TSI PRESS SERIES

INTELLIGENT AUTOMATION AND SOFT COMPUTING

TRENDS IN RESEARCH, DEVELOPMENT, AND APPLICATIONS

VOLUME 2

Proceedings of the First
World Automation Congress (WAC '94)
held August 14-17, 1994 in Maui, Hawaii, U.S.A.

EDITORS

Mohammad Jamshidi
University of New Mexico

Junku Yuh
University of Hawaii

Charles Nguyen
Catholic University of America

Ronald Lumia
National Institute of Standards and Technology

CAD LAB
Laboratory for Intelligent and Robotic Systems
University of New Mexico

WAC
WORLD AUTOMATION CONGRESS

HTDC
HIGH TECHNOLOGY
DEVELOPMENT CORPORATION

TSI Press | Albuquerque, New Mexico USA | 1994

About the Cover: The cover represents a reflection on the effects of intelligence on the industrial automation of the world. Seen here are many nodes and branches of a neural network and two Gaussian style fuzzy membership functions.

Cover Designers: Ava Jamshidi, Sandia Preparatory School, Albuquerque, New Mexico, USA and Virginia Henning, Desert Dreams Publishers, Albuquerque, New Mexico, USA.

P. O. Box 14393, Albuquerque, NM 87191-4394, USA
FAX: 1-505-291-0013

ISBN 0-962 7451-5-4

Proceedings of the
First World Automation Conference
Wailea, HI (August 1994).

ADAPTIVE FAULT DETECTION AND TOLERANCE FOR ROBOTS

M. L. Visinsky, J. R. Cavallaro and I. D. Walker
Department of Electrical and Computer Engineering
Rice University, Houston, Texas 77251-1892

ABSTRACT

In existing robot fault detection schemes, sensed values of the joint status (position, velocity, etc.) are typically compared against expected or desired values, and if a given threshold is exceeded, a fault is inferred. The thresholds tend to be empirically determined and held constant over a wide range of trajectories. This leads to false alarms when the threshold is too small to counter the error-inducing effects of model inaccuracy and to undetected faults when the threshold is too large for the given situation. This paper presents new methods for adaptively choosing fault detection thresholds, subject to sensing and modeling inaccuracies and the changing status of the robot. Our approach chooses optimal thresholds based on a Singular Value Decomposition (SVD) of a specialized error regressor format of the dynamics to minimize the possibility of false alarms or undetected failures. The thresholds vary dynamically with the changing trajectory and configuration of the robot and with the robot's failure status. Examples of the fault detection scheme for a non-planar 3 DOF robot are given.

INTRODUCTION

Fault detection algorithms monitor systems for failures by comparing measured system outputs with other similar outputs or with the corresponding expected output derived from a model of the system behavior. In previous work [8, 10], we have developed comparison tests for a variety of robots using analytical redundancy, a method which reveals the existing functional redundancy in a given system and allows us to perform fault detection without modifying or expanding the current hardware. In their pure form, the ideal tests derived from analytical redundancy cause an unacceptable number of false alarms due to the sensor and modeling errors which arise from linearization of the robot equations and inaccuracies in model parameters such as link inertia or mass. In addition, the effect of the model errors and sensor noise fluctuates dynamically as the motion of the robot changes and as failures occur. Dynamic thresholds must be used to mask out the modeling errors and cope with the ever changing robot status while remaining small enough so as not to hide real failures.

There are a variety of algorithms which focus on computing thresholds for fault detection in uncertain dynamic systems [1, 2, 4, 7, 12]. In [11], we introduced ThMB, a novel threshold generator tailored to the special needs of robotics. ThMB utilizes a technique of estimating parameter error effects similar to that used in the Reachable Measurement Intervals (RMI) algorithm [3, 11] while creating a faster, more efficient algorithm. In this paper, we first summarize the derivation of this new method for threshold generation and then present our new work in improving the ThMB algorithm. Initial results of the improved algorithm applied to the first three joints of a simulated PUMA 600 robot are given. Our conclusions and future work are presented in the final section.

