INTELLIGENT ASSEMBLY SYSTEMS

WORLD SCIENTIFIC SERIES IN ROBOTICS AND INTELLIGENT SYSTEMS

Editor-in-Charge: C J Harris (*University of Southampton*)

Advisor: T M Husband (*University of Salford*)

- Vol. 1: Genetic Algorithms and Robotics — A Heuristic Strategy for Optimization (*Y Davidor*)
- Vol. 2: Parallel Computation Systems for Robotics: Algorithms and Architectures (*Eds. A Bejczy and A Fijany*)
- Vol. 3: Intelligent Robotic Planning Systems (*P C-Y Sheu and Q Xue*)
- Vol. 4: Computer Vision, Models and Inspection (*A D Marshall and R R Martin*)
- Vol. 5: Advanced Tactile Sensing for Robotics (*Ed. H R Nicholls*)
- Vol. 6: Intelligent Control: Aspects of Fuzzy Logic and Neural Nets (*C J Harris, C G Moore and M Brown*)
- Vol. 7: Visual Servoing: Real-Time Control of Robot Manipulators Based on Visual Sensory Feedback (*Ed. K Hashimoto*)
- Vol. 8: Modelling and Simulation of Robot Manipulators: A Parallel Processing Approach (*A Y Zomaya*)
- Vol. 9: Advanced Guided Vehicles – Aspects of the Oxford AGV Project (*S Cameron and P Probert*)
- Vol. 10: Cellular Robotics and Micro Robotic Systems (*T Fukuda and T Ueyama*)
- Vol. 11: Recent Trends in Mobile Robots (*Ed. Y F Zheng*)
- Vol. 12: Intelligent Assembly Systems (*Eds. M Lee and J J Rowland*)

Forthcoming:

- Vol. 13: A Methodology of Systems Integration for Intelligent Autonomous Systems (*R J C Fraser and C J Harris*)
- Vol. 14: Intelligent Supervisory Control Based on Qualitative Bond Graph Reasoning (*D A Linkens and H Wang*)
- Vol. 15: Neural Adaptive Control Technology (*Eds. R Żbikowski and K J Hunt*)
- Vol. 16: Advances in Robotics & Automation for Hazardous Environment (*Eds. P Lever and F Y Wang*)

World Scientific Series in Robotics and Intelligent Systems – Vol.12

INTELLIGENT ASSEMBLY SYSTEMS

Editors

M H LEE & J J ROWLAND

Centre for Intelligent Systems
Department of Computer Science
University of Wales
Aberystwyth

World Scientific

Singapore • New Jersey • London • Hong Kong

Published by

World Scientific Publishing Co Pte Ltd

P O Box 128, Farrer Road, Singapore 912805

USA office: Suite 1B, 1060 Main Street, River Edge, NJ 07661

UK office: 57 Shelton Street, Covent Garden, London WC2H 9HE

Library of Congress Cataloging-in-Publication Data

Intelligent assembly systems / editors, M. H. Lee, J. J. Rowland.

p. cm. -- (World Scientific series in robotics and intelligent systems ; vol. 12)

Includes bibliographical references and index.

ISBN 981022494X

1. Robots, Industrial. 2. Assembly-line methods -- Automation.

I. Lee, M. H. II. Rowland, J. J. III. Series.

TS191.8.I5564 1995

670.42'72--dc20

95-41157

CIP

British Library Cataloguing-in-Publication Data

A catalogue record for this book is available from the British Library.

Copyright © 1995 by World Scientific Publishing Co. Pte. Ltd.

All rights reserved. This book, or parts thereof, may not be reproduced in any form or by any means, electronic or mechanical, including photocopying, recording or any information storage and retrieval system now known or to be invented, without written permission from the Publisher.

For photocopying of material in this volume, please pay a copying fee through the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, Massachusetts 01923, USA.

Printed in Singapore.

Preface

This book deals with a key area of industrial robotics that offers great potential benefit to manufacturing industry – the automation of small batch assembly. Assembly can be defined as an ordered sequence of physical handling tasks in which discrete parts and components are brought together and joined or mated to form a specified configuration. Assembly is an operation that gives enormous added value and converts components worth pence into products sold for many dollars. It is at first surprising, then, that such an important manufacturing process, that can take up to 30% of the manufacturing cost of a product, is still mainly performed by hand. If high levels of quality are to be achieved then assembly by human operator becomes expensive because much verification and inspection is needed to compensate for human inconsistencies. This is the case for small and medium batch sizes; for very large batches the technology of mass production has a long and successful history of sophisticated automation – so why is small batch assembly so difficult to automate?

