# Cognitive Systems Monographs

## Volume 8

Editors: Rüdiger Dillmann · Yoshihiko Nakamura · Stefan Schaal · David Vernon

Henrik Iskov Christensen,
Geert-Jan M. Kruijff,
and Jeremy L. Wyatt (Eds.)

# Cognitive Systems

Springer

**Rüdiger Dillmann,** University of Karlsruhe, Faculty of Informatics, Institute of Anthropomatics, Humanoids and Intelligence Systems Laboratories, Kaiserstr. 12, 76131 Karlsruhe, Germany

**Yoshihiko Nakamura,** Tokyo University Fac. Engineering, Dept. Mechano-Informatics, 7-3-1 Hongo, Bukyo-ku Tokyo, 113-8656, Japan

**Stefan Schaal,** University of Southern California, Department Computer Science, Computational Learning & Motor Control Lab., Los Angeles, CA 90089-2905, USA

**David Vernon,** Khalifa University Department of Computer Engineering, PO Box 573, Sharjah, United Arab Emirates

## Editors

Henrik Iskov Christensen
Georgia Tech
RIM@GT
801 Atlantic Dr.
Atlanta, GA 30332
USA
E-mail: hichristensen@gmail.com

Jeremy L. Wyatt
School of Computer Science
University of Birmingham
Birmingham B15 2TT
UK
E-mail: jlw@cs.bham.ac.uk

Geert-Jan M. Kruijff
DFKI GmbH
Stuhlsatzenhausweg 3
66123 Saarbrücken
Germany
gj@dfki.de

ISBN 978-3-642-11693-3    e-ISBN 978-3-642-11694-0

DOI 10.1007/978-3-642-11694-0

Cognitive Systems Monographs    ISSN 1867-4925

Library of Congress Control Number: 2010925225



*Typeset & Cover Design:* Scientific Publishing Services Pvt. Ltd., Chennai, India.

Printed on acid-free paper

5 4 3 2 1 0

springer.com

**Quotes from the CoSy Science Advisors**

While there is still a long way to come close to the objective of the European Commission Cognitive Systems initiative "to construct physically instantiated or embodied systems that can perceive, understand, ... and interact with their environments and evolve in order to achieve human-like performance" this book is about one of the funded projects in this initiative. It gives and excellent insight into the challenges and benefits of working in an large interdisciplinary team to better understand the human mind and in order to build intelligent machines.

Heinrich Bülthoff, MPIK

One of the great challenges of the 21st century is to build a robot that can perceive and act within its environment and communicate with people, while also exhibiting the cognitive capabilities that lead to performance like that of people. This book reports on the European Union project on Cognitive Systems. It offers detailed explanations of the exciting progress made on this challenge and serves as a foundation for the science of Cognitive Systems in the next part of this century.

Candance Sidner, BAE Systems

# Preface

The present volume is a report on the research results generated by the project "Cognitive Systems for Cognitive Assistants" (CoSy), which was sponsored by the European Commission during the period 2004–2008.

The CoSy team was assembled to study the problem of embodied cognitive systems for domestic service tasks such as guidance, fetch and carry, etc. The main aim of the project has been to study the core technologies needed to build such systems rather than mere application development. The key competencies needed are: systems architectures, scalable knowledge representation, adaptive embodiment,categorical perception, planning and error recovery, learning, and situated dialog systems. All of these aspects were studied in the context of CoSy and exemplified using two "demonstrator scenarios" that were conceived to allow studies / evaluation in an integrated context.

The volume is organized into 4 parts. The introduction outlines the overall problem domain and the CoSy approach to the problem. The second part contains a number of chapters that detail progress on topical problems across architectures, perception, learning, planning and dialog systems. These competencies were integrated into systems as described in the third part of the book. The final section provides a perspective on the results obtained and considers some possible issues for future research.

The project has published extensively throughout its life and links to publications can be found at the project web facility `www.cognitivesystems.org`, where copies of associated project deliverables also can be retrieved. The CoSy web facility contains also a page with material that supplements the book. The page has pointers to published material, associated datasets, videos and open software. The electronic version of the book also has embedded links to the web facility and published papers. I.e., referenced material published by the consortium can be accessed through embedded links.

The consortium would like to express our gratitude for the support the European Commission has provided for this research. In addition we are grateful for the guidance and feedback we have received from our scientific advisors: Prof. Heinrich Bülthoff - MPIK, Prof. Benjamin Kuipers - UT Austin, Dr. Candy Sidner - BAE Systems. We are also grateful for the support from the project reviewers: Prof. Igor Alexander - Univ. College London, Prof. Mark Steedman - Univ. of Edinburgh, Prof. John Tsotsos - York Univ. and Prof. Oliver Brock, UMASS. Finally, we appreciate the guidance from the associated EU project officer Cecile Huet.

Atlanta, Saarbrücken & Birmingham
November 2009

Henrik Iskov Christensen
Geert-Jan M. Kruijff
Jeremy L. Wyatt

# Contents

**Part I: Introduction**

**1 Cognitive Systems Introduction**
*Henrik I. Christensen, Aaron Sloman, Geert-Jan M. Kruijff, Jeremy L. Wyatt* . . . . . 3

**Part II: Component Science**

**2 Architecture and Representations**
*Nick Hawes, Jeremy L. Wyatt, Mohan Sridharan, Henrik Jacobsson, Richard Dearden, Aaron Sloman, Geert-Jan M. Kruijff* . . . . . 51

**3 The Sensorimotor Approach in CoSy: The Example of Dimensionality Reduction**
*David Philipona, J. Kevin O'Regan* . . . . . 95

**4 Categorical Perception**
*Mario Fritz, Mykhaylo Andriluka, Sanja Fidler, Michael Stark, Aleš Leonardis, Bernt Schiele* . . . . . 131

**5 Semantic Modelling of Space**
*Andrzej Pronobis, Patric Jensfelt, Kristoffer Sjöö, Hendrik Zender, Geert-Jan M. Kruijff, Oscar Martinez Mozos, Wolfram Burgard* . . . . . 165

**6 Planning and Failure Detection**
*Michael Brenner, Christian Plagemann, Bernhard Nebel, Wolfram Burgard, Nick Hawes* . . . . . 223

**7 Multi-modal Learning**
*Danijel Skočaj, Matej Kristan, Alen Vrečko, Aleš Leonardis, Mario Fritz, Michael Stark, Bernt Schiele, Somboon Hongeng, Jeremy L. Wyatt* . . . . . 265