MODEL-BASED THRESHOLD GENERATION

In this section, we summarize the development of ThMB, our new model-based threshold generation algorithm, which was introduced in [11]. The goal of the algorithm is to determine the maximum effects of modeling inaccuracies on the robot system. Modeling inaccuracies arise due to the fact that the estimates of the parameters in the controller vary from the actual parameters of the real robot. These inaccuracies are complicated by the linearization of the dynamics in most controllers. To develop the threshold algorithm, we must first derive an equation which represents the error in the system based on the errors in the parameters. We start with the inverse dynamics control law:

$$\underline{\tau} = M(\underline{q})[\ddot{\underline{q}}_d + K_d\dot{\underline{e}} + K_p\underline{e}] + N(\underline{q}, \dot{\underline{q}}), \tag{1}$$

where M is the inertia matrix, N contains the centripetal and coriolis terms, and $\underline{q}$ and $\dot{\underline{q}}$ are state variables. The desired acceleration is $\ddot{\underline{q}}_d$, K_d and K_p are the system gain matrices, and $\dot{e}$ and e represent the error in the state variables (ie: $\dot{q}_d - \dot{q}$ and $q_d - q$).

Equation (1) is based on the estimates of the system parameters (present in M and N). The actual parameters in the robot vary from these estimates and are assumed to be within given ranges. The robot equation is written as:

$$\ddot{\underline{q}} = \hat{M}^{-1}\underline{\tau} - \hat{M}^{-1}\hat{N}. \tag{2}$$

The actual robot parameters are represented by $\hat{M}$ and $\hat{N}$. By substituting the torque from (1) into (2) and manipulating the equations, we get the following error equation:

$$\ddot{\underline{e}} + K_d\dot{\underline{e}} + K_p\underline{e} = M^{-1}(\tilde{M}\ddot{\underline{q}} + \tilde{N}), \tag{3}$$

with $\tilde{M} = \hat{M} - M$ and $\tilde{N} = \hat{N} - N$.

Using the fact that, for a rigid link robot, the nonlinear equations of motion are linear with respect to the parameters, we can separate the parameter-based coefficients in the equation from the joint variables to produce the linear relationship:

$$\ddot{\underline{e}} + K_d\dot{\underline{e}} + K_p\underline{e} = M^{-1}[Y]\tilde{\underline{p}}. \tag{4}$$

The regressor matrix Y contains the known functions of the state variables $(\underline{q}, \dot{\underline{q}}, \ddot{\underline{q}})$ and $\tilde{\underline{p}}$ is a vector of the errors between the actual parameter coefficients $(\hat{p})$ and their estimates (p). This formulation of the dynamics, often used in adaptive control strategies, is discussed in more detail in [5, 11]. By extracting the unknown portions of the dynamics (the parameters) into the coefficient vector, $\tilde{p}$, the ThMB algorithm can more easily determine the effects of changes in these parameters on the system.

Because equation (4) is based on the real (measured) error (in $\dot{e}$ and e), we expect the thresholds to become invalid after the failure has occurred. This is verified in the results below but is not a major concern as the fault detection and tolerance algorithms of [8] are designed to detect the failure. After this immediate detection event, the incoming faulty data will be isolated and ignored. The reconfiguration strategy will attempt to find valid data for the controller from a backup sensor system, if possible, or redistribute the workload of the joint so that the robot will be able to continue on its desired path.

SVD

Determining the maximum effect of the parameter variations becomes more time consuming as the number of parameters increases. For the simplest possible example, that of a pendulum, there are two parameters (mass and length) and four possible combinations of the parameter extremes (2^2). For such a small number of combinations, the four different $\tilde{p}$ vectors can be computed directly and compared to find the maximum of the resulting $M^{-1}[Y]\tilde{p}$ values [11]. As the number of parameters increase, however, the number of possible combinations increases geometrically. For the non-planar 3 DOF robot used in the examples of this paper, there are nine variant parameters and thus $2^9 = 512$ combinations of the parameter extremes. This complexity is obviously not conducive to a full enumeration of all possible $\tilde{p}$ vectors. We are also not guaranteed that the extremes of the parameters will produce the extremes in equation (4). We therefore need to derive a method for computing the extremes that is more efficient for larger problems and more reliable.