Programming a robot to perform a repetitious cycle of spatial movements is very easy – it consists of a set of “move” commands with associated spatial locations. However, this superficial view, which is all that is seen by the casual observer, hides an enormous amount of design and development work which has to be undertaken in order to provide tolerance to the inevitable minor variations in the handled parts and to eliminate the possibility of consequent failures. Even when all the production engineering has been completed, the daily operation of a working robot system will be backed up by many hidden contingency plans: plans that it is hoped will never be needed! We have observed, in industrial systems, that as much as 90% of the control programs are concerned with error contingencies, the remaining 10% being the normal operating program.

To the uninitiated it might seem relatively easy to create an autonomous intelligent assembly system by simply coupling together modern software systems (such as expert systems and other techniques from Artificial Intelligence research) with the industrial robots that are readily available commercially. However the difficulties involved here stem not from technology but from our meagre understanding of the nature of the task and the inaccessibility of the knowledge we humans employ when we perform assembly. The vital word here is *flexibility* – a human characteristic that we take for granted, but which is so hard to reproduce in machines. Small batches, almost by definition, demand very flexible assembly agents. Consider a typical list

of characteristics for automated flexible assembly (with the human counterpart in parentheses):

- quick change of product (adaptable)
- ease of programming and reprogramming (trainable)
- minimal product-specific tooling (dextrous)
- resilience to manufacturing tolerances (compliant and skillful)
- error detection and recovery (observant and conscientious)

Each one of these requirements represents a major automation headache for the production engineer, and yet they are so natural for human operators that we hardly even notice them. We hope that the material presented here will illustrate some of the subtle difficulties that only come to light through experience and that our suggestions for dealing with them will help others when trying to engineer such systems.

Another awkward aspect of robotics research, that will be familiar to every researcher who has been faced with the problem of interfacing incompatible pieces of equipment in order to build an assembly cell, is the difficulty of building an effective software environment with powerful tools that enable rapid development of assembly systems. Commercial robotics systems are very manufacturer-dependent. There are few standards, with a great range of software and hardware mixtures available. Consequently, a great deal of time can be spent in simply acquiring and building the workcell, including configuring and writing software, before any useful work on the application can begin. In addition, the experience of building such workcells is hardly ever captured or reported and so the know-how gained is not passed on and others repeat the exercise, from scratch, time and time again. The work presented here addresses this issue and suggests ways in which the difficulties may be at least reduced through clean architectural designs and specific structures for interfacing and controlling sensory-actuation systems.

Perhaps the most profound difficulty with robotics work, not shared by other computing applications, is that the physical environment is an essential part of the system and cannot be easily simplified, unlike some industrial control problems, for example, where a few equations can approximate to the operating regime. This is because assembly occurs in a discrete and open world rather than in a continuous and finite domain. Steven Pinker has remarked that language and DNA structures are so much more powerful than continuous variable systems because they offer infinite variability by being based on discrete elements with unlimited combinatorial complexity. The discrete objects and events in assembly also introduce opportunities for similar complexity, as seen in the enormous range of possibilities for errors to occur. Of course, engineering environments are simpler and much more constrained than is the physical world in general but they still have enough variability and

openness that we must not forget the ever present opportunities for disturbances, interference, uncertainty and noise.

Two current approaches to the problem of dealing with the environment are (a) prediction and planning, a classical Artificial Intelligence method, and (b) the behaviour based approach, which rejects all planning and deals with events reactively through direct coupling between sensors and actuators. In the first case the control system contains models of the world which guide planning decisions and are modified to reflect changing situations. In behaviour based systems there is no plan other than in the actual structure of the machine which can be seen to be predisposed towards certain behavioural goals. Different behaviours can be designed for different goals and are often very tolerant to noise and disturbances. Such systems are very robust but it seems unlikely that multiple agents with their own goals would achieve an assembly task without some co-ordinating structure, at least not in a reliable and efficient manner as required in industry. On the other hand, planning systems find it hard to take account of uncertainties and assume perfect information and control. Our work described here has developed a task centred approach that views the details of sensory-actuator control as subservient to the application requirements of the assembly task. Using tasks to analyse assembly problems does not preclude either goal directed behaviour or planning and prediction but, we believe, offers a fundamentally better framework for understanding and implementing flexible assembly systems.

