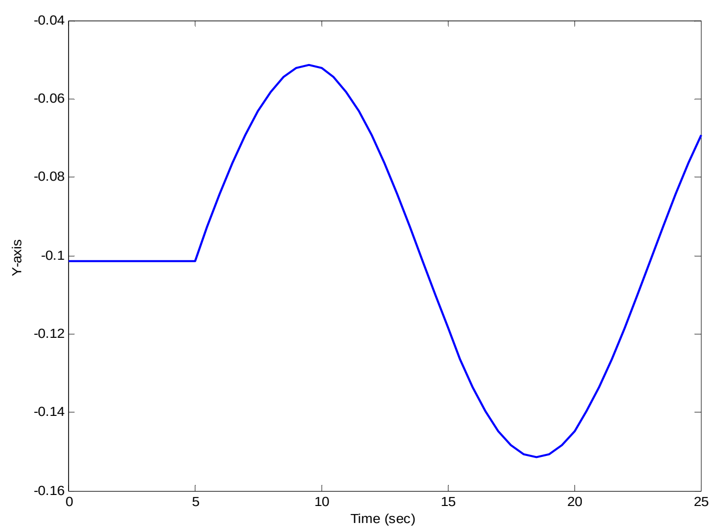


Figure 3-2. Desired X-axis trajectory

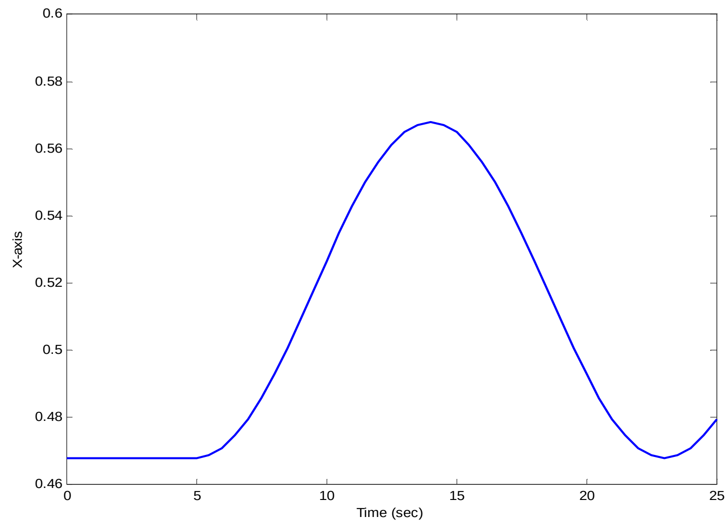


Figure 3-3. Desired Y-axis trajectory

While the manipulator tracked the trajectory we introduced senor and actuator faults (not at the same time) and monitored the sensor and actuator residuals. The endpoint of the PUMA was controlled by a PID controller with the following PID gains: P=400, I=5 and D=15. We should mention that the RNLAR residuals are independent of the choice of the controller. We compare our results with NLAR technique as described in [22] that does not consider MPM and disturbances in its formulation. We make the following comments before presenting the results:

1) In modeling the PUMA, the friction term is not considered. Also the parameter estimation for PUMA is not perfect. All these factors contribute to the MPM and the disturbances in the experiments.
2) We perform numerical differentiation using the derivative block, which uses forward differentiation technique, available in Matlab. Numerical differentiation of noisy sensor signal is well-known to be ill-posed in the sense

that a small noise in measurement data can induce a large error in the approximate derivatives [41]. Although we obtained reasonable results in the experiments using the standard numerical differentiation block provided by Matlab but one can use various low-pass-filters and regularization methods such as Savitzky-Golay smoothing filters [42] to reduce the effect of noise in differentiation if needed.

3) To calculate the residuals as in (19) we run the nominal model in parallel with the PUMA. We use the nominal model as in (3) to calculate the terms, G_s, H_s, and Γ_s while y_s and u_s are obtained directly from the experimental data.

4) Faults are considered detected if the magnitudes of the residuals cross some pre-determined threshold value. We design a threshold value as twice the absolute maximum value achieved in a fault-free run with the same parameters to demonstrate the effectiveness of the proposed RNLAR technique. Note that in order to minimize false alarms one would possibly choose an even larger threshold. In such a case, as will be seen from the results (described later), the RNLAR technique will outperform the NLAR technique even more significant manner.

5) Generating the RNLAR residuals for WMR require mathematical calculation of various terms. We use the Matlab symbolic toolbox and Mathematica to generate all the terms and combine them appropriately to create the RNLAR tests.

5.2 Actuator Fault Detection Results

Various types of actuator faults are discussed in [12]. We choose a partial actuator fault where one actuator generates only a part of the desired torque. This type of fault represents degradation in the actuator system (e.g., friction due to jamming, problems in transmission etc.). We introduced two kinds of partial fault: sudden partial fault and slow partial fault.

In the experimental set-up the sudden partial actuator faults were introduced by multiplying the controller calculated output by 0.75 after 11 seconds of operation. In order to demonstrate the robustness of the proposed RNLAR technique, we compare fault residuals generated from our proposed technique with that of the NLAR technique [21] for the same fault conditions. First, we present results when there is no fault. The NLAR and RNLAR residual results are given in Figures 3-4 and 3-5, respectively. These figures demonstrate the effect of error vector that captures MPM and disturbance on NLAR and RNLAR residuals, respectively. Next, we introduced a sudden partial fault on the first actuator. The fault detection result with the NLAR residual is presented in Figure 3-6 and that with the RNLAR residual is presented in Figure 3-7. The absolute maximum value of the NLAR signal in a fault-free run was 81.4. Thus the magnitude of the threshold value for NLAR residual was chosen as 162.8. It can be seen that before the fault occurred, the maximum value of the residual stayed within ± 81.4. Now observe that, in Figure 3-6 the absolute maximum value of NLAR signal was 143.56, which was less than the threshold value, 162.8. Hence, we can conclude that the fault was not detected for the given threshold. On the other hand, the absolute maximum value of RNLAR signal in a fault-free and faulty run was 11.32 and 3671.11 respectively. The magnitude of the threshold value was 22.64. Hence, the fault was detected clearly and rapidly (i.e., almost instantaneously) in the RNLAR test. It can be seen that the RNLAR residual is significantly more sensitive to the sudden partial actuator fault.

Next, the same experiment was repeated with slow partial fault in the first actuator. The slow partial actuator faults were introduced by multiplying the controller-calculated output with the function given in Figure 3-8. The residual test results are presented in Figure 3-9 for the NLAR technique and in Figure 3-10 for the RNLAR technique. In this case, the absolute maximum value of the NLAR signal in faulty run was 89.7, which was less than the threshold value, 162.8. The absolute maximum value for the RNLAR signal was 42.12 in a faulty run, which was more than the threshold value, 22.64. From these values we can conclude that with the chosen threshold the NLAR residual cannot detect the faults while RNLAR residual detects the fault with a time delay.

Next, we performed similar experiments with fault in the second actuator. Here we only present the comparison results for sudden second actuator fault. The fault detection result with the NLAR residual is presented in Figure 3-11 and that with the RNLAR residual is presented in Figure 3-12. In this case, the absolute maximum value of the NLAR signal in a faulty run was 92.7. The absolute maximum value for the RNLAR signal was 30800 in a faulty run. From these values we can conclude that with the chosen threshold the NLAR residual cannot detect the faults while RNLAR residual detects the fault clearly and quickly (i.e., almost instantaneously).

Finally, we introduced a fault in third actuator. Here, we present the RNLAR residuals under both the sudden and slow partial third actuator fault. The RNLAR residuals under sudden and slow partial actuator faults are shown in Figure 3-13 and 3-

14. We can observe that the fault was detected in both cases. It is clear from the above set of results that the presented RNLAR technique is useful in detecting actuator faults in the presence of MPM and disturbance. The fault detection using RNLAR technique is clear and fast.

5.3 Sensor Fault Detection Results

Various kinds of sensor faults in robotics are discussed in [36-37]. For experimental purpose partial sensor fault was considered where one encoder reflected only a fraction of the actual value. This type of fault occurs when there is an offset or bias in the sensor reading. We introduced two kinds of partial fault: sudden partial fault and slow partial fault.

In the experimental set-up the sudden partial sensor faults were introduced by multiplying the joint encoder with 0.80 after 11 seconds of operation. Here also, we compared both the NLAR and the RNLAR residuals. We introduced the sudden partial fault on first encoder. The residual test results are presented in Figure 3-15 for the NLAR technique and in Figure 3-16 for the RNLAR technique for the first encoder fault. In this case, the absolute maximum value of the NLAR signal in a fault-free run was 68.13 and in faulty run was 73.54. The magnitude of the threshold value for NLAR residual was chosen as 136.26; hence we conclude that the fault was not detected for the given threshold. The absolute maximum values for the RNLAR signal were 15.87 and 972.4 in a fault-free run and in a faulty run, respectively. From these values we can conclude that with the chosen threshold the NLAR residual cannot detect the sensor fault while RNLAR residual detects the fault clearly and quickly (i.e., almost instantaneously).

We repeated the same experiment with slow partial fault in the first encoder. The slow encoder fault was introduced in the same way as we did for the slow actuator fault. The fault detection results with the NLAR and RNLAR residuals are presented in Figure 3-17 and 3-18, respectively. The maximum value of the NLAR residual in a faulty run was 70.01, which was less than the threshold value, 142.19. Hence the fault was not detected. For the RNLAR residual, the absolute maximum value with fault was 69.84. Hence, the fault was detected with a time delay in the RNLAR test.

We introduced sudden partial fault in the second encoder. Here we only present the comparison results for sudden second encoder fault. The fault detection result with the NLAR residual is presented in Figure 3-20 and that with the RNLAR residual is presented in Figure 3-20. In this case, the absolute maximum value of the NLAR signal in a faulty run was 78.21. The absolute maximum value for the RNLAR signal was 29102 in a faulty run. From these values we can conclude that with the chosen threshold the NLAR residual cannot detect the faults while RNLAR residual detects the fault clearly and quickly.

Finally, we introduced a fault in third encoder. Here, we present the RNLAR residuals under both the sudden and slow partial third encoder fault. The RNLAR residuals under sudden and slow partial actuator faults are shown in Figures 3-21 and 3-22, respectively. We can clearly observe that the fault was detected in both cases. In conclusion, we have performed experiments with actuator and sensor faults on a PUMA 560. We presented the comparison results of NLAR and RNLAR. We observe that the RNLAR residuals successfully detected both the sensor and actuator faults for all cases under MPM and disturbances.

