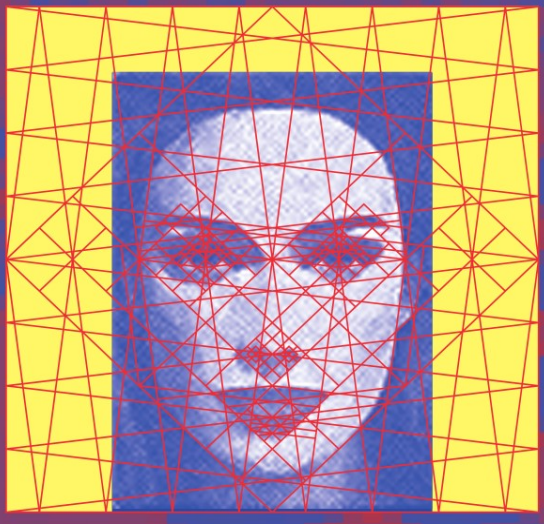
State-of-the-Art Survey

LNAI 5499

Giovanni Pezzulo
Martin V. Butz
Olivier Sigaud
Gianluca Baldassarre (Eds.)

Anticipatory Behavior in Adaptive Learning Systems

From Psychological Theories to Artificial Cognitive Systems



Springer

Lecture Notes in Artificial Intelligence 5499

Edited by R. Goebel, J. Siekmann, and W. Wahlster

Subseries of Lecture Notes in Computer Science

Giovanni Pezzulo Martin V. Butz
Olivier Sigaud Gianluca Baldassarre (Eds.)

Anticipatory Behavior in Adaptive Learning Systems

From Psychological Theories to Artificial Cognitive Systems

Springer

Series Editors

Randy Goebel, University of Alberta, Edmonton, Canada
Jörg Siekmann, University of Saarland, Saarbrücken, Germany
Wolfgang Wahlster, DFKI and University of Saarland, Saarbrücken, Germany

Volume Editors

Giovanni Pezzulo
Consiglio Nazionale delle Ricerche
Istituto di Linguistica Computazionale "Antonio Zampolli"
Via Giuseppe Moruzzi 1, 56124 Pisa, Italy
and Consiglio Nazionale delle Ricerche
Istituto di Scienze e Tecnologie della Cognizione
Via San Martino della Battaglia 44, 00185 Roma, Italy
E-mail: giovanni.pezzulo@cnr.it

Martin V. Butz
University of Würzburg, Department of Psychology III
COBOSLAB – Cognitive Bodyspaces: Learning and Behavior
Röntgenring 11, 97070 Würzburg, Germany
E-mail: mbutz@psychologie.uni-wuerzburg.de

Olivier Sigaud
Université Pierre et Marie Curie
Institut des Systèmes Intelligents et de Robotique (CNRS UMR 7222)
Pyramide Tour 55, 4 Place Jussieu, 75252 Paris Cedex 05, France
E-mail: Olivier.Sigaud@upmc.fr

Gianluca Baldassarre
Consiglio Nazionale delle Ricerche, Istituto di Scienze e Tecnologie della Cognizione
Laboratory of Computational Embodied Neuroscience
Via San Martino della Battaglia 44, 00185 Roma, Italy
E-mail: gianluca.baldassarre@istc.cnr.it

Library of Congress Control Number: 2009928424

CR Subject Classification (1998): I.2.11, I.2, F.1, F.2.2, J.4, I.6

LNCS Sublibrary: SL 7 – Artificial Intelligence

ISSN 0302-9743
ISBN-10 3-642-02564-1 Springer Berlin Heidelberg New York
ISBN-13 978-3-642-02564-8 Springer Berlin Heidelberg New York

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12696468 06/3180 5 4 3 2 1 0

Preface

Anticipatory behavior in adaptive learning systems continues to attract the attention of researchers in many areas, including cognitive systems, neuroscience, psychology, and machine learning. The ABiALS workshop series is now in its fourth edition – and it is very vital.

The 4th Workshop on Anticipatory Behavior in Adaptive Learning Systems (ABiALS 2008) was held in collaboration with the 5th Six-Monthly Meeting of euCognition: "The Role of Anticipation in Cognition" in Munich, June 26–27, 2008.

EuCognition, the European Network for the Advancement of Articial Cognitive Systems (FP6-26408), funded this stimulating two-day event which saw the participation of six invited speakers (four of whom contributed to this book) and over 50 researchers from several European nations and abroad. Over 20 papers were discussed, either in oral or poster presentations. We are grateful to the euCognition's Executive Committee, and in particular to David Vernon, for giving us the possibility of holding the fourth ABiALS meeting in collaboration with the euCognition meeting, and for generously sponsoring both events.

We are grateful to our Program Committee members for providing careful reviews of the contributions, and additional comments and suggestions, which have greatly enhanced the quality of this book.

Thanks to the numerous participants – with different backgrounds, but with converging interests – the workshop hosted an extremely stimulating discussion and comparison of ideas which touched numerous topics, including time scales in prediction, how anticipation relates to hierarchies in the control of action, in what sense anticipatory mechanisms of living organisms are related (or the same) across different domains, or what could be the foundations of artificial systems provided with anticipatory capabilities.

The numerous interactions we had during the two-day event testified an extremely vivid interest in basic issues related to prediction and anticipation in many disciplines and from several perspectives. This makes the sharing of a common language extremely important. For this reason, the introductory chapter of this volume revisits the current available terminology on anticipatory behavior and relates it to the available system approaches. In addition, the introductory chapter offers an overview of the contributions in this volume. The contributions have been grouped in six sections: "Anticipation in Psychology: Focus on the Ideomotor View," "Conceptualizations," "Anticipation and Dynamical Systems," "Computational Modeling of Psychological Processes in the Individual and Social Domains," "Behavioral and Cognitive Capabilities Based on Anticipation," and "Computational Frameworks and Algorithms for Anticipation, and Their Evaluation."

One remarkable aspect of this volume is that numerous papers encompass more than one discipline, and in particular study the close relationships between the study of living organisms and the realization of computational modeling of anticipatory mechanisms. This interaction is clearly bidirectional. Some papers start with psychological theories and empirical evidence to inform the study and realization of computational and robotic

models. Others apply insights from computer science or information theory to suggest novel ways to look at empirical phenomena, or to explain empirical data.

In addition to its role in producing scientific advancements and promoting cross-disciplinary discussions, ABiALS continues in its community-building activity, too. A novelty this year was the setting up of a Web portal focused on *anticipatory behavior*, with the intention of further disseminating ideas and fostering discussions and collaborations within and outside the ABiALS community:
http://www.anticipatorybehavior.org/

April 2009

Giovanni Pezzulo
Martin V. Butz
Olivier Sigaud
Gianluca Baldassarre

Organization

This work was supported by the EU-funded projects HUMANOBS: Humanoids That Learn Socio-Communicative Skills Through Observation, contract no. FP7-STREP-231453, and IM-CLeVeR - Intrinsically Motivated Cumulative Learning Versatile Robots, contract no. FP7-IST-IP-231722. Additional support from the Emmy Noether program of the German Research Foundation grant BU1335/3-1 is acknowledged.

Executive Committee

Giovanni Pezzulo

Istituto di Linguistica Computazionale "Antonio Zampolli",
Consiglio Nazionale delle Ricerche (ILC-CNR) Pisa, Italy and
Istituto di Scienze e Tecnologie della Cognizione,
Consiglio Nazionale delle Ricerche (ISTC-CNR), Rome, Italy

Martin V. Butz

COBOSLAB, Department of Cognitive Psychology III,
University of Würzburg, Würzburg, Germany

Olivier Sigaud

Institut des Systèmes Intelligents et de Robotique (CNRS UMR 7222)
Université Pierre et Marie Curie, Paris, France

Gianluca Baldassarre

Laboratory of Computational Embodied Neuroscience,
Istituto di Scienze e Tecnologie della Cognizione,
Consiglio Nazionale delle Ricerche (LOCEN-ISTC-CNR), Rome, Italy

Program Committee

Christian Balkenius	Lund University, Lund, Sweden
Andy Barto	University of Massachusetts, Amherst, Amherst, USA
Edoardo Datteri	Università Milano-Bicocca, Milan, Italy
Jason Fleischer	University of California, San Diego, USA
Oliver Herbort	University of Würzburg, Würzburg, Germany
Frederic Kaplan	Ecole Polytechnique Fédérale de Lausanne, Lausanne, Suisse
Pier Luca Lanzi	Politecnico di Milano, Milan, Italy
Pierre-Yves Oudeyer	INRIA, Bordeaux, France

Tony Prescott	University of Sheffield, Sheffield, UK
Alexander Riegler	Vrije Universiteit, Brussels, Belgium
Wolfram Schenck	University of Bielefeld - AG Technische Informatik, Bielefeld, Germany
Samarth Swarup	University of Illinois, Urbana-Champaign, USA
Marc Toussaint	Technische Universität Berlin, Berlin, Germany
Tom Ziemke	University of Skövde, Skövde, Sweden

Table of Contents

Computational Modelling of Psychological Processes in the Individual and Social Domains

Behavioral and Cognitive Capabilities Based on Anticipation

Computational Frameworks and Algorithms for Anticipation, and Their Evaluation

From Sensorimotor to Higher-Level Cognitive Processes: An Introduction to Anticipatory Behavior Systems

Giovanni Pezzulo[1,2], Martin V. Butz[3],
Olivier Sigaud[4], and Gianluca Baldassarre[2]

[1] Istituto di Linguistica Computazionale Antonio Zampolli,
Consiglio Nazionale delle Ricerche,
Via Giuseppe Moruzzi, 1 - 56124 Pisa, Italy
giovanni.pezzulo@cnr.it
[2] Istituto di Scienze e Tecnologie della Cognizione,
Consiglio Nazionale delle Ricerche,
Via San Martino della Battaglia 44, I-00185 Roma, Italy
gianluca.baldassarre@istc.cnr.it
[3] University of Würzburg, Röntgenring 11, 97070 Würzburg, Germany
mbutz@psychologie.uni-wuerzburg.de
[4] Institut des Systèmes Intelligents et de Robotique (CNRS UMR 7222)
Université Pierre et Marie Curie
Pyramide Tour 55, 4 Place Jussieu - 75252 PARIS cedex 05 France
Olivier.Sigaud@upmc.fr

Abstract. This book continues the enhanced post-workshop proceedings series on "Anticipatory Behavior in Adaptive Learning System" (ABiALS), published as Springer LNAI 2684 and LNAI 4520 [3,5]. The proceedings offer a multidisciplinary perspective on anticipatory mechanisms in cognitive, social, learning, and behavioral processes, with contributions from key researchers in psychology and computer science. This introduction offers a conceptual terminology on anticipatory mechanisms and involved predictive capabilities. Moreover, it provides an overview of the book contributions, highlighting some of their peculiarities and complementarities.

Keywords: Anticipation, anticipatory behavior, prediction, simulation, goal-directed behaviour.

1 Introduction

This book is the third volume of extended post-workshop proceedings on "Anticipatory Behavior in Adaptive Learning System" (ABiALS). The previous two volumes were published as Springer LNAI 2684 and LNAI 4520 [3,5]. The theme of anticipation and anticipatory behavior continues to gather attention from scholars of many disciplines, including computer science, psychology, neuroscience, and philosophy.

G. Pezzulo et al. (Eds.): ABiALS 2008, LNAI 5499, pp. 1–9, 2009.

Anticipatory mechanisms are increasingly recognized as a key research area. At the European level, the EU Commission has recognized the relevance of the theme of anticipation by funding the MindRACES (from Reactive to Anticipatory Cognitive Embodied Systems) project (FP6-511931) in the area of Cognitive Systems research. One of the final outputs of the MindRACES project is a collective book on anticipation [10]. Successively, the theme of anticipation has been targeted by numerous EU-funded projects. For instance, the Technical Background Notes for Proposers document of the EU Commission (relative to FP7-ICT CALL 4, Work Programme 2009-10, Challenge 2: Cognitive Systems, Interaction, Robotics, pag. 10) indicates the following key research question: *How can we predict (and anticipate) future events in their environment (including, where relevant, the behavior of other agents - human or not - operating in the same environment)?*

The last edition of the ABiALS workshop was sponsored and hosted by the EU-funded Coordination Action *euCognition*: The European Network for the Advancement of Artificial Cognitive Systems (FP6-26408). Thanks to the euCognition's Executive Committee, and in particular thanks to David Vernon, the athors were able to organize an extremely stimulating two-days event (held on 26 and 27 June 2008 at Munich), which combined ABiALS 2008 and the Fifth Six-Monthly Meeting of euCognition"The Role of Anticipation in Cognition". The two-days event saw the participation of six invited speakers (four of whom contributed to this book), over twenty papers discussed either in oral or poster presentations, and the participation of over fifty researchers coming from several European nations and from outside Europe. Most of the contributors of this book participated to this event.

Within this renewed interest on anticipation and anticipatory behavior, the goal pursued by the ABiALS workshop series is to foster interactions, mutual understanding, and effective communications and cooperations between scientists belonging to different disciplinary domains, which nonetheless work on the same subject. To do so, we believe that it is also necessary to continue developing the theoretical and computational foundations of the study of anticipation in natural and artificial agents, and to integrate insights from many disciplines into a principled approach for the study of living systems and the design of artificial systems – also termed the *anticipatory approach* [4,10].

The anticipatory approach aims to understanding and conceptualizing anticipation and anticipatory behavior in natural cognition and to implementing them in artificial systems. Anticipatory systems have capabilities that go far beyond those of purely reactive ones and anticipation is a strong prerequisite for various cognitive functions and for goal-directed behavior. This introductory chapter first revisits the terminology of anticipatory systems and then introduces the book's contributions and their key aspects.

2 Basic Terminology Revisited

Although anticipations and predictions are often used nearly as synonyms in natural language, in scientific realms there is a clear distinction between

predictive systems and anticipatory systems. Generally, anticipatory systems are those that use their predictive capabilities to optimize behavior and learning to the best of their knowledge. Rosen [11, ch. 6] might have been one of the first who put this idea into a useful definition. According to this author, an anticipatory system is:

> [...] a system containing a predictive model of itself and/or its environment, which allows it to change state at an instant in accord with the model's predictions pertaining to a latter instant.

More precisely, he also states that:

> An anticipatory system S_2 is one which contains a model of a system S_1 with which it interacts. This model is a predictive model; its *present* states provide information about *future* states of S_1. Further, the present state of the model causes a change of state in other subsystems of S_2; these subsystems are (a) involved in the interaction of S_2 with S_1, and (b) they do not affect (that is, are unlinked to) the model of S_1. In general, we can regard the change of state in S_2 arising from the model as an adaptation, or pre-adaptation, of S_2 relative to its interaction with S_1.

The most peculiar aspect of anticipatory systems is thus their dependence on (predicted) future states and not only on past states. Although the definition provided by Rosen may be too strong (it excludes systems that coordinate with future states without explicitly representing them – we call this form *implicit anticipation*), it describes the kinds of systems we are mainly interested in: those able to realize behavior mediated by explicitly formulated expectations (*explicit anticipation*). In order to produce explicit expectations, anticipatory systems need predictive mechanisms, which may have different realizations, but nevertheless share the common feature of predicting future states.

Thanks to their predictive mechanisms, anticipatory systems can employ anticipatory behavior, which may be defined according to [2, p. 3] as:

> [...] a process or behavior that does not only depend on past and present but also on predictions, expectations, or beliefs about the future.

It is this capability to formulate predictions and to use them for own purposes that distinguishes an anticipatory system from a merely reactive one. For example, anticipation plays a key role in goal-directed and proactive behavior, since patterns of actions can be selected depending on their expected outcomes and not (only) on stimuli that are available here and now. While reactive systems can be functionally described with STIMULUS → ACTION (S-A) behavioral patterns, anticipatory systems are instead based on EXPECTATION → ACTION (E-A) behavioral patterns, which are permitted by the explicit prediction of a stimulus or an action effect (STIMULUS → EXPECTATION (S-E), or STIMULUS ACTION → EXPECTATION (S-A-E)). However consider that, as it will be clearly shown by the works presented in the book, anticipatory behavior can have

different functional organization and can rely upon a multitude of different specific mechanisms.

A last important distinction needs to be drawn between "prediction" and "anticipation":

Prediction is a representation of a particular future event.
Anticipation involves processes underlying future-oriented action, decisions, and behaviors based on (implicit or explicit) prediction.

Thus, anticipation – the main focus of this book – includes prediction but goes beyond mere forecasting in that it refers to processes which use predictive knowledge to coordinate behavior and, more importantly, to act in a goal-directed fashion and pro-actively to realize achievable and desirable future states while avoiding unsuitable ones.

3 Overview of the Book

Due to the interdisciplinary nature of the theme of anticipation, and the variegated audience of the ABiALS workshop, the book includes a diverse range of contributions that vary from psychological theories to evaluation of computational frameworks based on anticipation and real-world applications. To ease reading, the book contributions, now reviewed one by one, are grouped into six categories.

A technical note before starting. This section references to all the works of the book in terms of authors and title of the respective book chapters, but a whole reference of the works can be searched for (for example) as:

> Pezzulo G., Butz M.V., Sigaud O., Baldassarre G. (2009). From sensorimotor to higher-level cognitive processes: An introduction to anticipatory behavior systems. In Pezzulo G., Butz M.V., Sigaud O., Baldassarre G. (Eds.), *Anticipatory Behavior in Adaptive Learning Systems – From Psychological Theories to Artificial Cognitive Systems*, LNAI 5499. Berling: Springer-Verlag.

3.1 Anticipation in Psychology: Focus on the Ideomotor Principle

The book includes two (invited) contributions by two leading cognitive psychologists, Joachim Hoffmann and Bernhard Hommel, that put forward an *ideomotor* view of goal-directed action. They present their comprehensive frameworks *ABC (Anticipative Behavioral Control)* [6] and *TEC (Theory of Event Coding)* [7], which place anticipation at the very core of cognition. In addition, both contributions discuss in depth how their ideas are being implemented in robotic systems.

The first paper, contributed by Joachim Hoffmann (ABC: A Psychological Theory of Anticipative Behavioral Control), offers a complete ideomotor model of goal-directed action and discusses in detail the roles of action-effect contingencies: how are they acquired and contextualized, and how can they be

arranged hierarchically to realized nested loops of control that permit the transformation of abstractly defined goals into motor patterns. The paper reviews recent empirical evidence in favor of the ABC theory, and discusses a recent computational implementation: the SURE_REACH architecture [1].

The second paper, contributed by Pascal Haazebroek and Bernhard Hommel (Anticipative Control of Voluntary Action: Towards a Computational Model), describes another theory based on the ideomotor principle: the Theory of Event Coding on human goal-directed action. Differently from Hoffmann's theory, one of the main tenets of the TEC is that integration of perceptual and motor codes is realized at a distal level; specifically, at the level of distal perceptual effects of actions, not their proximal ones such as reafferences. As a consequence, the TEC describes planning as operating on distal perceptual codes and on-line realization by different (automatic) processes. The paper discusses also a recent implementation of the TEC theory: the HiTEC computational model.

3.2 Theoretical and Review Contributions

The book continues with three theoretical and review contributions that explore multiple facets of anticipation and anticipatory behavior.

The first paper is a quite provocative invited contribution by Jurgen Schmidhuber (Driven by Compression Progress: A Simple Principle Explains Essential Aspects of Subjective Beauty, Novelty, Surprise, Interestingness, Attention, Curiosity, Creativity, Art, Science, Music, Jokes). The author introduces the principle of 'data compression progress', that is equivalent to an augmented capability to predict data, as a basic mechanism that makes data 'interesting in itself' and motivates exploratory behavior. This principle has the potential to be relevant for multiple domains, which range from art to science, and captures the slipping concept of 'beauty' in a rigorous and extremely interesting way.

The second paper is contributed by Fabio P. Bonsignorio (Steps to a Cyber-Physical Model of Networked Embodied Anticipatory Behavior). It sketches a modeling framework for embodied anticipatory behavior systems by using a wide range of formal notions such as entropy, complexity, and information. Although still rather preliminary, this paper introduces numerous essential issues toward the development of more autonomous artificial systems based on anticipatory capabilities.

The last paper is authored by Henrik Svensson, Anthony Morse, and Tom Ziemke (Neural Pathways of Embodied Simulation). This paper offers a deep discussion of the many facets of recent theories based on the idea of internal simulation. The main contributions of this paper are a comprehensive review of the multiple pathways in neural simulations and a discussion of the differences between procedural and declarative knowledge in covert simulation.

3.3 Anticipation and Dynamical Systems

Two papers explore anticipation in relation to internal agent dynamics, and discuss two kinds of anticipatory processes that are coupled to internal dynamics rather than responsible for the selection and triggering of overt behavior.

In the first paper, Michela Ponticorvo, Domenico Parisi, and Orazio Miglino (The Autopoietic Nature of the "Inner World": A Study with Evolved "Blind" Robots) explore the internal dynamics of an agent that is deprived of any external stimulation. In these extreme circumstances, the agent cannot rely on external stimuli to predict the effects of its actions, but only on self-generated stimuli – an 'inner world'. The paper shows that, even in these conditions, artificial agents can be evolved that show significant adaptivity thanks to the coupling of their 'inner worlds' and the external environment.