We can accomplish this task by applying the Singular Value Decomposition (SVD) to each row of the matrix $X = [M^{-1}Y]$. If X is $m \times p$, then the SVD of a single row will look like:

$$U\Sigma V^{\mathrm{T}} = [1][\sigma 0 \cdots 0]_{1\times p} \begin{bmatrix} v_{11} & \cdots & v_{1p} \\ \cdot & & \\ \cdot & & \\ v_{p1} & \cdots & v_{pp} \end{bmatrix}^{\mathrm{T}} . \tag{5}$$

The matrix V is unitary, implying that it will map a uniform p−dimensional sphere into a p−dimensional sphere [13]. Thus, if the elements of $\tilde{p}$ are all the same size, V will map the sphere defined by $\tilde{p}$ back onto itself. We can make the assumption, here, that the parameter errors are all of the same range. This may be a conservative estimate for smaller parameters but will produce acceptable results. When the decomposition of a row is used in (4), the error equation for the single joint corresponding to that row of X becomes:

$$\ddot{e} + K_d\dot{e} + K_p e = [1][\sigma 0 \cdots 0]\underline{W}. \tag{6}$$

The vector $W = V^{\mathrm{T}}\tilde{p}$ and the norm of W is the same as the norm of the chosen vector $\tilde{p}$ because V is unitary. If the maximum range of the parameter errors is

p_r, then the maximum error would be

$$\ddot{e}_{\max} = -K_d \dot{e}_r - K_p e_r + \sigma p_r. \tag{7}$$

The measured errors of the last iteration, $\dot{e}_r$ and e_r, are used in (7) as we assume that the last iteration was within the thresholds bounds [11]. Equation (7) thus becomes the basis from which $\dot{e}_{\max}$ and $e_{\max}$ are derived through numerical integration.

RESULTS

This section presents the initial results of the improved ThMB algorithm using the SVD computations discussed above. The algorithm was applied to the first three joints of the PUMA 600 robot using the parameter values defined in [6]. The regressor formulation and code details are presented in [9]. Figure 1 shows

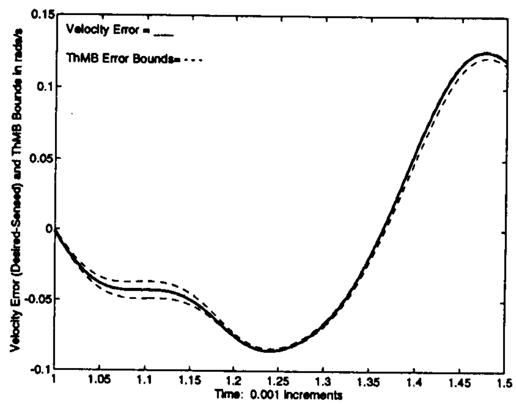

FIG. 1. Thresholds for Joint 1 Tachometer, Fault Free Case

the tachometer threshold bounds for joint 1 (dashed lines) produced by ThMB when all joints are directed to move along a sine wave trajectory. The variant parameters, which are inertias and centers of mass, are all assumed to range around their estimates by ±0.003. For the larger inertias, this range implies a possible error of ±1.624%. For the smallest parameter, the possible error might be as high as 83.3%. The actual parameter errors are selected randomly and for this example range between 0.00134 and −0.000985. The tracking errors (desired values minus the measured values) are also presented (solid line) in the figure for this joint. The figure shows that ThMB produces dynamic thresholds that vary in size as the joint moves through its trajectory and that bound the actual error during the fault free run.

Given that we assume the error of the last iteration is a fact, ThMB computes the farthest that the state can change from that point due to the parameter errors. This may not include a perfect system (ie: an error of zero) as can be seen in the figures because the dynamics of the robot might not enable it to move back from the given erroneous state to the desired value within one iteration even with the effects of the parameter errors. ThMB takes on the current state in its computations and therefore can only compute the effects based on this state. Moving to the zero error state may take more change than the current state and robot dynamics can produce.

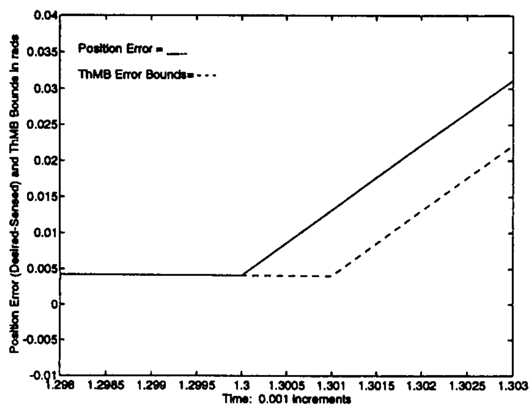

FIG. 2. Thresholds for Joint 1 Encoder when Encoder fails at time 1.3

Figure 2 presents the results when a failure occurs in the encoder for joint 1 at time 1.3. Note that there are actually two dashed lines representing the maximum positive and negative bound, but they are difficult to differentiate at this scale. The actual error (solid line) immediately exceeds the bounds when the failure occurs. Textual output alerts the operator that the bounds were exceeded at time 1.301. As mentioned earlier, the fault detection and tolerance code would respond to this failure when the error passes outside of the thresholds and the erroneous bounds produced by ThMB and shown in Figure 2 after time 1.3 would be ignored [8, 11]. More extensive results are presented in [9].