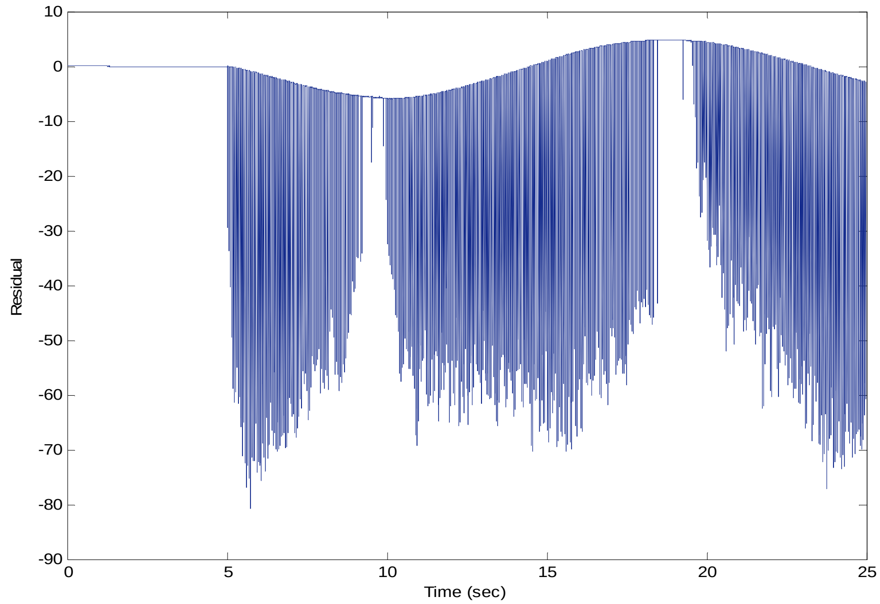


Figure 3-4. NLAR residual under no fault condition

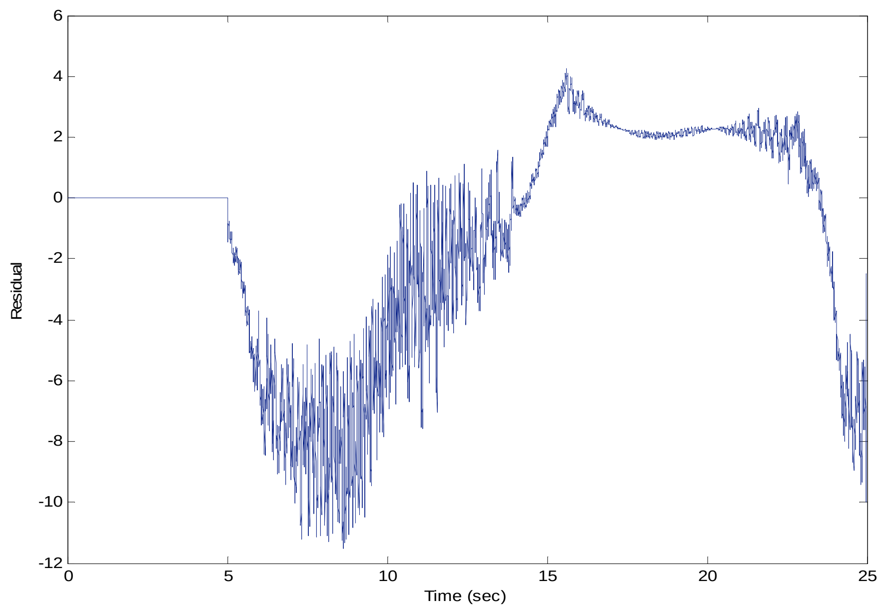


Figure 3-5. RNLAR residual under no fault condition

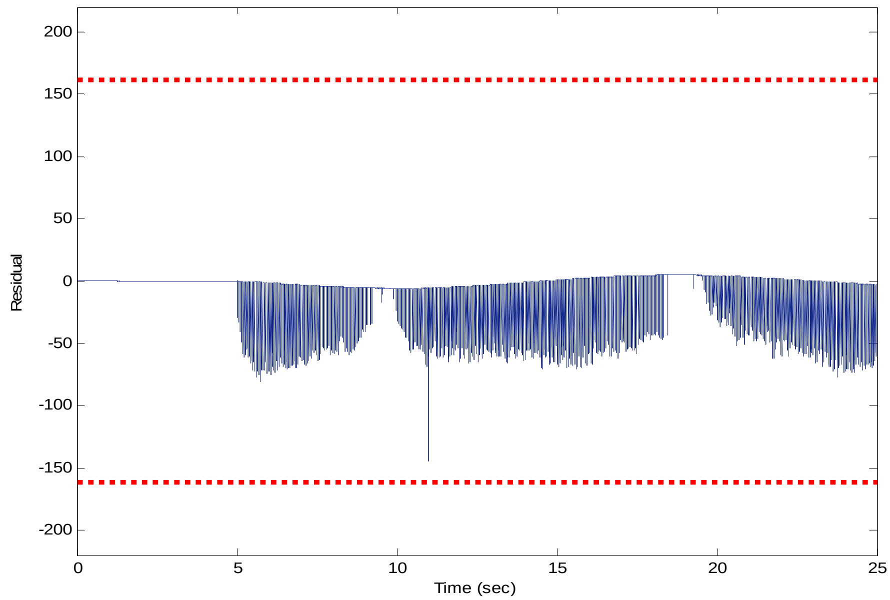


Figure 3-6. NLAR residual under sudden first actuator fault

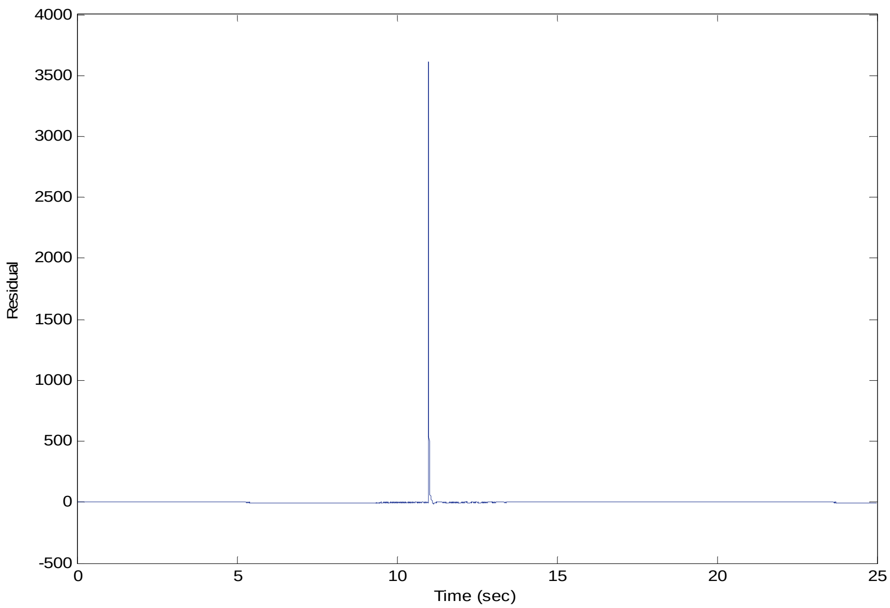


Figure 3-7. RNLAR residual under sudden first actuator fault

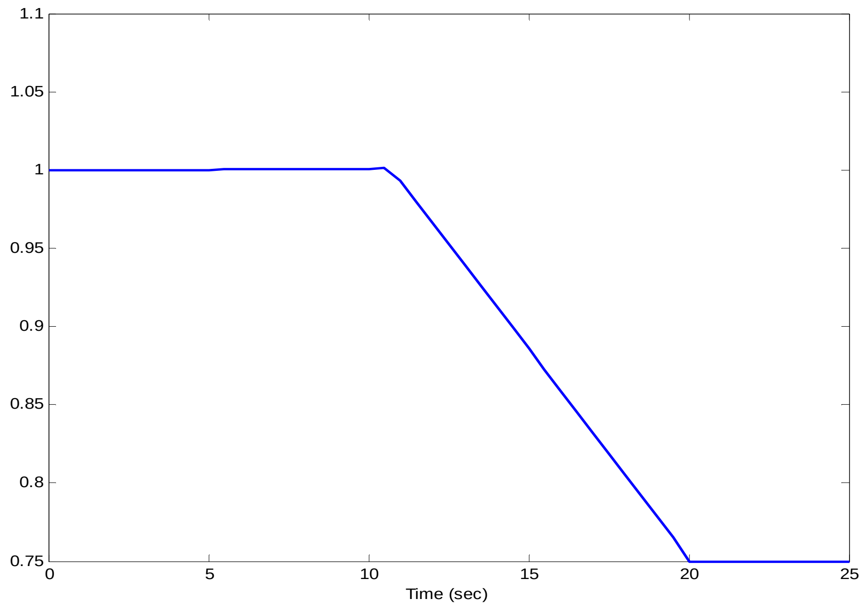


Figure 3-8. Multiplier for slow fault generation

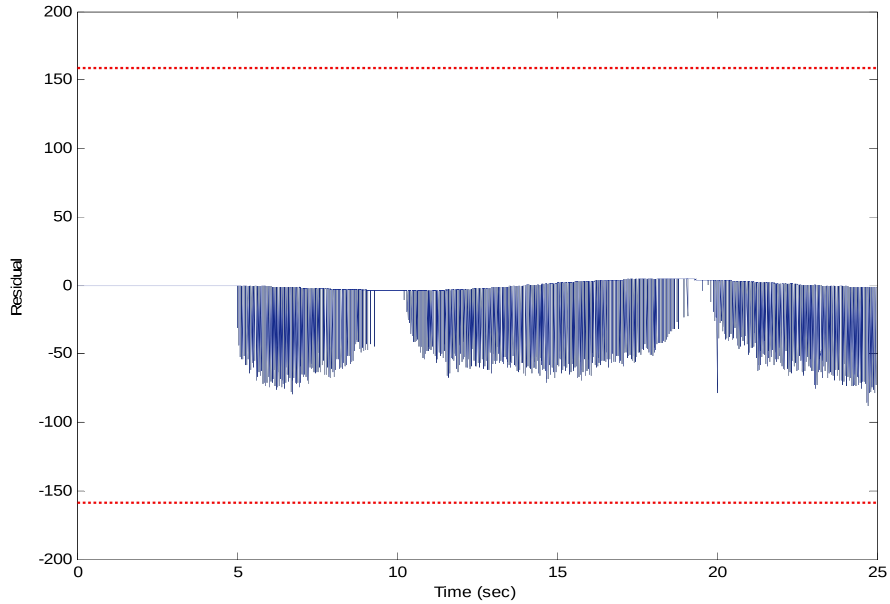


Figure 3-9. NLAR under slow first actuator fault

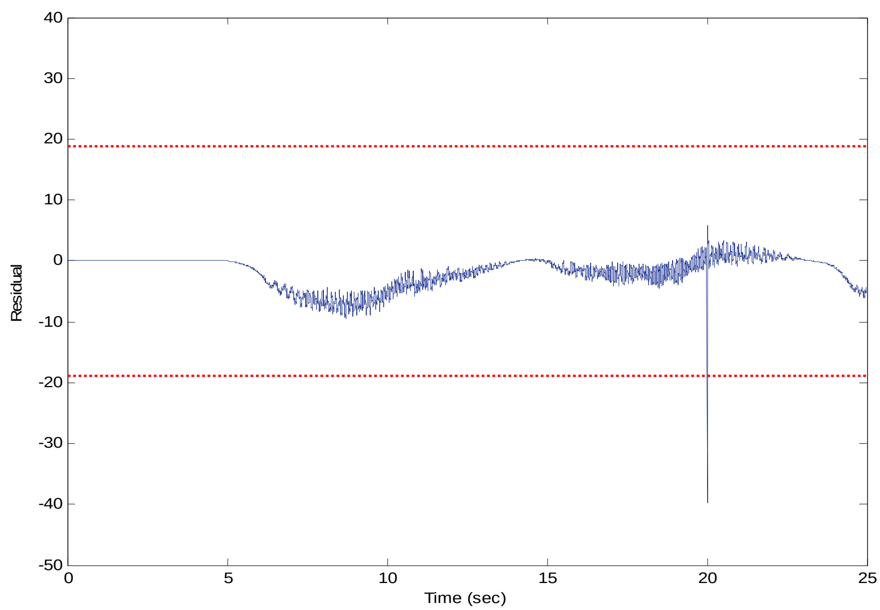


Figure 3-10. RNLAR under slow first actuator fault

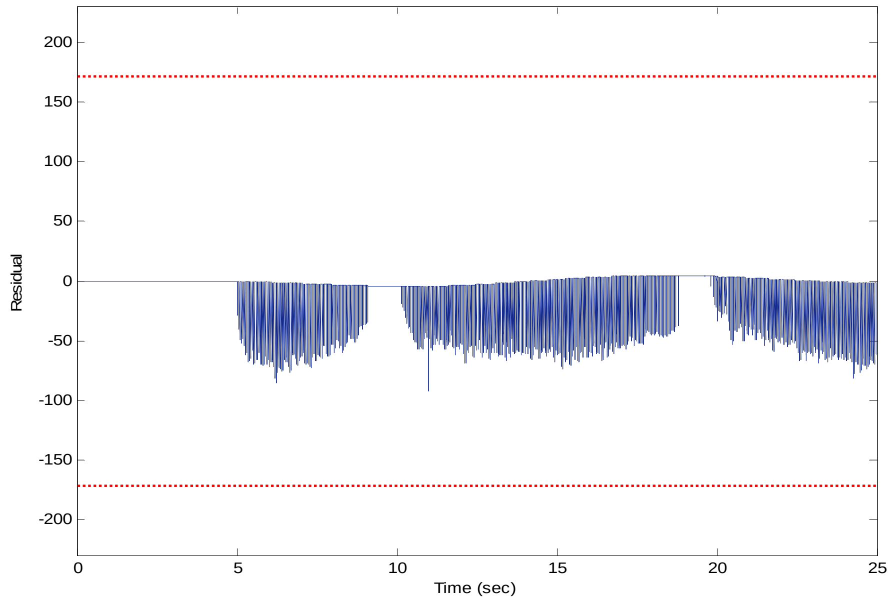


Figure 3-11. NLAR under sudden second actuator fault

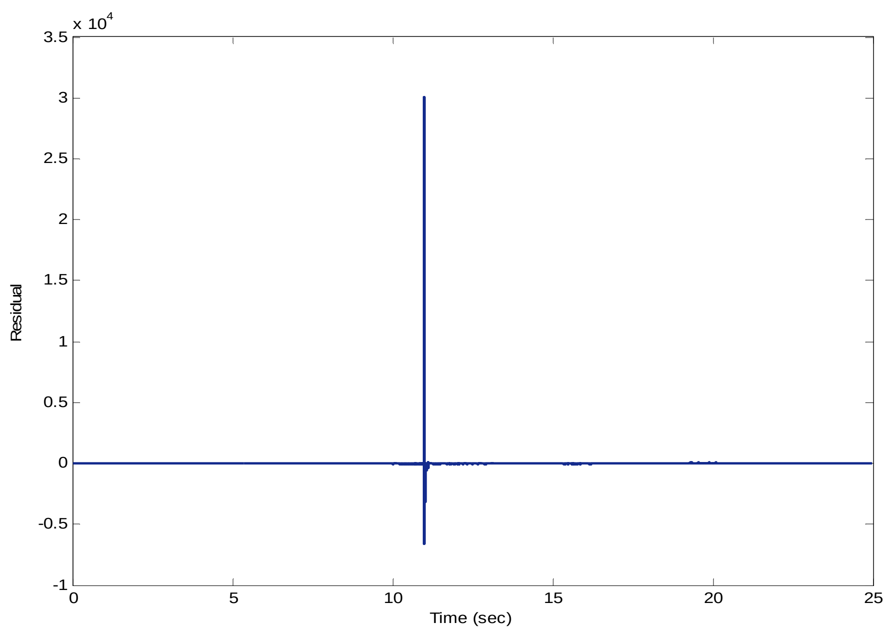


Figure 3-12. RNLAR under sudden second actuator fault

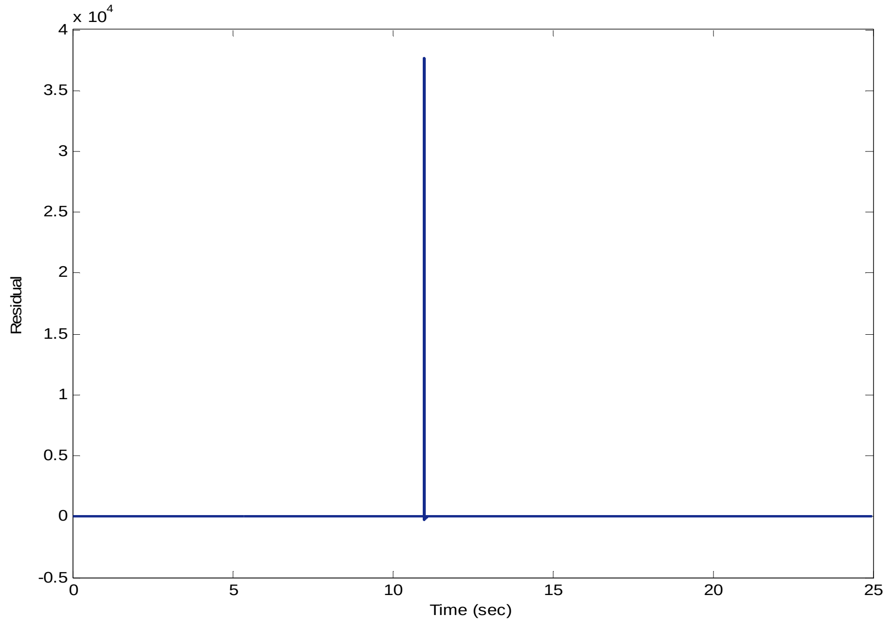


Figure 3-13. RNLAR under sudden third actuator fault

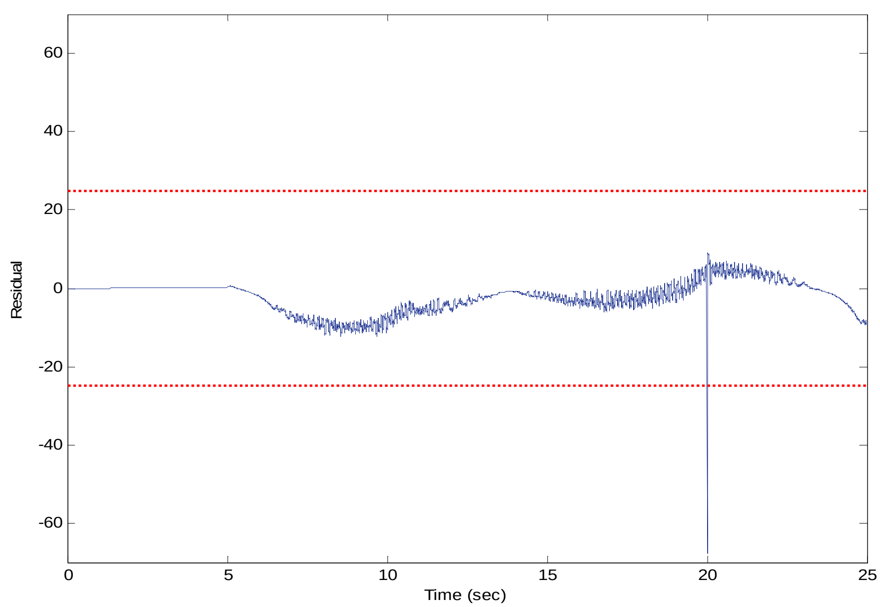


Figure 3-14. RNLAR under slow third actuator fault

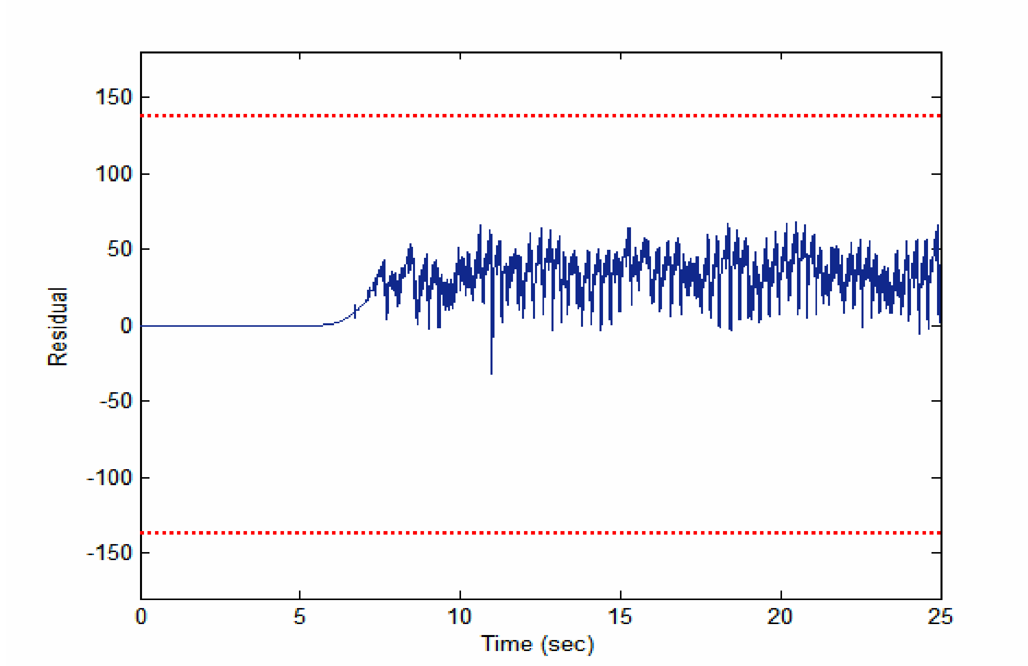


Figure 3-15. NLAR under sudden first encoder fault

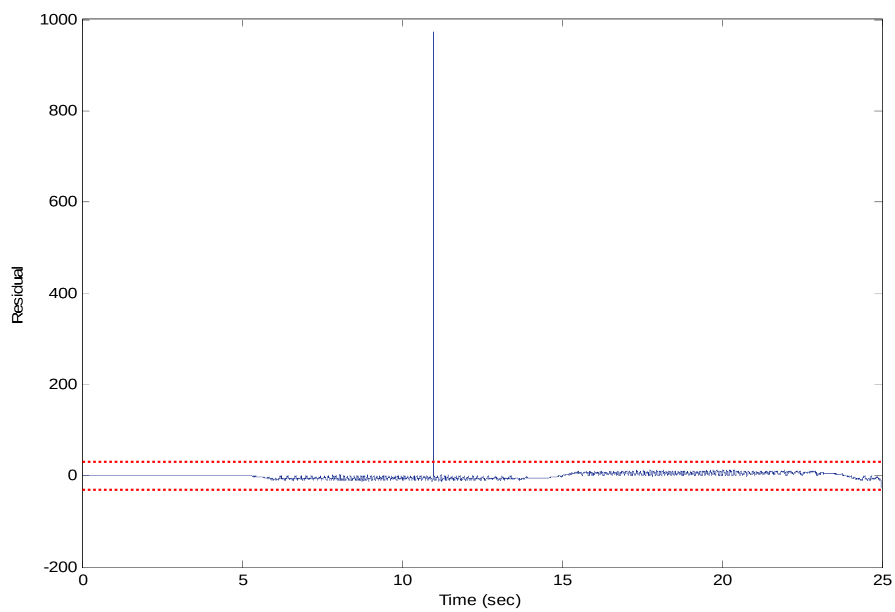


Figure 3-16. RNLAR under sudden first encoder fault

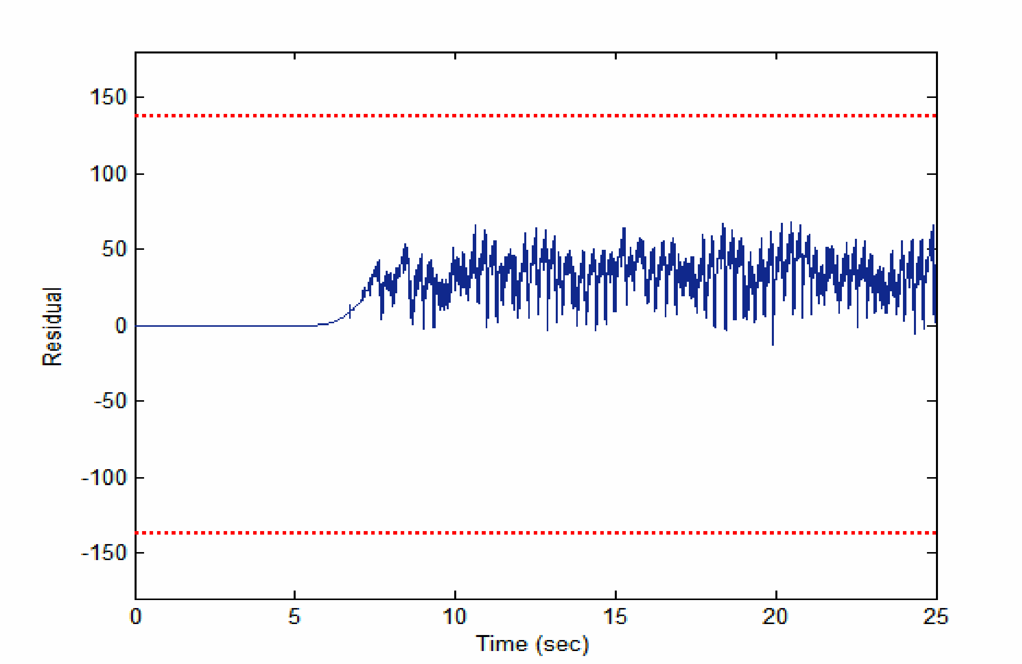


Figure 3-17. NLAR under slow first encoder fault

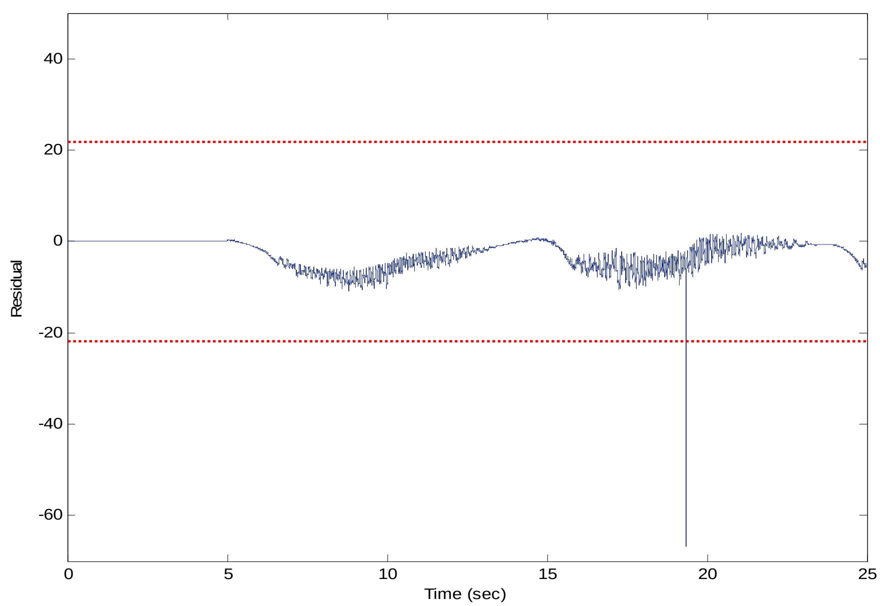


Figure 3-18. RNLAR under slow first encoder fault

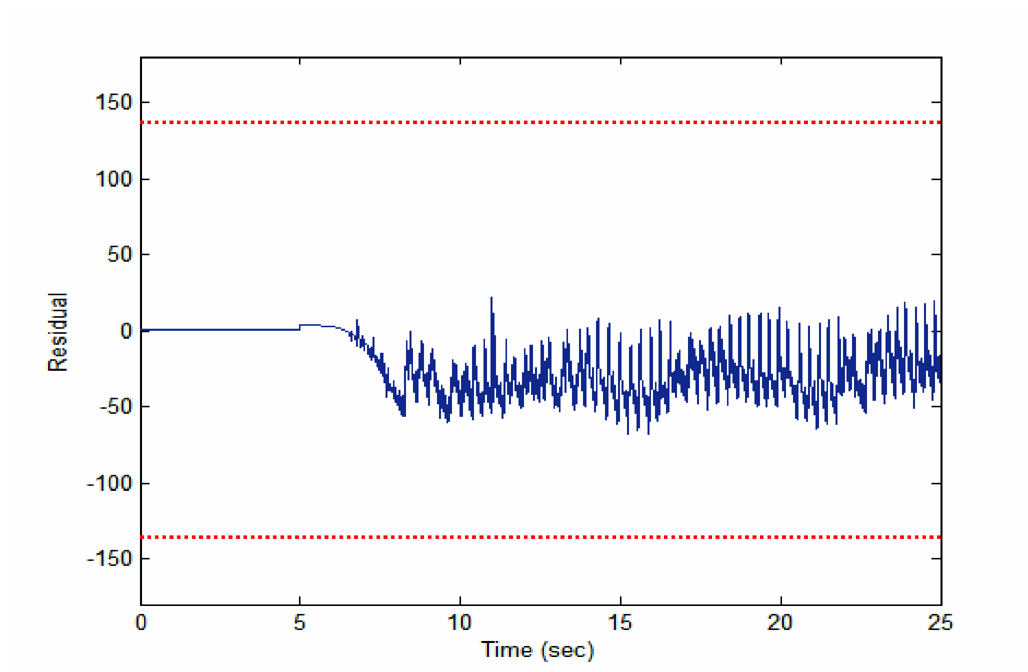


Figure 3-19. NLAR under sudden second encoder fault

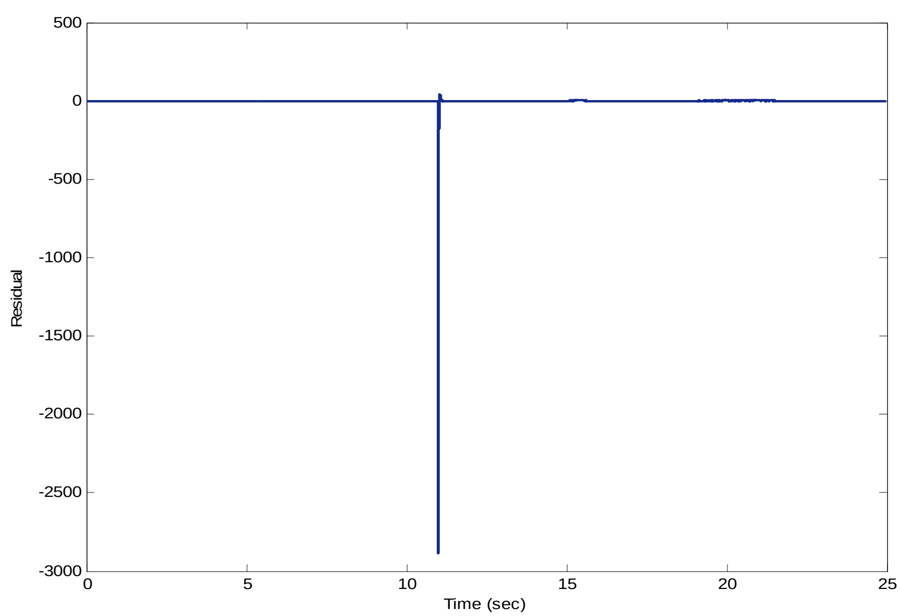


Figure 3-20. RNLAR under sudden second encoder fault

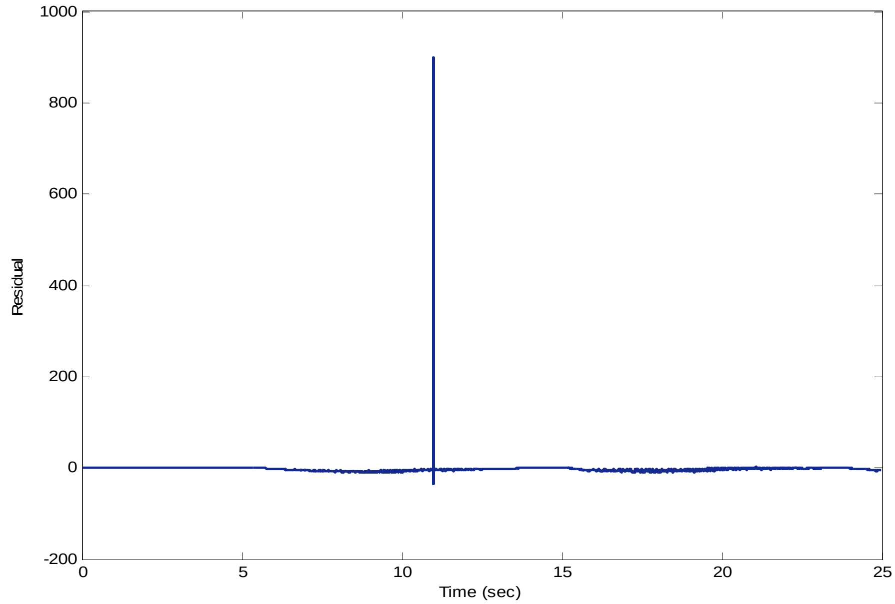


Figure 3-21. RNLAR under sudden third encoder fault

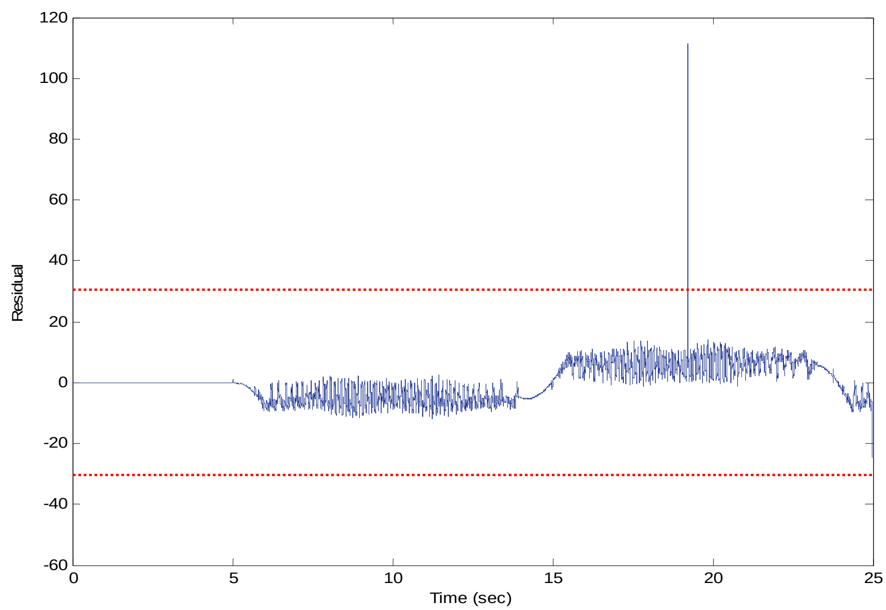


Figure 3-22. RNLAR under slow third encoder fault

6. Conclusion

A robust methodology for detecting sensor and actuator faults in multivariable input-affine nonlinear dynamic systems has been proposed in this paper. The presented robust nonlinear analytic redundancy (RNLAR) technique is an extension of the robustness idea used in linear domain into the nonlinear domain. It also extends the current state-of-the-art of nonlinear analytic redundancy (NLAR) techniques used for fault detection of nonlinear systems. We have shown that although residual output for actuator and sensor faults could not be made completely insensitive the MPM and disturbances, our presented RNLAR technique could minimize their effects to detect faults. We have presented a new theorem to this effect that provides a constructive technique to design such a PRV. We experimentally verified the presented methodology in relation to the sensor and actuator fault detection of a PUMA 560 manipulator.

References

[1] J. F. Engelberg, "Three million hours of robot field experience," The industrial robot, pp. 164-168, June 1974.

[2] S. Okina, K. Kawabata, T. Fuji, Y. kunii, H. Asama and I.Endo, " Self-diagnosis Systems of an Autonomous Mobile Robot Using Sensory Information" J. of Robotics and Mechatronics, 12, pp.72-77, 2000.