The second paper is contributed by Alberto Montebelli, Robert Lowe, and Tom Ziemke (The Cognitive Body: From Dynamic Modulation to Anticipation). Here the basic agent architecture is characterized by the presence of a motivational internal state, having slowly changing dynamics, which modulates the agent's activity. This architecture is then augmented with an anticipatory mechanism, which is directly coupled to the internal unit. The anticipatory mechanism significantly enhances the agent's adaptivity. The most peculiar aspect of this paper is that – in contrast to standard implementations – anticipation operates through bodily mediation by modulating the internal dynamics, rather than by triggering direct behavioral responses.

3.4 Computational Modeling of Psychological Processes in the Individual and Social Domains

Four papers are devoted to modeling and interpreting specific psychological experiments. In particular, they investigate sustained inattentional blindness, the foreperiod paradigm, collision avoidance behavior, and cooperative dynamics in the Iterated Prisoner's Dilemma game.

Anthony F. Morse, Robert Lowe, and Tom Ziemke (A Neurocomputational Model of Anticipation and Sustained Inattentional Blindness in Hierarchies) present a model of how sustained inattentional blindness results from a process of anticipation of task-relevant features. In a simulated 'input tracking' task, the authors find that anticipation enhances performance of the task and simultaneously degrades detection of unexpected features, thereby modeling the sustained inattentional blindness effect.

Johannes Lohmann, Oliver Herbort, Annika Wagener, and Andrea Kiesel (Anticipation of Time Spans: New Data from the Foreperiod Paradigm and the Adaptation of a Computational Model) first review empirical and theoretical literature on the foreperiod paradigm, where subjects are asked to react to events that occur at more or less unpredictable times after a warning stimulus (foreperiod). Then in a model they systematically vary predictability of the foreperiods and find adaptation to different probability distributions with a pronounced adaptation for the peaked more-predictable one. Finally the authors discuss their results in relation to the computational model proposed by Los and colleagues [8].

Janneke Lommertzen, Eliana Costa e Silva, Raymond H. Cuijpers, and Ruud G.J. Meulenbroek (Collision-Avoidance Characteristics of Grasping: Early Signs in Hand and Arm Kinematics) have studied prehension kinematics and

collision-avoidance strategies in grasping tasks. Their study shows that different forms of objects (small or large cylinders) elicits different approaching phases and subjects successfully avoid collisions by adapting the last phase of their movements (aperture overshoots) and by adjusting the movements of their distal joints. The authors relate their study to computational models of reaching and collision avoidance and succeed in replicating and interpreting the empirical results within a robotic set-up.

The last paper, contributed by Maurice Grinberg and Emilian Lalev (The Role of Anticipation on Cooperation and Coordination in Simulated Prisoner's Dilemma Game Playing) studies anticipatory strategies in a cooperative social task: the Iterated Prisoner's Dilemma game ("IPD"). The authors first describe the results of experiments on IPD obtained with humans and then investigate and interpret such data on the basis of a connectionist model. Within genetic simulations, the model shows how under certain circumstances anticipatory strategies emerge and lead to increased cooperation payoffs, therefore making the case that anticipation is a key ingredient for having a high level of cooperative coordination in simulated and real societies.

3.5 Behavioral and Cognitive Capabilities Based on Anticipation

As discussed in the introduction, anticipation plays a major role in the acquisition and use of multiple behavioral and cognitive capabilities. The three papers of this section address the role of anticipation in learning a stand-up posture, in space perception, and in planning.

The first paper, contributed by Camille Salaun, Vincent Padois, and Olivier Sigaud (A Two-level Model of Anticipation-based Motor Learning for Whole Body Motion) presents a model of motor learning that combines Operational Space Control and Optimal Control. The paper demonstrates the efficacy of the latter approaches in a simulated robotic task that consists in learning to stand-up. In the model anticipation has a two-fold important function, namely learning the dynamics model of the system and coordinating the two types of control.

The paper contributed by Wolfram Schenck (Space Perception through Visuokinesthetic Prediction) follows the "perception through anticipation" approach of [9] and demonstrates how objects can be localized by generating a visuokinesthetic (iterative) simulation of reaching with a robotic arm. Within this framework, space perception arises from the knowledge of how to move (e.g. push) an object. Anticipation is used for sensory prediction and novelty detection.

The last paper in this section, contributed by Irene Markelic, Tomas Kulvicius, Minija Tamosiunaite, and Florentin Worgotter (Anticipatory Driving for a Robot-Car Based on Supervised Learning) may be the most applicative of the book. The authors construct a database that couples look-ahead sensory information and action sequences. The constructed knowledge is the used to train a car-like trajectory planning robot that runs at real-time by issuing steering and velocity control commands in a human manner.

3.6 Computational Frameworks and Algorithms for Anticipation and Their Evaluation

The last section of the book includes three papers that discuss basic issues of anticipatory mechanisms and algorithms.

Birger Johansson and Christian Balkenius (Prediction Time in Anticipatory Systems) have run simulated robotic experiments in a guard-and-thieves scenario with the aim of assessing what is the best length of the future time interval in which thieves should anticipate the movement of the guard in order to successfully trick them and steal a treasure. Their results show that it is not always better to predict long into the future and that the best performance is indeed achieved when the time spent planning is comparable to the time it will take to perform the tasks.

The paper contributed by Matthias Rungger, Hao Ding, and Olaf Stursberg (Multiscale Anticipatory Behavior by Hierarchical Reinforcement Learning) presents a two-level hierarchical reinforcement learning scheme that combines a discrete representation (finite state automaton) at the higher layer, and a continuous representation at the lower layer. The results of the test of the model within in a robot grasping task show that the iteration between both layers permits to autonomously determine suitable solutions to new tasks.

Olivier Sigaud, Martin V. Butz, Olga Kozlova, and Christophe Meyer (Anticipatory Learning Classifier Systems and Factored Reinforcement Learning) compare both conceptually and empirically Factored Reinforcement Learning (FRL) and Anticipatory Learning Classifier System (ALCS) techniques. Their empirical comparison reveals that an instance of the latter (XACS) scales much better than an instance of the former (SPITI) in two benchmark problems. The authors conclude the work by analyzing what are the key mechanisms in XACS that permit better performance, and propose importing them into FRL systems.

4 Conclusions and Important Open Issues on Anticipation

As diverse as the included contributions are, as overarching their perspective and inclusive the highlighted aspects of anticipatory mechanisms and behavior. While the benefits of anticipatory mechanisms and the ubiquitous presence of anticipatory behavior in various levels of perceptual processing and motor control becomes increasingly clear, the challenge of combining these different anticipatory mechanisms appropriately and efficiently appears to be the next challenge in anticipatory processing systems.

In this respect, it is hoped that the next years will provide theoretical and implementation advances on the interactions between different anticipatory functions and mechanisms, with a particular emphasis on the different sensory-motor feedback loops and the learning mechanisms which might allow both the acquisition of predictive capabilities and their exploitation for guiding action. The book shows that several ideas do exist on these issues in various forms, so in the

near future it is paramount to invest research efforts to organize them within comprehensive frameworks and, more importantly, within whole integrated architectures.

Acknowledgments. This work was supported by the EU funded projects *HUMANOBS: Humanoids That Learn Socio-Communicative Skills Through Observation*, contract no. FP7-STREP-231453, and *IM-CLeVeR - Intrinsically Motivated Cumulative Learning Versatile Robots*, contract no. FP7-IST-IP-231722. Additional support comes from the Emmy Noether program of the German Research Foundation grant BU1335/3-1.

References

1. Butz, M.V., Herbort, O., Hoffmann, J.: Exploiting redundancy for flexible behavior: Unsupervised learning in a modular sensorimotor control architecture. Psychological Review 114, 1015–1046 (2007)
2. Butz, M.V., Sigaud, O., Gérard, P.: Anticipatory behavior: Exploiting knowledge about the future to improve current behavior. In: Butz, M.V., Sigaud, O., Gérard, P. (eds.) Anticipatory Behavior in Adaptive Learning Systems. LNCS, vol. 2684, pp. 1–10. Springer, Heidelberg (2003)
3. Butz, M.V., Sigaud, O., Gérard, P. (eds.): Anticipatory Behavior in Adaptive Learning Systems. LNCS (LNAI), vol. 2684. Springer, Heidelberg (2003)
4. Butz, M.V., Sigaud, O., Gérard, P.: Internal models and anticipations in adaptive learning systems. In: Butz, M.V., Sigaud, O., Gérard, P. (eds.) Anticipatory Behavior in Adaptive Learning Systems. LNCS, vol. 2684, pp. 86–109. Springer, Heidelberg (2003)
5. Butz, M.V., Sigaud, O., Pezzulo, G., Baldassarre, G. (eds.): ABiALS 2006. LNCS, vol. 4520. Springer, Heidelberg (2007); The book is a result from the third workshop on anticipatory behavior in adaptive learning systems, ABiALS 2006, Rome, Italy, September 30, 2006, colocated with SAB 2006
6. Hoffmann, J., Stöcker, C., Kunde, W.: Anticipatory control of actions. International Journal of Sport and Exercise Psychology 2, 346–361 (2004)
7. Hommel, B., Musseler, J., Aschersleben, G., Prinz, W.: The theory of event coding (TEC): A framework for perception and action planning. Behavioral and Brain Science 24(5), 849–878 (2001)
8. Los, S., Knol, D., Boers, R.: The foreperiod effect revisited: Conditioning as a basis for nonspecific preparation. Acta Psychologica 106, 121–145 (2001)
9. Möller, R., Schenck, W.: Bootstrapping cognition from behavior—a computerized thought experiment. Cognitive Science 32(3), 504–542 (2008)
10. Pezzulo, G., Butz, M.V., Castelfranchi, C., Falcone, R. (eds.): The Challenge of Anticipation. LNCS (LNAI), vol. 5225. Springer, Heidelberg (2008)
11. Rosen, R.: Anticipatory Systems. Pergamon Press, Oxford (1985)

ABC: A Psychological Theory of Anticipative Behavioral Control

Joachim Hoffmann

Department of Psychology
University of Würzburg
Röntgenring 11, D 97070 Würzburg
hoffmann@psychologie.uni-wuerzburg.de

Abstract. Almost all behavior is purposive or goal oriented. People behave, for example, in order to cross the street, to open a door, to ring a bell, to switch on a radio, to fill a cup with coffee, etc. Likewise, animals behave to attain various goals as for example to escape from a predator, to catch prey, to feed their offspring, etc. The ABC framework accords with the purposive character of almost all behavior by assuming that behavior is not determined by the current stimulation but by the desired or the 'to-be-produced' effects. For this to work, behavioral acts have to be connected to the effects they produce in such a way that anticipations of effects gain the power to address the behavior that brings them about (often called the ideo-motor principle). Moreover, if action-effect contingencies systematically depend on the situational context, the formed action-effect relations have to be contextualized. Accordingly, the ABC framework assumes the formation of representations that preserve information about which effects can be realized by which behavior under which conditions. In the present article we review some of the empirical evidence in favor of the ABC approach and discuss the structures by which sensory anticipations might be transformed into the motor patterns that move the body to bring the desired effects about.

1 The Limits of the Information Processing Approach

In the second half of the last century, information processing replaced behaviorism as the leading approach in theoretical and experimental psychology (cf. [44]). This development was induced by new insights in other sciences in particular in mathematics, communication, and system analyses: Norbert Wiener [63] established "Cybernetics" as a new science for the analysis of informational processes in machines and animals. One year later, Shannon and Weaver [54] provided a mathematical calculus for the measurement of information. Concurrently, Alan Turing [59] discussed intelligence as a feature of computing machines and John von Neumann [61] delivered the architecture for such intelligent machines. All these developments awaked the belief that also humans can be described and analyzed as information processing systems.

This belief in the applicability of the information processing approach on the analysis of psychic processes was strongly nurtured, when Hick [21] reported

G. Pezzulo et al. (Eds.): ABiALS 2008, LNAI 5499, pp. 10–30, 2009.

that the latencies of simple choice reactions increased linearly with the entropy of the presented stimulus. And when Newell and Simon [46] implemented the first computer program that was able to solve challenging problems, such as the 'Tower of Hanoi', many psychologists became convinced that higher mental processes must be studied from an information processing perspective. Thus, the information processing approach emerged and research efforts henceforward were concentrated on mental processes such as perception, attention, language, reasoning, and memory.

Although the information processing approach overcame the theoretical restrictions of behaviorism on merely stimulus-response relations, the now explored mental activities were still considered as determined or driven by stimulation. For example, in his seminal book "Cognitive Psychology", Ulric Neisser [45] defined cognition as referring ...to all the processes by which the sensory input is transformed, reduced, elaborated, stored, recovered, and used. Thus, also in the new information processing perspective, the unfortunate doctrine of behaviorism survived, which posits that 'all' starts with the impact of stimuli on the organism. The question of how the stimuli drive behavior was merely shifted to the question of how stimuli are processed in order to create an internal representation of the information they transmit.

In the present article, I propose that this view is misleading if not basically wrong. There are at least two arguments, which put the information processing approach into question.

1. The information processing approach suggests that stimulus information is processed to build a veritable mental representation of the 'information source', i.e. the 'environment'. However, there is no unique environment, which is to be represented. For example, if you look at Figure 1 you certainly will see, i.e. you will mentally re-present, this stimulus as being two interwoven squares. However, there are also eight triangles or the shape of a house with some extra brackets, etc. In general, stimulations from the environment contain information about countless properties from which we always perceive or process only an evanescent part. Thus, the question is not how we, or any other animal, process the given stimuli in order to create a veritable representation of the transferred information, but rather the question is what determines the particular information that is selected for processing.

   If one compares the perception of different species, it becomes obvious that what a species can perceive is primarily determined by the behavioral requirements the species has to face. For example, bats are in particular sensitive for sound waves of 50 kHz because they use such waves for echo-navigation and frogs are especially sensitive for small, fast moving dots in their visual field because the dots signal the presence of a potential prey etc. Thus, organism perception is above all determined by the information they need in order to behave successfully.
2. Any movement of our body produces changes in the sensory input the so called reafferences [60]. Whether you move your finger, your eyes, and even if

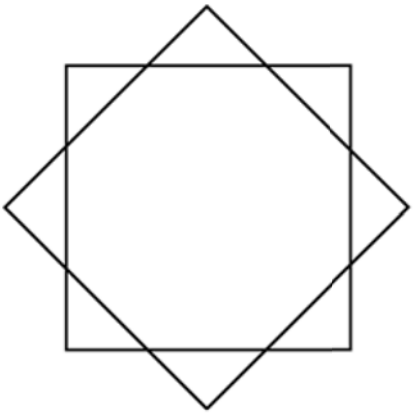


Fig. 1. Two squares, eight triangles, or the outline of a house with additional angles

you talk, you produce various new sensory input sorts of sensory stimulation. Accordingly, you have to distinguish which aspects of the current stimulation were produced by yourself and which ones might have other causes. And this is true for any and every animal, even for such primitive organisms like an earthworm: Without distinguishing the sensory consequences of one's own behavior from other sensory inputs, active organisms could not make at all any meaningful use of stimulus information. Thus, every active organism has to learn what the sensory consequences of its own behavior are.

Both arguments emphasize the importance of the interplay between stimuli and behavior instead of the relations between stimuli and representations. Accordingly, one may claim that the primary function of cognition is not the processing of stimulus information but rather the control of stimulus production (cf. also [47]). In the following, I will elaborate on this claim both theoretically and experimentally.

2 The Primacy of Action-Effect Learning Over Stimulus-Response Learning

According to classical behaviorism, all behavior is finally due to stimulus-response relations and stimulus-response learning is the basis of all behavioral changes [58,62]. The tenet is indeed supported by countless experiments in animal learning, in particular, by experiments in discriminative conditioning. In discriminative conditioning a particular behavior is reinforced only if stimulus A is present but the behavior is never reinforced if stimulus B is present. In the test it appears that only stimulus A but not stimulus B evokes the formerly reinforced behavior. Accordingly, it is concluded that an associative connection has been formed between stimulus A and the particular response so that the stimulus gained the power to evoke the associated response.

However, the conclusion is premature. For example, if one varies not the stimulus conditions and the reinforcer but the behavior and the reinforcer, it soon becomes obvious that behavior is not determined by stimulus-response but by response-effect relations. Imagine, for example, rats that experience that in the experimental cage chain pulling leads to a reinforcement by some food pellets and lever pressing leads to a reinforcement by a drop of sugar solution. After this experience has made one of the reinforcers, let us say the sugar solution, becomes devaluated by adding something that causes a mild nausea whenever the rats are drinking from it in their home cage. If then the rats again have access to the chain and the lever in the experimental cage, they clearly avoid pressing the lever, which would lead to the meanwhile devaluated reinforcer, whereas they do not hesitate to pull the chain, which would lead to the still valuable food pellets (cf. [9]).

The avoidance of respectively that behavior that would lead to the devaluated reinforcer allows to conclude (1) that the rats have formed associations between the actions and their respective outcomes, i.e. action-effect associations have been formed, and (2) that in the test the behavioral choice is determined by an anticipation of the respectively anticipated reinforcer and not by the current stimulation (i.e. the experimental cage).

Meanwhile countless experiments demonstrated that action-effect relations outrange stimulus-response relations in the determination of animal behavior (e.g. [49,10])[1]. Surprisingly, in humans' stimulus-response and response-effect learning has been rarely, if ever, directly compared except in a study by Stock & Hoffmann [57], which is shortly discussed next.

Participants get presented one start- and one goal-symbol on a computer screen, both selected from a set of four possible figures (a star, a hexagon, a rhombus, and a "sun", cf. Figure 2). Participants were instructed to find out which one of four possible response keys were to be pressed in order to attain the current "goals" in the presence of the current "start". Among others, we varied in one of a series of experiments the feedback (the reinforcer so to say): For half of the participants it was merely fed back whether the current response key led to a "hit" or a "failure". For the other half of the participants, the key presses triggered the presentation of another effect-symbol on the screen, which could either match (a hit) or could not match (a failure) the current goal-signal.

In both cases, simple stimulus-response relations had to be learned as to each of the four start-symbols one of the four keys was assigned, which was always successful, while all other keys failed in the presence of this particular start symbol. Thus, in the presence of a certain start-symbol there was a certain key to press in order to produce the feedback "hit" (in the former condition) or to trigger another presentation of the current goal-symbol whatever it was (i.e. a hit in the latter condition). This seemingly tiny manipulation of the feedback had dramatic consequences for the learning rate (cf. Figure 3): If only "hits" and "failures" are fed back, participants learn very fast that the key to select in order

[1] This does not mean that the current stimulation lose any influence on behavior as we will discuss in section 4 of this paper.

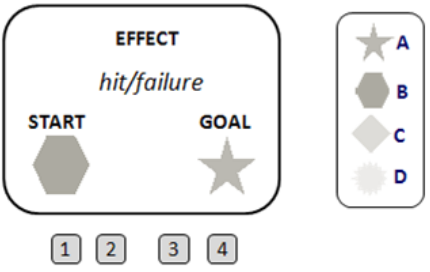


Fig. 2. Illustration of the conditions in an experiment by Stock & Hoffmann 2002. Only the feedback is shown which informs about hits and failures.

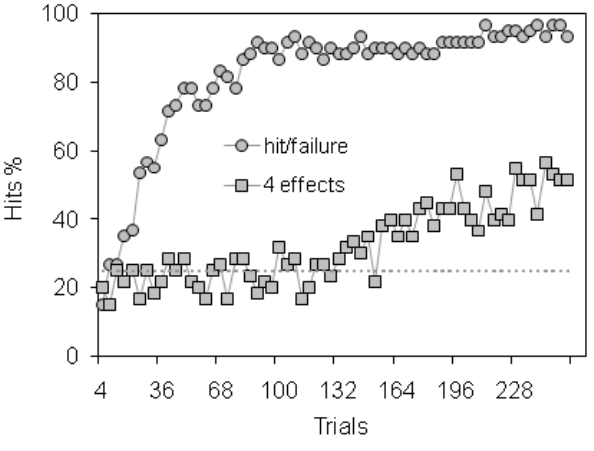


Fig. 3. The percentage of hits plotted against the number of learning trials in dependence on whether only hits and failures or four different effects are fed back

to produce a hit depends on the current start-symbol. However, if pressing the keys resulted into the presentation of another symbol on the screen only three of fifteen participants learned the critical start-key relations whereas all other participants despaired and were convinced to be fooled by the experimenter.

Figure 4 illustrates our account of this striking difference: Under the reduced feedback, participants have no other option than to strive for the feedback "hit" and they experience that every key sometimes produces a "hit" and sometimes it does not. Accordingly, participants try to find out the critical condition from which the success of each key may depend and they quickly learn that the success depends on the current start-symbol so that each start-symbol requires a certain key to press in order to launch a "hit".