**8 Situated Dialogue Processing for Human-Robot Interaction**
*Geert-Jan M. Kruijff, Pierre Lison, Trevor Benjamin, Henrik Jacobsson, Hendrik Zender, Ivana Kruijff-Korbayová, Nick Hawes* . . . . . . . . . . 311

**Part III: Integration and Systems**

**9 The PlayMate System**
*Nick Hawes, Jeremy L. Wyatt, Mohan Sridharan, Marek Kopicki, Somboon Hongeng, Ian Calvert, Aaron Sloman, Geert-Jan M. Kruijff, Henrik Jacobsson, Michael Brenner, Danijel Skočaj, Alen Vrečko, Nikodem Majer, Michael Zillich* . . . . . . . . . . 367

**10 The Explorer System**
*Kristoffer Sjöö, Hendrik Zender, Patric Jensfelt, Geert-Jan M. Kruijff, Andrzej Pronobis, Nick Hawes, Michael Brenner* . . . . . . . . . . 395

**11 Lessons Learnt from Scenario-Based Integration**
*Nick Hawes, Michael Zillich, Patric Jensfelt* . . . . . . . . . . 423

**Part IV: Summary and Outlook**

**12 Cross-Disciplinary Reflections: Philosophical Robotics**
*Aaron Sloman* . . . . . . . . . . 441

**13 Lessons and Outlook**
*Henrik I. Christensen* . . . . . . . . . . 485

**Author Index** . . . . . . . . . . 491

## List of Contributors

**Mykhaylo Andriluka**
TU Darmstadt
Multimodal Interactive Systems
Hochschulstrasse 10
D-64289 Darmstadt, Germany
andriluka@cs.tu-darmstadt.de

**Trevor Benjamin**
DFKI GmbH
Saarbrücken, Germany

**Michael Brenner**
Albert-Ludwigs Universität Freiburg
Department of Computer Science
Freiburg, Germany
brenner@informatik.uni-freiburg.de

**Wolfram Burgard**
Albert-Ludwigs Universität Freiburg
Department of Computer Science
Freiburg, Germany
burgard@informatik.uni-freiburg.de

**Ian Calvert**
University of Birmingham
School of Computer Science
Edgbaston
Birmingham, B15 2TT UK

**Henrik I. Christensen**
Georgia Institute of Technology
Robotics and Intelligent Machines
Atlanta, GA 30332-0280
hichristensen@gmail.com

**Richard Dearden**
University of Birmingham
School of Computer Science
Edgbaston
Birmingham, B15 2TT UK
rwd@cs.bham.ac.uk

**Sanja Fidler**
University of Ljubljana
Faculty of Computer and
Information Science
Visual Cognitive Systems Laboratory
Trzasaka 25
SE-1001 Ljubljana, Slovenia
sanja.fidler@fri.uni-lj.si

**Mario Fritz**
TU Darmstadt
Multimodal Interactive Systems
Hochschulstrasse 10
D-64289 Darmstadt, Germany
fritz@cs.tu-darmstadt.de

**Nick Hawes**
University of Birmingham
School of Computer Science
Edgbaston
Birmingham, B15 2TT UK
nah@cs.bham.ac.uk

**Somboon Hongeng**
University of Birmingham
School of Computer Science
Edgbaston
Birmingham, B15 2TT UK
S.Hongeng@cs.bham.ac.uk

**Henrik Jacobsson**
DFKI GmbH
Saarbrücken, Germany
henrikj@dfki.de

**Patric Jensfelt**
Royal Institute of Technology (KTH)
Center for Autonomous Systems
SE-100 44 Stockholm, Sweden
patric@csc.kth.se

**Marek Kopicki**
University of Birmingham
School of Computer Science
Edgbaston
Birmingham, B15 2TT UK
mzs@cs.bham.ac.uk

**Matej Kristan**
University of Ljubljana
Faculty of Computer and
Information Science
Visual Cognitive Systems Laboratory
Trzasaka 25
SE-1001 Ljubljana, Slovenia
matej.kristan@fri.uni-lj.si

**Geert-Jan Kruijff**
DFKI GmbH
Saarbrücken, Germany
gj@dfki.de

**Ivana Kruijff-Korbayova**
DFKI GmbH
Saarbrücken, Germany
ivana.kruijff@dfki.de

**Ales Leonardis**
University of Ljubljana
Faculty of Computer and
Information Science
Visual Cognitive Systems
Laboratory
Trzasaka 25
SE-1001 Ljubljana, Slovenia
ales.leonardis@fri.uni-lj.si

**Pierre Lison**
DFKI GmbH
Saarbrücken, Germany
Pierre.Lison@dfki.de

**Nikodem Majer**
TU Darmstadt
Multimodal Interactive Systems
Hochschulstrasse 10
D-64289 Darmstadt, Germany
majer@cs.tu-darmstadt.de

**Oscar Martinez Mozos**
Albert-Ludwigs Universität Freiburg
Department of Computer Science
Freiburg, Germany
omartine@informatik.
uni-freiburg.de

**Bernhard Nebel**
Albert-Ludwigs Universität Freiburg
Department of Computer Science
Freiburg, Germany
nebel@informatik.uni-freiburg.de

**J. Kevin O'Regan**
Laboratoire Psychologie
de la Perception,
Université Paris
Descartes and CNRS
Paris, France
jkrevin.oregan@gmail.com

**David Philipona**
Laboratoire Psychologie
de la Perception,
Université Paris
Descartes and CNRS
Paris, France

**Christian Plagemann**
Albert-Ludwigs Universität Freiburg
Department of Computer Science
Freiburg, Germany
plagem@informatik.
uni-freiburg.de

**Andrzej Pronobis**
Royal Institute of Technology (KTH)
Center for Autonomous Systems
SE-100 44 Stockholm, Sweden
pronobis@csc.kth.se

**Bernt Schiele**
TU Darmstadt
Multimodal Interactive Systems
Hochschulstrasse 10
D-64289 Darmstadt, Germany
schiele@cs.tu-darmstadt.de

**Kristoffer Sjöö**
Royal Institute of Technology (KTH)
Center for Autonomous Systems
SE-100 44 Stockholm, Sweden
krsj@csc.kth.se

**Danijel Skocaj**
University of Ljubljana
Faculty of Computer and
Information Science
Visual Cognitive Systems Laboratory
Trzasaka 25
SE-1001 Ljubljana, Slovenia
danijel.skocaj@fri.uni-lj.si