CONCLUSION

In this paper, we present our latest improvements to the ThMB detection threshold generation algorithm. The addition of the SVD-based computation for determining the maximum effects of parameter errors on the robot system make ThMB more efficient and responsive, especially for larger systems with more complex dynamics and more variant parameters. We present some preliminary results for a non-planar 3 DOF robot and show that the ThMB thresholds dynamically bound modeling errors during fault free runs while still enabling the detection of real failures in the system.

Acknowledgements

This work was supported in part by the National Science Foundation under grants MSS-9024391 and DDM-9202639, and by DOE Sandia National Laboratory Contract #18-4379A. Further support was provided by a Mitre Corporation Graduate Fellowship and NSF Graduate Fellowship RCD-9154692.

REFERENCES

1. X. Ding and P. M. Frank. Frequency Domain Approach and Threshold Selector for Robust Model-Based Fault Detection and Isolation. In *Proceedings IFAC Fault Detection, Supervision and Safety for Technical Processes*, pages 271–276, Baden-Baden, Germany, 1991.

2. A. Emami-Naeini, M. M. Akhter, and S. M. Rock. Effect of Model Uncertainty on Failure Detection: the Threshold Selector. *IEEE Transactions on Automatic Control*, 33(12):1106–1115, December 1988.

3. D. T. Horak. Experimental Estimation of Modeling Errors in Dynamic Systems. *AIAA Journal of Guidance, Control, and Dynamics*, 12(5):653–658, September-October 1989.

4. R. Isermann, W. Appel, B. Freyermuth, A. Fuchs, W. Janik, D. Neumann, Th. Reiss, and P. Wanke. Model Based Fault Diagnosis and Supervision of Machines and Drives. In *Proceedings IFAC 11th Triennial World Congress*, pages 1–12, Tallinn, Estonia, USSR, August 1990.

5. W. S. Lu and Q. H. Meng. Regressor Formulation of Robot Dynamics: Computation and Applications. *IEEE Transaction on Robotics and Automation*, 9(3):323–333, June 1993.

6. C.P. Neuman and J.J. Murray. The Complete Dynamic Model and Customized Algorithms of the Puma Robot. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-17(4):635–644, July/August 1987.

7. R. Rückwald. A Model-Based Fault Detection Concept for Mechanical Systems. In *Proceedings IFAC Fault Detection, Supervision and Safety for Technical Processes*, pages 205–210, Baden-Baden, Germany, 1991.

8. M. L. Visinsky. Fault Detection and Fault Tolerance Methods for Robotics. Master's thesis, Department of Electrical and Computer Engineering, Rice University, Houston, TX, December 1991.

9. M. L. Visinsky. *Dynamic Fault Detection and Intelligent Fault Tolerance Methods for Robotics.* PhD thesis, Department of Electrical and Computer Engineering, Rice University, Houston, TX, April 1994.

10. M. L. Visinsky, I. D. Walker, and J. R. Cavallaro. Layered Dynamic Fault Detection and Tolerance for Robots. In *1993 IEEE International Conference on Robotics and Automation*, pages 180–187, Atlanta, GA, May 1993.

11. M. L. Visinsky, I. D. Walker, and J. R. Cavallaro. New Dynamic Model-Based Fault Detection Thresholds for Robot Manipulators. In *1994 IEEE International Conference on Robotics and Automation*, San Diego, CA, May 1994. To appear.

12. J. Wünnenberg and P. M. Frank. Dynamic Model Based Incipient Fault Detection Concept for Robots. In *Proceedings IFAC 11th Triennial World Congress*, pages 61–66, Tallinn, Estonia, USSR, August 1990.

13. T. Yoshikawa. Analysis and control of robot manipulators with redundancy. In Brady and Paul, editors, *Robotics Research - The First International Symposium*, pages 735–747. MIT Press, 1985.