In contrast, under the elaborated feedback, participants strive to find out which key is to press in order to trigger another presentation of the current goal-symbol (i.e. a hit). If accidentally the correct key has been pressed, participants try to store the experienced successful key-effect relation, that is, they try

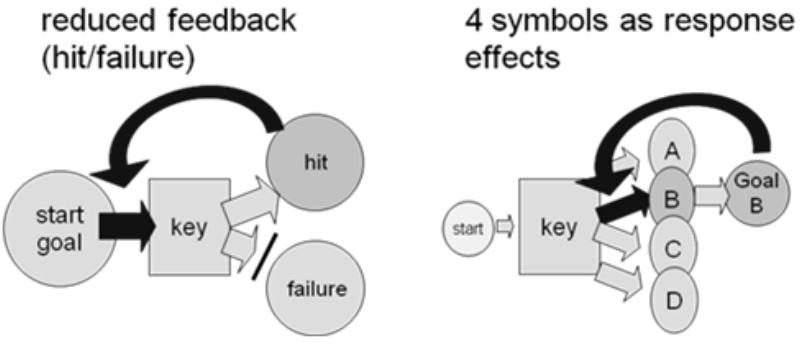


Fig. 4. An illustration of the impact different feedback has on learning: If only hits and failures are fed back associations between the successful keystrokes and the current stimuli are formed (left side). However, if the keystrokes produce distinctive effects, associations between the successful keystrokes and their current effects are formed (right side).

to store that the currently pressed key is appropriate to produce the currently presented goal-symbol as its effect[2]. The concurrently given start-key relations, however, are not noticed so that the participants remain blind for their regularity. In more general terms: If behavior results into different goal related effects, learning is primarily directed onto the acquisition of the proper action-effect contingencies, which in turn blocks learning of concurrent stimulus-response contingencies. Thus, the data nicely demonstrate that the primacy of action-effect learning over stimulus-response learning does not only hold for rats but also for humans.

3 Anticipations Even of Non-intended Effects are Indispensable in the Determination of Voluntary Behavior

According to the preceding discussion, voluntary behavior is primarily determined by anticipations of the sensory effects the behavior produces instead of being determined by the current stimulation. This insight can be traced back more than 150 years to scholars like Herbarth [17], Lotze [43] and [15]; cf. [56] for an overview). William James [32] finally used the term "ideo-motor principle" to denominate the notion that the motor output is determined by an idea (anticipation) of the desired outcome: "An anticipatory image ... of the sensorial consequences of a movement, ... is the only psychic state which introspection lets us discern as the forerunner of our voluntary acts." [32, p.1112].

[2] Remember that in the presence of a certain start-symbol each of the keys always triggered the presentation of the current goal-symbol again, whatever it was. Accordingly there were no systematic key-goal relations to detect.

If we define voluntary behavior as behavior by which organisms strive for a certain goal, it follows by definition that the goal has somehow to be represented in advance because otherwise the respective behavior would not be voluntary. Thus, in order to verify the ideo-motor principle it needs to be not only shown that anticipations of the intended outcomes precede the voluntary behavior (this is trivial) but that anticipations also of non-intended behavioral effects take active part in the determination of the respective behavior.

Recently, the integration of incidental behavioral effects in the control of simple voluntary acts like pressing a button has become subject of numerous studies, which preferred a methodological approach already suggested by Greenwald [14]: In reaction time tasks, participants practice responses that produce distinctive but unintended sensory effects. Concurrently or subsequently, it is tested whether the incidental effects have gained the power to address the actions they were effects of. The test is mostly conducted by presenting the experienced effects as the imperative stimuli to trigger either the responses they formerly were the effects of or to trigger responses they formerly did not follow as effects. The results typically show that responses are performed faster and less error prone if they are triggered by their former effect-stimuli compared to corresponding control conditions, which indicates that the incidental effects are not only associated with the preceding responses but that they become indeed involved in response generation (e.g. [3,11,12,20,26,28,29,30,65,66]).

The evidence discussed so far convincingly shows that the presentation of stimuli that have been experienced as response effects facilitates the generation of the responses they were previously the effect of. However, the ideo-motor principle claims that anticipations and not presentations of the effects determine voluntary behavior. Thus, the reported evidence is consistent with the ideo-motor principle but not yet "on the point".

Anticipations are subjective entities and are consequently difficult to control experimentally. However, if any access of a voluntary movement does indeed require an anticipation of its sensory effects, manipulations of the to-be-expected effects should have an impact on the access to the movement that produces these effects. Following this logic, Kunde [39] recently provided convincing evidence for the more specific claim of the ideo-motor principle that not only effect presentations but also effect anticipations contribute to the control of voluntary behavior.

Kunde [39] started from the well established stimulus-response compatibility effect: If in a choice reaction time experiment the imperative stimuli and the required responses vary on a common dimension (dimensional overlap), compatible S-R assignments are faster accomplished than incompatible assignments (cf. Kornblum, Hasbroucq, & Osman, 1990). Consider for example, spatial compatibility: if participants have to respond to left and right stimuli with the left and right hand, they respond faster with the left hand to a left stimulus and with the right hand to a right stimulus than vice versa (e.g. [55]). Ongoing from S-R compatibility, Kunde [39] proceeded to argue that if selecting and

initiating a response does indeed require the anticipation of its sensory effects, the same compatibility phenomena should appear between effects and responses as between stimuli and responses.

Imagine, for example, that participants are asked to press a button either softly or strongly in response to an imperative color signal. Each key press produces either a quiet or a loud effect tone. In the compatible assignment a soft key press produces a quiet tone and a strong key press produces a loud tone. In the incompatible case, the assignment is reversed. The results show that participants responded significantly faster if their responses triggered tones of compatible intensity than if they triggered incompatible tones. This response-effect compatibility phenomenon meanwhile has been proven to be a very robust one. The phenomenon occurs in the dimensions of space, time, and intensity [39,40,38,41,37]. As in all these experiments, the effects were not intended but appeared incidentally after the execution of the response. Their impact on response latencies proves that representations also of these non-intended effects were activated before the responses were selected and initiated. The use of response alternatives that differ in intensity additionally allowed a qualification of response execution. For example, if participants are required to complete a soft or a strong key press the peak force that is reached provides an appropriate measure of response execution, allowing to explore whether response-effect compatibility would affect not only reaction times but also response execution. This was indeed the case. The intensity of the effect-tones uniquely affected the peak forces of soft as well as of strong key presses in a contrast like fashion. As Figure 5 illustrates, loud effect-tones reduced and quiet effect-tones intensified the peak forces of intended soft key presses as well as of intended strong key presses.

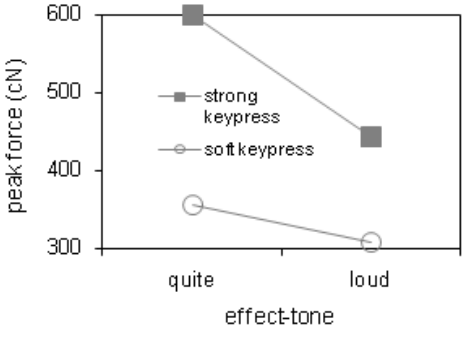


Fig. 5. The peak force for intended strong and soft key presses in dependence on the intensity of the effect-tones the keystrokes produced

For an appropriate account of the found contrast, it is to notice that peak forces indicate the intensity of the tactile feedback by which participants start to reduce the force of their hand because they feel the intended force

(strong or soft) to be reached. In this view, the data show that less strong tactile feedback is required to feel the intended force completed if a loud effect-tone follows and stronger tactile feedback is needed if a quiet effect-tone follows. Figure 6 illustrates two possible accounts for this contrast. A simple feedback loop for the execution of a prescribed pressure force is depicted: The imperative stimulus determines the set point (the proximal reference), i.e. the proprioceptive feeling is anticipated which has to be reached in order to realize either a strong or a soft key press. The difference between the set point and the current feeling (the current proximal feedback) determines the appropriate motor commands which are activated until the proprioceptive feedback from the finger tip and from the muscles signal that the set point is reached.

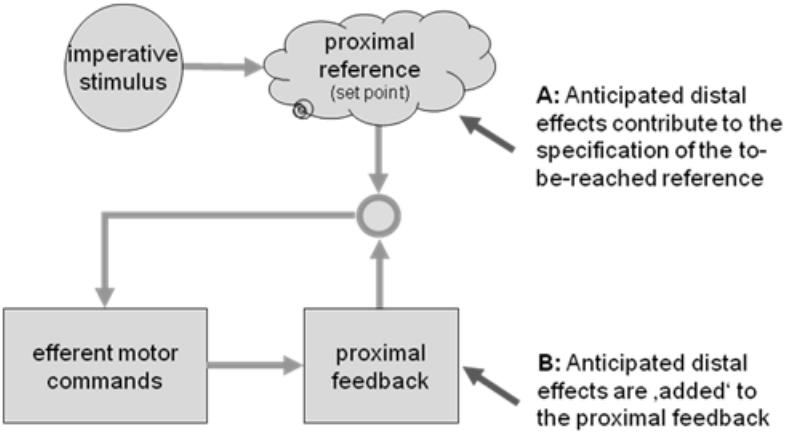


Fig. 6. Illustration of two possible points of action at which anticipated effects might affect behavioral control

Within this loop the additionally anticipated intensity of the distal effect-tone might on the one hand (A) influence the set point so that the set point is somewhat enhanced if a quiet tone is anticipated, and the set point is somewhat reduced if a loud tone is anticipated. In this way the intended force of the key press would be adjusted in order to compensate for the anticipated force of the effect tones. On the other hand (B) it might be that the anticipated intensity of the distal effect-tone is charged to the feedback so that an anticipated loud tone earlier evokes the feeling that the set point is reached and an anticipated quiet tone delays somewhat this feeling. Both mechanisms provide an account for the contrast effect and it might be that they both conjointly contribute to it. In any case, the present data provides profound evidence that anticipations even of unintended response effects are not only involved in the selection and initiation of voluntary actions but also take part in the control of their execution.

4 Anticipative Behavioral Control Becomes Conditioned to the Situational Context

As convincing the evidence for the determination of voluntary behavior by anticipations of its intended and non-intended effects might be, it would be silly to deny the contextual impact situations have on behavior. For example, if a bus driver who drives home in his private car stops at a bus stop, his behavior is obviously not determined by his goal to drive home but rather by perceiving the bus stop, which immediately evokes the habit to stop there [16]. Indeed, several theoretical conceptions in psychology acknowledged the fact that situations may attain the power to evoke associated behavior. For example, Lewin (1928) spoke in this context of the "Aufforderungscharakter" of objects, Ach [1] coined the term 'voluntive Objektion', and Gibson [13] argued that objects are not only to be characterized by their physical features but also by their 'affordances'. All these terms refer to the fact that suitable objects often afford us to do the things we mostly do with them and that they immediately trigger habitual behavior if one is already ready for doing it. For example, if one intends to post a letter, the sight of a mailbox immediately triggers the act of posting and in driving a car, flashing stop lights of the car ahead immediately evokes applying the brakes.

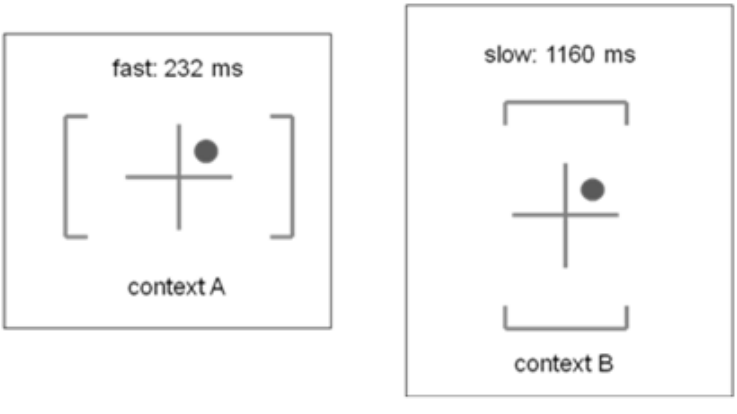


Fig. 7. Illustration of the experimental settings in Kiesel & Hoffmann [36]. Participants were told that the dot represents a "ball" and the brackets represent "goals" and that they are requested to push the ball as fast as possible into the respectively adjacent goal. When the goals were horizontally arranged, balls in the left quadrants had to be pushed with the left button and balls in the right quadrants had to be pushed with the right button whereas, when the goals were vertically arranged, the upper quadrants were assigned to the right and the lower quadrants to the left button. In all cases the ball moved to the respective goal as soon as the correct button was pressed. However, in order to vary a non-intended property of this visual effect, the ball moved quickly (in 232 ms) if the goals were horizontally arranged and the ball moved slowly (in 1160 ms) if they were vertically arranged. Accordingly, one and the same action resulted in either a slow or a fast ball movement depending on the current context.

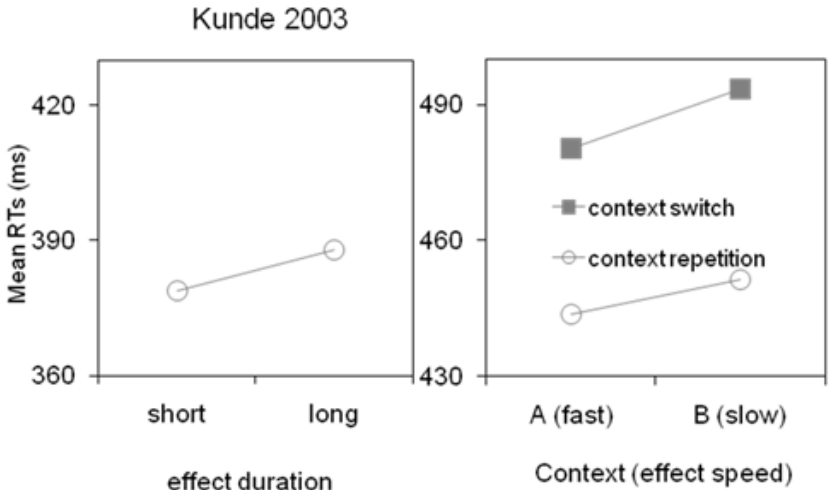


Fig. 8. Reaction times (RTs) in dependence of the duration of the sensory effects produced by the currently required response

In order to reach a more complete picture of the representations that underlie behavioral control, the integration of situational features also need to be considered. The situational context presumably becomes integrated into behavioral control either if a particular context repeatedly accompanies the attainment of a particular effect by a particular action or if situational conditions systematically modify action-effect contingencies (cf. [25]). Especially the latter deserves attention as it points to the frequent case that the effects of an action change with the situational context, as, for example, the effect of pressing the left mouse button may dramatically change with the position of the cursor. There is no doubt that people learn to take into account critical situational conditions in order to attain the intended effects, but the issue to what extent the situational context might also affect anticipations of situation-specific non-intended effects remains to be discussed.

If one is going to explore whether the same action is preceded by different effect anticipations in dependence on the situational context, first, one has to vary the effects of the same responses in different contexts and second, one has to show that respectively those effects are anticipated that correspond to the current context. A corresponding study was recently reported by Kiesel and Hoffmann [36]: In a choice reaction time experiment participants were presented with a cross and a dot in one of its quadrants framed by either two horizontally or two vertically arranged brackets (cf. Figure 7).

Experiments by Kunde [40] had shown that reaction times increase with the duration of the effect tone the currently required response produces. Figure 8 shows this finding. Thus, if the velocity of the ball-move would indeed be anticipated, the responses should be somewhat delayed when a slow movement is

to be expected compared to when a fast movement is to be expected. Exactly this result was found (see the right graph of Figure 8): When the context indicated a slow movement, reaction times were consistently increased in comparison to when the context indicated a fast movement of the ball. Additionally, it took some extra time if the context had been switched in comparison to the previous trial, but the influence of the effect duration clearly was independent of these "switch costs". Thus, the data confirmed that the very same actions were not only preceded but also affected by anticipations of either a fast or a slow movement depending on the current situational context, which indicates that participants acquired and used context-specific effect anticipations.

5 ABC: An Integrative Framework

In order to integrate the discussed relationships between voluntary actions, their effects, and situational contexts, we [22,23,24,27] proposed a tentative framework that takes into account the determination of voluntary behavior by effect anticipations and the conditionalization of action-effect relations on critical situational contexts as well. The framework is based on the following assumptions (cf. Figure 9):

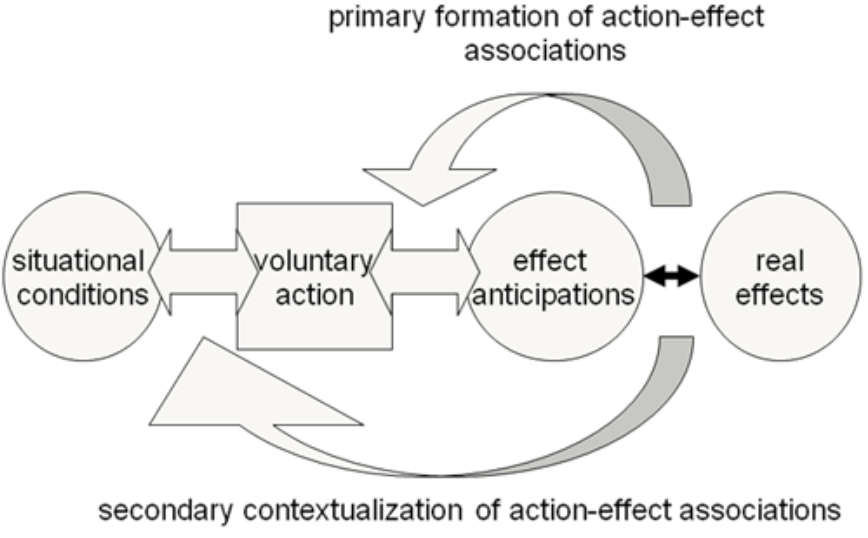


Fig. 9. Illustration of the ABC framework: The acquisition of anticipative structures for the control of voluntary behavior

1. A voluntary action is defined as performing an act to attain some desired outcome or effect. Thus, a desired outcome, as general and imprecise as it might be specified in the first place, has to be represented in some way before a voluntary action can be performed. Consequently, it is supposed that any voluntary act is preceded by corresponding effect anticipations.

2. The actual effects are compared with the anticipated effects. If there is sufficient coincidence between what was desired and what really happened, representations of the just-performed action and of experienced effects become interlinked, or an already existing link is strengthened. By this, action representations become linked to intended as well as non-intended effects provided that the effects are contingently experienced as outcomes of the preceding act. If there is no sufficient coincidence, no link is formed, or an already existing link is weakened. This formation of integrated action-effect representations is considered as being the primary learning process in the acquisition of behavioral competence.
3. It is assumed that situational contexts become integrated into action-effect representations, either if a particular action-effect episode is repeatedly experienced in an invariant context or if the context systematically modifies the contingencies between actions and effects. This conditionalization of action-effect relations is considered as being a secondary learning process.
4. An awakening need or a concrete desire activates action-effect representations whose outcomes sufficiently coincide with what is needed or desired. Thus, anticipations of effects address actions that are represented as being appropriate to produce them. If the activated action-effect representations are conditionalized, the coincidence between the stored conditions and the current situation is checked. In general, an action will be performed that in the current situational context most likely produces the anticipated effect.
5. Conditionalized action-effect representations can also be addressed by stimuli that correspond to the represented conditions. Thus, a certain situational context in which a certain outcome has been repeatedly produced by a certain action can elicit the readiness to produce this outcome by that action again.

The sketched framework integrates, still rather roughly, important aspects of the acquisition of behavioral competence: First, it considers the commonly accepted fact that behavior is almost always goal oriented instead of being stimulus driven. Second, it is assumed that any effect that meets an anticipated outcome will act as a reinforcer. Consequently, learning is not only driven by a satisfaction of needs but by anticipations, which can flexibly refer to any goal. Third, the framework considers the given evidence that voluntary behavior is primarily determined by action-effects instead of by stimulus-response associations. Finally, also stimulus driven habitual behavior is covered, as it is assumed that action-effect relations become conditionalized and can be evoked by the typical contexts in which they are experienced.

Although on a conceptual level the ABC framework is consistent with a huge body of empirical evidence it still fails to give an account on how sensory anticipations are transformed into the motor patterns, which let the body move so that the anticipated effects are really produced. We now discuss this concern in further detail.

6 How Sensory Anticipations Might Be Transformed into Appropriate Motor Patterns

It can be taken for granted that "ideo-motor" transformations comprise proprioceptive as well exteroceptive effects of the intended behavior. For example, if Cole [8] describes how a deafferented patient (i.e. a patient without any proprioception) is unable to maintain an upright posture in darkness, it becomes obvious that proprioceptive feedback is indispensable even for the simplest motor control (cf. also [33,2,52,7]. Also exteroceptive, especially visual feedback is fundamental even for simple and highly trained grasping movements. For example, blocking of visual feedback causes strong disturbances of a simple grasping movement despite the movement was extensively trained [48]. Thus, it appears that motor patterns are controlled by anticipations of the to-be-produced exteroceptive as well as proprioceptive effects. However, there are reasons to assume that exteroceptive and proprioceptive effects play a different role in the determination of concrete body movements.

Proprioceptive effects covary very systematically with the efferent activation patterns they are produced from, so that each of the various properties of a certain body movement as for example its strength or its velocity finds its counterpart in a corresponding proprioceptive feeling [50]. Accordingly, anticipations of proprioceptive effects can be specified to a degree which determines all parameter of a definite movement. In contrast, aspired exteroceptive effects like opening a door, switching on a device, grasping an object etc. are almost never accomplished by the very same movements. The same outcomes rather can and are typically attained by numberless body-movements. This is the well known redundancy problem in motor control [4]. Accordingly, anticipations of exteroceptive effects in most cases do not specify a definite movement but rather a whole set of possible movements (e.g. [5,51]). Finally, even if one has learned to attain a certain exteroceptive effect by a certain movement of one limb, the learned goal-movement relation can be easily transferred to another limb. For example if one trains to reach a goal by the left hand, the learned trajectory is immediately transferred to the untrained right hand (e.g. [42,53]).