**Aaron Sloman**
University of Birmingham
School of Computer Science
Edgbaston
Birmingham, B15 2TT UK
axs@cs.bham.ac.uk

**Mohan Sridharan**
Texas Tech at Abilene
302 Pine Street
Abilene, TX 79601, USA
mohan.sridharan@ttu.edu

**Michael Stark**
TU Darmstadt
Multimodal Interactive Systems
Hochschulstrasse 10
D-64289 Darmstadt, Germany
stark@cs.tu-darmstadt.de

**Jeremy Wyatt**
University of Birmingham
School of Computer Science
Edgbaston
Birmingham, B15 2TT UK
jlw@cs.bham.ac.uk

**Henrik Zender**
DFKI GmbH
Saarbrücken, Germany
zender@dfki.de

# Part I

# Introduction

# 1

# Cognitive Systems Introduction

Henrik I. Christensen[1], Aaron Sloman[2], Geert-Jan Kruijff[3], and Jeremy L. Wyatt[2]

[1] Robotics and Intelligent Machines, Georgia Institute of Technology, Atlanta, Ga. USA
hic@cc.gatech.edu
[2] Intelligent Robotics Lab, School of Computer Science, University of Birmingham, Birmingham, UK
{axs,jlw}@cs.bham.ac.uk
[3] DFKI GmbH, Saarbrücken, Germany
gj@dfki.de

## 1.1 Introduction

The CoSy project was setup under the assumption that the visionary FP6 objective

> *"To construct physically instantiated ... systems that can perceive, understand ... and interact with their environment, and evolve in order to achieve human-like performance in activities requiring context-(situation and task) specific knowledge"*

is far beyond the state of the art and will remain so for many years. From this vision several intermediate targets were defined. Achieving these targets would provide a launch pad for further work on the long term vision.

In particular it has been an objective to advance the *science* of cognitive systems through a multi-disciplinary investigation of *requirements*, *design options* and *trade-offs* for human-like, autonomous, integrated, physical (e.g. robot) systems, including requirements for architectures, for forms of representation, for perceptual mechanisms, for learning, planning, reasoning, motivation, action, and communication.

To validate science progress a succession of increasingly ambitious working systems are constructed to test and demonstrate the ideas. Devising demanding but achievable test scenarios, including scenarios in which a machine not only *performs* some task but shows that it *understands* what it has done, and why, is an integral part of the empirical study of cognitive systems.

In this chapter the basic objectives, expected results and organization of the project will be presented, whereas the remainder of the book present results that have been obtained during the CoSy project. The final chapters of the

H.I. Christensen et al. (Eds.): Cognitive Systems, COSMOS 8, pp. 3–48.
springerlink.com 

book will provide reflections on progress in terms of new insight and major lessons.

## 1.2 Objective of Project

### 1.2.1 The Problem

Despite impressive progress in many specific sub-topics in AI and Cognitive Science, the field as a whole moves slowly. Most systems able to perform complex tasks that humans and other animals can perform easily, for instance robot manipulators, or intelligent advisers, have to be carefully crafted. Whatever intelligence they have could be described as 'insect-like' insofar as they have capabilities that they do not understand, they do not know why they do things one way rather than another, they cannot explain what they are doing, they cannot improve their performance by taking advice from a human, and they cannot give advice or help to someone else doing similar tasks. Part of the reason for this is that over the last few decades research has become fragmented: with many individuals and research teams focusing their efforts on narrowly defined problems in vision, or learning, or language understanding, or problem solving, or mobile robotics, for instance.

### 1.2.2 The Way Forward

A key part of the CoSy effort has been to try to overcome some of these limitations by using ideas from relevant disciplines to investigate an ambitious vision of a highly competent robot, combining many different capabilities in a coherent manner, for instance a subset of the capabilities of a typical human 4-5 year old child. The scientific importance of this objective is that such a robot requires generic capabilities providing a platform for many different sorts of subsequent development, since a child of that age can develop in any human culture and benefit from many forms of education. However, we do not underestimate the profound difficulties of this challenge.

The research makes use of and feeds results into the various component disciplines of AI and cognitive science, for instance, new results on perception, learning, reasoning, language processing, memory, plan execution, and studies of motivation and emotion. Perhaps more importantly: the project not only benefits from other disciplines but has also tried to provide new substantive contributions to those disciplines in the form of new theories and working models. The detailed tasks of developing working systems generate new research questions for the contributing disciplines.

### 1.2.3 Steps to Success

The goal of producing a robot with many of the capabilities of a human child is unrealistic for a five year research project: it is an significant long term challenge. However, by analysing the many requirements for moving in that direction, one can derive sets of successively less challenging sub-goals that provide steps towards the distant goal. Some of these sub-goals are achievable in the time-frame of the project and form the main deliverables of the project.

An important part of the effort has been to consider two main kinds of deliverables: *theory* and *implementation*.

1. **Theory Deliverables**

   A body of theory, at different levels of abstraction, regarding requirements, architectures, forms of representation, kinds of ontologies, types of reasoning, kinds of knowledge, and varieties of mechanisms relevant to embodied, integrated, multi-functional intelligent systems. The results are expected to be useful both for enhancing scientific understanding of naturally occurring intelligent systems (e.g. humans and other animals) and for the design of artificial intelligent systems.

   The theory results are built around the core idea of a self-modifying architecture comprising different sorts of capabilities which develop over time. The results cover both analysis of *requirements* for such an architecture and also *design options* with their trade-offs.

   Key ideas for the architecture are informed by biological considerations, e.g. the notion of an architecture combining components which developed at different evolutionary epochs, and which in humans operate concurrently, performing different categories of tasks, for instance:

   - *reactive* components controlling the details of execution of skilled behaviours (these are evolutionarily old and use mechanisms shared with many species)
   - *deliberative* components supporting thought and reasoning about what might happen next or what should be done at some later stage (these are found in fewer species and require highly specialised mechanisms and forms of representation – including human language in some cases)
   - *self-reflective, meta-management* components that monitor, evaluate, and (partially) control and redirect the reactive and deliberative processes (these are probably rare in animals because they require the ability to represent, compare and evaluate information-processing tasks and strategies, as opposed to merely implementing them).

   The requirements for perceptual and motor systems that operate concurrently with, and in close coordination with, processes in all the different architectural layers are analysed and architectures are proposed for both perception (e.g. [122]) and action mechanisms.

   Learning processes are different within and between different parts of the architecture and current theories of learning have to be substantially extended to explain how, for instance (a) kinds of learning that extend the individual's ontology for perceiving and thinking about the environment, and (b) kinds of learning that develop fluency and speed in motor performance, e.g. because the reactive layer is *trained* by processes in the deliberative layer.