[3] H. Noura, D.Sauter, F. Hamelin, and T. Didier, "Fault tolerant control in dynamic systems: Application to a winding machine," IEEE Control Syst. Mag., vol. 20, pp. 33-49, 2000.

[4] R.J. Patton, J. Chen, and C.J. Lopez-Toribio, " Fuzzy observer for nonlinear dynamic systems fault diagnosis," in Proc 37th IEEE Conf. on Decision and control, 1998, vol. 1, pp. 84-89.

[5] E. N. Skoundrianos and S. G. Tzafestas, “Fault diagnosis on the wheels of a mobile robot using local model neural networks”, IEEE Robotics and Automation Magazine, pp. 83-90, 2004.

[6] N. Mehranbod, M. Soroush and C. Panjapornpon, “A method of sensor fault detection and identification ”, Journal of Process Control, Volume 15, Issue 3, April 2005, Pages 321-339 G.

[7] L. Zhang, L. B. Jack and A. K. Nandi, “Fault detection using genetic programming”, Mechanical Systems and Signal Processing, Volume 19, Issue 2, March 2005, Pages 271-289.

[8] R. Isermann, “Process fault detection based on modeling and estimation methods- a survey,” Automatica, 20, 387-404.

[9] J. J Gertler., “Analytic redundancy methods in fault detection and isolation – survey and synthesis”, Preprints of IFAC safeprocess conference, vol. 1, pp. 9-22. 1991.

[10] P.M. Frank, and X. Ding, “Survey of robust residual generation and evaluation method in observer-based fault detection system,” J. Proc. Cont. Vol. 7. No. 6. pp 403-424, 1997.

[11] J. J Gertler “Fault Detection and Diagnosis in Engineering systems”, 1 ed. Marcel Dekker , Inc.

[12] A. De. Luca and R. Mattone, “Actuator failure detection isolation using generalized momenta,” proceedings of the 2003 IEEE, ICRA, 634639, 2003.

[13] A. De Luca and R. Mattone, “An adapt-and-detect actuator FDI scheme for robot manipulators,” Proc. IEEE, ICRA, New Orleans, LA, April 2004.

[14] G. Antonelli, F. Caccavale, and L. Villani, “Adaptive discrete-time fault diagnosis for a class of nonlinear systems: application to a mechanical manipulator,” Proc. IEEE, ISIC, Houston, Texas, October 5-8, 2003.

[15] W. E. Dixon, I. D. Walker, D. M. Dawson, and J. P. Hartranft, “ Fault detection for robot manipulators with parametric uncertainty: A prediction error based approach,” Proc. IEE ICRA, San Francisco, CA, 2000.

[16] M. L. McLntyre, W. E. Dixon, D. M. Dawson, and I. D. Walker, “Fault detection and identification for robot manipulators,” Proc. IEEE ICRA, New Orleans, LA, 2004.

[17] J. Shin and J. Lee, “Fault detection and robust fault recovery control for robotic manipulators with actuator fault,” Proc. IEEE, ICRA, Detroit, Michigan, May 1999.

[18] G. Liu, "Control of robotic manipulators with consideration of actuator performance degradation and failure," Proc. IEEE ICRA, Seoul, Korea, May 21-26, 2001.

[19] L. Notash, "Kinematic solution for the effective implementation of parallel manipulators," PhD Dissertation, Dept of Mech Engg, Univ of Victoria, June 1995.

[20] G. Paviglianiti, F. Caccavale, M. Mattei, and F. Pierri, " Sensor fault detection and isolation for robot manipulator," Proc 13th Mediterranean conference on control and automation, Limassol, Cyprus, June 27-29, 2005.

