Adaptive Control of Mechanical Manipulators

John J. Craig
Silma, Inc.

Adaptive Control of Mechanical Manipulators

John J. Craig

Silma, Inc.

ADDISON-WESLEY PUBLISHING COMPANY
Reading, Massachusetts • Menlo Park, California • New York
Don Mills, Ontario • Wokingham, England • Amsterdam • Bonn
Sydney • Singapore • Tokyo • Madrid • Bogotá • Santiago • San Juan

Library of Congress Cataloging-in-Publication Data

Craig, John J., 1955–
Adaptive control of mechanical manipulators.

Bibliography: p.
Includes index.
1. Manipulators (Mechanisms)—Automatic control.
2. Adaptive control systems. 3. Robots, Industrial.
I. Title.
TJ211.C66 1988 629.8′92 86-22158
ISBN 0-201-10490-3

ABCDEFGHIJ-AL-8987

PREFACE

This book is concerned with one of the most interesting and challenging problems in controlling a robot manipulator—the use of adaptive control algorithms. The idea of adaptive or learning control for mechanical manipulators has certainly come to the mind of anyone who has considered the problem of controlling such a complex system. It is a compelling notion that one might build a system that can learn and improve its performance as it operates. This desire has led many researchers to investigate the application of adaptive strategies to general control problems, and recently, to the problem of robot control. The problem is not an easy one, and is not yet completely solved.

This book examines the problem in the nonlinear domain in which the robot control problem is set, rather than relying on the existing theory for adaptive control of linear systems. Along the way, the theory for linear systems is reviewed, previous research in adaptive control of robot manipulators is reviewed, and manipulator dynamics and nonadaptive control are presented.

New results presented in the text include a robustness result for the non-adaptive model-based robot controller, and an algorithm for the learning control of manipulators. The core of the book is the presentation of a new non-linear adaptive controller for mechanical manipulators that is rigorously proven globally stable.

This book is appropriate for mathematicians and engineers with an interest in manipulator dynamics and control. In particular, those interested in the topics of robustness, adaptation, and learning in the context of nonlinear systems such as robot manipulators may find the text interesting.

Acknowledgments

I wish to thank many people who contributed in various ways to this book. My principal thesis advisor, Prof. Bernard Roth, deserves special thanks for this technical guidance and wisdom. It was my pleasure to have Prof. Roth as an advisor and friend at Stanford. Many results in Chapter 5 are joint conclusions with my colleagues at the University of California at Berkeley, Ping Hsu and Prof. S. Shankar Sastry. At Stanford, I benefited from discussions with Dr. Oussama Khatib, Prof. Gene Franklin, and Prof. Stephen Boyd. I am thankful to Greg Martin who worked on the simulations appearing in Chapter 5. Jim Maples supplied many hours of help with the experiments at Adept Technology, Inc.

The help and enthusiasm of the following is gratefully acknowledged: Brian Armstrong, Joel Burdick, Madhusudan Raghaven, Dr. Ron Goldman, Dr. Jeff Kerr, Dr. Dave Marimont, John Hake, Dr. Bill Hamilton, Brian Carlisle, Prof. Bernard Widrow, and Prof. Gordon Kino.

I am grateful to the Systems Development Foundation for their support through contracts with Prof. Roth. Several years of support were provided by Prof. Tom Binford. Silma, Inc., provided support and was understanding in scheduling my time carefully during completion of the book. Adept Technology, Inc., provided the use of their facilities for the experiments of Chapter 5.

This book is essentially a reprint of my Ph.D. dissertation, but I am thankful to the editors and staff at Addison-Wesley for their input, which led to many manuscript corrections.

For moral and spiritual support, I must single out Monique Craig, who has inspired and pushed me. I also thank my parents, James and Harriet Craig, and Tom and Linda Craig, Connie Craig, Chris Goad, Donalda Speight, Tim Turner, Paul and Avi Munro, Al Barr, Warren Davis, Joe Levy, Dave Baras, Ned Kirchner, Peter Hochschild, Glenn Howland, and others. Finally, special thanks to those smart enough to transcend all this: Yoda, Zorro, Babar, and Kaza.

Palo Alto, California J.C.

CONTENTS

Chapter 1

INTRODUCTION

1.1 Introduction

Present-day industrial robots operate with very simple controllers, which do not yet take full advantage of the inexpensive computer power that has become available. The result is that these fairly expensive mechanisms are not being utilized to their full potential in terms of the speed and precision of their movements. With a more powerful control computer it is possible to use a dynamic model of the manipulator as the heart of a sophisticated control algorithm. This dynamic model allows the control algorithm to "know" how to control the manipulator's actuators in order to compensate for the complicated effects of inertial, centrifugal, Coriolis, gravity, and friction forces when the robot is in motion. The result is that the manipulator can be made to follow a desired trajectory through space with smaller tracking errors, or perhaps move faster while maintaining good tracking.