Different varieties of communication and social interaction are related to the different architectural layers: for instance, (a) dancing, fighting and moving heavy objects require coupled *reactive* systems; (b) linked collaborative actions spanning spatial and temporal gaps, e.g. in building houses and bridges, require *deliberative* capabilities; (c) the ability to empathise, exhort, persuade, may require extensions of self-understanding in the *meta-management* system to support other-understanding. (All of these influences can go both ways: e.g. meeting requirements for social developments may enhance individual capabilities.)

As different sorts of architectures with these general features are possible it is important to consider an analysis of architectural options and trade-offs.

2. **Implementation Deliverables**
Several implementations of working systems demonstrating successful application of the theory, e.g. in a robot capable of performing a collection of diverse tasks in a variety of scenarios described in the project, including visual and other forms of perception, learning, reasoning, and communication (with another robot or a human) both in order to collaborate on some task and in order to explain what it is doing and why. Our aim has been to create robotic cognitive systems that have a kind of autonomy that will justify describing it as having its own goals, which may change over time.

The distinctive features of such a robot include integration of subfunctions (e.g. vision and other senses can be combined in making sense of a scene, vision can be used to disambiguate an utterance by looking at what the utterance refers to, and learning processes can enhance different kinds of capabilities, including linguistic, visual, reasoning, planning, and motor skills).

Integration does not imply *homogeneity*, such as use of the same type of representation throughout the system. For instance, low level visual mechanisms finding edge-features, optical flow patterns, etc. use different forms of representation from higher level mechanisms e.g. those recognizing and describing whole objects. Likewise, planning mechanisms will need different forms of representation from fine-grained motor control mechanisms.

*Nature vs. Nurture:* A major research issue concerns how much should be programmed into such a robot and how much will have to be learnt by interacting with the environment, including teachers and other agents. There are several AI projects aiming to develop intelligent systems on the basis of powerful and general learning mechanisms starting from something close to a "Tabula rasa" (e.g. the COG project at MIT, the Cyberlife Research "Lucy" Project, and the Dav project at Michigan State

University). Their hope is that the structure of the environment will cause the learning mechanisms to induce all required information about the nature of the environment.

Such projects are likely to be defeated by explosive search spaces requiring evolutionary time-scales for success.

Biological evolution enables individuals to avoid this problem by providing large amounts of “innate” information in the genomes of all species. In the case of humans this seems to include meta-level information about what kinds of things are good to learn, helping to drive the learning processes as well as specific mechanisms, forms of representation, and architectures to enable them to work.

Although such debates are most commonly associated with requirements for language learning (e.g. [118] the issues are deep and general. For instance, Kant [65] argued two centuries ago that notions of space, time and causation are presupposed by and cannot be learnt from perceptual experiences. Similar points could be made about notions of meaning, purpose and action.

Instead of taking a dogmatic stance on what needs to be innate CoSy has explored various alternatives for amounts and types of innate knowledge and included a first analysis of the trade-offs.

## 1.3 A Motivating Example

At the core of the CoSy project have been the methods for construction of embodied artifacts, such as robots, with advanced cognitive functions. Such systems should be endowed with facilities for automatic interpretation of the environment (in terms of mapping of the environment, recognition of a large number of objects, etc.), adaptive acquisition of new skills and tasks in cooperation with a human user, methods for advanced manipulation to allow the system to perform extended missions on behalf of the users and reasoning methods to ensure advanced autonomy. The interpretation facilities can be used both to ensure autonomy and to verbalise knowledge and mission parameters to a user. Advanced service robots can be used for a large range of tasks in our everyday environment ranging from vacuuming to clearing the table after dinner. The tasks can also be in terms of mobility support for elderly and handicapped. To be of true utility to an average citizen it is essential that the system has facilities for automatic adaptation to the environment of the user, it must also be able to adapt to the habits of the owner, it must be able to understand the importance and consequences of instructions, and it must be sufficiently flexible to learn new skills that it was not endowed with from the factory. An example scenario is outlined below.

1. The Hartmut family decides to acquire a CoSy system to assist them in their everyday life. A system is acquired from the local Robs-R-Us chain of stores

2. Upon arrival at home the system is unpackaged and booted up. Upon wakeup the CoSy system greets the owners and asks them to give their names (Helga and Oskar). The owners are asked to give it a name (Fido). The system then asks for a tour of the home. As they traverse the home it builds up a map of the house, and objects of particular interest are named to the system (this is the refrigerator), ...
3. When asked to clean the table after dinner, Fido already has a built-in strategy for cleaning of tables, but it needs to know the location of the dishwasher (Oskar shows him the dishwasher – a Bosch 4590). Upon recognition of the dishwasher Fido downloads an operations manual from the Cosy corporate site. It then goes about its task.

   Oskar notices that Fido is removing cutlery and dishes but moves glasses to one side of the table. He asks: 'Why are you leaving the glasses?' Fido answers: 'My movements can be clumsy, so I'll put them in the dishwasher after all the objects that might break them.'
4. After dishes, cutlery, glasses and bottles have been cleared, a single object is left on the table. Fido asks Helga: What am I to do about this object left on the table? This is an Orrefors vase please leave it on the table! Fido inquires if it is Ok to turn on the dishwasher.
5. A few months later the Hartmut family acquires a new GE refrigerator. When Fido detects the new item, he inquires about the new objects. He is told that it is a GE-Economy Fridge. Upon learning this he downloads the latest skills from the CoSy skill database to allow him to operate the unit in terms of opening of the door and placing objects on the shelves, etc.
6. Later they replace the door to the bathroom with a sliding door, a new object that Fido has never seen before. He asks Oskar to demonstrate to him how to operate the door. Initially Oskar shows him how to operate the door (learning by demonstration). Fido tries to replicate the action and is corrected when he pushes to hard.
7. One day Julie, the 8 year old grandchild of Helga and Oskar, comes to visit. At first Julie is a little afraid of Fido and keeps her distance. Later when she is alone with Fido she says to Fido: Can you play Ludo? Fido says: What is Ludo? Julie: A game - I'll teach you. Will you play? Fido understands the general category of game, does not know Ludo, but reasons that since Julie wants him to play, and he is not required for anything else at that time, he should oblige. He considers downloading information about Ludo from the Cosy corporate site, but reasons that if Julie offers to teach him she would probably like to do that, and he wants to please her. So he does not download the information and says: Yes please. Show me what to do. They both enjoy the ensuing interaction.