The work reported in this book was all undertaken in the Centre for Intelligent Systems (previously known as the Artificial Intelligence and Robotics Research Group) in the Department of Computer Science at the University of Wales, Aberystwyth. All the work in the Centre involves extensive industrial collaboration and several companies devoted considerable resources to the work described. The chapters are based on projects that range from PhD work to large research council, and other, grants including a pan-European project (EUREKA/FAMOS) that involved nine academic and industrial organisations. These projects cover a period of ten years, up to 1992, and are selected to show the structured growth and development of certain research ideas, running as themes throughout, from early concepts through to full demonstrators that prove feasibility and commercial potential.

The field is interdisciplinary in that it covers at least three different domains: mechanical and production engineering, concerning the handling and manipulating of mechanical parts, electrical systems involved in the design and control of sensing and actuation, and software engineering which provides the environment for driving the system, monitoring execution, interpreting assembly activity, diagnosing errors and supporting programming and configuration.

The intended readership includes production engineers, manufacturing managers, and researchers who are concerned with manufacturing or are interested in applications of Artificial Intelligence in industry. Also the text could be used as case study material for various educational programmes such as postgraduate courses in advanced automation. We believe our treatment of robotics, sensing and software

design highlights important issues that will be of value to all those interested in the application of computers to difficult but important real life problems.

Chapter 1 sets the scene with an early investigation into the design and construction of a knowledge based robot supervisor. This innovative project developed an understanding of the nature and causes of errors and consequent task failure in industrial assembly applications. It explored the use of sensory monitoring and the production of diagnostic aids, and produced a framework that represents a methodology for assembly system design. A successful prototype system was implemented that demonstrated sensor actuator integration and incorporated effective error diagnosis. Although the technology available at that time has now been hugely superseded (as seen from the equipment used) this project provided many ideas and insights that have illuminated issues and set research agendas for later work. The value of chapter 1 should be seen as an exercise in exposing core problems and issues; some of the main ones can be listed as follows:

Capture of explicit workcell knowledge. There is much implicit information embedded in engineering systems. The products themselves do not disclose how they were designed or manufactured, and only part of this knowledge is recorded in manufacturers' documents – the rest is held as 'know-how' in the heads of the engineering staff. Chapter 1 shows how important assembly knowledge can be captured as explicit information and factored into three major groups: events, equipment and expectations.

A unified framework. The value of a structured, uniform, framework is considerable. It offers modularity which is so important to support reuse. This facilitates opportunities for reuse of knowledge across applications, as well as reuse of supervisory structures to enhance the control task. Also the integration of sensing and action, which is an important modern theme, is promoted by such an approach.

Higher levels of programming. A typical assembly cell may contain complex elements from different manufacturers and there will be a clutter of hardware and different software systems running mixed levels of code in various languages. To advance from this messy situation, we must promote the use of higher levels of assembly programming that allow us to describe workcell execution in terms of the task rather than in terms of specific equipment activity.

Sensory monitoring of context. The concept of sensory signatures, in which the *relevance* of sensory data is emphasised, is seen as an important approach for controlling sensory input and facilitating interpretation.

Explicit knowledge of errors. Error handling is vital for assembly systems, particularly for automated or autonomous systems. The first stage in error handling is error detection, which implies a sensory monitoring regime; then

follows diagnosis and possibly error recovery. This work shows the value of the incorporation of explicit information for error treatment.

Chapter 2 builds on the principles revealed in the previous chapter and concentrates on the problems of configuring and re-configuring a workcell for small batch production. As batches get smaller, so the amount of time and expertise spent on configuring the system for the next batch becomes dominant and must be reduced. Chapter 2 investigates how this can be done through the use of co-operating design tools forming the basis of an intelligent assistant for the small batch programmer. The separate co-operating agents support assembly sequence planning, for conversion of assembly processes into executable tasks, and the modelling of errors and uncertainties in the workcell. The work features a task description language, a system for reasoning about assembly sequences, and a stochastic package that models uncertainty and predicts errors. Communication between the agents was achieved via a novel method known as the 'knowledge bus'. This work involved experimentation on a real industrial workcell, this time using modern equipment but without vision sensing and using simpler industrial sensing methods, pneumatic devices and programmable logic controllers.