Altogether the preceding considerations convincingly suggest that a desired exteroceptive effect, an environmentally related goal so to say, almost never specifies a definite body movement to bring the effect about[3]. It rather appears to be likely that anticipated exteroceptive effects first are transferred into states of a body-related space to which all limbs have equal access. Accordingly, desired exteroceptive effects become recoded into desired bodily related but still

[3] In the "Theory of Event Coding" (TEC, [31]), the authors emphasize that actions are primarily represented by codes of the aspired exteroceptive or distal effects. However, TEC explicitly deals only with 'early' cognitive antecedents of actions that stand for, or represent, certain features of events that are to be generated in the environment. TEC does not consider the complex machinery of the "late" motor processes that subserve their realization (i.e., the control and coordination of movements). Here it is argued that anticipations of the proprioceptive effects of actions are indispensable to "translate" desired distal effects into appropriate motor patterns.

effector-unspecific effect. Only then, such effector-unspecific goal representations might be transformed into effector specific anticipations of to-be-produced proprioceptive feelings, which finally determine the corresponding movement of a certain limb.

According to this view, at least three modes of anticipations are involved in behavioral control: anticipations of to-be-reached states in the environment, anticipations of to-be-reached states in an effector-unspecific "body space", and finally anticipations of to-be-reached states of a definite effector. If we add the idea that for each of these modes sensory feedback is used in order to control the progress of goal achievement, dynamic aspects of action-control come into focus, which we have neglected so far. Because feedback needs time and because the required amount of time differs between the different modes, the slower loops must determine the faster ones in order to hold control steady. Accordingly, the picture of hierarchically organized feedback loops emerges (Figure 10; cf. [47] for a comparable account):

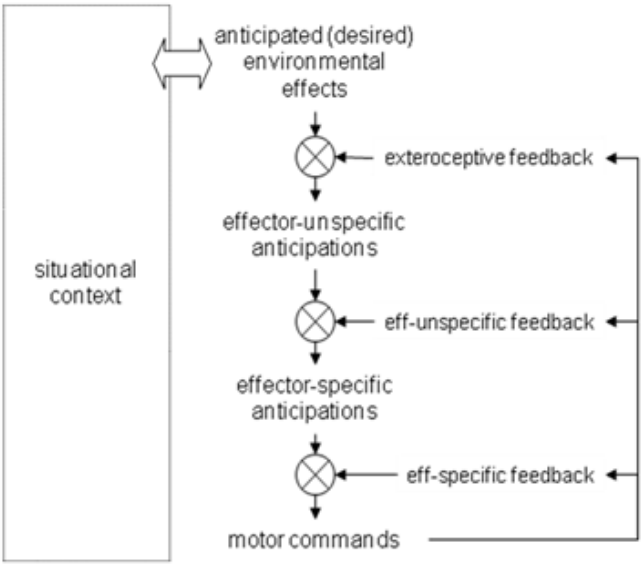


Fig. 10. A rough sketch of the assumption that anticipated sensory effects might be transformed into appropriate motor commands by a hierarchy of feedback loops or a cascade of inverse models

On the lowest level, we can think of fast (partial spinal) loops with which the length and the tension of muscles, joint angles, and postures might be controlled. At a higher level, destinations or trajectories in an effector-unspecific body space might be controlled. And finally, yet at another dimensional level, the attainment of environmental effects is controlled.

On each level it is assumed that the current deviation from the anticipated state determines the updating of the "set points" of directly subordinated loops.

Thus, at each level the respective desired (anticipated) and the currently given state (provided by sensory feedback) constitute the input, and the desired state of the subordinate level (the set point) constitutes the output. This "architecture" corresponds to the structure of an inverse model with the goal and the current state as input, and the action as output (e.g. [34,35,64]). Accordingly, instead of hierarchically organized feedback loops we can also speak of a cascade of inverse models. In such a structure, learning would refer to a continuous and simultaneous adjustment of the distributed inverse models: On each level the conversion of desired and perceived values into desired values for the next subordinated level would have to be adjusted, so that finally the emergence of an exteroceptive anticipation (an aspired goal state in the environment) automatically prompts the body to move in a way that brings the anticipated effects about.

The proposed structure of distributed feedback loops or adaptive inverse models, which are related among each other by sub- and super-ordinations, is still rather speculative and is subject to future exploration. However, recent simulations have shown that already a "cascade" of two control levels suffices for the modeling of goal oriented arm movements [5,19,18]. Figure 11 illustrates the basic structure of the SURE_REACH model, which consists of two modules. First, there is a posture memory, which accomplishes the transformation from an exteroceptive goal, represented as a desired hand location in an external space, into a set of all those arm postures that have been experienced as realizing the desired hand location. It thus transforms an exteroceptively defined goal into a set of proprioceptively defined postures. Second, there is a motor controller, which generates motor commands that move the arm toward the closest goal posture. Motor control is realized in two steps. First, the motor controller prepares a sensory-to-motor mapping, which provides suitable motor commands to achieve the desired hand location. It can be considered an online generated inverse model. Next, the sensory-to-motor mapping is used as a proprioceptive closed-loop feedback controller, which moves the hand to the target.

It is important to note that in SURE_REACH the mappings of desired hand locations into possible arm postures as well as the mappings of pairs of start- and goal postures into appropriate motor commands, which move the arm from the start to the goal, are learned from scratch by completely unsupervised learning mechanisms. In other words, SURE_REACH completely autonomously develops structures for the control of goal oriented arm movements by merely monitoring covariations between "visually" represented hand positions and proprioceptively represented arm postures on the one hand and proprioceptively represented changes of arm postures and motor commands on the other hand. Certainly, SURE_REACH is of minor complexity compared to the huge number of degrees of freedom natural behavior has to face. Nevertheless, the high flexibility and adaptability of the simulated behavior makes SURE_REACH a promising starting point for future elaborations.

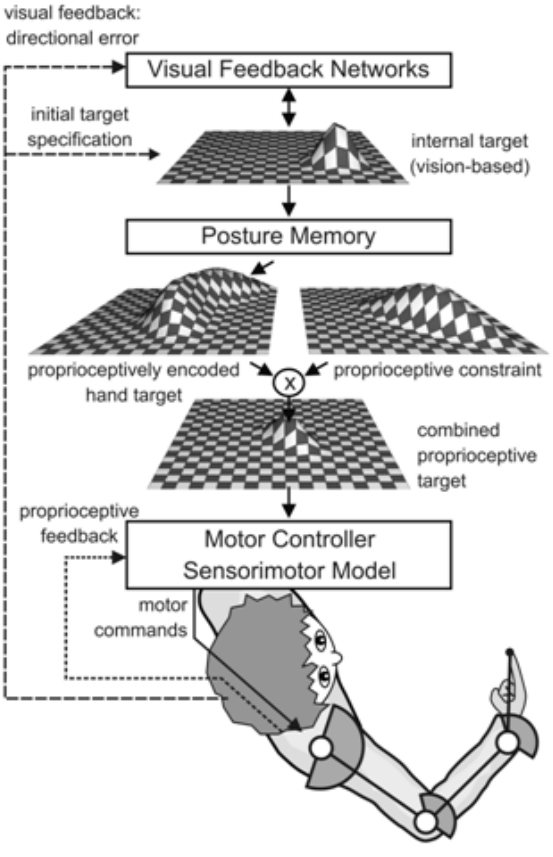


Fig. 11. The SURE_REACH model consists of a variety of target representations ("checkerboards") and neural controllers (boxes). In the cartoon, the target in the workspace of the hand is represented by visually defined population codes (top- most checkerboard). The Posture Memory converts the visually defined target into a set of appropriate arm postures which are represented by proprioceptively defined population codes (left checkerboard). The right checkerboard represents possible additional proprioceptive constraints of the target postures. The bottom-most checkerboard finally represents target postures that realize both, the desired hand location and the desired proprioceptive constraints. The Motor Controller then plans the transition from the initial posture to the nearest target posture based on a learned sensorimotor model. Proprioceptive feedback drives the execution of the movement plan. Visual Feedback Networks may adjust the visual goal representation if discrepancies between desired and actual hand locations cannot be corrected by the proprioceptive controller

7 Outlook

The replacement of the information processing approach by an "intentional approach", which acknowledges that cognition first and foremost serves the

control of goal oriented, voluntary behavior instead of serving the processing of stimulus information is still in its beginning. Substantial progress is already made in elucidating the anticipatory mechanisms by which simple voluntary acts are controlled [6]. These mechanisms already give a sense on how anticipations might shape perception and attention in accordance to behavioral requirements. However, to show, how from sensory-motor control higher cognitive abilities like planning, language, or reasoning emerge is still a long but promising way.

Acknowledgements

The reported research was supported by several grants from the German Research Community (DFG: HO 1301/4, 6, and 8) and the European Union (EU 823199-5, Mind Races Project). I am grateful to Martin Butz, Oliver Herbort, and two anonymous reviewers for valuable comments on an earlier version.

References

1. Ach, N.: Über den Willen. In: Ach, N. (ed.) Untersuchungen zur Psychologie und Philosophie. Quelle & Meyer, Leipzig (1913)
2. Bard, C., Turrell, Y., Fleury, M.: Deafferentation and pointing with visual double-step perturbations. Experimental Brain Research 125, 410–416 (1999)
3. Beckers, T., De Houwer, J., Eelen, P.: Automatic integration of non-perceptual action effect features: The case of the associative affective Simon effect. Psychological Research 66, 166–173 (2002)
4. Bernstein, N.A.: The co-ordination and regulation of movements. Pergamon Press, Oxford (1967)
5. Butz, M.V., Herbort, O., Hoffmann, J.: Exploiting redundancy for flexible behavior: Unsupervised learning in a modular sensorimotor control architecture. Psychological Review 114, 1015–1046 (2007)
6. Butz, M.V., Sigaud, O., Pezzulo, G., Baldassarre, G. (eds.): ABiALS 2006. LNCS, vol. 4520. Springer, Heidelberg (2007)
7. Cole, J., Paillard, J.: Living without touch and peripheral information about body position and movement: Studies with deafferented subjects. In: Bermudez, J.L., Marcel, A.J. (eds.) The body and the self, pp. 245–266. MIT Press, Cambridge (1995)
8. Cole, J.: Pride and a Daily Marathon. MIT Press, Cambridge (1995)
9. Colwill, R.M., Rescorla, R.A.: Instrumental responding remains sensitive to reinforcer devaluation after extensive training. Journal of Experimental Psychology: Animal Behavior Processes 11, 520–536 (1985)
10. Dickinson, A.: Instrumental conditioning. In: Mackintosh, N. (ed.) Animal learning and cognition, pp. 45–79. Academic Press, San Diego (1994)
11. Elsner, B., Hommel, B.: Effect anticipation and action control. Journal of Experimental Psychology: Human Perception and Performance 27, 229–240 (2001)
12. Elsner, B., Hommel, B.: Contiguity and contingency in action-effect learning. Psychological Research 68, 138–154 (2004)
13. Gibson, J.J.: The Ecological Approach to Visual Perception. Lawrence Erlbaum Associates, Mahwah (1979)

14. Greenwald, A.: Sensory feedback mechanisms in performance control: with special reference to the ideo-motor mechanism. Psychological Review 77, 73–99 (1970)
15. Harleß, E.: Der Apparat des Willens. Zeitschrift für Philosophie und philosophische Kritik 38, 50–73 (1861)
16. Heckhausen, H., Beckmann, J.: Intentional action and action slips. Psychological Review 97, 36–48 (1990)
17. Herbart, J.F.: Psychologie als Wissenschaft neu gegründet auf Erfahrung, Metaphysik und Mathematik. In: Zweiter, analytischer Teil, August Wilhem Unzer, Königsberg, Germany (1825)
18. Herbort, O., Butz, M.V., Hoffmann, J.: Multimodal goal representations and feedback in hierarchical motor control. In: International Conference on Cognitive Systems CogSys 2008 (2008)
19. Herbort, O., Butz, M.V., Hoffmann, J.: Towards an adaptive hierarchical anticipatory behavioral control system. In: Castelfranchi, C., Balkenius, C., Butz, M.V., Ortony, A. (eds.) From Reactive to Anticipatory Cognitive Embodied Systems: Papers from the AAAI Fall Symposium, pp. 83–90. AAAI Press, Menlo Park (2005)
20. Herwig, A., Prinz, W., Waszak, F.: Two modes of sensorimotor integration in intention-based and stimulus-based actions. Quarterly Journal of Experimental Psychlogy 60, 1540–1554 (2007)
21. Hick, W.E.: On the rate of gain of information. Quarterly Journal of Experimental Psychology 4, 11–26 (1952)
22. Hoffmann, J., Berner, M., Butz, M.V., Herbort, O., Kiesel, A., Kunde, W., Lenhard, A.: Explorations of anticipatory behavioral control (ABC): A report from the cognitive psychology unit of the University of Würzburg. Cognitive Processing 8, 133–142 (2007)
23. Hoffmann, J.: Vorhersage und Erkenntnis: Die Funktion von Antizipationen in der menschlichen Verhaltenssteuerung und Wahrnehmung. Anticipation and cognition: The function of anticipations in human behavioral control and perception. Hogrefe, Göttingen, Germany (1993)
24. Hoffmann, J.: Anticipatory behavioral control. In: Butz, M.V., Sigaud, O., Gérard, P. (eds.) Anticipatory Behavior in Adaptive Learning Systems. LNCS, vol. 2684, pp. 44–65. Springer, Heidelberg (2003)
25. Hoffmann, J., Sebald, A.: Lernmechanismen zum Erwerb verhaltenssteuernden Wissens [Learning mechanisms for the acquisition of knowledge for behavioral control]. Psychologische Rundschau 51, 1–9 (2000)
26. Hoffmann, J., Sebald, A., Stöcker, C.: Irrelevant response effects improve serial learning in serial reaction time tasks. Journal of Experimental Psychology: Learning, Memory, and Cognition 27, 470–482 (2001)
27. Hoffmann, J., Stöcker, C., Kunde, W.: Anticipatory control of actions. International Journal of Sport and Exercise Psychology 2, 346–361 (2004)
28. Hommel, B.: The cognitive representation of action: Automatic integration of perceived action effects. Psychological Research 59, 176–186 (1996)
29. Hommel, B.: Perceiving one's own action - and what it leads to. In: Jordan, J.S. (ed.) Systems theory and apriori aspects of perception, pp. 143–179. North Holland, Amsterdam (1998)
30. Hommel, B., Alonso, D., Fuentes, L.J.: Acquisition and generalization of action effects. Visual cognition 10, 965–986 (2003)
31. Hommel, B., Müsseler, J., Aschersleben, G., Prinz, W.: The theory of event coding (TEC): A framework for perception and action planning. Behavioral and Brain Sciences 24, 849–878 (2001)

32. James, W.: The principles of psychology. Dover Publications, New York (1890)
33. Jeannerod, M.: The neural and behavioral organization of goal-directed movements. Clarendon Press (1988)
34. Kalveram, K.T.: The inverse problem in cognitve, perceptual and proprioceptive control of sensorimotor behaviour: Towards a biologically plausible model of the control of aiming movements. International Journal of Sport and Exercise 2, 255–273 (2004)
35. Kawato, M.: Internal models for motor control and trajectory planning. Current Opinion in Neurobiology 9, 718–727 (1999)
36. Kiesel, A., Hoffmann, J.: Variable action effects: Response control by context-specific effect anticipations. Psychological Research 68, 155–162 (2004)
37. Koch, I., Kunde, W.: Verbal response-effect compatibility. Memory and Cognition 30, 1297–1303 (2002)
38. Kunde, W., Hoffmann, J., Zellmann, P.: The impact of anticipated action effects on action planning. Acta Psychologica 109, 137–155 (2002)
39. Kunde, W.: Response-effect compatibility in manual choice reaction tasks. Journal of Experimental Psychology: Human Perception and Performance 27, 387–394 (2001)
40. Kunde, W.: Temporal response-effect compatibility. Psychological Research 67, 153–159 (2003)
41. Kunde, W., Koch, I., Hoffmann, J.: Anticipated action effects affect the selection, initiation, and execution of actions. The Quarterly Journal of Experimental Psychology. Section A: Human Experimental Psychology 57, 87–106 (2004)
42. Lenhard, A., Hoffmann, J., Sebald, A.: Intra- and intermanual transfer of adaptation to unnoticed virtual displacement under terminal and continuous visual feedback. Diploma Thesis, University of Würzburg (2002)
43. Lotze, H.R.: Medicinische Psychologie oder Physiologie der Seele. Weidmannsche Buchhandlung, Leipzig (1852)
44. Mandler, G.: Cognitive Psychology. An essay in cognitive science. Erlbaum, Hillsdale (1985)
45. Neisser, U.: Cognitive psychology. Appleton, New York (1967)
46. Newell, A., Simon, H.A.: GPS, a program that simulates human thought. In: Billing, H. (ed.) Lernende Automaten, pp. 109–124. Oldenbourg, München (1961)
47. Powers, W.T.: Behavior: The Control of Perception. Aldine de Gruyter, New York (1973)
48. Proteau, L., Marteniuk, R.G., Girouard, Y., Dugas, C.: On the type of information used to control and learn an aiming movement after moderate and extensive training. Human Movement Science 6, 181–199 (1987)
49. Rescorla, R.A.: Associative relations in instrumental learning: The eighteenth Bartlett memorial lecture. Quarterly Journal of Experimental Psychology 43, 1–23 (1991)
50. Restat, J.: Kognitive Kinästhetik: Die modale Grundlage des amodalen räumlichen Wissens. Pabst, Lengerich (1999)
51. Rosenbaum, D.A., Meulenbroek, R.G.J., Vaughan, J., Jansen, C.: Posture-based motion planning: Applications to grasping. Psychological Review 108, 709–734 (2001)
52. Sainburg, R.L., Poizner, H., Ghez, C.: Loss of proprioception produces deficits in interjoint coordination. Journal of Neurophysiology 70, 2136–2147 (1993)
53. Sainburg, R.L., Wang, J.: Interlimb transfer of visuomotor rotations: independence of direction and final position information. Experimental Brain Research 145, 437–447 (2002)

54. Shannon, C.E., Weaver, W.: The mathematical theory of communication. The University of Illinois Press, Urbana (1949)
55. Simon, J.R., Rudel, A.P.: Auditory S-R compatibility: The effect of an irrelevant cue on information processing. Journal of Applied Psychology 51, 300–304 (1967)
56. Stock, A., Stock, C.: A short history of ideo-motor action. Psychological Research 68, 176–188 (2004)
57. Stock, A., Hoffmann, J.: Intentional fixation of behavioral learning or how R-E learning blocks S-R learning. European Journal of Cognitive Psychology 14, 127–153 (2002)
58. Thorndike, E.L.: Animal intelligence. an experimental study of the associative processes in animal. Psychological Monographs 2(4, whole No. 8) (1898)
59. Turing, A.M.: Computing machinery and intelligence. Mind 59, 433–460 (1950)
60. von Holst, E., Mittelstaedt, H.: Das Reafferenzprinzip. Naturwissenschaften 37, 464–476 (1950)
61. von Neumann, J.: Die Rechenmaschine und das Gehirn. Oldenbourg, München (1958)
62. Watson, J.B.: Psychology as a behaviorist views it. Psychological Review 20, 158–177 (1913)
63. Wiener, N.: Cybernetics or control and communication in the animal and the machine. Wiley, New York (1948)
64. Wolpert, D.M., Ghahramani, Z.: Computational principles of movement neuroscience. Nature Neuroscience 3, 1212–1217 (2000)
65. Ziessler, M., Nattkemper, D.: Effect anticipation in action planning. In: Hommel, W.P. (ed.) Attention and Performance XIX: Common mechanisms in perception and action, pp. 645–673. Oxford University Press, Oxford (2002)
66. Ziessler, M., Nattkemper, D., Frensch, P.A.: The role of anticipation and intention for the learning of effects of self-performed actions. Psychological Research 68, 163–175 (2004)

Anticipative Control of Voluntary Action: Towards a Computational Model

Pascal Haazebroek[1] and Bernhard Hommel[2,*]

[1] Leiden University
Cognitive Psychology Unit & Leiden Institute for Brain and Cognition
Leiden, The Netherlands
[2] Leiden University
Department of Psychology
Cognitive Psychology Unit
Wassenaarseweg 52
2333 AK Leiden, The Netherlands
hommel@fsw.leidenuniv.nl

Abstract. Human action is goal-directed and must thus be guided by anticipations of wanted action effects. How anticipatory action control is possible and how it can emerge from experience is the topic of the ideomotor approach to human action. The approach holds that movements are automatically integrated with representations of their sensory effects, so that reactivating the representation of a wanted effect by "thinking of it" leads to a reactivation of the associated movement. We present a broader theoretical framework of human perception and action control—the Theory of Event Coding (TEC)—that is based on the ideomotor principle, and discuss our recent attempts to implement TEC by means of a computational model (HiTEC) to provide an effective control architecture for artificial systems and cognitive robots.