[21] M. L. Leuschen, Ian D. Walker, Joseph R. Cavallaro, "Fault residual generation via nonlinear analytic redundancy",IEEE trans. On control systems technology, vol. 13. no. 3, , May, 2005.

[22] M. L. Leuschen, "Derivation and application of nonlinear analytic techniques with application on robotics", PhD, dissertation, Rice University. Texas, 2001.

[23] M. Goel, A. A. Maciejewski, and V. Balakrishnan, "An analysis of the post-fault behavior of robotic manipulator," Proc. IEEE ICRA, Albuquerque, New Mexico, April 1997.

[24] C. L. Lewis and A. A. Maciejewski, "An example of failure tolerant operation of a kinematically redundant manipulators," Proc of IEEE ICRA, pp. 1380-1387, 1994.

[25] A. A. Maciejewski, "Fault tolerant control properties of kinematically redundant manipulators," Proc of IEEE ICRA, pp. 638-642, 1990.

[26] G. Yang, I-Ming Chen, "A novel kinematic calibration algorithm for reconfigurable robotic systems," IEEE Transaction on Robotics and Automation, Vol. 4, pp. 3197-3202, 2000.

[27] B. Armstrong, O. Khatib, and J. Burdick," The explicit dynamic model and inertial parameters to the PUMA 560 arm," Robotics and Automation. Proceedings, vol. 3, pp- 510-518, 1986.

[28] E. Chow and A. Willsky, "Analytic redundancy and the design of robust failure detection systems", IEEE trans. on Automatic Control, Vol. AC-29, No. 7. July 1984.

[29] X., Lou, A. Willsky, and G. Verghese, "Optimally robust redundancy relations for failure detection in uncertain systems", Automatica, 22, 333-344, 1986.

[30] J. J Gertler and Kunwer, M, "Optimal residual decoupling for robust fault diagnosis" , International Journal of Control, 61, 395-421, 1995.

[31] F. Hamelin and D. Sauter, “Robust fault detection in uncertain dynamic systems”, Automatica, 36, 1747-1754, 2000.

[32] Z. Han, W. Li, and S. L. Shah, “Fault detection and isolation in the presence of process uncertainties”, Control engineering practice, 13, 587-599, 2005.

[33] A. Shumsky, “Robust analytical redundancy relations for fault diagnosis in nonlinear systems,” Asian journal of control, vol. 4, No. 2, pp. 159-170, June 2002.

[34] A. Isidori. Nonlinear control systems. Springer-Verlag, London, UK, 1995.

[35] H. K. Khalil. Nonlinear Systems. Second edition, 1996, pp.148.

[36] Li. Weihua and S. Sirish, “Structured residual vector-based approach to sensor fault detection and isolation”, Journal of process control 12, 429-443, 2002.

[37] S.I. Raumeliotis and G. S. Sukhatme, “Fault detection and identification in a mobile robot using multiple model estimation” , Robotics and automation, vol. 3, 2223-2228, 1998.

[38] J. J Gertler,“ Diagnosis parametric fault – from identification to parity relations”, Proc. American Control Conference, pp. 143-156.

[39] M. Staroswiecki, J. P. Cassar, and G. Comtet-Varga, “Analytic redundancy relations for state affine systems.” In proceedings of the fourth European control conferences, Brussels, Belgium, 1997.

[40] A.N. Zhirabok and O.V. Prebragenskaya, “Instrument fault detection in nonlinear dynamic systems.” In proceedings of the 1993 IEEE international conferences on systems, Man , and cybernetics, pp. 114-119, Touquet, France, oct 1993.

[41] T. Wei, Y.C. Hon, and Y. B. Wang, “Reconstruction of numerical derivatives from scattered noisy data,” Institute of Physics publishing, pp. 657-672, 2005.

[42] W. Press et al. “Numerical recipes in C,” Cambridge University press, 1997.

CHAPTER IV: MANUSCRIPT 3

IMPACT OF THE ORDER OF REDUNDANCY RELATION IN ROBUST FAULT DETECTION OF ROBOTIC SYSTEMS

Bibhrajit Halder[1] Nilanjan Sarkar[2]

(Submitted to Journal of Control Engineering Practice)

Abstract

This paper presents a new approach, called robust nonlinear analytic redundancy (RNLAR) technique to actuator fault detection for input-affine nonlinear multivariable dynamic systems that include most robotic systems. Robust fault detection is important because of the universal existence of model uncertainties and process disturbances in most systems. Analytic redundancy, which is a basis for residual generation to detect fault, is primarily used in the linear domain. In this paper, we characterize the order of redundancy relation for nonlinear systems in terms of robustness. We propose and prove that an increase in the order of redundancy relation increases the robustness in the sense

[1]Graduate Research Assistant, Department of Mechanical Engineering, Vanderbilt University, 2301 Vanderbilt Place, Nashville, TN-37235. Email: bibhrajit.halder@vanderbilt.edu Phone: 1-615-343-6472, Fax: 1-615-343-6687.

[2]Associate Professor, Department of Mechanical Engineering, Vanderbilt University, 2301 Vanderbilt Place, Nashville, TN-37235. Email: nilanjan.sarkar@vanderbilt.edu Phone: 1-615-343-7219, Fax: 1-615-343-6687.

of a performance index defined in this paper. We further develop an algorithm to select the redundancy relation order and design robust nonlinear fault detection residuals. Experimental results on a PUMA 560 robotic arm are presented to verify the claim.

Keywords: Nonlinear fault detection, order of redundancy, robustness, robotic systems

1. Introduction

During the last decade, as the applications of robots steadily expanded, there is significant research activity in the area of robot reliability and fault tolerance [1]. One way to address these needs is to design a fault tolerant control system (FTCS) for robotic systems. Generally, a FTCS consists of two major components: a fault detection and isolation (FDI) scheme, and a fault accommodation mechanism. In this paper we focus on the fault detection part of FTCS.

Considerable research effort has been invested in model-based fault detection methods since 1970s. Among them the parity relation-based schemes have been very successful. Some important survey papers in this area are given in [2]-[5]. The fundamental formulation of parity relation for linear systems is presented in [6], which was based on analytic redundancy (AR) of the system. More detail is given in [7]. Robustness is an important aspect in the fault detection method. To address the robustness issue, in [6] the authors have proposed an optimization method to select a parity vector from the parity space. This work was later extended by various researchers in [8][9]. Most recently in [10][11] the authors designed optimal primary residual, which considered both the model-plant-mismatch (MPM) and process disturbances for linear systems.

The journey from linear AR residual generation methods to nonlinear analytic redundancy (NLAR) residual generation methods started with the use of linearized model of the nonlinear system to derive the AR residuals [14][15]. The AR concept was later extended to nonlinear systems without linearization. In [16] the authors proposed a nonlinear analytic redundancy scheme based on parity relation method.

It was pointed out in [9] that the selection of the order of redundancy relation has an influence on the optimization performance. In fact, it is proved in [12] that increasing the order of redundancy relation leads to an increase in the dimension of the parity space, which in turn provides greater flexibility in residual generation as well as improves robustness. Note that the above-discussed conclusions regarding the increase in order of redundancy relation have been proven for linear systems. There are no equivalent results available in the literature for nonlinear systems. The objective of this paper is to extend the above results for nonlinear systems.

Recently we proposed a new approach, called robust nonlinear analytic redundancy (RNLAR) technique [13]. We extended the robustness idea, used in [10] for linear systems, into the nonlinear domain. In this paper we prove that an increase in the order of redundancy relation increases the robustness of the nonlinear residuals. This result is compatible with its linear counter part as given in [9] [12]. We further provide experimental verification of this claim using a Unimation PUMA 560 robotic arm as a test-bed. Finally, we summarize our contributions.

2. RNLAR Residual Generation

Consider the nonlinear system (1)

$$\dot{x} = f(x) + \sum_{i=1}^{q} g_i(x) u_i + d(x,u); \; y = Cx \tag{1}$$

where the state x is defined on an open subset U of $\Re^n$; $u = [u_1 \quad u_2 \quad \ldots \quad u_q]^T \in \Re^q$ is the process input; $y \in \Re^m$ is the process output; C is $m \times n$ output matrix; $d(x,u)$ represents an unmeasured deterministic process disturbance vector. The functions f, $g_1, \ldots, g_q$ are $\Re^n$ valued smooth mappings defined on the open set U, and $g = [g_1 \; g_2 \; \ldots g_q]$. In the presence of faults the system is represented by

$$\dot{x} = f^n(x) + g^n(x) u^g + e(x,u) + g^n(x) \, u^f, \quad y = Cx + o \tag{2}$$

where $e(x,u) = f^u(x) + g^u(x) \left(u^g + u^f\right) + d(x,u)$, $f^n(x), f^u(x), g^n(x)$, and $g^u(x)$ represent the nominal and uncertain part of the mappings f and g, respectively. $u^g \in \Re^q$ represents the fault-free input vector, $u^f \in \Re^q$ represents the actuator fault vector, and o represents a Gaussian-distributed white noise vector. It is assumed that u^g is available for computation but u^f and o are not. The magnitude of the noise is assumed to be significantly smaller than the magnitude of faults. We design robust residuals for the nonlinear systems given by (2). By robust we mean the residual will need to be sensitive to the faults but insensitive to the MPM and disturbances of the system, i.e., insensitive to $e(x,u)$ as much as possible. We proposed a new approach, called robust nonlinear analytic redundancy (RNLAR), to minimize the effect of error and accentuate the effect of faults. The details of RNLAR method are given in chapter II and III. Here we mention the steps that are important in this work. We take the derivative of the system output for s times, where s is the order of the redundancy relation as defined in [102]. s describes the 'memory span' of the redundancy relation. We define a performance, index, J_s, to quantify the robustness, as follows:

$$J_s(x,u) = \frac{w_s{}^T G_s G_s{}^T w_s}{w_s{}^T H_s H_s{}^T w_s} \tag{3}$$

where w_s, from the parity space W_s defined by $W_s = \{w_s : w_s \Gamma_s \equiv [0]\}$, The matrices Γ_s, G_s, and H_s are defined in chapter II and III. The subscript s represents the order of redundancy relation. We formulate the robust problem as follows: Find a w_s from the parity space such that $J \le K(x_e) \quad \forall x_e \in U_r$ for some predefined $0 < K(x_e) < 1 \quad \forall x_e \in U_r$. The choice of $K(x_e)$ determines the sensitivity of residual to the actuator fault and insensitivity to the error term. A smaller values of K implies more robustness. We propose a constructive theorem to find a w_s, under suitable condition. It was pointed out in chapter II that an increase in the order of redundancy relation, s, increases the robustness.

3. Robustness Theorem

We formulate the theorem that shows that increasing the order of redundancy relations improves the system robustness in the sense of performance index J_s, where the subscript s represents the order of the redundancy relation. This is the main contribution of this paper. We state the theorem as follows:

Theorem I:

Given the states $x \in \Re^n$ and inputs $u \in \Re^q$, consider an open set $U_r \in \Re^{n+q}$ such that the states and the inputs are restricted in, U_r i.e., $X = (x,u) \in U_r$. Let

$$\alpha_s = \min_{w_s \in W_s} \max_{X \in U_r} J_s(X) \ , \quad \alpha_{s+1} = \min_{w_{s+1} \in W_{s+1}} \max_{X \in U_r} J_{s+1}(X) \tag{4}$$

then $\alpha_s > \alpha_{s+1}$ for all $s>0$.

*Proof***:**

For a given s, let $X_{cs} \in U_r$ and $w_{cs} \in W_s$ be the optimal choice such that

$$\alpha_s = \frac{w_{cs}{}^T G_s G_s{}^T w_{cs}}{w_{cs}{}^T H_s H_s{}^T w_{cs}} \tag{5}$$

To prove the inequality $\alpha_s > \alpha_{s+1}$, it is sufficient to show that there exists a vector $w_{s+1} \in W_{s+1}$ such that

$$\frac{w_{s+1}{}^T G_{s+1} G_{s+1}{}^T w_{s+1}}{w_{s+1}{}^T H_{s+1} H_{s+1}{}^T w_{s+1}} < \alpha_s \tag{6}$$

for $X_{cs} \in U_r$. That implies

$$w_{s+1}{}^T \left(G_{s+1} G_{s+1}{}^T - \alpha_s H_{s+1} H_{s+1}^T \right) w_{s+1} < 0 \tag{7}$$

We can express G_{s+1} and H_{s+1} in terms of G_s and H_s, respectively. G_s is a $m(s+1) \times ns$ matrix while the dimension of G_{s+1} is $m(s+2) \times n(s+1)$. G_{s+1} has m more rows and n more columns than that of G_s, which carries the information of ($s+1$)$^{\text{th}}$ order differentiation of the output equation. Hence, we can express G_{s+1} as follows:

$$G_{s+1} = \begin{bmatrix} \underset{m(s+1)\times ns}{G_s} & \underset{m(s+1)\times n}{0} \\ \underset{m \times ns}{\Lambda_g} & \underset{m \times n}{\Delta_g} \end{bmatrix}_{m(s+2)\times n(s+1)} \tag{8}$$

where Λ_g and Δ_g contain the ($s+1$)$^{\text{th}}$ order differentiation of the output equation that is used with the error term. In a similar fashion, H_{s+1} can be expressed in terms of H_s as follows:

$$H_{s+1} = \begin{bmatrix} \underset{m(s+1)\times qs}{H_s} & \underset{m(s+1)\times q}{0} \\ \underset{m\times qs}{\Lambda_h} & \underset{m\times q}{\Delta_h} \end{bmatrix}_{m(s+2)\times q(s+1)} \tag{9}$$

where Λ_h and Δ_h contain the $(s+1)^{\text{th}}$ order differentiation. Substituting (8) and (9) into (7) gives

$$w_{s+1}{}^T \left(\begin{bmatrix} \underset{m(s+1)\times ns}{G_s} & \underset{m(s+1)\times n}{0} \\ \underset{m\times ns}{\Lambda_g} & \underset{m\times n}{\Delta_g} \end{bmatrix} \begin{bmatrix} \underset{m(s+1)\times ns}{G_s} & \underset{m(s+1)\times n}{0} \\ \underset{m\times ns}{\Lambda_g} & \underset{m\times n}{\Delta_g} \end{bmatrix}^T - \alpha_s \begin{bmatrix} \underset{m(s+1)\times qs}{H_s} & \underset{m(s+1)\times q}{0} \\ \underset{m\times qs}{\Lambda_h} & \underset{m\times q}{\Delta_h} \end{bmatrix} \begin{bmatrix} \underset{m(s+1)\times qs}{H_s} & \underset{m(s+1)\times q}{0} \\ \underset{m\times qs}{\Lambda_h} & \underset{m\times q}{\Delta_h} \end{bmatrix}^T \right) w_{s+1}$$

$$= w_{s+1}{}^T \left(\begin{bmatrix} \underset{m(s+1)\times m(s+1)}{G_s G_s^T} & \underset{m(s+1)\times m}{G_s \Lambda_g^T} \\ \underset{m\times m(s+1)}{\Lambda_g G_s^T} & \underset{m\times m}{\Lambda_g \Lambda_g^T + \Delta_g \Delta_g^T} \end{bmatrix} - \alpha_s \begin{bmatrix} \underset{m(s+1)\times m(s+1)}{H_s H_s^T} & \underset{m(s+1)\times m}{H_s \Lambda_h^T} \\ \underset{m\times m(s+1)}{\Lambda_h H_s^T} & \underset{m\times m}{\Lambda_h \Lambda_h^T + \Delta_h \Delta_h^T} \end{bmatrix} \right) w_{s+1}$$

$$= w_{s+1}{}^T \begin{pmatrix} G_s G_s^T - \alpha_s H_s H_s^T & G_s \Lambda_g^T - \alpha_s H_s \Lambda_h^T \\ \Lambda_g G_s^T - \alpha_s \Lambda_h H_s^T & \Lambda_g \Lambda_g^T + \Delta_g \Delta_g^T - \alpha_s \left(\Lambda_h \Lambda_h^T + \Delta_h \Delta_h^T \right) \end{pmatrix} w_{s+1} \tag{10}$$

We construct w_{s+1} as follows $w_{s+1} = \begin{bmatrix} w_{cs} \\ \gamma w_e \end{bmatrix}_{m(s+2)\times 1}$ where γ is a scalar constant and w_e is a $m \times 1$ vector. We will construct w_e such a way that (7) is satisfied and $w_{s+1}^T \Gamma_{s+1} = 0$. Substituting the above choice of w_{s+1} in (10) gives (11).