The above scenario is deliberately futuristic, but it does at the same time illustrate "automatic mapping", "recognition of objects", "recognition of limitations in knowledge or skills", "natural interaction using speech and gestures", "never ending dialogues" to resolve ambiguities, and affective states and processes.

These are key issues that have been studied in the CoSy projects. Through integration of perception, action generation, autonomy, flexible user interfaces it has, however, been possible to approach the problem of building truly flexible cognitive systems that can be used to implement scenarios as outlined above.

## 1.4 Organization of the Research/Research Questions

In the study of cognitive systems for cognitive assistants a number of key research challenges were identified: architectures, representations, learning, perception-action modelling, communication, and planning & failure handling. In addition to addressing these issues at a fundamental level there is also a need to consider the integration of the techniques into complete systems. Two scenarios were identified for the study of integrated systems: i) exploration/mapping of space, and ii) models of objects and concepts. Initially we will discuss the research challenges (Sections 1.4.1 – 1.4.7) followed by a discussion of the integrated scenarios (Sections 1.4.9 – 1.4.10).

The overall work has been organised according to two major milestones

1. "Using intermodality and affordances for the acquisition of concepts, categories and language"

   The goal of the first milestone was to develop an agent that was capable of exploring unknown spaces and objects in that space, learning representations of spaces and objects based on a known ontology. The agent can determine what it does not know (or about which it is not clear), and can carry out a dialogue with another agent to learn more. Fundamental to the latter effort was the acquisition of strategies for representing, coordinating and understanding multi-modal action & dialog acts.
2. "Introspection of models & representations; planning for autonomy – goal seeking"

   Based on the idea of an embodied cognitive agent that is able to explore its environment, and communicate with other agents about descriptions for spaces and objects in that environment, the goal of the second milestone iwas to develop an agent that can perceive of its environment in terms of affordances, acquire knowledge about such affordances, and use such knowledge to understand the intentions of other embodied agents in that environment. Based on such understanding, the agent is able to (re-)plan its own actions and dialogues to achieve its own goals, possibly requiring communication with other agents to request or negotiate cooperation.

### 1.4.1 Architecture

For many years, research in AI and computational cognitive science focused on forms of representation, algorithms to operate on them, and knowledge to be encoded and deployed or derived. In the last decade or two it has become clear that there is also a need to investigate alternative ways of putting pieces together into a complex functioning system, possibly including parts that operate concurrently and asynchronously on different sub-tasks, for instance, perception, action, reasoning and communicating.

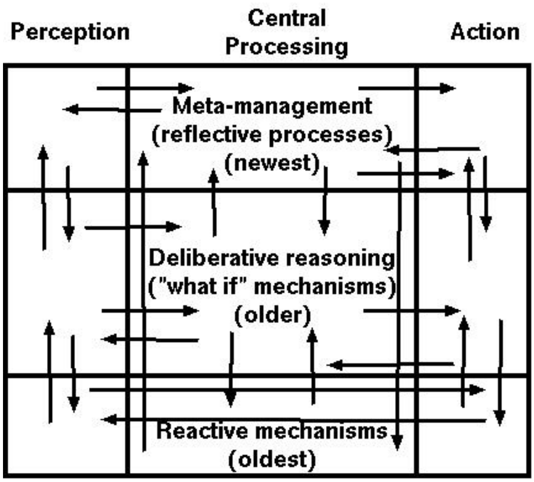


**Fig. 1.1.** The CogAff schema: superimposing 2 functional divisions, giving 9 categories of possible sub-mechanisms and many possible connections – not all shown.

Unfortunately this has led to a plethora of architectures being proposed including, for example SOAR, ACT (and its successors), PRODIGY, ICARUS, 3T, APEX, CLARION, CIRCA, EPIC, Subsumption architectures, H-COGAFF, and Minsky's emotion machine. Many of these were discussed at a DARPA/NSF-funded workshop in Stanford in March 2003.[1] One reason why this is a problem is that there is no agreement on what the space of possible architectures is like, nor on the terminology for describing architectures or on criteria for evaluating and comparing them.

One of the tasks for this project, therefore, has been to produce a framework for describing and comparing architectures. A first draft and relatively simple example of such a framework is the *CogAff schema* developed at the University of Birmingham, and described in [121], partly inspired by [89]. This

[1] Most of the presentations are online here: `http://www.isle.org/symposia/cogarch/`, including a useful survey paper by Langley and Laird. Another useful overview can be found in [40].

provides a way of classifying components of an architecture in terms of their functional role, starting with a crude three-way division between perceptual, central and action components, and another three-way division between components concerned with reactive, deliberative or meta-management functions. Superimposing these modes of division gives a grid of nine types of components which may or may not be present in an architecture, and which may be connected in various ways to other components as shown in Figure 1.1.

In recent years many researchers influenced by the work of Brooks [23] have attempted to design robots using only the bottom layer of the CogAff grid, the reactive layer, arguing that either features of the environment or emergent interactions between many individuals will produce effects that were thought to require deliberative and other mechanisms. Others have challenged this view arguing that it suffices only for simple organisms and insect-like robots.

Our approach has not been to engage in such battles but to try to understand under which conditions the various types of architectural components are useful. For example if building something at location A requires materials known to be at location B, and the agent does not have innately determined reactive behaviours that cause fetching of materials from B, then the deliberative ability to consider and evaluate a possible journey to B in advance of doing it will be useful. Of course this requires suitable forms of representation for learning and storing re-usable generalizations and perceptual mechanisms that can 'chunk' the environment into categories that support learning of useful generalisations. Without discretization, planning several steps ahead in a purely continuous space would be very difficult, although discretization of percepts may not be needed for certain kinds of reactive behaviours involving continuous control and no predictions.

One of the particularly interesting issues to explore is whether the kind of self-understanding that so many AI systems lack can be provided on the basis of a *meta-management* architectural layer permitting observation, classification, evaluation and possibly some control of internal states and processes, especially deliberative processes that are capable of getting stuck in loops, wasting resources be repeating sub-tasks or in other ways performing suboptimally. An important form of learning might include detecting such cases and finding out how to prevent them or reduce their effects. An early example of this sort of thing was demonstrated 30 years ago by Sussman [128], but never developed further. One of the issues to be explored is how our notion of meta-management relates to notions of "executive function" used in psychology and psychiatry. It is possible that empirical research on executive functions in humans can contribute design ideas for the construction of artificial cognitive systems. Likewise our design work may shed new light on puzzles about the nature of executive functions in humans.