1 Introduction

Human behavior is commonly proactive rather than reactive. That is, people do not await particular stimulus events to trigger certain responses but, rather, carry out planned actions to reach particular goals. Planning an action ahead and carrying it out in a goal-directed fashion requires prediction and anticipation: in order to select an action that is suited to reach a particular goal presupposes knowledge about relationships between actions and effects, that is, about which goals can be realized by what action. Under some circumstances this knowledge might be generated ad hoc. For instance, should your behavior ever make a flight attendant to drop you by parachute in a desert, your previously acquired knowledge may be insufficient to select among reasonable action alternatives, so you need to make ad hoc predictions to find out where to turn to. But fortunately, most of the situations we encounter are much more familiar and, thus, much easier to deal with. We often have a rough idea about what actions may be suitable under a given goal and in a particular context, simply because

* Corresponding author.

G. Pezzulo et al. (Eds.): ABiALS 2008, LNAI 5499, pp. 31–47, 2009.

we have experience: we have had and reached the same or similar goals and acted in the same or similar situations before.

How experience with one's own actions generates knowledge that guides the efficient selection of actions, and how humans carry out voluntary actions in general, was the central issue in ideomotor approaches to human action control. Authors like Lotze (1852), Harless (1861), and James (1890) were interested in the general question of how the mere thought of a particular action goal can eventually lead to the execution of movements that reach that goal in the absence of any conscious access to the responsible motor processes (*executive ignorance*). Key to the theoretical conclusion they came up with was the insight that actions are means to generate perceptions (of wanted outcomes) and that these perceptions can be anticipated. If there would be an associative mechanism that integrates motor processes (m) with representations of the sensory effects they produce (e), and if the emerging association between movements and effect representations would be bidirectional (m←→e), reactivating the representation of the effect by voluntarily "thinking of it" may suffice to reactivate the associated motor processes (e→m). In other words, integrating movements and their sensory consequences provides a knowledge base that allows for selecting actions according to their anticipated outcomes—for anticipative action control that is.

After a flowering period in the second half of the 19th century ideomotor approaches were effectively eliminated from the scientific stage (Prinz, 1987; Stock & Stock, 2004). A major reason for that was the interest of ideomotor theoreticians in conscious experience and the relationship between conscious goal representations and unconscious motor behavior, a topic that did not meet scientific criteria in the eyes of the behaviorist movement gaining power in the beginning of the 20th century (cf., Thorndike, 1913). Starting with an early resurrectional attempt by Greenwald (1970), ideomotor ideas have recently regained scientific credibility and explanatory power however. In their Theory of Event Coding (TEC), Hommel, Müsseler, Aschersleben, and Prinz (2001) have even suggested that the ideomotor principle may represent a firm base on which a comprehensive theory of human perception and anticipatory action control can be built. In the following, we will elaborate on what such a theory may look like. In particular, we will briefly discuss the basic principles and basic assumptions of TEC and then go on to describe our recent attempts to implement these principles and assumptions by means of a computational model of human perception and action control—a model we coined HiTEC (Haazebroek & Hommel, submitted).

2 TEC

The core idea underlying TEC (Hommel et al., 2001) is that perception and action are in some sense the same thing and must therefore be cognitively represented in the same way—the notion of *common coding* (Prinz, 1990). According to the ideomotor principle, action consists in intentionally producing wanted effects, that is, in the execution of motor processes for the sake of creating particular sensory events. In contrast to action, perception is commonly conceived of as the passive registration of sensory input. However, Hommel et al. (2001) argue that this conception is incorrect and misleading, as sensory input is commonly actively produced (Dewey, 1896; Gibson, 1979). For

instance, even though visual perception needs light hitting the retina, we actively move our eyes, head, and body to make sure that our retina is hit by the light that is reflecting the most interesting and informative events. That is, we actively search for the information we are interested in and move our receptive surfaces to optimize the intake of that information. This is even more obvious for the tactile sense, as almost nothing would be perceived by touch without systematically moving the sensor surface across the objects of interest. Hence, we perceive by executing motor processes for the sake of creating particular sensory events. Obviously, this is exactly the way we just defined action, which implies that action and perception are one process.

The second central assumption of TEC is that cognitive representations are *composites* of feature codes (Hommel, 2004). Our brain does not represent events through individual codes or neurons but by widely distributed feature networks. For instance, the visual cortex consists of numerous representational maps coding for various visual features, such as color, orientation, shape, or motion (DeYoe & Van Essen, 1988) and similar feature maps have been reported for other modalities. Likewise, action plans are composites of neural networks coding for various action features, such as the direction, force, or distance of manual actions (Hommel & Elsner, 2009). One implication of the assumption that cognitive event representations are composites is that binding operations are necessary to integrate the codes referring to the same event, and another is that different events can be related to, compared with, or confused with each other based on the features they do or do not share. For instance, TEC implies that stimuli and responses can be similar to each other, in the sense that the binding representing the stimulus and the binding representing the response can include the same features, such as location or speed, and can thus prime each other (which for instance explains effects of stimulus-response compatibility) or interact in other ways.

The third main assumption of TEC is that the cognitive representations that underlie perception and action planning code for *distal* but not *proximal* aspects of the represented events (Prinz, 1992). In a nutshell, this means that perceived and produced events are coded in terms of the features of the external event *as external event* (i.e., as objectively or inter-subjectively definable) but not with respect to the specifics of the internal processing, such as retinal or cortical coding characteristics, or particular muscle parameters. This terminology goes back to Heider (1926, 1930), who discussed the problem that our conscious experience refers to objective features of visual objects (the distal attributes), even though the intermediate processing steps of the physical image on the retina and the physiological response to it (the proximal attributes) are not fully determined by the distal attributes. Brunswik (1944) extended this logic to action and pointed out that goal representations refer to distal aspects of the goal event and, thus, do not fully determine the proximal means to achieve it.

To summarize, TEC assumes that perceived events are represented by activating and integrating feature codes—codes that represent the distal features of the event. Given that perceptions are actively produced, these bindings are likely to also include action features, that is, codes that represent the features of the action used to produce that perception. In turn, action plans are integrated bindings of codes representing the distal features of the action. As actions are carried out to create sensory events, action plans also comprise of feature codes referring to these events. In other words, both perceived and produced events are represented by sensorimotor bindings or "event files" (Hommel, 2004). However, not all features of a perceived or a produced event

are relevant in a particular context. To account for that, TEC assumes that feature codes are "intentionally weighted" according to the goal or task at hand. For instance, if you are searching for a particular color, or if what matters for your actions is the location of your fingertip, color and location codes would be weighted higher, respectively, and thus affect perception and action planning more strongly. TEC was very helpful in interpreting and integrating available findings in a coherent manner, as well as in stimulating numerous experiments and studies on various topics and perception-action phenomena. However, as Hommel et al. (2001) pointed out, TEC only provides a general framework and the theoretical concepts needed to get a better understanding of higher level perception, action, and their relationship. Deeper insight and theoretical advancement calls for more detail and additional assumptions. To meet this challenge we began developing HiTEC, a computational implementation of TEC's basic principles and assumptions. In the following, we provide a brief overview of the main strategies guiding our implementation, but refer to Haazebroek and Hommel (submitted) for a broader treatment.

3 HITEC

HiTEC (Haazebroek & Hommel, submitted) is an attempt to translate the theoretical framework of TEC (Hommel et al, 2001) into a runnable computational model. Our ambition is to develop a broad, cognitive architecture that can account for a variety of empirical effects related to stimulus-response translation and that can serve as a starting point for a novel control architecture for cognitive robots in the PACO-PLUS project (www.paco-plus.org).

From a modeling perspective TEC provides a number of constraints; some of them enforce structural elements while others impose the existence of certain processes. First, we describe the general structure of HiTEC. Next, we elaborate on the processes operating on this structure, following the two-stage model (Elsner and Hommel, 2001) for the acquisition of voluntary action control. Finally, we discuss how the mechanisms of HiTEC might operate in a real life scenario and show that anticipation plays a crucial role in quickly generating and controlling appropriate responses.

4 HITEC's Structure and Representations

HiTEC is architected as a connectionist network model that uses the basic building blocks of parallel distributed processing (PDP; e.g., McClelland, 1992; Rumelhart, Hinton, & McClelland, 1986). In a PDP network model processing occurs through the interactions of a large number of interconnected elements called units or nodes. Nodes may be organized into higher structures, called modules, each containing a number of nodes. Modules may be part of a larger processing pathway. Pathways may interact in the sense that they can share common modules.

Each node has an activation value indicating local activity. Processing occurs by propagating activity through the network; that is, by propagating activation from one node to the other, via weighted connections. When a connection between two nodes is positively weighted, the connection is excitatory and the nodes will increase each other's activation. When the connection is negatively weighted, it is inhibitory and the

nodes will reduce each other's activation. Processing starts when one or more nodes receive some sort of external input. Gradually, node activations will rise and propagate through the network while interactions between nodes control the flow of processing. Some nodes are designated output nodes. When activations of these nodes reach a certain threshold (or when the time allowed for processing has passed), the network is said to produce the corresponding output(s).

In HiTEC, the elementary units are codes. As illustrated in Figure 1, codes are organized into three main systems: the sensory system, the motor system and the common coding system. Each system will now be discussed in more detail.

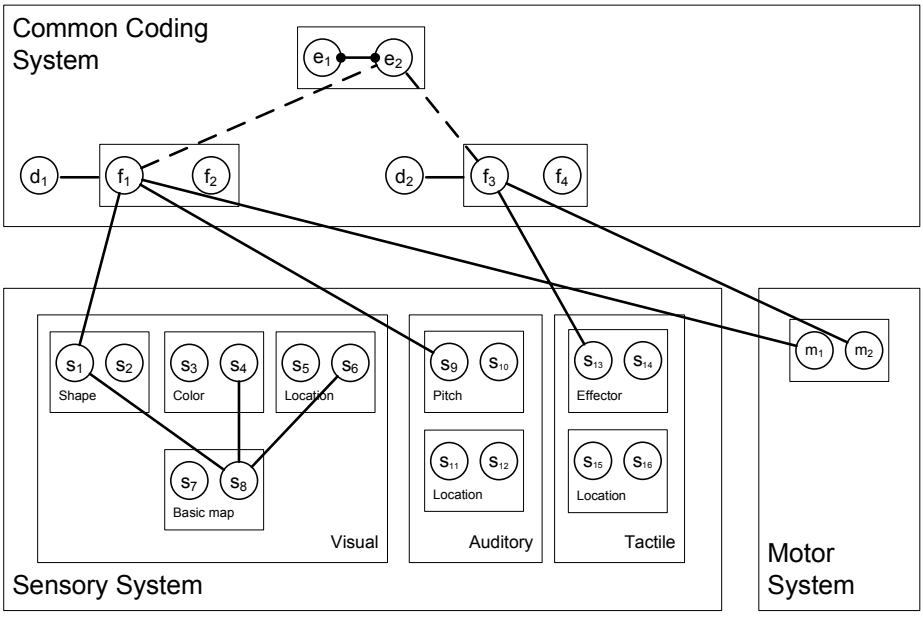


Fig. 1. General architecture of HiTEC

4.1 Sensory System

As already mentioned, the primate brain encodes perceived objects in a distributed fashion: different features are processed and represented across different cortical maps (e.g., Cowey, 1985; DeYoe & Van Essen, 1988). In HiTEC, different modalities (e.g., visual, auditory) and different dimensions within each modality (e.g., visual color and shape, auditory location and pitch) are processed and represented in different sensory maps. Each sensory map is a module containing a number of sensory codes that are responsive to specific sensory features (e.g., a specific color or a specific pitch). Note that Figure 1, shows only two sensory codes per map for clarity.

In the visual brain, there are two major parallel pathways (Milner & Goodale, 1995) that follow a common preliminary basic feature analysis step. The ventral pathway is seen as crucial for object recognition and consists of a hierarchy of sensory maps coding for increasingly complex features (from short line segments in the

lower maps to complex shapes in higher maps) and increasingly large receptive field (from a small part of the retina in the lower maps to anywhere on the retina in higher maps). The second pathway, the dorsal pathway, is seen as crucial for action guidance as it loses color and shape information but retains information about contrast, location of objects, and other action-related features.

In HiTEC, a common visual sensory map codes for basic visual parts of perceptual events. This common basic map projects to both the ventral and the dorsal pathways. The ventral pathway consists of sensory maps coding for combinations (such as more specific shapes) or abstractions (e.g., object color). The dorsal pathway is currently simply a sensory map coding for visual location—to be extended for processing other action-related features in a later version of HiTEC.

Distributed processing allows a system to dramatically increase its representational capacity as it no longer requires each combination of features to have its own dedicated representational structure but can rather encode a specific combination on demand in terms of activating a collection of constituting feature structures. On the downside, in typical scenarios, this inevitably results in binding problems (Treisman, 1996). For instance, when multiple objects are perceived and they are both represented in terms of activating the structures coding for their constituting features, how to tell which feature belongs to which object? This clearly calls for an integration mechanism that can tell them apart.

Recent studies in the visual modality have shown that this problem can, partly, be solved by employing local interactions between feed-forward and feed-back processes in the ventral and dorsal pathways (Van der Velde & De Kamps, 2001). It is true that higher ventral sensory maps do not contain information on location and that higher dorsal sensory maps do not contain information on object shape or color, but these pathways can interact using the common basic visual feature map as a visual blackboard (Van der Velde, De Kamps, & Van der Voort van der Kleij, 2004). For instance: when a specific color is activated in a higher sensory map, it can feed back activation to lower sensory maps, thereby modulating the activity of these sensory codes in a way that those codes that code for simple parts of this color are enhanced. This can modulate the processing in the dorsal pathway as well resulting in enhanced activation of those codes in the location map that code for the location(s) of objects of the specified color.

This principle also works the other way round: activating a specific location code in the location map can modulate the sensory codes in the lower sensory maps that code for simple parts at this location. This can modulate the processing in the ventral pathway, resulting in enhanced activation of the more complex or abstract features of the object at the specified location. In HiTEC, this is the way the visual sensory system can be made to enhance the processing of objects with specific features or on a specific location. For now, we assume the following sensory maps in the HiTEC architecture: visual basic features map, visual color map, visual shape map, visual location map, auditory pitch map, auditory location map, tactile effector (i.e., hands or feet) map and tactile location map.

4.2 Motor System

The motor system contains motor codes, referring to proximal aspects of movements. Motor codes can also be organized in maps, following empirical evidence that

suggests distributed representations at different cortical locations in the motor domain (e.g., Andersen, 1988; Colby 1998). For example, cortical maps can be related to effector (e.g., eye, hand, arm, foot) or movement type (e.g., grasping, pointing). It makes sense to assume that there is some sort of hierarchical structure as well in motor coding. However, in the present version of HiTEC, we consider only one basic motor map with a set of motor codes. As our modeling efforts in HiTEC evolve, its motor system may be extended further.

It is clear that motor codes, even when structured in multiple maps, can only specify a rough outline of the motor action to be performed as some parameters depend strongly on the environment. For instance, when grasping an object, the actual object location is not represented by a motor code (this would lead to an explosion of the number of necessary motor codes, even for a very limited set of actions). So it makes sense to interpret a motor program as a blueprint of a motor action that needs to be filled in with this specific, on line, information, much like the schemas put forward by Schmidt (1975) and Glover (2004). In our discussion of HiTEC processes we will discuss this issue in more detail.

4.3 Common Coding System

According to TEC both perceived events and action generated events are coded in one common representational domain (Hommel et al, 2001). In HiTEC, this domain is the common coding system that contains common feature codes. Feature codes refer to distal features of objects, people and events in the environment. Example features are distance, size and location, but on a distal, descriptive level, as opposed to the proximal features as coded by the sensory codes and motor codes.

Feature codes may be associated to both sensory codes and motor codes and are therefore truly sensorimotor. They can combine information from different modalities and are in principle unlimited in number. Feature codes are not given but they evolve and change. In HiTEC simulations, however, we usually assume a set of feature codes to be present initially, to bootstrap the process of extracting sensorimotor regularities in interactions with the environment.

Feature codes are contained in feature dimensions. As feature dimensions may be enhanced as a whole, for each dimension an additional dimension code is added that is associated with each feature code within this dimension. Activating this code will spread activation towards all feature codes within this dimension, making them more sensitive to stimulation originating from sensory codes.

4.4 Associations

In HiTEC, codes can become associated, both for short term and for long term. Short term associations between feature codes reflect that these codes 'belong together in the current task or context' and their binding is actively maintained in working memory. In Figure 1, these temporary bindings are depicted as dashed lines. Long term associations can be interpreted as learned connections reflecting prior experience. For now, we do not differentiate between episodic and semantic memory—even though later versions are planned to distinguish between a "literal" episodic memory that stores event files (see below) and a semantic memory that stores rules abstracted from

episodic memory (O'Reilly & Norman, 2002). At present, both types of experience are modeled as long term associations between (any kind of) codes and are depicted as solid lines in Figure 1.

4.5 Event File

Another central concept in the theory of event coding is the event file (Hommel, 2004). In HiTEC, the event file is modeled as a structure that temporarily associates to feature codes that 'belong together in the current context' in working memory. The event file serves both the perception of a stimulus as well as the planning of an action. Event files can compete with other event files.

5 HITEC's Processes

How do associations between codes come to be? What mechanisms result of their interactions? And how do these mechanisms give rise to anticipation based, voluntary action control? Elsner and Hommel (2001) proposed a two-stage model for the acquisition of voluntary action control. At the first stage, the cognitive system observes and learns regularities in motor actions and their effects. At the second stage, the system uses the acquired knowledge of these regularities to select and control its actions. For both stages, we now discuss in detail how processes take place in the HiTEC architecture. Next, we discuss some additional process related aspects of the architecture.

Stage 1: Acquiring Action-Effect Associations

The framework of event coding assumes that feature codes are grounded representations as they are derived by abstracting regularities in activations of sensory codes. However, the associations between feature codes and motor codes actually signify a slightly different relation: feature codes encode the (distal) perceptual effect of the action that is executed by activating the motor codes. Following the ideomotor principle, the cognitive system has no innate knowledge of the actual motor action following the activation of a certain motor code. Rather, motor codes need to become associated with their perceptual action effects so that by anticipating these effects, activation can propagate via these associations to those motor codes that actually execute the corresponding movement.

Infants typically start off with a behavioral repertoire based on stimulus-response (SR) reflexes (Piaget, 1952). As the infant exhibits these stimulus-response reflexes, as well as random behaviors (e.g., motor babbling), its cognitive system learns the accompanying response-perceptual effect (RE) regularities that will serve as some sort of database of 'what action achieves what environmental effect'. Following Hommel (1996), we assume that any perceivable action effect is automatically coded and integrated into an action concept, which is, in the HiTEC architecture, an event file consisting of feature codes. Although all effects of an action become integrated automatically, intentional processes do affect the relative weighting of integrated action effects—TEC's intentional-weighting principle.

Taken together, action – effect acquisition is modeled in HiTEC as follows: motor codes m_i are activated, either because of some already existing associations or simply

because of network noise. This leads to a change in the environment (e.g., the left hand suddenly touches a cup) which is picked up by sensory codes s_i. Activation propagates from sensory codes towards feature codes f_i. And eventually, these feature codes are integrated into an event file e_i which acts as an action concept. Subsequently, the cognitive system learns associations between the feature codes f_i belonging to this action concept and the motor code m_i that just led to the executed motor action. Crucially, task context can influence the learning of action effects. Not by selecting which effects are associated but by weighting the different effect features. Nonetheless, this is an interactive process that does not exclude unintended but utterly salient action effects to become involved in strong associations as well.

Stage 2: Using Action Effect Associations

Once associations between motor codes and feature codes exist, they can be used to select and plan voluntary actions. Thus, by anticipating desired action effects, feature codes become active. Now, by integrating the feature codes into an action concept, the system can treat the features as constituting a desired state and propagate their activation towards associated motor codes. Crucially, anticipating certain features needs integration to tell them apart from the features that code for the currently observed environment. Once integrated, the system has 'a lock' on these features and can use these features to select the right motor action.

Initially, multiple motor codes m_i may become active as they typically fan out associations to multiple feature codes f_i. However, some motor codes will have more associated features that are also part of the active action concept and some of the m_i - f_i associations may be stronger than others. Taken together, the network will – in PDP fashion – converge towards one strongly activated motor code m_i which will lead to the selection of that motor action.

In addition to the mere selection of a motor action, feature codes also form the actual action plan that specifies (in distal terms) how the action should be executed: namely, in such a way the intended action effects are realized. By using anticipated action effects to choose an action, the action actually is selected because the cognitive system intended this, not because of a reflex to some external stimulus. Thus, in HiTEC, using anticipation is the key to voluntary action.

5.1 Task Context

Task context can modulate both action-effect learning and the usage of these links. This can help focus processing to action alternatives that 'make sense' in the current context. In real life this is necessary as the action alternatives are often rather unconstrained. Task context comes in different forms. One is the overall environment, the scene context in which the interaction takes place. The cognitive system may just have seen other objects in the room, or the room itself, and feature codes that code for aspects of this context may still have some activation. This can, in principle, influence action selection. As episodic and semantic memory links exist as well, this influence may also be less salient: the presence of a certain object might recall memories of previous encounters or similar contexts that influence action selection in the current task.