There are two reasons why such sophisticated control algorithms have not found use outside of research laboratories. The first is the economics of supplying sufficient computing power to the robot controller. Recently this problem has diminished greatly, and will continue to do so. The second and more serious problem is that of imprecise dynamic models. Developing a correct dynamic model (in the form of a set of coupled differential equations) for a multidegree-of-freedom manipulator is a difficult task. Recent work has made developing the structure of these equations more or less straightforward for the cases where the links are modeled as rigid bodies. However, a problem that remains is that of unknown parameters that appear in the model, and also of effects such as friction and flexibilities, which are left out of the model formulation. Not only are parameters unknown or only poorly known, but they also may be subject to change as the manipulator goes about its tasks.

This book addresses the control of mechanical manipulators in cases where the physical models that describe the manipulators are not well known. Incorrectness or uncertainty in a dynamic model can be split into two portions. *Structured uncertainty* is what we will call the case of a correct structural model with all uncertainty due to incorrect parameter values. That is, there exists a correct (but unknown) set of values for the parameters such that the model will match the actual system. *Unstructured uncertainty* is the name given to unmodeled effects, some of which may be state-dependent and some of which are external disturbances. Unstructured uncertainty arises from sources not considered by the designer, or those that are too complex to model.

Much of this book addresses the case where modeling error is largely due to structured uncertainty. However, the reality of external disturbances is considered throughout, and in Chapter 6 a special learning algorithm is developed specifically for the case of unmodeled dynamic effects that lack a parametric model.

1.2 Historical Summary

In the past several years a great number of papers have been published about various aspects of robotics. Even confining ourselves only to those papers dealing with the control problem for mechanical manipulators, the volume of published work makes a concise review difficult. We will therefore mention only early work, and only in the areas of kinematics, path generation, dynamics, and control. We will neglect many important areas such as sensors, robot programming languages, locomotion, etc. Although we believe that we have cited the most important contributors, it is possible that we are unaware of some work.

Today's industrial robots have their roots in numerically controlled (NC) machines and the early master–slave teleoperators used in the nuclear industry. In 1947 work started at Argonne National Laboratory on master–slave systems. Originally these were simply mechanical linkages; later electrically and hydraulically powered systems were developed, some with "force-reflecting" capability [1].

Based on George Devol's ideas, Unimation Inc. developed the first industrial robot in 1959, and installed the first robot in a U.S. factory in 1961 [2]. In 1961 Ernst [3] developed a computer-controlled mechanical hand with tactile sensors, called "MH-1," which was coupled with an Argonne National Laboratory manipulator and a computer. It was capable of stacking blocks under computer control.

At Stanford University an early laboratory was established in 1965 by John McCarthy and others [4]. One of the first six degree-of-freedom, electric, computer-controlled manipulators was designed and built by Scheinman in 1969 [5], and became known as the Stanford Scheinman arm.

Early work in robotics was largely concerned with the basic problems of representing spatial information [6], and the manipulator's kinematic equations and their solution [7, 8]. Another early focus of research was in generating trajectories and controlling the manipulator to move along them [9-13].

The first application of dynamic analysis to the particular problem of a multidegree-of-freedom mechanical manipulator seems to have been by Kahn and Roth [12], based on Uicker's work [14] on linkages. This early work was not particularly concerned with efficiency and resulted in a computational algorithm that was $O(n^4)$ in complexity, where n is the number of manipulator joints. Renaud [15] and Liegois et al. [16] made early contributions concerning formulating the mass-distribution descriptions of the links. While studying the modeling of human limbs, Stepanenko and Vukobratovic [17] began investigating a "Newton–Euler" approach to dynamics instead of the somewhat more traditional Lagrangian approach. This work was revised for efficiency by Orin et al. [18] in an application to the legs of walking robots. Orin's group improved the efficiency somewhat by writing the forces and moments in the local link reference frames instead of the inertial frame. They also noticed the sequential nature of calculations from one link to the next, and speculated that an efficient recursive formulation might exist. Armstrong [19] and Luh, Walker, and Paul [20] paid close attention to details of efficiency and published an algorithm that is $O(n)$ in complexity. This was accomplished by setting up the calculations in an iterative (or recursive) nature and by expressing the velocities and accelerations of the links in the local link frames. Hollerbach [21] and Silver [22] further explored various computational algorithms. Hollerbach and Sahar [23] showed that for certain specialized geometries the complexity of the algorithm would further reduce. Finally, several authors have published articles showing that for any given manipulator customized closed-form dynamics are more efficient than even the best of the general schemes [24-29].