Another interesting and important issue to consider is how such an architecture might develop. This relates to questions about the pros and cons of various degrees of complexity of the innate mechanisms given to a robot by its designers, or to animals by evolution.

*Architecture Deliverables:*

The study of architectures has continued throughout the project. It has three sorts of outputs:

- A succession of theoretical papers analysing the variety of possible architectures and their trade-offs in various contexts
- experimental designs for working systems to help investigate the theories and demonstrate the ideas

*Software Tools:*

A major requirement for the success of a project like this is the availability of tools that support construction of a complex architecture with many interacting, concurrently active components doing different things, e.g. concurrently active mechanisms involved in perception (concurrently processing at different levels of abstraction – eg. feature detection and perception of affordances), reasoning, motive generation, planning, plan execution, communication with other agents, evaluation of whatever's going on, learning of various kinds, etc.

Cosy has developed a platform for rapid prototyping and rapid deployment of intelligent systems combining multiple software and hardware resources. There were quite a lot of toolkits already available but mostly they tend either to be committed to a particular sort of architecture (e.g. SOAR, ACT-R, PRS, etc.) or else aimed at multi-agent systems composed of lots of relatively simple agents perhaps distributed over many machines. More general and open ended toolkits include the SimAgent toolkit developed at Birmingham [123], the Cogent toolkit developed at Birkbeck College and the Mozart toolkit developed in Saarbrücken and elsewhere. The project has evaluated these and other tools and developed a new improved tool - CAST that is presented in Chapter 2.

A cognitive systems project aimed at producing an integrated physically embodied agent with diverse but cooperative concurrently active and possibly strongly interacting capabilities needs tools that can be used to explore and test out ideas about different sorts of architectures – preferably rapidly and easily.

One of the requirements that arises in connection with architectures that include a meta-management layer is the need for mechanisms that allow self-observation during program execution. The SimAgent toolkit provides some support for that, used in work by Kennedy on self-monitoring intrusion detectors [66].

For all these explorations, merely using a powerful programming language, e.g. Prolog, or Lisp or Java is clearly not enough. For instance, different sorts of languages have been required for the different components (e.g. low-level vision, reasoning about where to go next, understanding or composing sentences, etc.) Facilities to support concurrency, self-observation and self-modification of running systems should if possible be general and re-usable. Rapid prototyping, i.e. rapidly producing new experimental implementations that can be run, is essential for research and exploration, as opposed to formal specification and verification, which are often the main concerns of software engineers: that only makes sense if the problem has already been understood, so that researchers know what they are trying to verify.

Often, discovering requirements is the hardest part of the research task in AI and exploratory design is a major tool for doing that (you soon find out that things you have built are no good for surprising reasons, narrowing the search space for what a working system needs to be able to do.)

Moreover, as the prototypes get more complex the human interface requirements for interactive debugging, testing and interrogating parts of the system, also get more complex and those requirements have to be met as well as support for whatever is being built.

*Note on Representations:*

There have been many debates in recent years about whether animals or robots need representations (E.g. see [23]). Our view on this is that anything that senses the environment and on the basis of what is sensed uses its own energy stores in order to select among possible actions is using information to determine what to do. Biological evolution discovered many variations on that theme depending on the kind of information acquired, how it is processed, how it is used, when it is used (e.g. long term storage may be required) how it is transformed, how it is combined with other information, and how it is communicated. In all cases there is some *medium* used for the information, but there are great differences between different media, including whether they are discrete or continuous, one-dimensional or multidimensional, what sorts of structures they can have, and so on. Some people may be inclined to argue about whether some of them are *really* representations or not, but we avoid such disputes by investigation *what kinds* of representations they are, and what their costs and benefits are to the organism.

### 1.4.2 Representations

A very important issue in the project has been the design of representations for objects, scenes, actions, dialogues and affordances. The representations must take into account visual, haptic, proprioceptive, and auditory — particularly linguistic — information, capturing temporal as well as static aspects. Many of these representations also need to be suitable for higher-level cognitive processes, such as deliberative and reflective processing. Others may be used only for reactive control. To accomplish the tasks, that formed for the basis for the project, the representations should

- enable integration of representations of objects, scenes, actions, events, causal relations and affordances; these representations should be linked together to form a consistent framework.
- allow incremental updating or sometimes correction.

- allow different types of learning (supervised, unsupervised, reinforcement).
- allow integration of various modalities, of very different input signals (visual, tactile, auditory, etc.), into a common framework of representations.
- deal with multiple cues, enable multiple representations of the same object, scene, event or action.
- be suitable for recognition and categorization in the presence of a cluttered background and variable illumination.
- take into account the context of an object, an action or a scene
- be scalable; they should enable efficient learning, storing of the representations, and recognition even when the number of objects, scenes and actions increases (in fact, with an increase in the number of entities, learning should become more efficient).
- accommodate semantics (in terms of language); they should be linked with symbolic descriptions.
- be suitable for acquiring, describing, and using spatial relationships, both quantitatively and qualitatively.
- be suitable for higher level cognitive tasks, such as reasoning or planning.
- enable the formation of concepts and hierarchies of concepts.
- allow introspection leading to refinement of the representations, and the construction of new representations at new levels of abstraction (e.g. capturing and exploiting temporal, or geometric regularities not explicitly programmed in by the designer).

Many of these issues have been addressed prior to CoSy. They were, however, addressed separately, considering only one, or a few aspects at a time. They have never been tackled together, in a common framework. This has been an important part of this project. The representations (and, consequently, the learning and usage of the representations) of objects, scenes, actions, events, causes and affordances should be linked together in a unifying framework. The representations may be obtained in different ways, but processes employing different representations need to able to exchange information, and to update some representations on the basis of others.

*Task Specific vs. General Representations*

For a cognitive system there is a need for representations which can, on the one hand, be quite general, to serve different tasks and goals, and, on the other hand, be tailored for particular tasks. The advantage with task specific representations is that they are efficient to use, and often compact. The problem with specific representations is that they can only be used for the task that they were trained for. One example is task specific recognition or classification e.g., a system trained to distinguish faces with glasses from faces without glasses would be inappropriate for other tasks e.g., to distinguish male from female faces. Another example would be a reactive controller for plugging a socket into a power point, which is not directly of use for another task such as grasping a cup.