A task can also be very specific, as given by a tutor or instructor in terms of a verbal description. In HiTEC, it is assumed that feature codes can be activated by means of verbal labels. Thus, when a verbal task is given, this could directly activate feature codes. The cognitive system integrates these codes into an event file that is actively maintained in working memory. For example, when approached with several options to respond differently to, different event files e_i are created for the different options. Due to the mutual inhibitory links between event files, they will compete with each other. Because of the efficiency the cognitive system can now display, one could state that a cognitive reflex has been prepared (Hommel, 2000) that anticipates certain stimuli features. The moment these features are actually perceived, the reflex 'fires' and - by propagating activation to event codes and subsequently to other feature codes - quickly anticipates the correct action effects, which results in the selection and execution of the correct motor action.

5.2 Online vs. Offline Processing

In HiTEC, action selection and action planning are interwoven, but on a distal feature level. This leaves out the necessity of coding every minute detail of the action, but restricts action planning to a ballpark idea of the movement. Still, a lot has to be filled in by on line information. Currently, this falls outside the scope of HiTEC, but one could imagine that by activating distal features, the proximal sensory codes can be top down moderated to 'focus their attention' towards specific aspects of the environment (e.g., visual object location), see Hommel (in press). In addition, actions need still not to be completely specified in advance, as they are monitored and adjusted while they are performed—which in humans seems to be the major purpose of dorsal pathways (Milner & Goodale, 1995)

5.3 Action Monitoring

The anticipated action effects are a trigger for action selection, but also form an expectation of the perceptual outcome of the action. Differences between this expectation and reality lead to adjusting the action on a lower sensorimotor level than is currently modeled in HiTEC. What matters now, is that the feature codes are interacting with the sensory codes, making sure that the generated perception is within the set parameters, as determined by the expected action outcome. If this is not (well enough) the case, the action should be adjusted.

However, when a discrepancy of this expectation drastically exceeds 'adjustment thresholds', it may actually trigger action effect learning (stage 1). Apparently, the action-effect associations were unable to deliver an apt expectation of the actual outcome. Thus, anticipating the desired outcome falsely led to the execution of this action. This may trigger the system to modify these associations, so that the motor codes become associated with the correct action effect features.

Crucially, having anticipations serve as expectations, the system is not forced into two distinct operating modes (learning vs. testing). With anticipation as retrieval cue for action selection and as expectation of the action outcome, the system has the means to self-regulate its learning by making use of the discrepancy between actual effects and these anticipations.

6 Model Implementation

The HiTEC model is implemented using neural network simulation software that facilitates the specification and simulation of interactive networks. In interactive networks, connections are bidirectional and the processing of any single input occurs dynamically during a number of cycles. Each cycle, the network is gradually updated by changing the activation of each node as a result of its interactions with other nodes.

6.1 Code Dynamics

HiTEC aims at a biologically realistic implementation of network dynamics. In the human brain, local interactions between neurons are largely random, but when looking at groups of neurons (i.e., neuron populations) their average activation can be described using mean field approximation equations (Wilson and Cowan, 1972). In HiTEC, a single code is considered to be represented by a neuron population. Its dynamics can therefore be described using differential equations such as:

$$\frac{dA}{dt} = -A + \sum_k w_k F(A_k) + N$$

This equation states that the change in activation A of a code is a result of a decay term and the weighted sum of the outputs of those nodes k that it connects to. Also, each node receives additional random noise input N. Node output is computed using an activation function $F(A)$ that translates node activation into its output as governed by the following logistic function:

$$F(A) = \frac{1}{1 + \exp(-A)}$$

The simulator uses numerical integration to determine the change of activation for each node in each cycle.

6.2 Codes

Currently, in our simulations we hard code all sensory codes including their receptive field specification (e.g., whether a code is responsive to a red or a blue color). Also, feature codes are assumed to exist, as well as all connections between sensory codes and feature codes reflecting prior experience with sensory regularities. In the future it may become an interesting endeavor to learn the grounding of feature codes in terms of proximal sensory codes, possibly by means of self organizing map methods that can be moderated by HiTEC processes (e.g., failing to predict an action outcome may signal relevant novelty and moderate the creation or update of a feature code). Also, for now, we assume a limited set of motor programs that are simply represented by fixed motor codes. Thus, in simulations we currently focus on the interactions between perception and action and how task context influence these interactions, rather than on the grounding of codes per se.

6.3 Action-Effect Learning

Learning action effects is reflected by creating long term connections between feature codes and motor codes. This is currently done by simple associative, Hebbian learning, as described by the following equation:

$$\frac{dw_{ij}}{dt} = \gamma A_j (A_i - w_{ij})$$

Thus, the change of the connection strength is determined by the activation of the nodes *i* and *j* that are connected. This way, feature codes that were activated more strongly will become more strongly connected to the motor code that caused the perceptual effect. Surely, this type of learning is known to be limited but serves our current purposes.

6.4 Short Term Associations and Event File Competition

Crucial in HiTEC is the short term memory component. A task instruction is represented using short term connections between feature codes and event files. In the current set up, an event file is simply a node that is created on demand, as a result of the task instruction, and temporarily connects to those feature codes that were activated by the task instruction (i.e., via verbal labels). An event file has an enhanced baseline activation, reflecting its task relevance. Moreover, event files compete with each other by means of lateral inhibition (i.e., they are interconnected with negative connections) resulting in a winner-take-all mechanism: as activation gradually propagates from feature codes to event files (and back), their activation changes as well. Due to the lateral inhibition, only one event file will stand as the 'winner', while weakening the other event files. This results in selective activation at the feature code level and subsequently in action selection at the motor code level.

6.5 Related Work

We must note that we do not advertise the associative learning method used in HiTEC as a competitive alternative to highly specialized machine learning techniques that are traditionally used in classification tasks (e.g., Hiddden Markov Models, Support Vector Machines et cetera) or reward based learning tasks (e.g., Reinforcement learning, Q-learning et cetera). However, we do focus on the context of learning: the interplay between (the coding of) task context and action effect anticipation and perception triggers and mediates learning. In particular, we stress that the cognitive system employs anticipation as reflection of both its learned knowledge so far and its interpretation of the current context. Anticipation can subsequently mediate learning by influencing which features engage in learning (and even further: what features to look for in the sensory input) and how strongly these features may be associated to motor codes, thereby constraining whatever (machine) learning technique used to actually create or change the associations.

Moreover, failing to correctly anticipate an action effect may be a major trigger to update the learned knowledge. In the future we may add this as a reinforcement learning component that drives on biologically plausible reward mechanisms (e.g., dopamine moderated learning).

Finally, we stress that although simulations may be set up in terms of instruction, train and test phases, the HiTEC model itself does not artificially 'switch' between two modes of operation: learning occurs on line as a result of perceiving action effects.

7 Examplary Scenario: Responding to Traffic Lights

In order to clarify the co-operation of the different processes and mechanisms in HiTEC on a functional level, the following example real life scenario is presented: learning to respond to traffic lights. In this example, s_i denotes sensory codes, f_i denotes feature codes and m_i denotes motor codes in the HiTEC architecture. Figure 2 shows a scenario-specific version of the HiTEC architecture.

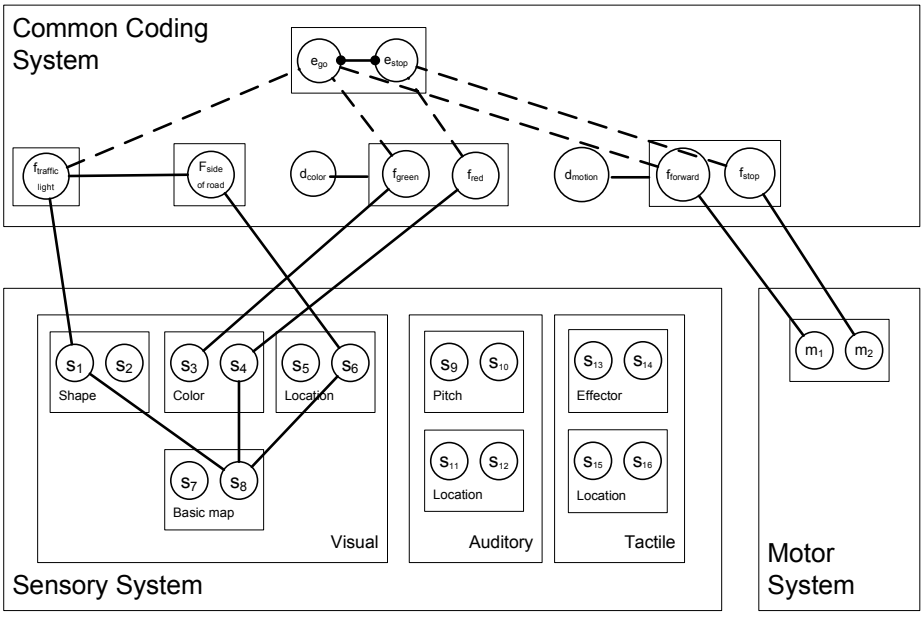


Fig. 2. Learning to respond to traffic lights in HiTEC

7.1 Action Effect Acquisition

Let's say you are a student driver who has never paid attention to the front seat before and this is your first driving lesson. You climb behind the steering wheel and place your feet above the pedals. Now, the instructor starts the car for you and you get the chance of playing around with the pedals. After a while, you get the hang of it: it seems that pressing the right pedal results in a forward movement of the car, and pressing the left one puts the car on hold.

From a HiTEC perspective, you just have tried some motor codes and learned that m_1 (pressing the gas pedal) results in a forward motion, coded by $f_{forward}$ and m_2 in standing still, coded by f_{stop}. In other words: you acquired these particular action-effect

associations. Note that we assume that you have been able to walk before, so it is fair to say that $f_{forward}$ and f_{stop} are already present as feature codes in your common coding system.

7.2 Using Action Effect Associations

Now, in your next lesson you actually need to take cross roads. The instructor tells you to pay attention to these colored lights next to the road. When the red light is on, you should stop, and when the green light is on, you can go forward.

In HiTEC, this verbal instruction is modeled as creating two event files that hold short term associations in working memory: $e_{stop\ for\ red\ light}$ for the 'stop' condition, and $e_{go\ at\ green\ light}$ for the 'forward' condition. The event file $e_{stop\ for\ red\ light}$ contains bindings of feature codes f_{red}, $f_{traffic\ light}$, f_{stop} and the event file $e_{go\ at\ green\ light}$ relates to the feature codes f_{green}, $f_{traffic\ light}$, $f_{forward}$.

These event files are activated and their activation spreads to their associated feature codes which will become increasingly receptive for interaction with related sensory codes. In addition to the specific features, the feature dimensions these features are contained in (d_{color}, d_{motion}) are weighted as well. The anticipation of traffic lights also serves as a retrieval cue for prior experience with looking at traffic lights. As traffic lights typically stand at the side of the road, one could expect associations between $f_{traffic\ light}$ and $f_{side\ of\ road}$ to exist in episodic or semantic memory. Consequently, anticipating a traffic light activates $f_{traffic\ light}$ and propagates activation automatically towards $f_{side\ of\ road}$, which makes the system more sensitive to objects located on the side of the road.

Ok, there it goes... you start to drive around, take some turns, and there it is… your very first cross road with traffic lights!

Now, from a HiTEC perspective, the following takes place: the visual scene consists of a plethora of objects, like road signs, other cars, houses and scenery, and of a cross road with traffic lights at the side. The sensory system encodes the registration of these objects by activating the codes in the sensory maps. This leads to the classical binding problem: multiple shapes are registered, multiple colors and multiple locations. However, we now have a top down 'special interest' for traffic lights. As mentioned above, this has resulted in increased sensitivity of the $f_{traffic\ light}$ feature code, that now receives some external stimulation from related sensory codes. Also, from prior experience we look more closely at $f_{side\ of\ road}$ locations in the sensory location maps.

The interaction between this top down sensitivity and the bottom up external stimulation results in an interactive process where the sensory system uses feedback signals to the lower level visual maps where local interactions result in higher activation of those sensory codes that code for properties of the traffic light, including its color. In the visual map for object color, the traffic light color will be more enhanced than colors relating other objects. On the feature code level, the color dimension already was enhanced because of the anticipation of features in the d_{color} dimension, resulting in fast detection of f_{red} or f_{green}.

Meanwhile, the event files $e_{stop\ for\ red\ light}$ and $e_{stop\ for\ red\ light}$ are still in competition. When the sensory system collects the evidence, activation propagates towards feature codes and event codes, quickly converging into a state that where either $f_{forward}$ or f_{stop}

is activated more strongly than the other. This activation is propagated towards the motor codes m_1 or m_2 via associations learned in your first drivers lesson. This results in the selection and execution of the correct motor action.

It is clear that by preparing the cognitive system for perceiving a traffic light color and producing a stop-or-go action allows the system to effectively attend its resources to the crucial sensory input and already pre-anticipate the possible action outcome. This way, upon perceiving the actual traffic light color, the system can quickly respond with the correct motor action.

Luckily, for your safety and that of all your fellow drivers on the road, practicing this task long enough will also result in long term memory bindings between f_{red}, $f_{traffic\ light}$ and f_{stop} that will also be retrieved during action selection and bias you towards pressing the brake pedal, even when no instructor is sitting next to you.

8 Conclusions

We have introduced HiTEC's three main modules: the sensory system, the motor system, and the emergent common coding system. These systems interact with each other. In the common coding system anticipations are formed that have a variety of uses in the architecture, allowing the system to be more flexible and adaptive. In action selection, anticipation acts as a rich retrieval cue for associated motor programs. At the same time, forming this anticipation reflects the specification of an action plan that can be used during action execution.

One of the drawbacks of creating anticipations is that it might not be worth the costs (Butz & Pezzulo, 2008). However, from a real life scenario perspective, the number of possible action alternatives is enormous. Creating anticipations at a distal level seems as a necessity to constrain the system in its actions to select from. Doing this, as we propose in HiTEC, not only aids action selection but also delivers the rudimentary action plan at the same time.

Another concern often mentioned is the inaccuracy of predictions. Following the framework of event coding, events – including action plans – are coded in distal terms that abstract away from the proximal sensory values. Only inaccuracies on the distal level could disturb the use of anticipations in action selection and planning. The feature codes on this distal level are based on sensorimotor regularities that are stable over time. Thus minor inaccuracies in sensors should be relatively easily overcome.

Actions are usually selected and planned in a task context. When forced with different behavioral alternatives to choose from, multiple anticipations of features are created and compete with each other. When features are actually perceived, anticipatory activation quickly propagates to the correct action effects, which results in the selection and execution of the correct motor action.

In action monitoring, anticipation serves as the representation of expected and desired action effects that helps adjusting the movement during action execution. In action evaluation, this expectation acts as a set of criteria for success of the action. If the actual action effect can no longer – on a lower sensorimotor level - be adjusted to fulfill the expected action effect, the existing action-effect associations are considered insufficient and learning is triggered. During action-effect learning, anticipation also may weight the different action effect features in the automatic integration into action concepts, influencing the action-effect association weights.

In conclusion, anticipation plays a crucial role in virtually all aspects of action control within the HiTEC architecture. Just as it does in real life.

Acknowledgments

Support for this research by the European Commission (PACO-PLUS, IST-FP6-IP-027657) is gratefully acknowledged.

References

1. Andersen, R.A.: The neurobiological basis of spatial cognition: Role of the parietal lobe. In: Stiles-Davis, J., Krtichivsky, M., Belugi, U. (eds.) Spatial cognition: Brain bases and development. Erlbaum, Mahwah (1988)
2. Brunswik, E.: Distal focussing of perception: Size constancy in a representative sample of situations. Psychological Monographs 56(1) (1944)
3. Butz, M.V., Pezzulo, G.: Benefits of Anticipations in Cognitive Agents. In: Pezzulo, G., Butz, M.V., Castelfranchi, C., Falcone, R. (eds.) The Challenge of Anticipation. LNCS, vol. 5225, pp. 45–62. Springer, Heidelberg (2008)
4. Colby, C.L.: Action-oriented spatial reference frames in the cortex. Neuron 20, 15–20 (1998)
5. Cowey, A.: Aspects of cortical organization related to selective attention and selective impairments of visual perception: A tutorial review. In: Poster, M.I., Marin, O.S.M. (eds.) Attention and performance XI, pp. 41–62. Erlbaum, Hillsdale (1985)
6. Dewey, J.: The reflex arc concept in psychology. Psychological Review 3, 357–370 (1896)
7. DeYoe, E.A., Van Essen, D.C.: Concurrent processing streams in monkey visual cortex. Trends in Neuroscience 11, 219–226 (1988)
8. Elsner, B., Hommel, B.: Effect anticipation and action control. Journal of Experimental Psychology: Human Perception and Performance 27 (2001)
9. Gibson, J.J.: The ecological approach to visual perception. Houghton Mifflin, Boston (1979)
10. Glover, S.: Separate visual representations in the planning and control of action. Behavioral and Brain Sciences 27, 3–24 (2004)
11. Greenwald, A.: Sensory feedback mechanisms in performance control: With special reference to the ideomotor mechanism. Psychological Review 77, 73–99 (1970)
12. Harless, E.: Der Apparat des Willens. Zeitschrift fuer Philosophie und philosophische Kritik 38, 50–73 (1861)
13. Haazebroek, P., Hommel, B.: HiTEC: A computational model of the interaction between perception and action (submitted)
14. Heider, F.: Thing and medium. Psychological Issues, Monograph 3 (original work published 1959) (1926/1959)
15. Heider, F.: The function of the perceptual system. Psychological Issues, 1959, Monograph, 371–394 (1930/1959) (original work published 1930)
16. Hommel, B.: The cognitive representation of action: Automatic integration of perceived action effects. Psychological Research 59, 176–186 (1996)
17. Hommel, B.: The prepared reflex: Automaticity and control in stimulus-response translation. In: Monsell, S., Driver, J. (eds.) Control of cognitive processes: Attention and performance XVIII, pp. 247–273. MIT Press, Cambridge (2000)

18. Hommel, B.: Event files: Feature binding in and across perception and action. Trends in Cognitive Sciences 8, 494–500 (2004)
19. Hommel, B.: Grounding attention in action control: The intentional control of selection. In: Bruya, B.J. (ed.) Effortless attention: A new perspective in the cognitive science of attention and action. MIT Press, Cambridge
20. Hommel, B., Elsner, B.: Acquisition, representation, and control of action. In: Morsella, E., Bargh, J.A., Gollwitzer, P.M. (eds.) Oxford handbook of human action, pp. 371–398. Oxford University Press, New York (2009)
21. Hommel, B., Müsseler, J., Aschersleben, G., Prinz, W.: The Theory of Event Coding (TEC): A framework for perception and action planning. Behavioral and Brain Sciences 24, 849–937 (2001)
22. James, W.: The principles of psychology. Dover Publications, New York (1890)
23. Lotze, R.H.: Medicinische Psychologie oder die Physiologie der Seele. Weidmann'sche Buchhandlung, Leipzig (1852)
24. McClelland, J.L.: Toward a theory of information processing in graded, random, and interactive networks. In: Meyer, D.E., Kornblum, S. (eds.) Attention and performance XIV: Synergies in experimental psychology, artificial intelligence and cognitive neuroscience – A Silver Jubilee Volume. MIT Press, Cambridge (1992)
25. Milner, A.D., Goodale, M.A.: The visual brain in action. Oxford University Press, Oxford (1995)
26. O'Reilly, R.C., Norman, K.A.: Hippocampal and neocortical contributions to memory: Advances in the complementary learning systems framework. Trends in Cognitive Sciences 6, 505–510 (2002)
27. Piaget, J.: The origins of intelligence in childhood. International Universities Press (1952)
28. Prinz, W.: Ideo-motor action. In: Heuer, H., Sanders, A.F. (eds.) Perspectives on perception and action. Erlbaum, Hillsdale (1987)
29. Prinz, W.: A common coding approach to perception and action. In: Neumann, O., Prinz, W. (eds.) Relationships between perception and action, pp. 167–201. Springer, Berlin (1990)
30. Prinz, W.: Why don't we perceive our brain states? European Journal of Cognitive Psychology 4, 1–20 (1992)
31. Rumelhart, D.E., Hinton, G.E., McClelland, J.L.: A general framework for parallel distributed processing. In: Rumelhart, D.E., McClelland, J.L., The PDP Research Group (eds.) Parallel distributed processing: Explorations in the microstructure of cognition, vol. 1, pp. 45–76. MIT Press, Cambridge (1986)
32. Schmidt, R.A.: A schema theory of discrete motor skill learning. Psychological Review 82, 225–260 (1975)
33. Stock, A., Stock, C.: A short history of ideo-motor action. Psychological Research 68, 176–188 (2004)
34. Thorndike, E.L.: Ideo-motor action. Psychological Review 20, 91–106 (1913)
35. Treisman, A.: The binding problem. Current Opinion in Neurobiology 6, 171–178 (1996)
36. Van der Velde, F., De Kamps, M.: From knowing what to knowing where: Modeling object-based attention with feedback disinhibition of activation. Journal of Cognitive Neuroscience 13(4), 479–491 (2001)
37. Van der Velde, F., De Kamps, M., Van der Voort van der Kleij, G.: Clam: Closed-loop attention model for visual search. Neurocomputing 58-60, 607–612 (2004)
38. Wilson, H., Cowan, J.: Excitatory and inhibitory interactions in localized populations of model neurons. Biophysics Journal 12, 1–24 (1972)

Driven by Compression Progress: A Simple Principle Explains Essential Aspects of Subjective Beauty, Novelty, Surprise, Interestingness, Attention, Curiosity, Creativity, Art, Science, Music, Jokes*

Jürgen Schmidhuber

TU Munich, Boltzmannstr. 3, 85748 Garching bei München, Germany &
IDSIA, Galleria 2, 6928 Manno (Lugano), Switzerland
juergen@idsia.ch
http://www.idsia.ch/~juergen

Abstract. I argue that data becomes temporarily interesting by itself to some self-improving, but computationally limited, subjective observer once he learns to predict or compress the data in a better way, thus making it subjectively simpler and more *beautiful.* Curiosity is the desire to create or discover more non-random, non-arbitrary, regular data that is novel and *surprising* not in the traditional sense of Boltzmann and Shannon but in the sense that it allows for compression progress because its regularity was not yet known. This drive maximizes *interestingness,* the first derivative of subjective beauty or compressibility, that is, the steepness of the learning curve. It motivates exploring infants, pure mathematicians, composers, artists, dancers, comedians, yourself, and (since 1990) artificial systems.