The net effect of the developments in computing the dynamic model of a manipulator, coupled with the increasing power of computers, was that the model could be computed sufficiently quickly for use in real-time control. The use of the nonlinear dynamic model of a manipulator in a control algorithm apparently has its roots in the work of several researchers. Perhaps the earliest is the work of Freund [30, 31], in which he uses tools from Lie Algebra to discuss the decoupling and linearizing of nonlinear systems. The work of Bejczy [32], Lewis [33], and Markiewicz [34], seems to be responsible for the term "computed torque method" by which the general approach is sometimes known. Other early work was done by Zabala-Iturralde [35], Khatib et al.[36], and Liegois et al. [37]. A closely related approach was given by Luh, Walker, and Paul in [38].

Later, there were many different papers published on various approaches to manipulator control. Of these methods, we will confine our discussion to adaptive techniques. A complete review of adaptive control applied to manipulators is defered until Chapter 4, but the earliest work seems to have been done by Timofeyev and Ekalo [133], Dubowsky and DesForges [83], and Horowitz and Tomizuka [90].

1.3 Contributions of This Book

This book addresses the control of mechanical manipulators in cases where the physical models that describe the manipulators are not well known. We adopt the view that the nonlinear, model-based method (or "computed torque method") of manipulator control is, in theory, a good approach to manipulator control. We then investigate methods of compensating for the fact that a perfect dynamic model is never available.

The contributions of this book are in three areas:

(1) The robustness of the model-based servo in the presence of poorly known parameters is investigated. A sufficient condition for stability of the overall system in the presence of parameter errors is developed.

(2) A parameter-adaptive control scheme is developed in the form of a set of adaptive laws that can be added to the nonlinear model-based controller.

The scheme is unique in that it is designed specifically for this model-based controller, and is rigorously proven stable in the full nonlinear setting.

(3) A learning control scheme is developed that can be added to the model-based controller in order to "learn" compensation for friction and other effects that are difficult to model parametrically.

1.4 Preview of the Book

In Chapter 2 we examine some properties of the dynamics of manipulators and then examine a couple of control stategies that are in use or have been proposed for their control. One of these methods, the so-called computed torque servo, or nonlinear model-based control scheme, will be the focus of our attention. This scheme represents an excellent way to use a dynamic model of a manipulator (if it were perfectly known) in a controller.

In Chapter 3 we analyze some robustness properties of the nonlinear model-based control scheme. The basic question is When the parameters appearing in the model are not well known, does the control scheme still perform well?

In Chapter 4 we review the notion of adaptive control as a methodology for compensating for unknown or loosely known parameters. We review the adaptive control stategies for mechanical manipulators that have been proposed by other researchers.

In Chapter 5 we derive a new adaptive control scheme for manipulators and discuss its properties. The scheme is novel in that no assumptions of plant linearity are made in the development of the adaptation laws or in the stability proof. The scheme can be viewed as an extension of the existing theory of adaptive control for linear systems to a class of nonlinear systems that includes rigid-body models of manipulators. We also consider the questions of persistent excitation and robustness to bounded disturbances.

In Chapter 6 we present a method for learning control of manipulators. This scheme may be used on its own, or in conjunction with the adaptive scheme of Chapter 5. Dynamic effects for which a parametric model are unavailable may be handled by this scheme.

In Chapter 7 we present some conclusions.

In Appendix A there is a brief introduction to norms and normed spaces. Appendix B presents a brief introduction to Lyapunov stability theory. In Appendix C the notion of a strictly positive real transfer function is introduced.

Throughout the book, theories are illustrated by results from simulations, and, in Chapter 5, by results from actual experiments with an industrial manipulator.

Chapter 2

CONTROL OF MECHANICAL MANIPULATORS

2.1 Introduction

In this chapter we present the basics of the trajectory-control problem of mechanical manipulators. First we state our underlying assumptions, which will hold throughout the book. These help to outline the scope of the book by stating what is and what is not assumed to be true of the manipulator system. Next, the dynamic equations that describe the motion of a manipulator are presented, along with some notes on the inherent structure of these equations. We look briefly at one simple method before introducing the scheme on which we will focus, the so-called *computed torque* method of manipulator control.

Here we state some assumptions that will hold throughout this book.