A number of unsupervised learning methods produce generative representations which allow partial approximate reconstruction, or hallucination, of the input data. According to Grossberg [51], reconstruction is central for a two-way processing cognitive system. Memory traces in the short term memory could activate the long term memory forming a reconstruction, which gets compared to the momentary input. This mechanism allows categorization and establishment of a reference frame, forming a set of high-level expectations. These expectations influence further processing in a top-down manner, by limiting the contextual frame, checking for consistency, providing missing data etc. Cognition is therefore performed as a two-way, bottom-up and top-down process. Comparing the input to the "hallucination" can reinforce the good hypotheses in the interpretation and inhibit inconsistent parts such as noise and occlusions. A two-way interaction also enables the processing and organization of a huge amount of data, in contrast to the one-way processing with no feedback. While being significantly faster [104], the lack of interaction with the higher levels of a one-way processing schema could lead to a combinatorial explosion of complexity.

Nevertheless, supervised non-generative methods can produce simpler representations, which can be more efficient for specific tasks they are tuned for. Our goal should be to combine both, unsupervised generative methods and supervised non-generative methods, in order to obtain representations, which are general enough to enable a two-way processing and robust recognition, and to complement them with features, which are tailored for specific tasks.

### 1.4.3 Learning

*Modes of Learning*

Learning can be considered using a number of different modes:

Tutor Driven. A user (tutor) shows to the system an object or an action and explains to the cognitive system what he/she is showing or doing. The tutor can also provide figure-ground segmentation (e.g., showing the object on a black background), which facilitates the learning process and enables the creation of a reliable representation. In a similar manner, the tutor can guide the agent around the space and give it additional information (e.g. coordinates) to facilitate the creation of a map. In such a learning scenario, the user has a high degree of involvement.

Tutor Supervised. A cognitive system detects a new object, an action, event, affordance or a scene by itself and builds its representation in an unsupervised manner. Then it asks the user to provide additional information about the object or action (e.g., What is this object, action or room?). The involvement of the user in the learning process is now significantly reduced, yet it still assures that the produced representations are reliable/consistent.

Exploratory. A cognitive system detects a new instance of an object, action, event or scene, tries to recognize (categorize) it (using previously acquired knowledge, statistical properties, etc.), and then updates the representations accordingly. Such a learning scenario does not require any involvement on the part of the user.

To speed up the initial phase of the learning process and to enable development of consistent basic concepts, one could start with mainly tutor-driven learning with many user interactions. These concepts would consequently be used to detect new concepts with limited help from the user (an important role will be played by the dialogue between the cognitive system and a user). Later on in the process, when the ontology is sufficiently large, many new concepts could be acquired without user interaction.

Learning is embedded in the perception-action loop but must go beyond simple percept-react cycles involving deliberative processes such as reasoning and planning. This is particularly important in tutor supervised and exploratory modes of learning, where the agent must plan how to interact with the world so as to learn effectively. This will in turn require reflective processes to modify the learning strategy, as well as a model of motivation and attention.

*Continuous Learning*

As mentioned in Section 1.2.3 in order to avoid evolutionary time-scales in the development of the system, it has to initially encompass a certain level of pre-defined functionality and knowledge. It then has to be able to build new concepts upon these. It needs to do this throughout its lifetime.

It is therefore important that the representations employed allow the learning to be a continuous, open-ended, life-long process. The representations should not be learned once and for all in a training stage and then used in their fixed form for recognition, planning and acting. They should be continuously updated over time, adapting to the changes in the environment, new tasks, user reactions, user preferences, etc. So there will be no strict distinction between the activities of learning, planning, recognising, acting and introspection: these activities will be interleaved.

This is a non-trivial challenge. For example, most of the state-of-the-art algorithms for visual learning and recognition do not consider continuous learning. They follow the standard paradigm, dividing the off-line learning stage and the recognition stage [141, 139, 111, 79, 80, 63, 2, 42, 73]. Most of these approaches are not designed in a way which would enable efficient incremental learning, which is a basic prerequisite for continuous learning.

When having a hierarchical or multi-layered organization of representations, one has to ensure, when performing continuous learning, that the representations are updated on all levels and that also all mutual dependencies are updated accordingly, to assure the consistency of the representations. Some of these updates would need to be performed on-line (thus they have to be simple and fast) while some of them may be performed off-line from time to time (during the agent's "sleeping mode") by processing several updates at once.

There are two issues which are very important for reliable continuous learning. First, the representations have to be carefully chosen, such that they must allow efficient upgrading with the newly acquired data. Second, it is important how new data is extracted and prepared. When this is performed under the user's supervision, this operation is risk-free in the sense that the algorithm can update the model with high confidence. On the other hand, when the information, which is to be added into the representation, is autonomously extracted by the agent in an unsupervised manner (e.g., using a recognition procedure) there exists a possibility of propagating an erroneous extraction from the data through the learning process. Consequently, the representation could be corrupted with false data, resulting in poorer performance and a less reliable representation. Robust mechanisms, which prevent such propagation of errors, play an important role in the process of continuous learning.

Similarly, the already learned concepts should facilitate learning of new concepts. For instance, it should be easier to learn to recognise a new object after a hundred objects have previously been learned than only after a few objects have been considered. In the former case, information is captured in the representations of already learned objects, which should make learning of the new object easier. Such behavior (which is in accordance with some of the properties of human perception) is not a characteristic of many state-of-the-art algorithms for visual learning.

### 1.4.4 Perception-Action Modelling

In planning and acting approaches fall into two broad classes, although there are now points of contact. On the one hand there are abstract relational representations of the effects of actions as used in classical AI planning. On the other there are probabilistic models of action effects in continuous spaces as used in robot localisation and mapping. The former are general, and non-task specific, but assume observability of the world, and a hand-constructed abstraction from sensor data. Reasoning with them is also inefficient. The latter capture uncertainty in both action and observation, and are tractable for localisation and path planning in continuous spaces. They are, however, typically tied to geometric representations of space, and in some cases to quite strong assumptions about the sensors used. One of the tasks of this projects has been to try to connect these different types of representation in such a way that updates to one representation can be propagated to other representations.

In neither case are the representations of actions entirely suitable for continuous learning. Some attempts have been made to achieve this. Early techniques like chunking attempted to capture reusable patterns of behaviour, and probabilistic models of actions allow the natural integration of new experiences into existing action models. But our agents also need to learn to identify and acquire new action models on different time-scales, recognise new events, introspect about their causes and thus identify affordances for specific tasks.

Analogously in the literature on planning and learning to act there has been some work on automatic relevance detection when learning action models. This can be seen as a simple form of affordance learning, and has been studied in, for example, reinforcement learning. These algorithms learn action models that are more efficient because they are task specific. These are acquired by identifying features that are relevant to predicting the outcome on the task. The state-of-the-art techniques, however, work only with unstructured or propositional representations, and rely on statistical methods to detect the affordances. Another issue has been how knowledge be transferred from one task specific representation to another.