$$\begin{bmatrix} w_{cs}^T & \gamma w_e^T \end{bmatrix} \begin{pmatrix} G_s G_s^T - \alpha_s H_s H_s^T & G_s \Lambda_g^T - \alpha_s H_s \Lambda_h^T \\ \Lambda_g G_s^T - \alpha_s \Lambda_h H_s^T & \Lambda_g \Lambda_g^T + \Delta_g \Delta_g^T - \alpha_s \left(\Lambda_h \Lambda_h^T + \Delta_h \Delta_h^T \right) \end{pmatrix} \begin{bmatrix} w_{cs} \\ \gamma w_e \end{bmatrix} \quad (11)$$

Let us define $A = G_s \Lambda_g^T - \alpha_s H_s \Lambda_h^T$, $B = \Lambda_g G_s^T - \alpha_s \Lambda_h H_s^T$ and $C = \Lambda_g \Lambda_g^T + \Delta_g \Delta_g^T - \alpha_s \left(\Lambda_h \Lambda_h^T + \Delta_h \Delta_h^T \right)$. Substituting the above definition we get

$$\begin{bmatrix} w_{cs}^T & \gamma w_e^T \end{bmatrix} \begin{pmatrix} G_s G_s^T - \alpha_s H_s H_s^T & A \\ B & C \end{pmatrix} \begin{bmatrix} w_{cs} \\ \gamma w_e \end{bmatrix} \quad (12)$$

$$= w_{cs}^T \left(G_s G_s^T - \alpha_s H_s H_s^T \right) w_{cs} + \gamma w_e^T B w_{cs} + w_{cs}^T A \gamma w_e + \gamma w_e^T C \gamma w_e \quad (13)$$

$$= \gamma w_e^T B w_{cs} + w_{cs}^T A \gamma w_e + \gamma w_e^T C \gamma w_e \quad (14)$$

because $w_{cs}^T \left(G_s G_s^T - \alpha_s H_s H_s^T \right) w_{cs} = 0$ from (5). Now, for $m \geq 2$, we can always select w_e such a way that $\gamma w_e^T B w_{cs} + w_{cs}^T A \gamma w_e + \gamma w_e^T C \gamma w_e < 0$ and $w_{s+1}^T \Gamma_{s+1} = 0$ as two constraints can be satisfied with two free variables. With the above selection of w_e, finally we have

$$\gamma w_e^T B w_{cs} + w_{cs}^T A \gamma w_e + \gamma w_e^T C \gamma w_e < 0 \quad \Rightarrow w_{s+1}{}^T \left(G_{s+1} G_{s+1}{}^T - \alpha_s H_{s+1} H_{s+1}^T \right) w_{s+1} < 0$$

$$\Rightarrow \frac{w_{s+1}{}^T G_{s+1} G_{s+1}{}^T w_{s+1}}{w_{s+1}{}^T H_{s+1} H_{s+1}{}^T w_{s+1}} < \alpha_s \quad (15)$$

For $m = 1$, w_e is a scalar quantity. Hence we can write (14) as

$$\gamma w_e B w_{cs} + \gamma w_e w_{cs}^T A + \gamma^2 w_e w_e C \quad (16)$$

We can choose the constant γ such a way that $\gamma w_e B w_{cs} + \gamma w_e w_{cs}^T A + \gamma^2 w_e w_e C < 0$ when both A and B are non zero, which is the case here. Then we can select w_e such a way that $w_{s+1}^T \Gamma_{s+1} = 0$. With the above choices we get (15) for $m = 1$ as well. This concludes the proof of Theorem I.

Based on Theorem I, we give a step-by-step procedure for optimal search of parity vector in the parity space for actuator fault detection.

Step 1: Set the order of redundancy relation *s* and choose a desired value for *K* as defined before.

Step 2: Find the nature of eigenvalues of *R* based on the choice of *K* and *s*.

Step 3: If there are more then two distinct non-positive eigenvalues, then calculate w_s.

Step 4: If the above condition does not satisfy, then increase the value of *s* and go to Step 2.

4. Experimental Results

A Unimation PUMA 560 is used to experimentally verify the claim of *Theorem I*. We use the first three joints of the manipulator for our experiments. We have replaced the microcontroller board of the PUMA to develop an open architecture system. This allows us to implement the controllers that are essential for this experiment. In addition, we have interfaced the robot with Matlab and Real-time Workshop to allow fast and easy system development. The joint angles of the robot are measured using encoders. The encoder readings are acquired with a sample time of 0.001seconds from a Measurement Computing PCI-QUAD04 card. The torque output to the robot is given with a Measurement Computing PCIM-DDA06/16 card with the same sample time. The encoder outputs are used for calculating the residuals in the experiment. Armstrong et al. [20] experimentally determined the relevant parameters of the PUMA 560 and derived its dynamic model. The dynamic equation of the PUMA 560 is given in Chapter III. The output of the system are the joint angles and is represented as

$$y = [\theta_1 \quad \theta_2 \quad \theta_3]^T \tag{17}$$

We run the model in parallel to the Puma 560 and use the model to calculate the terms, G_s, H_s, and Γ_s while y_s and u_s comes from Puma 560.

4.1 Results

We design experiments to detect actuator faults using different residuals: one RNLAR residual with $s = 2$; and one RNLAR residual with $s = 3$. In these experiments, the PUMA was asked to track a straight-line trajectory in the *x* direction with *y* and *z* coordinates were kept constant at -*.029m* and -*.034m*, respectively. Thus Joint 1 did not need to move in these experiments. The trajectory for *x* direction starts after 5 seconds. While it tracked the trajectory we introduced actuator faults and monitored the residuals. The endpoint of PUMA was controlled by a PID controller with the following PID gains: *p*=400, *I*=5 and *D*=15. We should mention that the residuals are independent of the choice of controller. Since Joint 1 did not need to move in our experiments, we only present the residuals for the Joints 2 and 3. For each actuator 2 and 3, we present four results: one RNLAR residual output without any faults and three different residual outputs as described before in the presence of fault.

Various types of actuator fault are discussed in [21]. We chose two common actuator faults for the experiments. First one is a partial actuator fault where one actuator generates only a part of the desired torque. This type of fault represents degradation in the actuator system (e.g., friction due to jamming, problems in transmission etc.). The second actuator fault that we consider is a constant torque output. This may occur due to constant polarization of the actuator, called actuator bias. We apply the partial actuator fault in the second joint and bias actuator fault in the third joint but not at the same time.

In the experimental set-up the partial actuator faults were introduced in the second joint where the joint torque was reduced by 80% after 11 seconds of operation. Faults are considered detected if the magnitudes of the residuals cross some pre-determined threshold value. We use a standard threshold design as outlined in [16] where the threshold value is considered twice the absolute maximum value achieved in a fault-free run with the same parameters First, we present the output of RNLAR residual with $s = 2$ and $s = 3$ without any fault in the system to demonstrate the effect of the MPM and process disturbance on the residuals. The residual output is shown is Figure 4-1 and Figure 4-2. We can observe that the absolute maximum value of the residual under no-fault condition with $s = 2$ is 3.15 and with s=3 is 11.63. We set the threshold value as 6.30 for residual with $s = 2$ and 23.26 with $s = 3$. RNLAR residuals with $s = 2$, and $s = 3$ under partial second actuator fault are shown is Figure 4-3 and 4-4, respectively. In Figure 4-3, the threshold value is shown in red dotted line. The peak value of the residual output for $s = 2$ is 33.67, which is more than the threshold value. Hence the fault is considered detected. Finally, the peak value for RNLAR residual with $s = 3$ is 3612, which is 100 times more than the threshold value. We conclude from the above results that among the two residuals, the residual with $s = 3$ is more sensitive to fault in the presence of identical MPM and disturbance and hence more robust.. As can be seen, our experimental results support the claim of Theorem I that increasing the redundancy order increases the robustness in fault detection.

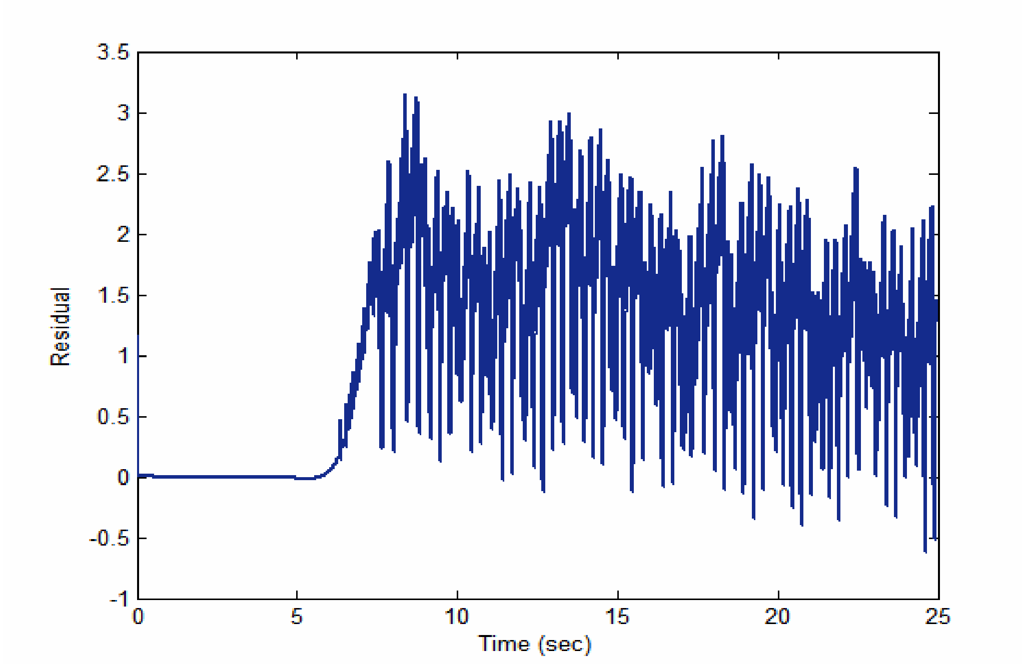


Figure 4-1. First RNLAR residual for $s = 2$ without any fault

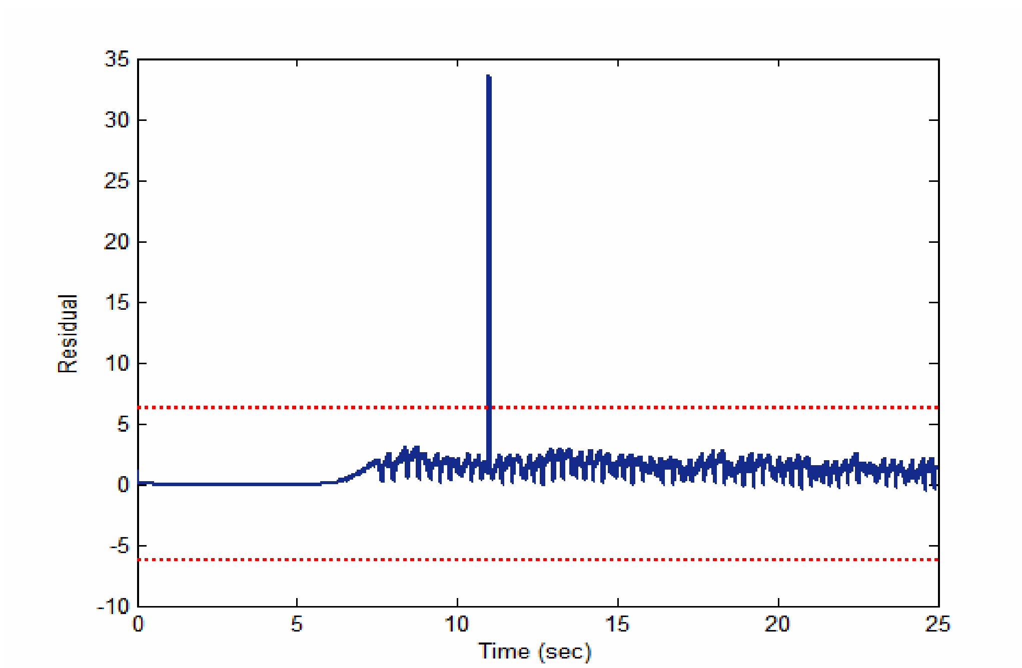


Figure 4-2. First RNLAR residual for $s = 2$ under partial second actuator fault

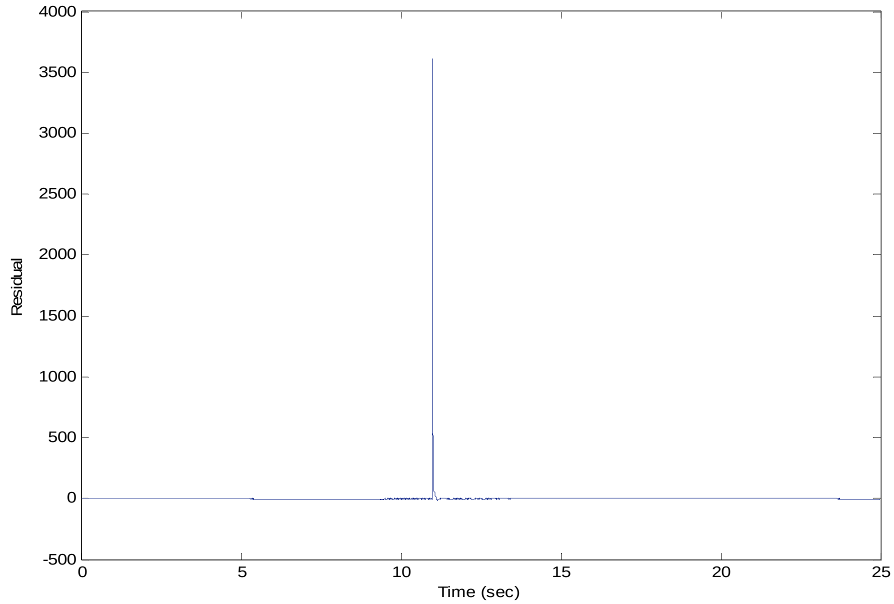


Figure 4-3. First RNLAR residual for $s = 3$ under partial second actuator fault

Next, we design another RNLAR residual to detect the fault in third actuator. We introduce the bias fault in the third joint actuator. A constant $\tau_3 = 0.1$ is introduced to the third actuator at *t=11s*. The output of RNLAR residual with $s = 2$ without any fault is shown in Figure 4-4. The absolute maximum value of the residual in no-fault condition is 4.68. Thus we set the threshold value to be 9.36. RNLAR residuals with $s = 2$, and $s = 3$ under partial second actuator are shown is Figure 4-5 and 4-6 respectively. The peak value of the residual output for $s = 2$ is 42.19, which is more than the threshold value. Hence the fault is detected. Finally, the peak value for RNLAR residual with $s = 3$ is 274.94. Thus once again we notice that increasing the redundancy order increases the robustness in fault detection.