Chapter 3 deals with a different area – that of reasoning about mechanical components and their interactions. Our early work kindled an interest in the wider issues of diagnosis as an essential aspect of error recovery. In diagnosis, complex reasoning must take place if we are to deduce the causes of a failure situation from a few sparse symptoms. In general, an *ad hoc* collection of symptom-cause rules in an expert system is inadequate for such tasks and our search for better methods led into investigations of model-based systems using the new techniques of qualitative reasoning. This is now a very active research area with a range of new industrial applications in evidence. Chapter 3 shows how detailed (non-numerical) modelling can provide some of the explicit knowledge necessary to allow assembly systems to reason about their behaviour.

Chapter 4 deals with the sense of touch. The senses of sight and sound are nowadays easily incorporated into computer systems, with a range of available software packages for image processing and signal recognition. However the sense of touch has not been very well developed in robotics mainly because of the inadequacies of the sensory devices currently available. Many ingenious research prototypes have been built but one of the main difficulties with tactile sensing is that considerable wear is associated with industrial processes involving contact and consequently the tactile surface must be very sturdy. The ideal sensor would consist of a flexible skin with high resolution sensing, yet with very robust physical properties.

Nowadays robots can be fitted with (a) *internal sensors* for measuring, for example, velocity, acceleration, stress and force and (b) *external sensors* such as vision, ultrasonic and infrared devices. We believe tactile sensing is the missing *surface based* sensory modality that represents a major gap in the armoury of sensors available to a robotic system. Tactile sensors can feel textures, shapes and grasping patterns – exciting new kinds of robotic sensory data. Although some of our robot

workcells have used tactile sensing, and we have developed new high resolution tactile sensors, this work is not reported here. Instead, chapter 4 introduces tactile sensing techniques, describes their features and discusses the basic issues; it illustrates the potential, and the importance, of tactile sensing for future assembly systems.

Chapter 5 returns to some of the themes developed in chapters 1 and 2, and describes further work on sensor integration for flexible assembly. The concept of virtual sensors is developed and their rôle in supervisory systems is described and discussed. A combination of the emerging virtual sensor concepts and the tactile sensor development led to an interest in the issues of virtual actuators, where actuation and sensing could be combined to form 'intelligent' actuation devices that can present uniform control and sensing interfaces to higher level controllers. These ideas are illustrated through the design of an experimental gripper controller which is based on the principles of virtual sensing and the integration of sensing and action through tight coupling in the software modules.

Many of the ideas in earlier chapters are in evidence in chapter 6 where a large demonstrator of a fully integrated flexible assembly machine is described. This novel industrial prototype was built by a multinational multidisciplinary team and was targeted on market sector requirements. Chapter 6 discusses the design and implementation of the task specification, execution and sensor integration software for the machine, with particular emphasis on identifying the issues that underlie the creation of versatile and robust software for a production version of such a machine. The machine's control architecture was based on multiple transputers and this had a major influence on the software design strategy that we adopted.

All aspects of configuration and operation of this machine, i.e. the design of assembly sequences, parts feeding, parts transport, mating and error handling, are processed by the system software whose user interface was designed for use by production staff, rather than robotics specialists. The system shows how the ideas of sensory signatures can be exploited, how the concepts of high level programming become developed into supervisory user-friendly menu-based systems and how the ideas of task context, that were expounded earlier, become important constraints in the control and simplification of the process. A key lesson learned from this prototype machine is that a valuable structure is gained when sensory systems, supervisory software, and user interfaces are designed as communicating independent modules based on a coherent and uniform machine architecture.

Chapter 7 is a theoretical discussion on robotics research. Much of robotics research is driven by experimental application engineering, at the expense of the direct development of scientific theory. Sound theoretical grounding must be present if progress is to be made, so that engineers are more able to design quality systems to match given performance criteria. To this end, a formal model-based approach to the design sensory robot systems is introduced. It addresses fundamental questions about the rôle of sensors, for example identifying when sensory data are **useful, relevant, necessary, or critical** for a task. By shifting to a *task oriented* view, a formal model is developed which has resonances with some of the earlier ideas that

were expressed informally. We hope the task-based view will point the way forward in the design and construction of assembly machines. It will not solve all problems but at least will raise the right sort of questions to clarify the feasibility or otherwise of a particular task being solved by a particular arrangement of assembly system.