### 1.4.5 Continuous Planning and Acting in Dynamic Multiagent Environments

*Continuous Planning*

Planning and acting in realistic environments poses a number of difficulties to artificial agents many of which are due to the dynamic nature and partial observability of such environments: Other agents' actions as well as naturally occurring events (e.g. sunset) may change the agent's surroundings in ways it cannot foresee, control or even perceive. It is thus crucial that during plan execution agents react according to the perceived changes (e.g. stop moving when there is a human standing in your way). Unfortunately, on the one hand not all plans can easily be repaired during plan execution, on the other hand with increasing dynamics of the world an agent's knowledge will become less accurate and its plans more likely to be not executable or to achieve undesired effects. However, it is also not possible to give up planning at all or to use purely reactive forms of planning [3] since its lack of flexibility does not allow to account for diverse and varying goals and its lack of knowledge may prevent an agent from selecting a "good" action to perform without some lookahead planning.

Thus contingencies have to be taken into account already during the planning phase. Traditionally, research in AI Planning has addressed this problem in the subfields of *conditional planning* [105, 14, 56] and *probabilistic planning* [20, 77]. The problem solved by conditional and probabilistic planning algorithms is to find plans that will work not only for a given initial state but under all circumstances imaginable, a fact that makes the problem computationally hard [105, 77]. Considering the large number of unobservable features and of possible contingencies in dynamic multiagent environments it is clear that even small-sized problems will be hard to solve by conditional and probabilistic planners.

Luckily, in realistic domains (like those considered in the CoSy project) there is often no need to devise universally executable plans before acting as agents will continuously plan, act, monitor their actions, and learn in the same environment. Therefore, in this project techniques have been developed that allow agents to postpone the resolution of contingencies and handle them only when they actually occur. Ideally, an agent learns from past experiences which contingencies it can easily solve by re-planning at execution time. During the planning phase the agent can then simply "ignore its ignorance" about those contingencies. Instead of a conditional plan fragment, agents use so-called *assertions* in their plans, non-conditional pseudo actions achieving facts whose actual achievement will be planned for later. Assertions consist of preconditions and effects just like normal actions and special *planning trigger conditions* describing under which conditions in later phases of a planning-execution cycle detailed planning for the assertion's effect will be carried out. Trigger conditions mainly consist in a set of state variables whose values are unknown to the agent in the current state. As soon as the agent can observe the actual values of these variables a detailed sub-plan achieving the assertion is searched for.

The planning agent research goes beyond classical planning in integrating planning and execution into a continuous planning agent. It contrast to related approaches [142, 8, 144] the research goal of this project is not focused on plan repair or re-planning[2] but on planning under incomplete information and partial observability. Unlike the conditional and probabilistic planners presented earlier contingency resolution is based on postponing the solution of parts of the planning problem to later points in time. In their use of abstract pseudo actions that are decomposed into executable actions the resulting plans resemble hierarchical task networks [145, 38, 39]. In our approach, however, the abstraction hierarchy is not explicitly given, but the decompositions can be planned by the agent itself during a later phase of the planning-execution-monitoring cycle. Also the purpose of the abstraction is different from HTN planning: while HTN decompositions embody knowledge about how to solve subtasks, in our project assertions essentially represent a way to reason under imperfect knowledge.

The approach to automatically interleaved planning and execution taken in this project relies on explicit modeling of an agent's perception. Assertion achievement is planned for only when the observations specified by the triggering conditions can be made. It is therefore an important part of the proposed project to model different perceptors and sensing capabilities in a general way. There are two possibilities for modeling sensing: automatic sensing and active sensing by executing sensory actions [109] which both are allowed in our representation. Making perception explicit in this manner allows agents to plan their own sensing, i.e. to actively try to reach states in which observations can be made and thus to reduce their ignorance. Both kinds of sensing are modeled as *perceptive events*, active sensing events being triggered by explicit sensing actions of the agent while automatic sensing events "fire" similarly to

[2] Modern planners like FF [57] allow fast re-planning whenever it is necessary.

domain axioms [130] (also sometimes called derived predicates, causal rules, or indirect effects). In contrast to earlier approaches to planning with sensing [41, 67, 50, 9] the use of perceptive events allows to separate modeling of the physical effects of events from their perception by an agent. This separation is especially important for active failure diagnosis (cf. section 1.4.5), but also to describe when several agents with different perceptors are present in a common environment (cf. section 1.4.5).

*Active Failure Diagnosis*

In the recent years, tremendous advances have been achieved in the field of mobile robotics. The majority of the work, however, has focused on navigation aspects such as path planning, localization and mapping. In most approaches it is typically assumed that the sensors and actuators of the robot are reliable in the sense that their input always corresponds to the expected input and that there is no malfunction of the sensors or actuators. Accordingly, these systems are prone to failure in situations in which a sensor breaks and provides erroneous data that does not correspond to the assumptions.

Recently, several techniques have been developed to allow mobile robots to detect or reason about potential failures. For example, Murphy and Hershberger [84] present an architecture that allows a mobile robot to exploit causal models for generating tests about potential failures or changes in the environment. Leuschen et al. [76] use a mixture of methods to analyze the reliability of a robot. However, they do not provide means for fault detection. Scheding et al. [110] analyze the effects of faults to the innovation of a Kalman filter. They introduce a metric for determining the detectability of faults. Washington [138] uses a combination of Kalman filters and Markov decision processes to identify faults on a mobile platform. Whereas Kalman filters are efficient means for detecting faults, their drawback lies in the limited capabilities to represent ambiguous situations in which multiple failures could case the same effect. To also deal with ambiguous situations recently particle filters have been used with great success for fault detection in technical systems [31, 30, 83, 136, 137, 135]

One important problem in the context of applying particle filters for fault diagnosis in complex systems is caused by the high dimensionality of the state space. In practice, a huge number of particles is needed to track all potential fault states of the system. Therefore, recent approaches have especially addressed the problem of how to reduce the number of samples during the sampling process [83, 137, 135].

All these approaches, however, are passive in the sense that they do not exploit the actuators of the robot to identify potential faults. By generating appropriate actions, such as moving for example, the robot can actively generate measurements that make the robot more certain about potential failures, i.e. a malfunction of its laser range scanner. This way, the complexity of the state tracking problem can significantly be reduced. In particular, we have investigated techniques allowing mobile robots to identify and to deal with