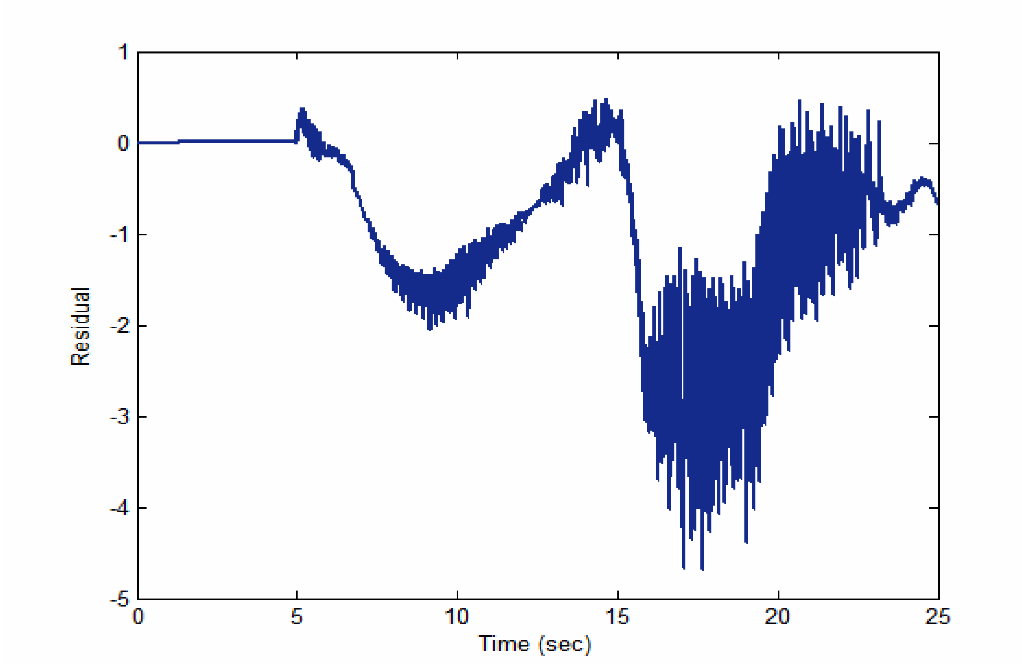


Figure 4-4. Second RNLAR residual for $s = 2$ without any fault

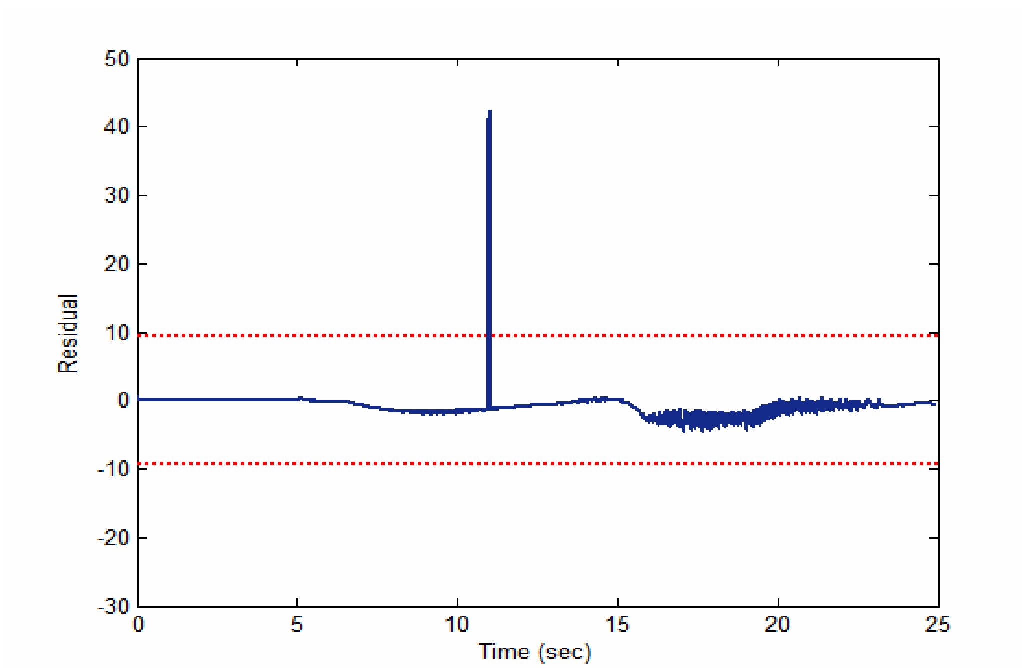


Figure 4-5. Second RNLAR residual for $s = 2$ under bias third actuator fault

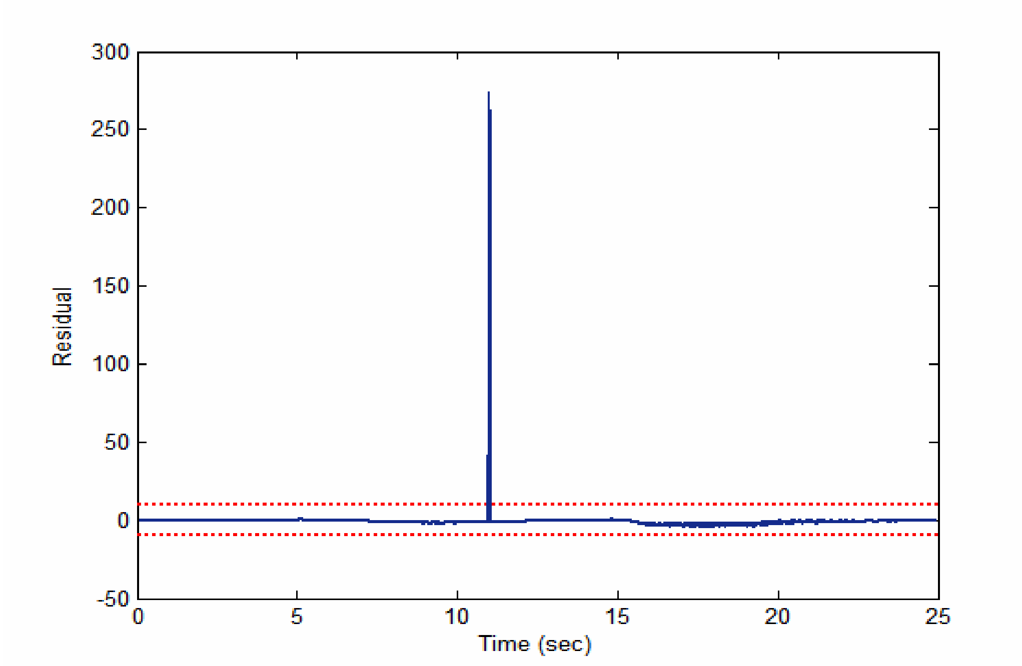


Figure 4-6. Second RNLAR residual for $s = 3$ under bias third actuator fault

5. Conclusion

In this paper, we have studied the relation between order of redundancy relation and robustness of the system. We have presented the RNLAR residuals generation procedure for multivariable input-affine nonlinear dynamic systems. The main contribution of this paper has been to formulate and prove the theorem that increasing the order of redundancy relation improves the system robustness. The proposed theorem is an extension of the similar results obtained in linear systems. Based on the theorem, an algorithm has been proposed to determine the optimal redundancy relation order. We have experimentally verified the claim on a PUMA 560 robotic arm. A comparative experimental study has been presented to demonstrate the effect of robust residuals.

References

[1] M.L. Visinsky, J. R. Cavallaro, and I. D. Walker, “ Robotic fault detection and fault tolerance: A survey,” *Rel. Eng. Syst. Safety*, vol. 46, pp. 139-158, 1994.

[2] R. Isermann, “Process fault detection based on modeling and estimation methods- a survey,” Automatica, 20, 387-404.

[3] J. J Gertler, “Analytic redundancy methods in fault detection and isolation – survey and synthesis”, Preprints of IFAC safeprocess conference, vol. 1, pp. 9-22. 1991.

[4] P.M Frank, and X. Ding, “Survey of robust residual generation and evaluation method in observer-based fault detection system,” J. Proc. Cont. Vol. 7. No. 6. pp 403-424, 1997.

[5] P.M Frank, and X. Ding, “Frequency domain approach to optimally robust residual generation and evaluation for model-based fault diagnosis”, Automatica, 30, 789-804, 1994.

[6] E. Chow, and A. S. Willsky, “Analytic redundancy and the design of robust failure detection systems”, IEEE trans. on Automatic Control, Vol. AC-29, No. 7. July 1884.

[7] J. J Gertler, “Fault Detection and Diagnosis in Engineering systems”, 1 ed. Marcel Dekker, Inc.

[8] J. J Gertler, “Diagnosis parametric fault – from identification to parity relations”, Proc. American Control Conference, pp. 143-156.

[9] X. C. Lou, A. S. Willsky, and G. L. Verghese, “Optimally robust redundancy relation for failure detection in uncertain systems,” *Automatica*, vol. 22, pp. 333-344, 1986.

[10] Z. Han, W. Li, and S. L. Shah, “Fault detection and isolation in the presence of process uncertainties”, Control engineering practice, 13, 587-599, 2004.

[11] C. Kwan and R. Xu, “A note on simultaneous isolation of sensor and actuator faults,” IEEE transaction of control systems technology, vol. 1, no. 1, Jan 2004.

[12] X. Ding, L. Guo, and T. Jeinsch, “A characterization of parity space and its application to robust fault detection,” IEEE transaction on automatic control, VOL. 44, No. 2, February 1999.

[13] B. Halder and N. Sarkar, “Robust fault detection based on nonlinear analytic redundancy technique with application to robotics,” Proceedings of IMECE’05, nov 5-11, 2005, Orlando, Fl.

[14] M. Staroswiecki, J. P. Cassar, and G. Comtet-Varga, “Analytic redundancy relations for state affine systems.” In proceedings of the fourth European control conferences, Brussels, Belgium, 1997.

[15] A.N. Zhirabok and O.V. Prebragenskaya, “Instrument fault detection in nonlinear dynamic systems.” In proceedings of the 1993 IEEE international conferences on systems, Man , and cybernetics, pp. 114-119, Touquet, France, oct 1993.

[16] M. L. Leuschen, I. D. Walker, and J. R. Cavallaro, “Fault residual generation via nonlinear analytic redundancy.” IEEE Trans. On control systems technology, vol. 13, No. 3, May 2005, pp. 452-458

[17] P. M. Frank and J. Wunnenberg,, “Robust fault detection using unknown input observer schemes”, in fault diagnosis in dynamic systems: theory and application, R. Patton, P. M. Frank and R. Clark, Eds: Prentice hall, 1989.

[18] H. K. Khalil. Nonlinear Systems. Second edition, 1996, pp.148.

[19] J. J. E. Slotine, W. Li, Applied nonlinear control, 1991.

[20] B. Armstrong, O. Khatib, and J. Burdick,” The explicit dynamic model and inertial parameters to the PUMA 560 arm,” Robotics and Automation. Proceedings, vol. 3, pp- 510-518, 1986.

[21] A. D. Luca and R. Mattone, “Actuator failure detection isolation using generalized momenta,” proceedings of the 2003 IEEE, ICRA, 634639, 2003.

CHAPTER V: MANUSCRIPT 4

ROBUST FAULT DETECTION AND ISOLATION IN MOBILE ROBOT

Bibhrajit Halder[1] Nilanjan Sarkar[2]

(Accepted by the International Journal of Automation and Computing)

Abstract

Robust nonlinear analytical redundancy (RNLAR) technique is used to detect and isolate actuator and sensor faults in a mobile robot. Both model-plant-mismatch (MPM) and process disturbance are considered during fault detection. The RNLAR is used to design primary residual vectors (PRV), which are highly sensitive to the faults and less sensitive to MPM and process disturbance, for sensor and actuator fault detection. The PRVs are transformed into set of structured residual vectors (SRV) for fault isolation. Experimental results on a Pioneer 3-DX are presented to justify the effectiveness of the RNLAR scheme.

---

[1]Graduate Research Assistant, Department of Mechanical Engineering, Vanderbilt University, 2301 Vanderbilt Place, Nashville, TN-37235. Email: bibhrajit.halder@vanderbilt.edu Phone: 1-615-343-6472, Fax: 1-615-343-6687.

[2]Associate Professor, Department of Mechanical Engineering, Vanderbilt University, 2301 Vanderbilt Place, Nashville, TN-37235. Email: nilanjan.sarkar@vanderbilt.edu Phone: 1-615-343-7219, Fax: 1-615-343-6687.

Keywords: Fault detection, fault isolation, nonlinear systems, robustness, uncertainty

1. Introduction

The demand for automation in modern society is increasing steadily during the last few decades. Mobile robots play an important role in automation industries that include planetary exploration, search and rescue, mine mapping, demining and nuclear waste cleanup to name a few. With this widespread applicability of mobile robots, a major concern is the reliability of the system. Fault detection and identification (FDI) are important problems in the development of reliable, robust mobile robots.

A substantial research effort has been invested in model-based FDI during the last few decades. Some important survey papers in this area are given in [1-3]. The fundamental concept of model-based fault detection is analytical redundancy (AR). The basic idea of AR is the comparison of the actual behavior of the monitored plant with the behavior of a mathematical plant. Implementation methods of AR can be classified into two groups: 1) indirect implementation, based on diagnostic observers, and 2) direct implementation based on parity relation technique [4].

The original idea of observer-based fault detection came from [5]. A survey paper [6] and the book [7] give the details about this method. Most of the methods were proposed for linear systems. Also, early FDI methods assumed the existence of an accurate model of the monitored system. However, model-plant-mismatch (MPM) and process disturbances almost always exist in practical systems. A model dependent fault detection scheme may not be useful under considerable MPM and process disturbances. Recently, various methods are used to design the observer for nonlinear system to accommodate the MPM and process disturbances [8-9].

Conceptually, the direct implementation based on parity relation is more straightforward than the observer based approach. Most research results on parity based fault detection techniques are for linear systems [9-14]. In [15] the authors proposed a nonlinear analytic redundancy scheme based on parity relation method. A robust fault detection method for nonlinear systems using a mathematical technique, called algebra of functions, was presented in [16]. It is assumed in this work that modeling uncertainty can be specified in the form of unknown constant or slowly varying system parameters.

Recently, a new robust nonlinear analytic redundancy (RNLAR) technique for fault detection was developed [17], which accommodates both the MPM and process disturbances for nonlinear multivariable dynamic systems. In this paper, RNLAR method is further developed to generate robust PRV, which is fault-accentuated signal, for fault detection in a mobile robot. In addition, we present results on fault isolation by generating a set of robust SRVs from these PRVs. We also verify the theoretical results by conducting experiments on a Pioneer 3-DX mobile robot.

2. Mobile Robot Model

The three-wheeled robot, Pioneer 3-DX (Figure 5-1), is used for experiments. The front two wheels are actuated independently by high-speed, high-torque, reversible-DC motors, which enable differential steering. The rear wheel is a passive caster. The Pioneer 3-DX has both holonomic and nonholonomic constraints. The kinematics of the Pioneer 3-DX is characterized by three constraints on the coordinates. The first one is the knifeedge constraint, i.e., the mobile robot cannot move in a lateral direction. We can represent this constraint as follows:

$$\dot{y}_c \cos\phi - \dot{x}_c \sin\phi - \dot{\phi} \cdot d = 0 \tag{1}$$

Figure 5-1. Pioneer 3-DX mobile robot

The other two constraints are that the two driving wheels satisfy pure rolling and do not slip, which implies:

$$\dot{x}_c \cos\phi + \dot{y}_c \sin\phi + b\dot{\phi} = r\dot{\theta}_r \tag{2}$$

$$\dot{x}_c \cos\phi + \dot{y}_c \sin\phi - b\dot{\phi} = r\dot{\theta}_l \tag{3}$$

where, (x_c, y_c) is the center of mass of mobile robot, ϕ is the heading angle measured from the *x*-axis, $d = 46mm$ is the distance from the center of mass of the mobile robot to the intersection of the axis of symmetry with the driving wheel axis, θ_r and θ_l are angular positions of the two driving wheels, $r = 97.5mm$ is the radius of the wheel, $b = 190mm$ is the distance between the driving wheel and the axis of symmetry. Each DC motor is equipped with a high-resolution optical quadrature shaft encoder for precise position and speed sensing. The total linear speed, v, and the angular velocity, $\omega = \dot{\phi}$, are two kinematic inputs to the Pioneer 3-DX.

The dynamic model of Pioneer 3-DX mobile robot is formulated using Lagrangian formulation. From the Lagrangian method we get the following

$$\frac{d}{dt}\left(\frac{\partial K}{\partial \dot{\theta}_s} - \frac{\partial K}{\partial \theta_s}\right) + \frac{\partial P}{\partial \theta_s} = Q_s \tag{4}$$

where K is the kinematic energy of the system, θ_s is the *s-th* generalized coordinate of the system, P is the potential energy and Q_s is the corresponding generalized force. Neglecting the wheel dynamics, which is small, compared to the dynamics of the body of the robot, we write the kinematic energy of the Pioneer mobile robot.

$$K = M_c cd(\dot{\theta}_r - \dot{\theta}_l)(\dot{x}_o \cos\phi - \dot{y}_o \sin\phi) + \frac{1}{2} J_c c^2 (\dot{\theta}_r + \dot{\theta}_l)^2 \tag{5}$$

The detail of each term of (5) is given in Appendix I. Substituting the kinetic energy to (4), we obtain

$$M(q)\ddot{q} + V(q, \dot{q}) = E(q)u^g - A^T(q)\lambda \tag{6}$$

where $q_x = [x_c \quad y_c \quad \theta_r \quad \theta_l]^T$, $u^g = [\tau_r^g \quad \tau_l^g]^T$ are the given torques applied to the two wheels, λ is the Lagrangian multiplier $M(q_x) \in R^{4\times4}$ is the symmetric, positive definite inertia matrix and $V(q_x, \dot{q}_x)$ is the vector of centrifugal and Coriolis forces.

Equation (6) can be represented into the state space as follows:

$$\dot{x} = f^n(x) + g^n(x)u \tag{7}$$

where $x = \begin{bmatrix} x_c & y_c & \theta_r & \theta_l & \dot{\theta}_r & \dot{\theta}_l \end{bmatrix}^T$ is the state vector. The details of each term of (7) are given in Appendix II. Equation (6) represents the nominal model of the Pioneer 3-DX mobile robot. We use the nominal model as in (7) to calculate the fault detection and isolation residuals.