1 Store and Compress and Reward Compression Progress

If the history of the entire universe were computable [123, 124], and there is no evidence against this possibility [84], then its simplest explanation would be the shortest program that computes it [65, 70]. Unfortunately there is no general way of finding the shortest program computing any given data [34, 37, 106, 107]. Therefore physicists have traditionally proceeded incrementally, analyzing just a small aspect of the world at any given time, trying to find simple laws that allow for describing their limited observations better than the best previously known law, essentially trying to find a program that compresses the observed data better than the best previously known program. For example, Newton's law of gravity can be formulated as a short piece of code which allows for substantially compressing many observation sequences involving falling apples and other objects. Although its predictive power is limited—for example, it does

* First version of this preprint published 23 Dec 2008; revised April 2009. Variants are scheduled to appear as references [90] and [91] (short version), distilling some of the essential ideas in earlier work (1990-2008) on this subject: [57, 58, 59, 60, 61, 68, 72, 76, 108] and especially recent papers [81, 87, 88, 89].

G. Pezzulo et al. (Eds.): ABiALS 2008, LNAI 5499, pp. 48–76, 2009.

not explain quantum fluctuations of apple atoms—it still allows for greatly reducing the number of bits required to encode the data stream, by assigning short codes to events that are predictable with high probability [28] under the assumption that the law holds. Einstein's general relativity theory yields additional compression progress as it compactly explains many previously unexplained deviations from Newton's predictions.

Most physicists believe there is still room for further advances. Physicists, however, are not the only ones with a desire to improve the subjective compressibility of their observations. Since short and simple explanations of the past usually reflect some repetitive regularity that helps to predict the future as well, *every* intelligent system interested in achieving future goals should be motivated to compress the history of raw sensory inputs in response to its actions, simply to improve its ability to plan ahead.

A long time ago, Piaget [49] already explained the explorative learning behavior of children through his concepts of assimilation (new inputs are embedded in old schemas—this may be viewed as a type of compression) and accommodation (adapting an old schema to a new input—this may be viewed as a type of compression improvement), but his informal ideas did not provide enough formal details to permit computer implementations of his concepts. How to model a compression progress drive in artificial systems? Consider an active agent interacting with an initially unknown world. We may use our general Reinforcement Learning (RL) framework of artificial curiosity (1990-2008) [57,58,59,60,61,68,72,76,81,87,88,89,108] to make the agent discover data that allows for additional compression progress and improved predictability. The framework directs the agent towards a better understanding the world through active exploration, even when external reward is rare or absent, through *intrinsic reward* or *curiosity reward* for actions leading to discoveries of previously unknown regularities in the action-dependent incoming data stream.

1.1 Outline

Section 1.2 will informally describe our algorithmic framework based on: (1) a continually improving predictor or compressor of the continually growing data history, (2) a computable measure of the compressor's progress (to calculate intrinsic rewards), (3) a reward optimizer or reinforcement learner translating rewards into action sequences expected to maximize future reward. The formal details are left to the Appendix, which will elaborate on the underlying theoretical concepts and describe discrete time implementations. Section 1.3 will discuss the relation to external reward (external in the sense of: originating outside of the brain which is controlling the actions of its "external" body). Section 2 will informally show that many essential ingredients of intelligence and cognition can be viewed as natural consequences of our framework, for example, detection of novelty & surprise & interestingness, unsupervised shifts of attention, subjective perception of beauty, curiosity, creativity, art, science, music, and jokes. In particular, we reject the traditional Boltzmann / Shannon notion of surprise, and demonstrate that both science and art can be regarded as by-products of the desire to create / discover more data that is compressible in hitherto unknown ways. Section 3 will give an overview of previous concrete implementations of approximations of our framework. Section 4 will apply the theory to images tailored to human observers, illustrating the rewarding learning process leading from less to more subjective com-

pressibility. Section 5 will outline how to improve our previous implementations, and how to further test predictions of our theory in psychology and neuroscience.

1.2 Algorithmic Framework

The basic ideas are embodied by the following set of simple algorithmic principles distilling some of the essential ideas in previous publications on this topic [57, 58, 59, 60, 61, 68, 72, 76, 81, 87, 88, 89, 108] As mentioned above, formal details are left to the Appendix. As discussed in Section 2, the principles at least qualitatively explain many aspects of intelligent agents such as humans. This encourages us to implement and evaluate them in cognitive robots and other artificial systems.

1. **Store everything.** During interaction with the world, store the entire raw history of actions and sensory observations including reward signals—the data is *holy* as it is the only basis of all that can be known about the world. To see that full data storage is not unrealistic: A human lifetime rarely lasts much longer than 3×10^9 seconds. The human brain has roughly 10^{10} neurons, each with 10^4 synapses on average. Assuming that only half of the brain's capacity is used for storing raw data, and that each synapse can store at most 6 bits, there is still enough capacity to encode the lifelong sensory input stream with a rate of roughly 10^5 bits/s, comparable to the demands of a movie with reasonable resolution. The storage capacity of affordable technical systems will soon exceed this value. If you can store the data, do not throw it away!
2. **Improve subjective compressibility.** In principle, any regularity in the data history can be used to compress it. The compressed version of the data can be viewed as its simplifying explanation. Thus, to better explain the world, spend some of the computation time on an adaptive compression algorithm trying to partially compress the data. For example, an adaptive neural network [8] may be able to learn to predict or postdict some of the historic data from other historic data, thus incrementally reducing the number of bits required to encode the whole. See Appendix A.3 and A.5.
3. **Let intrinsic curiosity reward reflect compression progress.** The agent should monitor the improvements of the adaptive data compressor: whenever it learns to reduce the number of bits required to encode the historic data, generate an intrinsic reward signal or curiosity reward signal in proportion to the learning progress or compression progress, that is, the number of saved bits. See Appendix A.5 and A.6.
4. **Maximize intrinsic curiosity reward** [57, 58, 59, 60, 61, 68, 72, 76, 81, 88, 87, 108]. Let the action selector or controller use a general Reinforcement Learning (RL) algorithm (which should be able to observe the current state of the adaptive compressor) to maximize expected reward, including intrinsic curiosity reward. To optimize the latter, a good RL algorithm will select actions that focus the agent's attention and learning capabilities on those aspects of the world that allow for finding or creating new, previously unknown but learnable regularities. In other words, it will try to maximize the steepness of the compressor's learning curve. This type of *active unsupervised learning* can help to figure out how the world works. See Appendix A.7, A.8, A.9, A.10.

The framework above essentially specifies the objectives of a curious or creative system, not the way of achieving the objectives through the choice of a particular adaptive compressor or predictor and a particular RL algorithm. Some of the possible choices leading to special instances of the framework (including previous concrete implementations) will be discussed later.

1.3 Relation to External Reward

Of course, the real goal of many cognitive systems is not just to satisfy their curiosity, but to solve externally given problems. Any formalizable problem can be phrased as an RL problem for an agent living in a possibly unknown environment, trying to maximize the future external reward expected until the end of its possibly finite lifetime. The new millennium brought a few extremely general, even universal RL algorithms (universal problem solvers or universal artificial intelligences—see Appendix A.8, A.9) that are optimal in various theoretical but not necessarily practical senses, e. g., [29, 79, 82, 83, 86, 85, 92]. To the extent that learning progress / compression progress / curiosity as above are helpful, these universal methods will automatically discover and exploit such concepts. Then why bother at all writing down an explicit framework for active curiosity-based experimentation?

One answer is that the present universal approaches sweep under the carpet certain problem-independent constant slowdowns, by burying them in the asymptotic notation of theoretical computer science. They leave open an essential remaining question: If the agent can execute only a fixed number of computational instructions per unit time interval (say, 10 trillion elementary operations per second), what is the best way of using them to get as close as possible to the recent theoretical limits of universal AIs, especially when external rewards are very rare, as is the case in many realistic environments? The premise of this paper is that the curiosity drive is such a general and generally useful concept for limited-resource RL in rare-reward environments that it should be prewired, as opposed to be learnt from scratch, to save on (constant but possibly still huge) computation time. An inherent assumption of this approach is that in realistic worlds a better explanation of the past can only help to better predict the future, and to accelerate the search for solutions to externally given tasks, ignoring the possibility that curiosity may actually be harmful and "kill the cat."

2 Consequences of the Compression Progress Drive

Let us discuss how many essential ingredients of intelligence and cognition can be viewed as natural by-products of the principles above.

2.1 Compact Internal Representations or Symbols as by-Products of Efficient History Compression

To compress the history of observations so far, the compressor (say, a predictive neural network) will automatically create internal representations or *symbols* (for example, patterns across certain neural feature detectors) for things that frequently repeat themselves. Even when there is limited predictability, efficient compression can still

be achieved by assigning short codes to events that are predictable with high probability [28, 95]. For example, the sun goes up every day. Hence it is efficient to create internal symbols such as *daylight* to describe this repetitive aspect of the data history by a short reusable piece of internal code, instead of storing just the raw data. In fact, predictive neural networks are often observed to create such internal (and hiearchical) codes as a by-product of minimizing their prediction error on the training data.

2.2 Consciousness as a Particular by-Product of Compression

There is one thing that is involved in all actions and sensory inputs of the agent, namely, the agent itself. To efficiently encode the entire data history, it will profit from creating some sort of internal *symbol* or code (e. g., a neural activity pattern) representing the agent itself. Whenever this representation is actively used, say, by activating the corresponding neurons through new incoming sensory inputs or otherwise, the agent could be called *self-aware* or *conscious.*

This straight-forward explanation apparently does not abandon any essential aspects of our intuitive concept of consciousness, yet seems substantially simpler than other recent views [1, 2, 105, 101, 25, 12]. In the rest of this paper we will not have to attach any particular mystic value to the notion of consciousness—in our view, it is just a natural by-product of the agent's ongoing process of problem solving and world modeling through data compression, and will not play a prominent role in the remainder of this paper.

2.3 The Lazy Brain's Subjective, Time-Dependent Sense of Beauty

Let $O(t)$ denote the state of some subjective observer O at time t. According to our *lazy brain theory* [67, 66, 69, 81, 87, 88], we may identify the subjective beauty $B(D, O(t))$ of a new observation D (but not its interestingness - see Section 2.4) as being proportional to the number of bits required to encode D, given the observer's limited previous knowledge embodied by the current state of its adaptive compressor. For example, to efficiently encode previously viewed human faces, a compressor such as a neural network may find it useful to generate the internal representation of a prototype face. To encode a new face, it must only encode the deviations from the prototype [67]. Thus a new face that does not deviate much from the prototype [17, 48] will be subjectively more beautiful than others. Similarly for faces that exhibit geometric regularities such as symmetries or simple proportions [69, 88]—in principle, the compressor may exploit any regularity for reducing the number of bits required to store the data.

Generally speaking, among several sub-patterns classified as *comparable* by a given observer, the subjectively most beautiful is the one with the simplest (shortest) description, given the observer's current particular method for encoding and memorizing it [67, 69]. For example, mathematicians find beauty in a simple proof with a short description in the formal language they are using. Others like geometrically simple, aesthetically pleasing, low-complexity drawings of various objects [67, 69].

This immediately explains why many human observers prefer faces similar to their own. What they see every day in the mirror will influence their subjective prototype face, for simple reasons of coding efficiency.

2.4 Subjective Interestingness as First Derivative of Subjective Beauty: The Steepness of the Learning Curve

What's beautiful is not necessarily interesting. A beautiful thing is interesting only as long as it is new, that is, as long as the algorithmic regularity that makes it simple has not yet been fully assimilated by the adaptive observer who is still learning to compress the data better. It makes sense to define the time-dependent subjective *Interestingness* $I(D, O(t))$ of data D relative to observer O at time t by

$$I(D, O(t)) \sim \frac{\partial B(D, O(t))}{\partial t}, \tag{1}$$

the *first derivative* of subjective beauty: as the learning agent improves its compression algorithm, formerly apparently random data parts become subjectively more regular and beautiful, requiring fewer and fewer bits for their encoding. As long as this process is not over the data remains interesting and rewarding. The Appendix and Section 3 on previous implementations will describe details of discrete time versions of this concept. See also [59, 60, 108, 68, 72, 76, 81, 88, 87].

2.5 Pristine Beauty and Interestingness vs. External Rewards

Note that our above concepts of beauty and interestingness are limited and *pristine* in the sense that they are *not a priori* related to pleasure derived from external rewards (compare Section 1.3). For example, some might claim that a hot bath on a cold day triggers "beautiful" feelings due to rewards for achieving prewired target values of external temperature sensors (external in the sense of: outside the brain which is controlling the actions of its external body). Or a song may be called "beautiful" for emotional (e.g., [13]) reasons by some who associate it with memories of external pleasure through their first kiss. Obviously this is not what we have in mind here—we are focusing solely on rewards of the intrinsic type based on learning progress.

2.6 True Novelty and Surprise vs. Traditional Information Theory

Consider two extreme examples of uninteresting, unsurprising, boring data: A vision-based agent that always stays in the dark will experience an extremely compressible, soon totally predictable history of unchanging visual inputs. In front of a screen full of white noise conveying a lot of information and "novelty" and "surprise" in the traditional sense of Boltzmann and Shannon [102], however, it will experience highly unpredictable and fundamentally incompressible data. In both cases the data is boring [72, 88] as it does not allow for further compression progress. Therefore we reject the traditional notion of surprise. Neither the arbitrary nor the fully predictable is *truly* novel or surprising—only data with still *unknown* algorithmic regularities are [57, 58, 61, 59, 60, 108, 68, 72, 76, 81, 88, 87, 89]!

2.7 Attention / Curiosity / Active Experimentation

In absence of external reward, or when there is no known way to further increase the expected external reward, our controller essentially tries to maximize *true novelty* or *interestingness,* the *first derivative* of subjective beauty or compressibility, the

steepness of the learning curve. It will do its best to select action sequences expected to create observations yielding maximal expected future compression *progress*, given the limitations of both the compressor and the compressor improvement algorithm. It will learn to focus its attention [96, 116] and its actively chosen experiments on things that are currently still incompressible but are expected to become compressible / predictable through additional learning. It will get bored by things that already are subjectively compressible. It will also get bored by things that are currently incompressible but will apparently remain so, given the experience so far, or where the costs of making them compressible exceed those of making other things compressible, etc. [57, 58, 61, 59, 60, 108, 68, 72, 76, 81, 88, 87, 89].

2.8 Discoveries

An unusually large compression breakthrough deserves the name *discovery*. For example, as mentioned in the introduction, the simple law of gravity can be described by a very short piece of code, yet it allows for greatly compressing all previous observations of falling apples and other objects.

2.9 Beyond Standard Unsupervised Learning

Traditional unsupervised learning is about finding regularities, by clustering the data, or encoding it through a factorial code [4, 64] with statistically independent components, or predicting parts of it from other parts. All of this may be viewed as special cases of data compression. For example, where there are clusters, a data point can be efficiently encoded by its cluster center plus relatively few bits for the deviation from the center. Where there is data redundancy, a non-redundant factorial code [64] will be more compact than the raw data. Where there is predictability, compression can be achieved by assigning short codes to those parts of the observations that are predictable from previous observations with high probability [28, 95]. Generally speaking we may say that a major goal of traditional unsupervised learning is to improve the compression of the observed data, by discovering a program that computes and thus explains the history (and hopefully does so quickly) but is clearly shorter than the shortest previously known program of this kind.

Traditional unsupervised learning is not enough though—it just analyzes and encodes the data but does not choose it. We have to extend it along the dimension of active action selection, since our unsupervised learner must also choose the actions that influence the observed data, just like a scientist chooses his experiments, a baby its toys, an artist his colors, a dancer his moves, or any attentive system [96] its next sensory input. That's precisely what is achieved by our RL-based framework for curiosity and creativity.

2.10 Art and Music as by-Products of the Compression Progress Drive

Works of art and music may have important purposes beyond their social aspects [3] despite of those who classify art as superfluous [50]. Good observer-dependent art deepens the observer's insights about this world or possible worlds, unveiling previously unknown regularities in compressible data, connecting previously disconnected

patterns in an initially surprising way that makes the combination of these patterns subjectively more compressible (art as an eye-opener), and eventually becomes known and less interesting. I postulate that the active creation and attentive perception of all kinds of artwork are just by-products of our principle of interestingness and curiosity yielding reward for compressor improvements.

Let us elaborate on this idea in more detail, following the discussion in [81, 88]. Artificial or human observers must perceive art sequentially, and typically also actively, e.g., through a sequence of attention-shifting eye saccades or camera movements scanning a sculpture, or internal shifts of attention that filter and emphasize sounds made by a pianist, while surpressing background noise. Undoubtedly many derive pleasure and rewards from perceiving works of art, such as certain paintings, or songs. But different subjective observers with different sensory apparati and compressor improvement algorithms will prefer different input sequences. Hence any objective theory of what is good art must take the subjective observer as a parameter, to answer questions such as: Which sequences of actions and resulting shifts of attention should he execute to maximize his pleasure? According to our principle he should select one that maximizes the quickly learnable compressibility that is new, relative to his current knowledge and his (usually limited) way of incorporating / learning / compressing new data.

2.11 Music

For example, which song should some human observer select next? Not the one he just heard ten times in a row. It became too predictable in the process. But also not the new weird one with the completely unfamiliar rhythm and tonality. It seems too irregular and contain too much arbitrariness and subjective noise. He should try a song that is unfamiliar enough to contain somewhat unexpected harmonies or melodies or beats etc., but familiar enough to allow for quickly recognizing the presence of a new learnable regularity or compressibility in the sound stream. Sure, this song will get boring over time, but not yet.

The observer dependence is illustrated by the fact that Schönberg's twelve tone music is less popular than certain pop music tunes, presumably because its algorithmic structure is less obvious to many human observers as it is based on more complicated harmonies. For example, frequency ratios of successive notes in twelve tone music often cannot be expressed as fractions of very small integers. Those with a prior education about the basic concepts and objectives and constraints of twelve tone music, however, tend to appreciate Schönberg more than those without such an education.

All of this perfectly fits our principle: The learning algorithm of the compressor of a given subjective observer tries to better compress his history of acoustic and other inputs where possible. The action selector tries to find history-influencing actions that help to improve the compressor's performance on the history so far. The interesting musical and other subsequences are those with previously unknown yet learnable types of regularities, because they lead to compressor improvements. The boring patterns are those that seem arbitrary or random, or whose structure seems too hard to understand.

2.12 Paintings, Sculpture, Dance, Film etc.

Similar statements not only hold for other dynamic art including film and dance (taking into account the compressibility of controller actions), but also for painting and sculpture, which cause dynamic pattern sequences due to attention-shifting actions [96, 116] of the observer.

2.13 No Objective "Ideal Ratio" between Expected and Unexpected

Some of the previous attempts at explaining aesthetic experiences in the context of information theory [7, 41, 6, 44] emphasized the idea of an *"ideal"* ratio between expected and unexpected information conveyed by some aesthetic object (its *"order"* vs its *"complexity"*). Note that our alternative approach does not have to postulate an objective ideal ratio of this kind. Instead our dynamic measure of interestingness reflects the *change* in the number of bits required to encode an object, and explicitly takes into account the subjective observer's prior knowledge as well as the limitations of its compression improvement algorithm.

2.14 Blurred Boundary between *Active* Creative Artists and *Passive* Perceivers of Art

Just as observers get intrinsic rewards for sequentially focusing attention on artwork that exhibits new, previously unknown regularities, the *creative* artists get reward for making it. For example, I found it extremely rewarding to discover (after hundreds of frustrating failed attempts) the simple geometric regularities that permitted the construction of the drawings in Figures 1 and 2. The distinction between artists and observers is blurred though. Both execute action sequences to exhibit new types of compressibility. The intrinsic motivations of both are fully compatible with our simple principle.