To summarise, this book develops three main concepts: **a task-centred approach** to assembly which we hope will lead to a more scientific understanding of robotics issues, **explicit reasoning** methods for fault diagnosis and error handling, and **sensor-actuator integration**, including virtual sensors. We hope readers will find these ideas as stimulating as we do and that they will be encouraged to apply them in their own field.

Acknowledgements

Much of the work described here was funded through various grants from the Engineering and Physical Sciences Research Council (previously the Science and Engineering Research Council). We particularly acknowledge the help of the regional coordinators from the CDP (previously ACME) directorate for their advice and assistance throughout. We also thank Mr. John Gillis and his staff at the DTI for their help and support in relation to our involvement in EUREKA/FAMOS.

We are grateful for the considerable additional support given by our industrial collaborators. We have worked with many companies; their involvement has been of great benefit and has ensured the industrial relevance of the research. Particular thanks are due to David Mott and Tony Williams who, over several projects (and several companies!) became special friends of the Aberystwyth team.

Many research associates and research students have contributed to this work. In particular we would like to acknowledge the contributions of those who worked as research associates on the projects, in addition to those who are chapter authors; thanks are due to Mark Allan, Yan Cheung, John Hunt, Stephen Kingston, Shahwar Rehman, Vivien Robinson, Kevin Sharp, and Adrian Shaw.

We also thank our colleagues, past and present, for many and varied forms of help and for making the department such an amenable research environment. We thank particularly the secretarial and administrative staff, technicians and computer officers; their work is invaluable but often goes unrecognised.

This book was prepared using L^AT_EX. We are grateful to Howard Nicholls who, with his own book *Advanced Tactile Sensing for Robotics* (World Scientific, 1992), pioneered the particular route to camera ready copy that we have used.

Mark Lee
Jem Rowland
University of Wales
Aberystwyth

July 1995

This page is intentionally left blank

List of Contributors

D.P. Barnes

Department of Electronic and Electrical Engineering,
University of Salford, Salford M5 4WT, England, U.K.

N.W. Hardy

Centre for Intelligent Systems, Department of Computer Science,
University of Wales, Aberystwyth SY23 3BD, Wales, U.K.

M.H. Lee

Centre for Intelligent Systems, Department of Computer Science,
University of Wales, Aberystwyth SY23 3BD, Wales, U.K.

H.R. Nicholls

Industrial Research Ltd.,
175 Roydvale Avenue, Bishopdale, Christchurch, New Zealand.

C.H. Orgill

71 Dale Road, Buxton, Derbyshire SK17 6NE, England, U.K.

J.J. Pinkava

Pixar,
1001 W. Cutting Blvd., Richmond, California 94804, U.S.A.

C.J. Price

Centre for Intelligent Systems, Department of Computer Science,
University of Wales, Aberystwyth SY23 3BD, Wales, U.K.

J.J. Rowland

Centre for Intelligent Systems, Department of Computer Science,
University of Wales, Aberystwyth SY23 3BD, Wales, U.K.

This page is intentionally left blank

Summary of Contents

1. Knowledge-based Robot Supervision D.P. Barnes and N.W. Hardy	1
2. Software Tools for Robot Supervisors C.H. Orgill	47
3. Qualitative Reasoning for Diagnosis of Mechanical Systems C.J. Price	83
4. Tactile Sensing for Automation H.R. Nicholls	131
5. Sensing and Actuation for Flexible Assembly J.J. Rowland and H.R. Nicholls	165
6. Supervisory and Sensing Software for an Integrated Flexible Assembly machine N.W. Hardy, J.J. Rowland and H.R. Nicholls	181
7. Towards a Theory of Sensory Robotics J.J. Pinkava	209