It is worth to mention that (7) represent the nominal model of mobile robot. In the presence of faults, the actual actuator input and the observed sensor output can be represented in general form by

$$u = u^g + u^f; \quad y^o = Cx + y^f + o \tag{8}$$

where $u^g \in \Re^2$ represents the fault-free input vector, $u^f \in \Re^2$ represents the actuator fault vector, $y^o \in \Re^3$ represents the observed output vector, $y^f \in \Re^3$ represents the sensor fault vector and o represents a Gaussian-distributed white noise vector and C is the output matrix. It is assumed that u^g and y^o are available for computation but u^f, y^f, and o are not. Magnitude of the noise is assumed to be significantly smaller then the magnitude of faults. Under the nominal fault-free condition, u^f and y^f are zero vectors. However, when either a sensor and/or an actuator fault occur in the system, u^f and y^f will become non-zero. Model-plant-mismatch is represented by

$$f(x) = f^n(x) + f^u(x), \; g(x) = g^n(x) + g^u(x) \tag{9}$$

where $f^n(x), f^u(x), g^n(x)$, and $g^u(x)$ represent the nominal and uncertain part of the mappings f and g, respectively. Combining (7), (8) and (9), the overall system with faults is represented by

$$\dot{x} = g^n(x)u^g + e(x,u) + g^n(x)\,u^f \tag{10}$$

where $e(x,u) = f^{u}(x) + g^{u}(x)\left(u^{g} + u^{f}\right) + d(x,u)$. The vector $e(x,u)$ is called an error vector, which contains both the uncertainty of the model and the disturbances.

3. Robust Fault Detection

We use robust nonlinear analytic redundancy (RNLAR) method to design the primary residual vectors (PRV) to detect sensor and actuator faults in mobile robot. The detail of this method is given in Chapter II and III. Here we mention the important steps and equations that would be used for fault isolation as well.

We take the derivative of y^{o} for s times and stack them together, where s is the order of the redundancy relation.

$$\begin{bmatrix} y^{o} \\ \dot{y}^{o} \\ \ddot{y}^{o} \\ \cdot \\ \cdot \\ y^{so} \end{bmatrix} = \begin{bmatrix} Cx + y^{f} + o \\ C\dot{x} + \dot{y}^{f} + \dot{o} \\ C\ddot{x} + \ddot{y}^{f} + \ddot{o}. \\ \cdot \\ \cdot \\ Cx^{s} + y^{sf} + o^{s} \end{bmatrix} \tag{11}$$

The right-hand side of (11) is grouped into three major components: collection of the error terms, collection of the input terms, and collection of the states. This leads to the following compact form:

$$y_{s} = H_{s}u_{s} + \Gamma_{s} + y_{s}^{f} + G_{s}e_{s} + H_{s}u_{s}^{f} + o_{s} \tag{12}$$

where G_{s}, and H_{s} are the coefficient of error terms and input terms respectively. For the detail expression of H_{s}, Γ_{x}, and G_{s}, please refer to Chapter II and III. We defined O_{NDD} for RNLAR as follows:

$$\mathrm{O}_{NDD} = y_{s} - H_{s}u_{s} \tag{13}$$

First we only consider the sensor fault, this simplifies (12) into

$$\mathrm{O}_{NDD} = \Gamma_x + y_s^f + G_s e_s + o_s \tag{14}$$

We select a transformation matrix, W_r, which is located in the left null space of Γ_x, i.e., $W_r \Gamma_x \equiv [0]$. Pre-multiplying both sides of (14) with W_r results:

$$R_r = W_r \mathrm{O}_{NDD} = W_r y_s^f + W_r G_s e_s + W_r o_s \tag{15}$$

R_r is defined as the PRV for sensor fault detection.

Under the actuator fault (12) can be simplifies as

$$\mathrm{O}_{NDD} = \Gamma_x + G_s e_s + H_s u_s^f + o_s \tag{16}$$

We select a transformation matrix, W_a, which is located in the left null space of Γ_x, i.e., $W_a \Gamma_x \equiv [0]$. Pre-multiplying both sides of (16) with W_a results:

$$R_a = W_a (y_s - H_s u_s) = W_a \left(G_s e_s + H_s u_s^f + o_s \right) \tag{17}$$

R_a is defined as the PRV for actuator fault detection.

We can observe from (15) and (17) that both the sensor and actuator residuals are sensitive to the faults and the uncertainty of the system. It is desirable that R_r and R_a should be highly sensitive to the sensor and actuator faults respectively and mostly insensitive to the error terms in order to be able to detect actuator fault in the presence of modeling uncertainty. The above desired property can be translated mathematically into the following statement: $\|W_a G_s\|$ is less than $\|W_a H_s\|$ for actuator residuals and $\|W_r G_s\|$ is less than $\|W_r\|$ for sensor residuals. Both G_s and H_s are system dependent matrices. However, W_a can be chosen independently from the feasible options to satisfy the above requirement. Hence the problem becomes, select W_a in such a way that $\|W_a G_s\|$ is less than $\|W_a H_s\|$ for actuator residuals and $\|W_r G_s\|$ is less than $\|W_r\|$ for sensor residuals. We proposed a constructive theorem for designing the residuals in Chapter II and III. We define two performance functions,

$J = \dfrac{W^T G G^T W}{W^T H H^T W}$ for actuator fault residual, R_a and

$J = \dfrac{W^T G G^T W}{W^T I I^T W}$ for sensor fault residual, R_r, where *I* is the identity matrix.

Based on the theorem we can state the following algorithm to find W:

Step 1. Choose a small value for *K*, such that $J \le K$.

Step 2. Using the theorem check there exists any *W* that satisfy the conditions for the choice of *K*.

Step 3. If there exists any *W* that satisfy $J \le K$, then calculate *W* using the method given in the theorem.

Step 4. If the above condition is not satisfied, then increase the value of *K* and go to Step 2.

We design five PRVs for sensor faults, R_r^i, and five PRVs for actuator faults, R_a^i, for Pioneer 3-DX using the nominal model as given in (7). We stack the RPVs into a vector as follows:

$$R_r^{PRV} = \begin{bmatrix} R_r^1 & R_r^2 & R_r^3 & R_r^4 & R_r^5 \end{bmatrix}^T \quad (18)$$

$$R_a^{PRV} = \begin{bmatrix} R_a^1 & R_a^2 & R_a^3 & R_a^4 & R_a^5 \end{bmatrix}^T \quad (19)$$

We use the PRVs in the next step for fault isolation.

4. Robust Fault Isolation

Once faults were detected, we need to isolate the faults. This is achieved by transforming the set of PRV into a set of structured residual vector (SRV). The SRVs are designed such that each SRV is insensitive to a subset of faults but most sensitive to the other faults. We discuss the sensor fault isolation first.

We select an incidence matrix to characterize the SRVs. It is pointed out in [13] that the selection of incidence matrix is dependent on the number of faults to be isolated, the system order n, and the number of outputs m, and is not unique. For faults, we assume that only one is present in the system at a time. The occurrence of faults, in general, is not very frequent, and also we assume that any fault gets repaired before another one appears. The Pioneer mobile robot is represented using six states as in (7) and has two inputs. For Pioneer sensor fault detection we chose an incidence matrix as given in Table 5-1. A "0"at an intersection in the incidence matrix indicates that one SRV is insensitive to a specific sensor fault, while "1" indicates that the SRV is most sensitive. The number of rows corresponding to the SRVs is selected to be three, because there are three sensors in the Pioneer mobile robot. It is pointed out in [14] that for the isolation of a single faulty sensor the number of SRVs is usually selected to be equal to be the number of total sensors. If the former is less than the latter, each faulty sensor may not be isolated. With such an incidence matrix, a faulty sensor can be isolated by observing how the three SRVs respond to the fault. For instance, if SRV1 is unaffected but SRV2 and SRV3 are affected by the fault, then it can be inferred that the first sensor is faulty.

Table 5-1: Incidence matrix for sensor fault isolation

	Sensor 1	Sensor 2	Sensor 3
SRV 1	0	1	1
SRV2	1	0	1
SRV3	1	1	0

The SRVs are calculated by pre-multiplying a transformation matrix, S_s with (18)

$$r_s^i = S_s^i R_r^{PRV} = S_s^i W_r^{PRV} \mathrm{O}_{NDD} = S_s^i W_r^{PRV} y_s^f + S_s^i W_r^{PRV} G_s e_s + S_s^i W_r^{PRV} o_s \qquad (20)$$

where W_r^{PRV} is the stacked vector and the indices i indicates the i^{th} SRV. Designing a set of SRVs is equivalent to selecting a transformation matrix, S_s such that i^{th} SRV is unaffected by the i^{th} sensor fault, highly sensitive to the rest of the sensor faults and mostly insensitive to the error terms. For a general system define,

$W_{r,i} = [W_r(:,i) \;\; W_r(:,i+m) \;\; W_r(:,i+ms)] \;\; \forall i = [1,m]$ where $W_r(:,j)$ for $1 \le j \le ms$ is the j^{th} column of W_r. Also, $W_{r,i}^{\perp}$ represent the W_r matrix without $W_{r,i}$ columns. Mathematically, we can write the SRV design conditions as follows:

1) $S_s^i W_{r,i}^{PRV} = 0$, i.e., i^{th} SRV is unaffected by the i^{th} sensor fault

2) $\left\| S_s^i W_r^{PRV} G_s \right\|$ is less than $\left\| S_s^i W_{r,i}^{\perp} \right\|$, i.e., transformation matrix, S_s is more sensitive to the rest of the sensor faults and mostly insensitive to the error terms.

The condition (2) is the robustness problem in SRV design. We define a performance index, J_s given as below, to characterize this robust problem.

$$J_s = \frac{\left\| S_s^i W_r^{PRV} G_s \right\|}{\left\| S_s^i W_{r,i}^{\perp} \right\|}$$

We need to find a suitable S_s^i that minimizes J_s and satisfy the condition (1). The algorithm of calculating W_r can be directly applied here to the calculation of a suitable S_s^i with W, G, and I replaced with S_s^i, $W_r^{PRV} G_s$, and $W_{r,i}^{\perp}$ respectively.

A similar idea is used for actuator fault isolation. There are two inputs to the Pioneer mobile robot. Hence, two SRVs will be sufficient to isolate two actuator faults. The incidence matrix we chose for actuator fault isolation is given in Table 5-2. The SRVs for actuator fault isolation are calculated by pre-multiplying a transformation matrix, S_a of (19)

$$r_a^i = S_a^i R_a^{PRV} = S_a^i W_a^{PRV} \mathrm{O}_{NDD} = S_a^i W_a^{PRV} H_s u_s^f + S_a^i W_a^{PRV} G_s e_s + S_a^i W_a^{PRV} o_s \quad (21)$$

where W_a^{PRV} is the stacked vector and the indices *i* indicates the i^{th} SRV.

Table 5-2: Incidence matrix for actuator fault isolation

	Actuator 1	Actuator 2
SRV1	0	1
SRV2	1	0

We Define, $W_a^{PRV} H_s = Q$, $Q_i^{PRV} = [Q(:,i) \;\; Q(:,i+m) \;\; Q(:,i+ms)] \;\; \forall i = [1,m]$, and $Q_i^{\perp}$ represent the Q matrix without Q_i columns. The SRV design conditions for actuator fault isolation are as follows:

1) $S_a^i Q_i^{PRV} = 0$ i.e., i^{th} SRV is unaffected by the i^{th} actuator fault

2) $\left\| S_a^i W_a^{PRV} G_s \right\| < \left\| S_a^1 Q_i^{\perp} \right\|$ i.e., transformation matrix S_a is more sensitive to the rest of the actuator faults and mostly insensitive to the error terms.

We define the following performance index for actuator fault isolation

$$J_a = \frac{\left\| S_a^i W_r^{PRV} G_s \right\|}{\left\| S_a^i Q_i^{\perp} \right\|}$$

The algorithm of calculating S_s^i can be directly applied here to the calculation of a suitable S_a^i with S_s^i, $W_r^{PRV} G_s$, and $W_{r,i}^{\perp}$ replaced with S_a^i, Q_i^{PRV}, and $Q_i^{\perp}$ respectively.

5. Experimental Results

We perform experiments on Pioneer 3-DX mobile robot to detect both the sensor and actuator faults. We make the following remarks before presenting the results:

1) We assume only one fault happen at a time, be it sensor or actuator fault.
2) We developed the dynamic model of the mobile robot in Section 2 as given in (7). We use (7) to calculate the PRV and SRV for fault detection and isolation respectively. The imprecise calculation of kinematic value of mobile robot, and the mismatch dynamics contribute to the error term and the friction in the system contributes to the disturbances.
3) To calculate the residuals as in (15) and (17) we run the model (7) in parallel with the Pioneer mobile robot. We use the nominal model (7) to calculate the terms, G_s, H_s, and Γ_s while the input and the output are obtained directly from the experimental data. The input to the mobile robot is velocity but we need the

torque input for the residual calculation. The controller in the mobile robot server calculates the error velocity based on the desired and actual velocity and multiple with constant for PID controller to generate the input to the driving motors. We select the following PID gains: P=20, I=15 and D=7 for the experiments. Based on this PID gains and the velocity error term we calculate the input torque to the mobile robot and use that value for residual calculation.

4) Player [18] is used to control Pioneer 3-DX mobile robot.
5) In the experiments, the mobile robot performed a straight-line trajectory-tracking task.
6) In the experiments the mobile robot tracks a straight line trajectory using a kinematic level PID controller with the following PID gains: P=12, I=0 and D=0. We should mention that the residuals are independent of the choice of the controller.

5.1 Sensor Fault Isolation Results

There can be different kind of faults in the sensor. In [13] some of the common sensor faults are mentioned. We consider partial sensor faults, which is very common in practical situation, to demonstrate the fault detection and isolation methods describe in this paper. A partial sensor fault is the one where encoder reflects only a fraction of the actual value. This type of fault occurs when there is an offset or bias in the sensor reading.

There are three sensor outputs, x_c, y_c, and ϕ. We introduce partial sensor faults in each of them but not at the same time. The partial sensor faults were introduced by multiplying the sensor outputs with 0.75 after 11 seconds of operation. We design five different fault detection residuals for sensor faults using the dynamic model of the mobile robot. Based on the fault detection residual we design three different SRVs. We present the SRV outputs under each sensor faults. A threshold value is assigned for each SRV.

First we introduce partial fault in the x_c output at 11 seconds. The three SRV outputs are shown in Figure 5-2. The response of the three SRVs to the first sensor fault can be characterized by the $[0 \quad 1 \quad 1]$, where "0" indicates SRV values are under the threshold value, and "1" indicates that the SRV values are above the threshold value. Using Table 5-1, we can conclude that the first sensor is faulty.

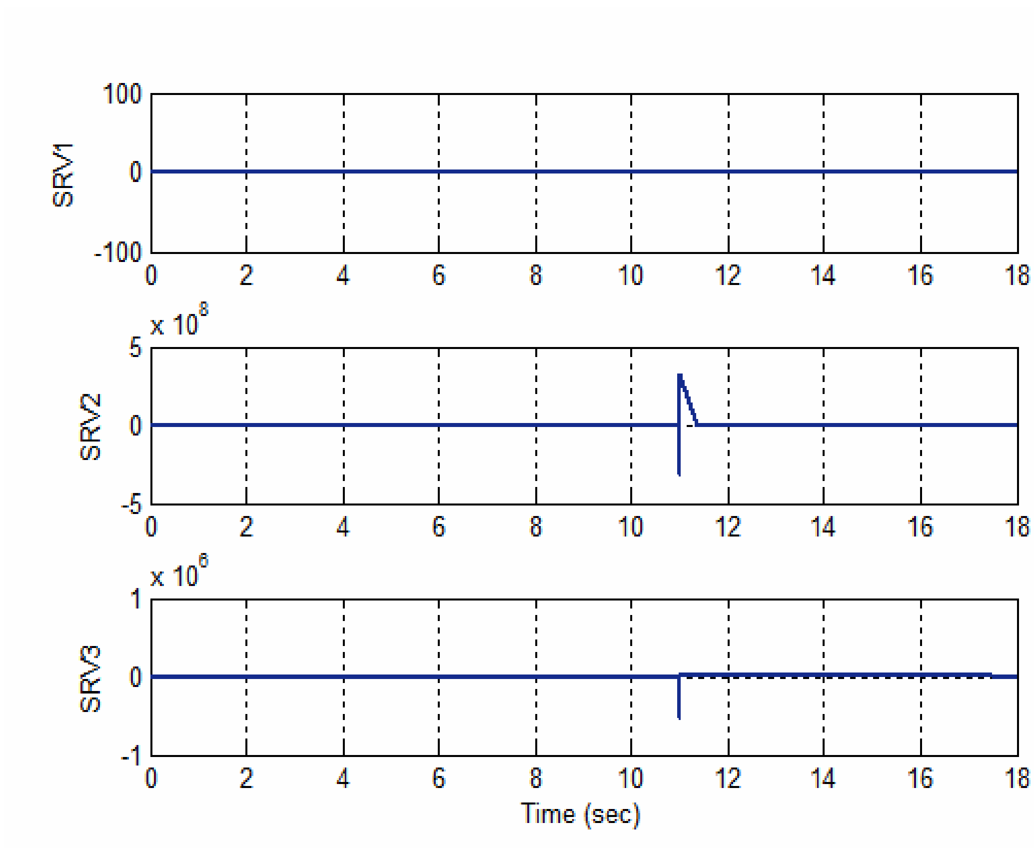


Figure 5-2. SRV outputs with first sensor fault

Next, we introduce the fault at 11 seconds in second sensor, i.e. y_c. We run the experiments under the same condition as before and register the output from the three SRVs. The SRV outputs are given in Figure 5-3. We characterize the response of the three SRVs to the second sensor fault by $[1 \quad 0 \quad 1]$, where "0" and "1" indicate that the SRV values are under and above the threshold value respectively. Based on Table 5-1, we can conclude that the fault in second sensor is isolated correctly.