(1) Manipulators are modeled as jointed rigid bodies. Link flexibility will not be addressed as such. Often the analysis will provide for a bounded "disturbance torque," T_d, acting at the joints. This may in some circumstances represent some of the torques attributable to flexibility, but not in any rigorous sense, because T_d will be assumed to be uncorrelated with the manipulator state, an assumption not necessarily true for torques caused by flexible bending modes in the linkages.

(2) Continuous-time analysis is employed. Discrete-time sampling and control effects will not be addressed in the analysis. On the other hand, some numerical simulations do include discrete sampling and control effects. Also, actual experiments with a physical manipulator will be

presented, for which, obviously, the control was performed by a computer in discrete time. Hence, analysis is performed in continous time, and simulations and experiments are used to verify empirically that the theory is implementable.

(3) Manipulators under discussion will be assumed to be open kinematic chains with revolute joints. With a very small amount of effort, all analyses performed can be extended to open chains with mixed revolute and prismatic joints, and with somewhat more effort, results can be extended to systems that contain closed kinematic chains.

(4) The desired manipulator joint trajectories will be assumed to be known, including first and second derivatives. Such trajectories could be computed by any of several well-known methods. We will assume that the desired trajectories are smooth, meaning that desired angular velocities and angular accelerations are bounded.

These assumptions will be in effect in the sequel unless otherwise stated.

2.2 Structure of the Manipulator Dynamic Equations

This section discusses the structure of the dynamic equations of motion for a mechanical manipulator. Results from this section will be useful throughout the remainder of the book.

The manipulator is modeled as a set of n moving rigid bodies connected in a serial chain with one end fixed to the ground and the other end free (Figure 2.1). The bodies are jointed together with revolute joints, there is a torque actuator and friction acting at each joint. The vector equation of motion of such a device can be written in the form [42]

$$T = M(\Theta)\ddot{\Theta} + V(\Theta,\dot{\Theta}) + F(\dot{\Theta}) + G(\Theta) + T_d, \tag{2.1}$$

where T is the $n \times 1$ vector of joint torques supplied by the actuators, and Θ is the $n \times 1$ vector of joint positions, with $\Theta = [\theta_1, \theta_2, \ldots, \theta_n]^T$. The matrix, $M(\Theta)$, is an $n \times n$ matrix, sometimes called the manipulator mass matrix. The vector $V(\Theta, \dot{\Theta})$ represents torques arising from centrifugal and Coriolis

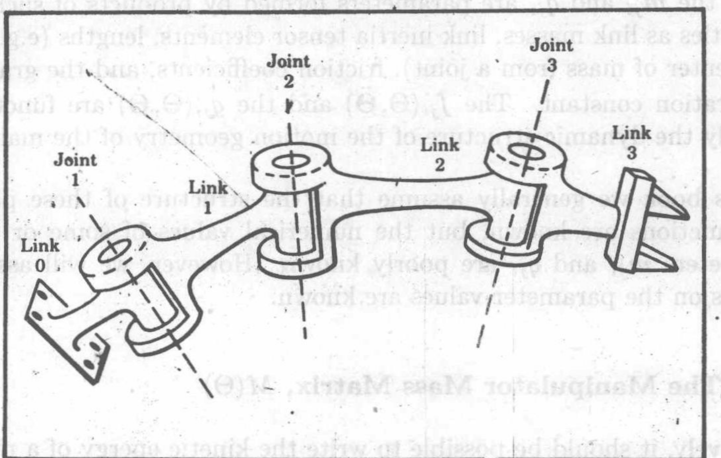

Figure 2.1 *An Articulated Chain of Rigid Bodies*

forces. The vector $F(\dot{\Theta})$ represents torques due to friction acting at the manipulator joints. The vector $G(\Theta)$ represents torques due to gravity, and T_d is a vector of unknown signals due to unmodeled dynamics and external disturbances.

Also, we will sometimes write the dynamics in the more compact form

$$T = M(\Theta)\ddot{\Theta} + Q(\Theta, \dot{\Theta}) + T_d, \tag{2.2}$$

where the vector $Q(\Theta, \dot{\Theta})$ represents torques arising from centrifugal, Coriolis, gravity, and friction forces.

It will be convenient from time to time to write the vector of velocity terms in the matrix–vector product form:

$$V(\Theta, \dot{\Theta}) = V_m(\Theta, \dot{\Theta})\dot{\Theta}, \tag{2.3}$$

where the subscript m stands for "matrix."

The jth element of (2.2) can be written in the sum-of-products form

$$\tau_j = \sum_{i=1}^{u_j} m_{ji} f_{ji}(\Theta, \ddot{\Theta}) + \sum_{i=1}^{v_j} q_{ji} g_{ji}(\Theta, \dot{\Theta}) + \tau_{dj}, \tag{2.4}$$