This page is intentionally left blank

Contents

Preface	v
1 Knowledge-based Robot Supervision	1
1.1 Introduction	1
1.1.1 The Perceived Problems	2
1.2 Why Use IKBS for Robot Workcells?	3
1.2.1 The Implicit Knowledge Approach	6
1.2.2 Robot User Requirements	6
1.2.3 A Proposed Knowledge-based Approach	8
1.3 What is Supervisory Knowledge?	9
1.3.1 Events	10
1.3.2 Equipment	11
1.3.3 Expectations	13
1.4 Knowledge Representation Paradigm	15
1.5 AFFIRM Knowledge Representation	16
1.5.1 Event Frames	17
1.5.2 Equipment Frames	21
1.5.3 Expectation Frames	24
1.5.4 Frame Association	26
1.6 The Frame Hierarchy	27
1.6.1 A Useful Analogy	32
1.7 AFFIRM Knowledge Manipulation	33
1.8 Implementation	35
1.8.1 The Run Time System	35
1.8.2 The AFFIRM Suite	36
1.9 The Achievements of AFFIRM	36
1.9.1 AFFIRM's Intelligent Knowledge Base Design	36
1.9.2 "Programming" AFFIRM	38
1.9.3 AFFIRM as a Supervisory System	38
1.9.4 AFFIRM's Use in Error Diagnosis	41
1.9.5 Possible Enhancements	43
References	43

2	Software Tools for Robot Supervisors	47
2.1	Introduction	47
2.1.1	Supervision	48
2.2	The Assembly Workcell	49
2.2.1	The Actuators and Sensors	49
2.3	TDL-MODEL, TDL-SEQ, WCE	51
2.3.1	TDL-MODEL and TDL-SEQ	51
2.3.2	WCE	61
2.4	The CRITIC Agent	61
2.4.1	CRITIC's Object Model	61
2.4.2	The Planner	66
2.4.3	Causality: Manufacturing Uses?	68
2.4.4	Plan Selection Heuristics	69
2.5	The FATE Agent	69
2.5.1	The Entropic Model	70
2.6	Putting it all Together: <i>The Knowledge Bus</i>	75
2.6.1	The kBus in Use	78
2.7	Conclusions	79
	References	80
3	Qualitative Reasoning for Diagnosis of Mechanical Systems	83
3.1	Introduction	83
3.2	Model-based Reasoning	84
3.2.1	Goals for the Representation	84
3.2.2	Modelling Mechanical Devices	85
3.2.3	Qualitative Reasoning	89
3.2.4	Conclusions about Modelling Mechanical Devices	91
3.3	Representation of the Device Structure	91
3.4	Animating the Representation	96
3.4.1	Describing the Behaviour of Components	99
3.4.2	Example One: Pressing on the Button	101
3.4.3	Example Two: Pulling on the Latch	103
3.4.4	Using Single State-steps	104
3.5	Implementing the Device Simulator	106
3.5.1	Initializing the Device Simulator	107
3.5.2	Performing a Single State Change	108
3.5.3	Creating a State Map	108
3.6	Evaluation of the Simulator	111
3.6.1	Limitations of the Simulator	112
3.6.2	Fitness for Purpose	116
3.6.3	Other Mechanical Device Simulation Work	117
3.7	Predicting Device Behaviour	121
3.7.1	Model-based Prediction	122

3.7.2	Performing Prediction with the RIDDL Simulator	123
3.7.3	Limitations of the RIDDL System for Prediction	124
3.7.4	Final Comments on Prediction	126
	References	127
4	Tactile Sensing for Automation	131
4.1	Sensing and Robotics	131
4.2	A Sensing Taxonomy	132
4.3	What is Tactile Sensing?	133
4.4	Some Attributes of the Tactile Domain	134
4.5	Haptic Sensing	137
4.6	Tasks for Tactile Sensors	138
4.7	Processing and Using Tactile Sensor Data	140
4.8	Basic Tactile Sensing Issues	140
4.8.1	Parameters of Contact	140
4.8.2	Grasping and Tactile Sensing	141
4.8.3	Stable Grasping with Intrinsic Tactile Sensing	144
4.9	Object Recognition Using a Single Tactile Image	146
4.10	Active Tactile Sensing	150
4.10.1	Shape Description without Specific Models	151
4.10.2	Model Matching	152
4.10.3	Haptic Sensing and Haptic Perception	154
4.11	Conclusion	157
	References	157
5	Sensing and Actuation for Flexible Assembly	165
5.1	Origins	165
5.2	The Virtual Sensor Concept	166
5.3	Design Issues	168
5.4	Supervisors and Virtual Sensing	170
5.5	Integrating Sensors with Actuators	171
5.5.1	The Command and Status Interface to the Controller	173
5.5.2	Active Sensing	176
5.6	Summary: Capabilities of Virtual Sensors	177
	References	178
6	Supervisory and Sensing Software for an Integrated Flexible Assembly Machine	181
6.1	Background	181
6.2	The Assembly Machine	182
6.3	Architecture	184
6.4	Sensor Integration	186
6.4.1	The Virtual Sensors	187
6.4.2	The Sensor Command Language	188