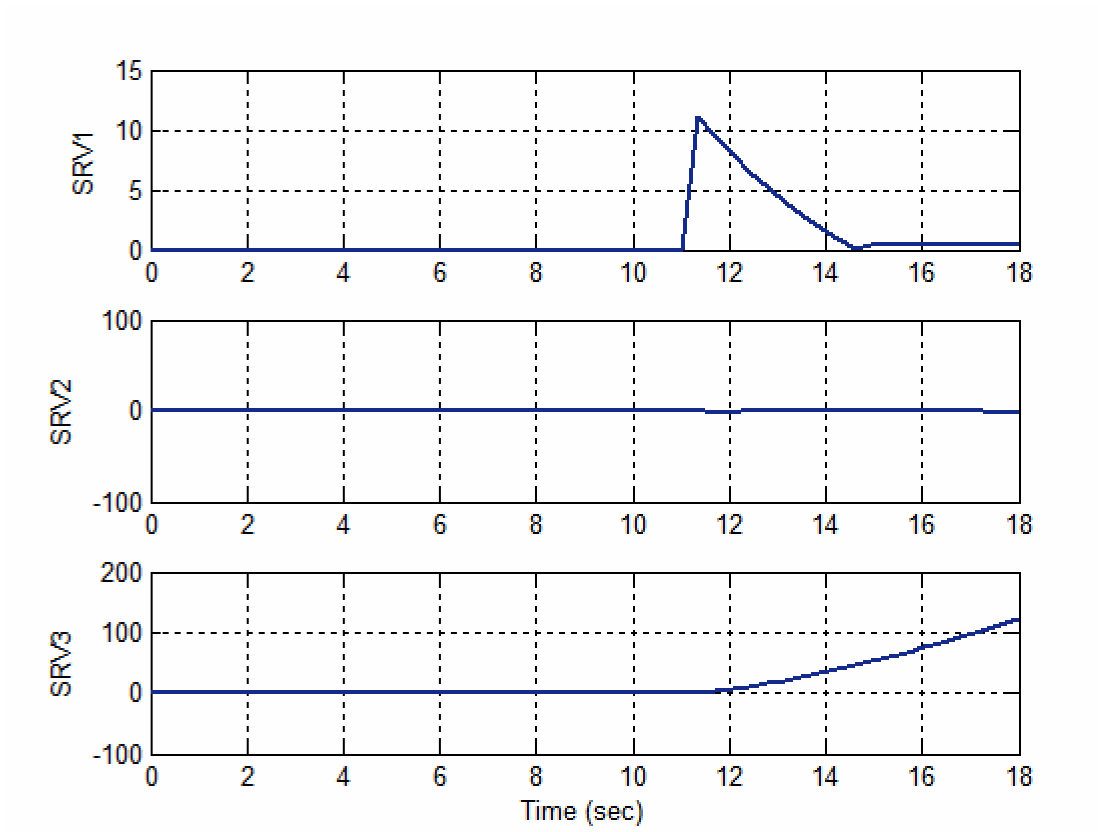


Figure 5-3. SRV outputs with second sensor fault

Finally, we introduce the fault in third sensor, ϕ. The three SRV outputs for third sensor fault are shown in Figure 5-4. The response of the three SRVs to the third sensor fault can be characterized by the $[1 \quad 1 \quad 0]$. In this case as well we isolate the third sensor fault correctly. The above results successfully demonstrate the effectiveness of the fault isolation method in a mobile robot.

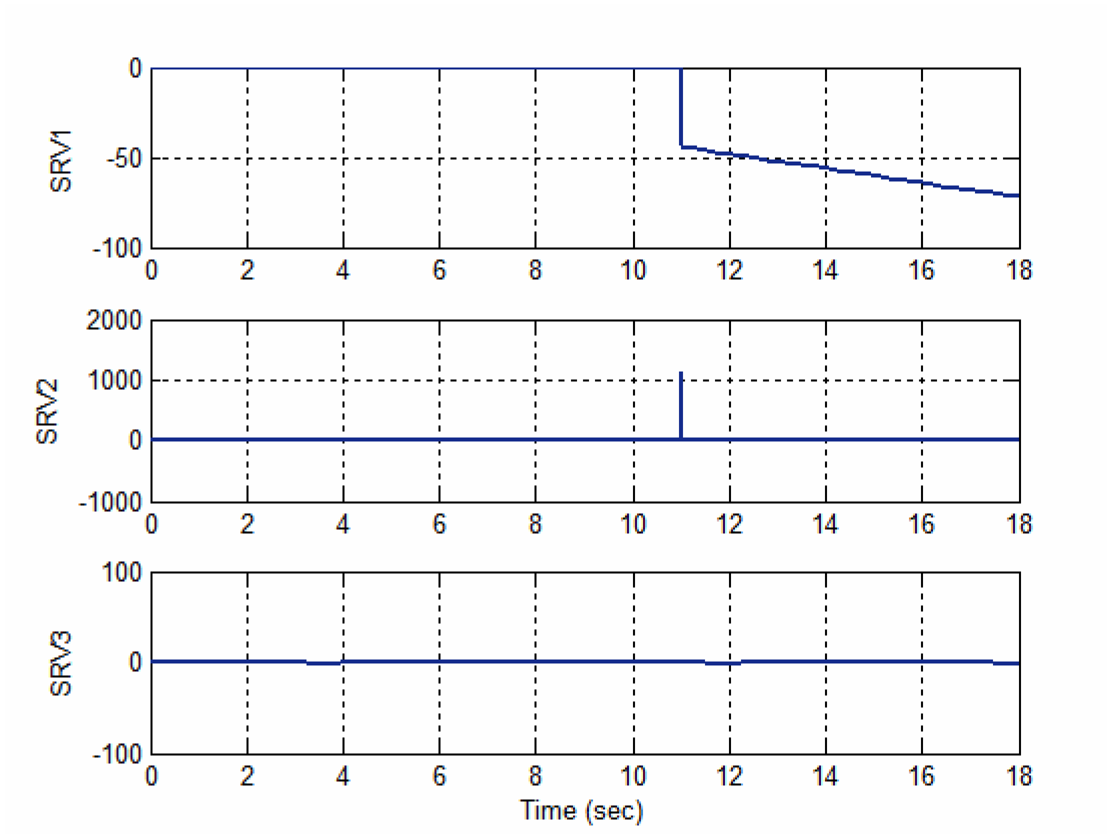


Figure 5-4. SRV outputs with third sensor fault

5.2 Actuator Fault Isolation Results

In [19] various types of actuator faults are discussed that are relevant for a mobile robot operation. We choose partial actuator fault where one actuator generates only a part of the desired torque to demonstrate the proposed fault isolation method. This type of fault represents degradation in the actuator system (e.g., friction due to jamming, problems in transmission etc.). Two DC motors are used to actuate the Pioneer mobile robot. We introduce partial actuator faults both in right and left motors but not at the same time. In the experimental set-up the partial actuator faults were introduced by changing the velocity inputs by 0.85% after 11 seconds of operation. We design two SRVs to isolate two actuator faults.

We introduce the fault at 11 seconds in the first actuator, which is the right side motor. The two SRV outputs are shown in Figure 5-5. We characterize the SRV outputs as $[0 \quad 1]$. Based on the SRV outputs and Table 5-2 we can observe that the first actuator fault is clearly isolated.

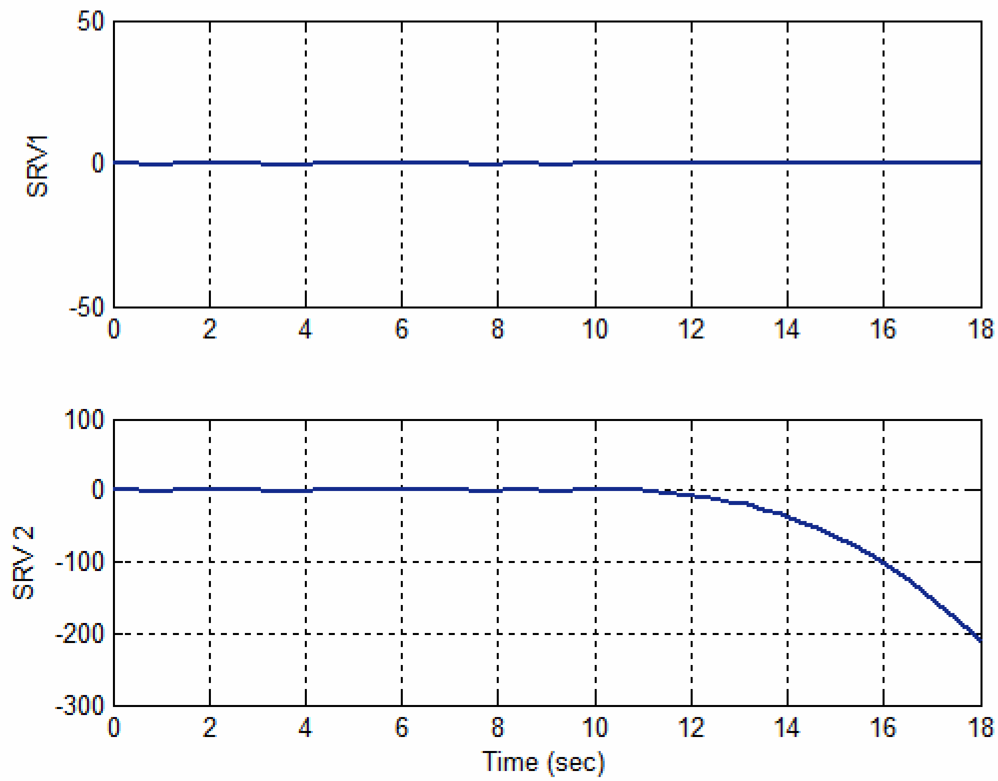


Figure 5-5. SRV outputs with first actuator fault

Finally, we introduce the fault in second actuator at 11 seconds. The SRV outputs under second actuator fault are given in Figure 5-6. Under the second actuator fault the SRV outputs is characterized as $[0 \quad 1]$. The results demonstrate that the second actuator fault is isolated clearly. We conclude that the presented fault detection and isolation method is effective under MPM and disturbance and applicable to the mobile robot.

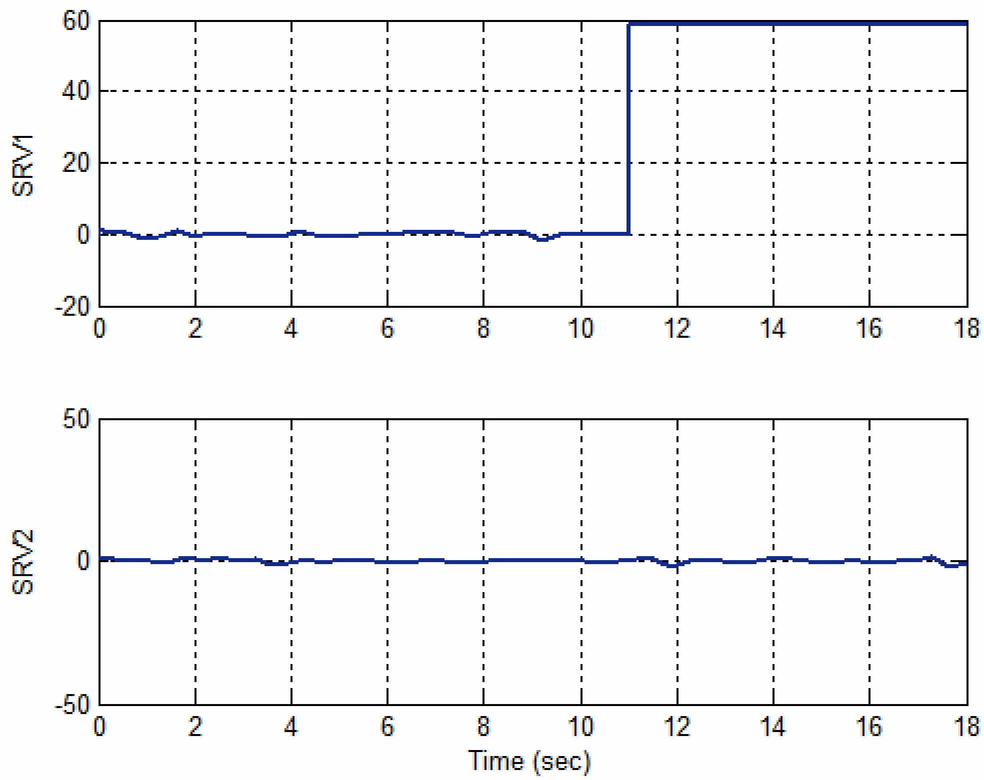


Figure 5-6. SRV outputs with second actuator fault

6. Conclusion

A robust method for the detection and isolation of sensor and actuator faults is presented in this paper. The proposed robust nonlinear analytic redundancy method was experimentally verified on a Pioneer 3-DX mobile robot. The results show that both sensor and actuator fault detection and isolation are possible in the presence of MPM and disturbances. Future work includes detection and isolation of multiple and incipient faults.

References

[1] R. Isermann, "Process fault detection based on modeling and estimation methods- a survey." *Automatica*, 20, 387-404, 1984.

[2] J. Gertler, "Analytic redundancy methods in fault detection and isolation – survey and synthesis." *Preprints of IFAC safeprocess conference*, 1, 9-22. 1991.

[3] P.M. Frank and X Ding, "Survey of robust residual generation and evaluation method in observer-based fault detection system." *Journal of Process Control*, 7, 403-424. 1997.

[4] J. Gertler, "*Fault Detection and Diagnosis in Engineering systems,* 1 ed" Marcel Dekker. Inc, New York. 1998

[5] R. V. Bread, "Failure Accommodation in Linear Systems Through Self-reorganization." PhD, dissertation, Massachusetts Inst. Technology, Cambridge, MA. 1971.

[6] P.M. Frank, and X. Ding, "Frequency domain approach to optimally robust residual generation and evaluation for model-based fault diagnosis" *Automatica*, 30, 789-804 1994.

[7] J. Chen, and R. J.Patton, *Robust model-based fault diagnosis for dynamic systems*. Dordrecht: Kluwer Academic Press, 1999.

[8] M. Junzheng, and W. Shoukun, "Robust fault detection using iterative learning observer for nonlinear systems." *Proc. 5th world conf. intelligent control and automation*, June 15-19, 2004.

[9] M. M. Polycarpou, and A. B. Trunov, "Learning approach to nonlinear fault diagnosis: detectability analysis." *IEEE Transactions on Automatic Control*, 45, 806-812, 2000.

[10] E. Chow and A. Willsky, "Analytic redundancy and the design of robust failure detection systems." *IEEE Transactions on Automatic Control*, 29, 603-614, 1984.

[11] J. Gertler and D. Singer, "A new structural framework for parity equation based failure detection and isolation." *Automatica*, 26, 381-388, 1990.

[12] J. Gertler, and M. Kunwer, "Optimal residual decoupling for robust fault diagnosis." *International Journal of Control*, 61, 395-421, 1995.

[13] W. Li, and S. Shah, "Structured residual vector-based approach to sensor fault detection and isolation." *Journal of Process Control*, 12, 429-443, 2002.

[14] Z. Han, W. Li, and S. Shah, "Fault detection and isolation in the presence of process uncertainties." *Control engineering practice*, 13, 587-599, 2005.

[15] M. L. Leuschen, "Derivation and application of nonlinear analytic techniques with application on robotics." PhD, dissertation, Rice University, 2001.

[16] A Shumsky, "Robust analytical redundancy relations for fault diagnosis in nonlinear systems." *Asian journal of control*, 4, 159-170, 2002.

[17] B. Halder, and N. Sarkar, "Robust fault detection based on nonlinear analytic redundancy techniques with Application to Robotics." Proceedings of IMECE, Orlando, Florida, November 5-11, 2005.

[18] B. P. Gerkey, R. T Vaughan, and A. Howard. The player/Stage Project: Tools for Multi-Robot and Distributed Sensor Systems. In proc. Of Intl Conf. on Advanced Robotics (ICRA), 317-323, 2003.

[19] A. D. Luca and R. Mattone, "Actuator failure detection isolation using generalized momenta," proceedings of the 2003 IEEE, ICRA, 634639, 2003.