6.4.3	Implementation and Testing	192
6.5	The Supervisor	194
6.5.1	Classes of Generic Task	196
6.5.2	List of Classes	197
6.5.3	Task Motion Parameters	199
6.6	An Example	200
6.6.1	Inventory Data	201
6.6.2	Task Specification	201
6.7	Simulation and Task Execution	202
6.7.1	Command Acknowledgement and Error Handling	203
6.8	In Conclusion	204
6.8.1	The Outcome	205
	References	206

7 Towards a Theory of Sensory Robotics 209

7.1	Introduction	209
7.2	Research Trends in Sensory Robotics	210
7.2.1	Research Categories	211
7.2.2	A Biased View of the Research	212
7.2.3	Summary of Research Classification	216
7.3	The Case for a Theoretical Bias	216
7.3.1	Difficulties with Theory	217
7.3.2	Benefits of Formal Theory	217
7.3.3	Appropriate Theoretical Bias	218
7.3.4	Summary of the Case for Theoretical Bias	221
7.4	Elements of a Robot Sensing Problem	222
7.4.1	Task	223
7.4.2	Mechanism	224
7.4.3	Behaviour	225
7.4.4	Sensing	227
7.4.5	Control System – the Abstract Robot	228
7.4.6	Summary of General Model	228
7.5	Conclusion	229
7.5.1	Directions for Theoretical Work	230
	References	230

Index 235

List of Figures

1.1	Case study system configuration.	4
1.2	Simple “pick and place” example.	5
1.3	Simplified VAL example program.	7
1.4	Some typical assembly events (after Nevins & Whitney, 1979).	11
1.5	Example of robot level event knowledge.	12
1.6	Example of how sensor equipment might relate to pick and place events.	14
1.7	Example of equipment frames forming an associative net.	18
1.8	An event frame.	19
1.9	An object frame.	22
1.10	A sensor hardware frame.	23
1.11	A signature frame.	25
1.12	A location frame.	26
1.13	An overview of AFFIRM frame type associations.	27
1.14	The event frame hierarchy.	29
1.15	An overview of sensor, command and interface event frames.	31
1.16	Conceptual analogue of event frame hierarchy.	33
1.17	A functional overview of the AFFIRM run time system.	34
1.18	The AFFIRM software suite.	37
1.19	Run time agenda example.	39
1.20	Key to ARTS agenda.	40
1.21	Run time agenda example 2.	42
2.1	The supervisory package.	48
2.2	Internal structure of TDL.	52
2.3	Internal structure of CRITIC.	62
2.4	Internal structure of FATE.	72
2.5	FATE’s diagnostic engine.	74
3.1	Weights on a balance.	90
3.2	Simple doorlock for sliding door.	92
3.3	Declarations for doorlock device.	93
3.4	3D axes for representation.	94
3.5	Simple device history.	104
3.6	Part of an attainable envisionment.	105

3.7	Network for doorlock.	108
3.8	Force propagation for doorlock.	109
3.9	Screen dump of part of doorlock state map.	110
3.10	Excerpt from doorlock state map.	111
3.11	Force propagation ambiguity.	113
3.12	Some possible mechanical devices.	117
4.1	Types of sensor (reproduced from Lee (1989)).	132
4.2	Example tasks for tactile sensors.	138
5.1	Virtual sensing: an example organised by sensory function.	169
5.2	Block diagram of the gripper controller.	173
5.3	Sensory feedback for gripper control and task progress.	174
5.4	Data from the active spring sensor.	177
6.1	A cutaway sketch of the InFACT machine.	183
6.2	The InFACT machine software hierarchy.	185
6.3	The hierarchy of virtual sensor classes.	189
6.4	The life cycle of a state sensor	190
6.5	The life cycle of an event sensor	191
6.6	Sensor integration architecture.	193
6.7	The railroad diagram.	196
6.8	A sequence of tasks.	199
6.9	A single task (not a Move).	200
7.1	Dependencies between elements of a robot system	222
7.2	A pick-and-place task	224
7.3	A point mechanism in widget space	226
7.4	Different mechanisms, same task	227
7.5	Elements of the general model	228