Some artists, of course, crave *external* reward from other observers, in form of praise, money, or both, in addition to the *intrinsic* compression improvement-based reward that comes from creating a truly novel work of art. Our principle, however, conceptually separates these two reward types.

2.15 How Artists and Scientists Are Alike

From our perspective, scientists are very much like artists. They actively select experiments in search for simple but new laws compressing the resulting observation history. In particular, the *creativity* of painters, dancers, musicians, pure mathematicians, physicists, can be viewed as a mere by-product of our curiosity framework based on the compression progress drive. All of them try to create new but non-random, non-arbitrary data with surprising, previously unknown regularities. For example, many physicists invent experiments to create data governed by previously unknown laws allowing to further compress the data. On the other hand, many artists combine well-known objects in a subjectively novel way such that the observer's subjective description of the result is shorter than the sum of the lengths of the descriptions of the parts, due to some previously unnoticed regularity shared by the parts.

What is the main difference between science and art? The essence of science is to *formally nail down* the nature of compression progress achieved through the discovery of a new regularity. For example, the law of gravity can be described by just a few symbols. In the fine arts, however, compression progress achieved by observing an artwork combining previously disconnected things in a new way (art as an eye-opener) may be *sub*conscious and not at all formally describable by the observer, who may *feel* the progress in terms of intrinsic reward without being able to say exactly which of his memories became more subjectively compressible in the process.

The framework in the appendix is sufficiently formal to allow for implementation of our principle on computers. The resulting artificial observers will vary in terms of the computational power of their history compressors and learning algorithms. This will influence what is good art / science to them, and what they find interesting.

2.16 Jokes and Other Sources of Fun

Just like other entertainers and artists, comedians also tend to combine well-known concepts in a novel way such that the observer's subjective description of the result is shorter than the sum of the lengths of the descriptions of the parts, due to some previously unnoticed regularity shared by the parts.

In many ways the laughs provoked by witty jokes are similar to those provoked by the acquisition of new skills through both babies and adults. Past the age of 25 I learnt to juggle three balls. It was not a sudden process but an incremental and rewarding one: in the beginning I managed to juggle them for maybe one second before they fell down, then two seconds, four seconds, etc., until I was able to do it right. Watching myself in the mirror (as recommended by juggling teachers) I noticed an idiotic grin across my face whenever I made progress. Later my little daughter grinned just like that when she was able to stand on her own feet for the first time. All of this makes perfect sense within our algorithmic framework: such grins presumably are triggered by intrinsic reward for generating a data stream with previously unknown regularities, such as the sensory input sequence corresponding to observing oneself juggling, which may be quite different from the more familiar experience of observing somebody else juggling, and therefore truly novel and intrinsically rewarding, until the adaptive predictor / compressor gets used to it.

3 Previous Concrete Implementations of Systems Driven by (Approximations of) Compression Progress

As mentioned earlier, predictors and compressors are closely related. Any type of partial predictability of the incoming sensory data stream can be exploited to improve the compressibility of the whole. Therefore the systems described in the first publications on artificial curiosity [57,58,61] already can be viewed as examples of implementations of a compression progress drive.

3.1 Reward for Prediction Error (1990)

Early work [57,58,61] described a predictor based on a recurrent neural network [115, 120,55,62,47,78] (in principle a rather powerful computational device, even by today's

machine learning standards), predicting sensory inputs including reward signals from the entire history of previous inputs and actions. The curiosity rewards were proportional to the predictor errors, that is, it was implicitly and optimistically assumed that the predictor will indeed improve whenever its error is high.

3.2 Reward for Compression Progress through Predictor Improvements (1991)

Follow-up work [59,60] pointed out that this approach may be inappropriate, especially in probabilistic environments: one should not focus on the errors of the predictor, but on its improvements. Otherwise the system will concentrate its search on those parts of the environment where it can always get high prediction errors due to noise or randomness, or due to computational limitations of the predictor, which will prevent improvements of the subjective compressibility of the data. While the neural predictor of the implementation described in the follow-up work was indeed computationally less powerful than the previous one [61], there was a novelty, namely, an explicit (neural) adaptive model of the predictor's improvements. This model essentially learned to predict the predictor's changes. For example, although noise was unpredictable and led to wildly varying target signals for the predictor, in the long run these signals did not change the adaptive predictor parameters much, and the predictor of predictor changes was able to learn this. A standard RL algorithm [114,33,109] was fed with curiosity reward signals proportional to the expected long-term predictor changes, and thus tried to maximize information gain [16, 31, 38, 51, 14] within the given limitations. In fact, we may say that the system tried to maximize an approximation of the (discounted) sum of the expected first derivatives of the data's subjective predictability, thus also maximizing an approximation of the (discounted) sum of the expected changes of the data's subjective compressibility.

3.3 Reward for Relative Entropy between Agent's Prior and Posterior (1995)

Additional follow-up work yielded an information theory-oriented variant of the approach in non-deterministic worlds [108] (1995). The curiosity reward was again proportional to the predictor's surprise / information gain, this time measured as the Kullback-Leibler distance [35] between the learning predictor's subjective probability distributions before and after new observations - the relative entropy between its prior and posterior.

In 2005 Baldi and Itti called this approach "Bayesian surprise" and demonstrated experimentally that it explains certain patterns of human visual attention better than certain previous approaches [32].

Note that the concepts of Huffman coding [28] and relative entropy between prior and posterior immediately translate into a measure of learning progress reflecting the number of saved bits—a measure of improved data compression.

Note also, however, that the naive probabilistic approach to data compression is unable to discover more general types of *algorithmic* compressibility [106, 34, 37, 73]. For example, the decimal expansion of π looks random and incompressible but isn't: there is a very short algorithm computing all of π, yet any finite sequence of digits will occur in π's expansion as frequently as expected if π were truly random, that is, no simple statistical learner will outperform random guessing at predicting the next digit

from a limited time window of previous digits. More general *program* search techniques (e.g., [36, 75, 15, 46]) are necessary to extract the underlying algorithmic regularity.

3.4 Zero Sum Reward Games for Compression Progress Revealed by Algorithmic Experiments (1997)

More recent work [68, 72] (1997) greatly increased the computational power of controller and predictor by implementing them as co-evolving, symmetric, opposing modules consisting of self-modifying probabilistic programs [97, 98] written in a universal programming language [18, 111]. The internal storage for temporary computational results of the programs was viewed as part of the changing environment. Each module could suggest experiments in the form of probabilistic algorithms to be executed, and make confident predictions about their effects by betting on their outcomes, where the *'betting money'* essentially played the role of the intrinsic reward. The opposing module could reject or accept the bet in a zero-sum game by making a contrary prediction. In case of acceptance, the winner was determined by executing the algorithmic experiment and checking its outcome; the money was eventually transferred from the surprised loser to the confirmed winner. Both modules tried to maximize their money using a rather general RL algorithm designed for complex stochastic policies [97, 98] (alternative RL algorithms could be plugged in as well). Thus both modules were motivated to discover *truly novel* algorithmic regularity / compressibility, where the subjective baseline for novelty was given by what the opponent already knew about the world's repetitive regularities.

The method can be viewed as system identification through co-evolution of computable models and tests. In 2005 a similar co-evolutionary approach based on less general models and tests was implemented by Bongard and Lipson [11].

3.5 Improving Real Reward Intake

Our references above demonstrated experimentally that the presence of intrinsic reward or curiosity reward actually can speed up the collection of *external* reward.

3.6 Other Implementations

Recently several researchers also implemented variants or approximations of the curiosity framework. Singh and Barto and coworkers focused on implementations within the option framework of RL [5, 104], directly using prediction errors as curiosity rewards [57, 58, 61] —they actually were the ones who coined the expressions *intrinsic reward* and *intrinsically motivated* RL. Additional implementations were presented at the 2005 AAAI Spring Symposium on Developmental Robotics [9]; compare the Connection Science Special Issue [10].

4 Visual Illustrations of Subjective Beauty and Its *First Derivative* Interestingness

As mentioned above (Section 3.3), the probabilistic variant of our theory [108] (1995) was able to explain certain shifts of human visual attention [32] (2005). But we can also

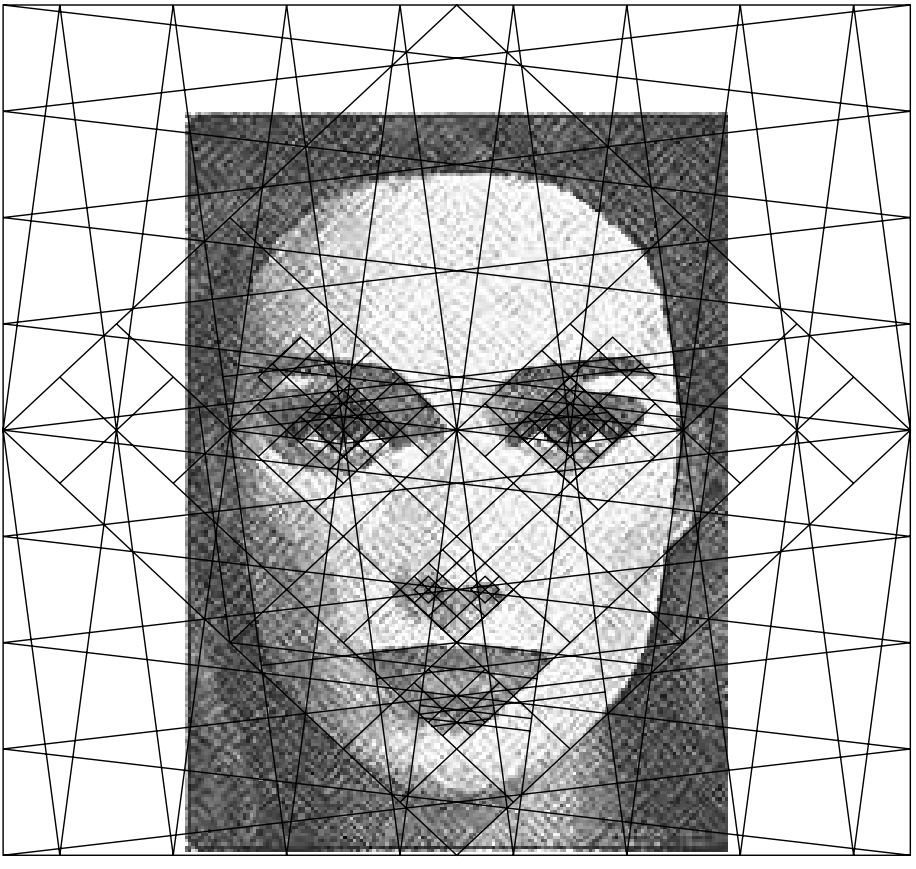


Fig. 1. Previously published construction plan [69, 88] of a female face (1998). Some human observers report they feel this face is 'beautiful.' Although the drawing has lots of noisy details (texture etc) without an obvious short description, positions and shapes of the basic facial features are compactly encodable through a very simple geometrical scheme, simpler and much more precise than ancient facial proportion studies by Leonardo da Vinci and Albrecht Dürer. Hence the image contains a highly compressible algorithmic regularity or pattern describable by few bits of information. An observer can perceive it through a sequence of attentive eye movements or saccades, and consciously or subconsciously discover the compressibility of the incoming data stream. How was the picture made? First the sides of a square were partitioned into 2^4 equal intervals. Certain interval boundaries were connected to obtain three rotated, superimposed grids based on lines with slopes ± 1 or $\pm 1/2^3$ or $\pm 2^3/1$. Higher-resolution details of the grids were obtained by iteratively selecting two previously generated, neighboring, parallel lines and inserting a new one equidistant to both. Finally the grids were vertically compressed by a factor of $1 - 2^{-4}$. The resulting lines and their intersections define essential boundaries and shapes of eyebrows, eyes, lid shades, mouth, nose, and facial frame in a simple way that is obvious from the construction plan. Although this plan is simple in hindsight, it was hard to find: hundreds of my previous attempts at discovering such precise matches between simple geometries and pretty faces failed.

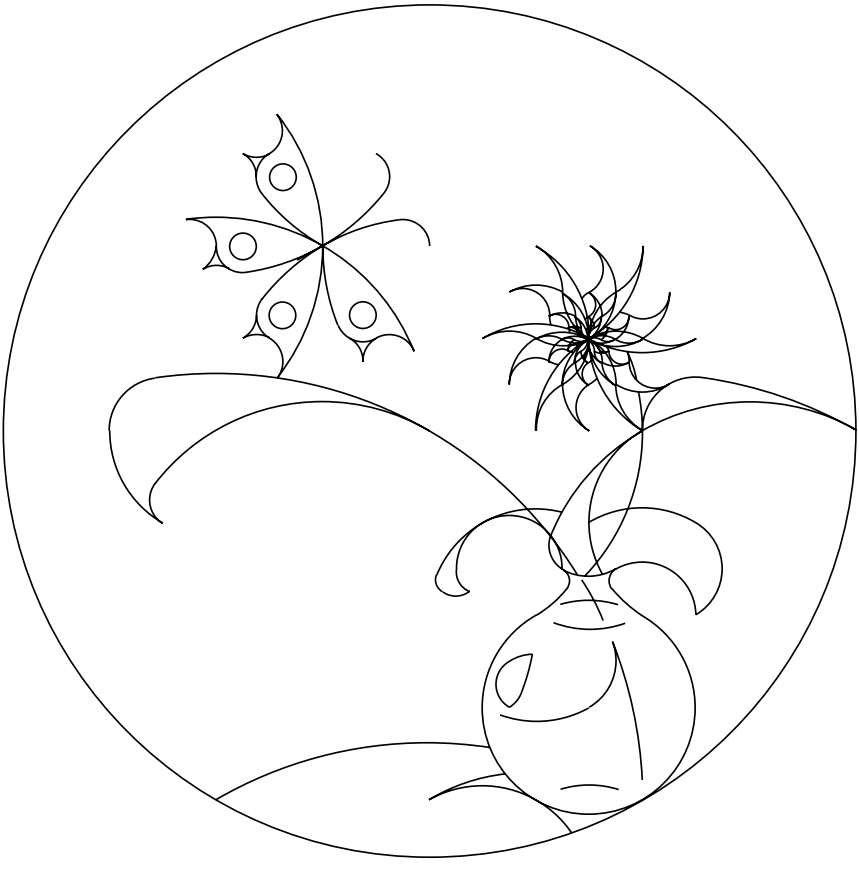


Fig. 2. Image of a butterfly and a vase with a flower, reprinted from *Leonardo* [67, 81]. An explanation of how the image was constructed and why it has a very short description is given in Figure 3.

apply our approach to the complementary problem of *constructing* images that contain quickly learnable regularities, arguing again that there is no fundamental difference between the motivation of creative artists and passive observers of visual art (Section 2.14). Both create action sequences yielding interesting inputs, where interestingness is a measure of learning progress, for example, based on the relative entropy between prior and posterior (Section 3.3), or the saved number of bits needed to encode the data (Section 1), or something similar (Section 3).

Here we provide examples of subjective beauty tailored to human observers, and illustrate the learning process leading from less to more subjective beauty. Due to the nature of the present written medium, we have to use visual examples instead of acoustic or tactile ones. Our examples are intended to support the hypothesis that unsupervised *attention* and the *creativity* of artists, dancers, musicians, pure mathematicians are just by-products of their compression progress drives.

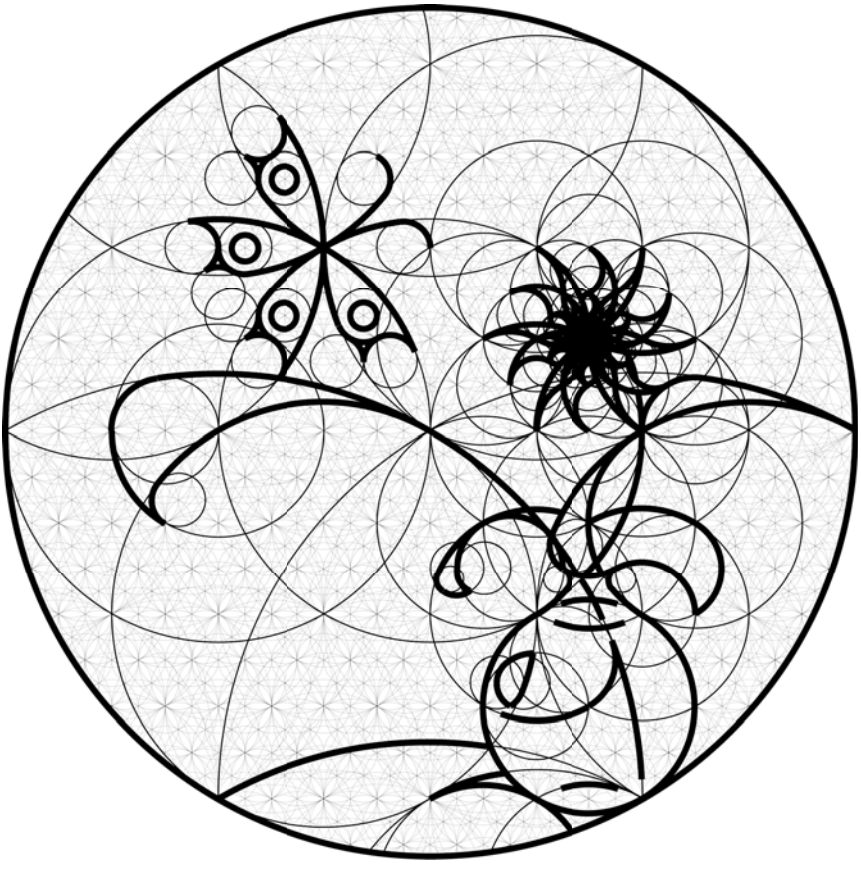


Fig. 3. Explanation of how Figure 2 was constructed through a very simple algorithm exploiting fractal circles [67]. The frame is a circle; its leftmost point is the center of another circle of the same size. Wherever two circles of equal size touch or intersect are centers of two more circles with equal and half size, respectively. Each line of the drawing is a segment of some circle, its endpoints are where circles touch or intersect. There are few big circles and many small ones. In general, the smaller a circle, the more bits are needed to specify it. The drawing is simple (compressible) as it is based on few, rather large circles. Many human observers report that they derive a certain amount of pleasure from discovering this simplicity. The observer's learning process causes a reduction of the subjective complexity of the data, yielding a temporarily high derivative of subjective beauty: a temporarily steep learning curve. (Again I needed a long time to discover a satisfactory and rewarding way of using fractal circles to create a reasonable drawing.)

4.1 A Pretty Simple Face with a Short Algorithmic Description

Figure 1 depicts the construction plan of a female face considered *'beautiful'* by some human observers. It also shows that the essential features of this face follow a very simple geometrical pattern [69] that can be specified by very few bits of information.

That is, the data stream generated by observing the image (say, through a sequence of eye saccades) is more compressible than it would be in the absence of such regularities. Although few people are able to immediately see how the drawing was made in absence of its superimposed grid-based explanation, most do notice that the facial features somehow fit together and exhibit some sort of regularity. According to our postulate, the observer's reward is generated by the conscious or subconscious discovery of this compressibility. The face remains interesting until its observation does not reveal any additional previously unknown regularities. Then it becomes boring even in the eyes of those who think it is beautiful—as has been pointed out repeatedly above, beauty and interestingness are two different things.

4.2 Another Drawing That Can Be Encoded By Very Few Bits

Figure 2 provides another example: a butterfly and a vase with a flower. It can be specified by very few bits of information as it can be constructed through a very simple procedure or algorithm based on fractal circle patterns [67]—see Figure 3. People who understand this algorithm tend to appreciate the drawing more than those who do not. They realize how simple it is. This is not an immediate, all-or-nothing, binary process though. Since the typical human visual system has a lot of experience with circles, most people quickly notice that the curves somehow fit together in a regular way. But few are able to immediately state the precise geometric principles underlying the drawing [81]. This pattern, however, is learnable from Figure 3. The conscious or subconscious discovery process leading from a longer to a shorter description of the data, or from less to more compression, or from less to more subjectively perceived beauty, yields reward depending on the first derivative of subjective beauty, that is, the steepness of the learning curve.

5 Conclusion and Outlook

We pointed out that a surprisingly simple algorithmic principle based on the notions of data compression and data compression *progress* informally explains fundamental aspects of attention, novelty, surprise, interestingness, curiosity, creativity, subjective beauty, jokes, and science & art in general. The crucial ingredients of the corresponding *formal* framework are (1) a continually improving predictor or compressor of the continually growing data history, (2) a computable measure of the compressor's progress (to calculate intrinsic rewards), (3) a reward optimizer or reinforcement learner translating rewards into action sequences expected to maximize future reward. To improve our previous implementations of these ingredients (Section 3), we will (1) study better adaptive compressors, in particular, recent, novel RNNs [94] and other general but practically feasible methods for making predictions [75]; (2) investigate under which conditions learning progress measures can be computed both accurately and efficiently, without frequent expensive compressor performance evaluations on the entire history so far; (3) study the applicability of recent improved RL techniques in the fields of policy gradients [110,119,118,56,100,117], artificial evolution [43,20,21,19,22,23,24], and others [71,75].

Apart from building improved *artificial* curious agents, we can test the predictions of our theory in psychological investigations of *human* behavior, extending previous studies in this vein [32] and going beyond anecdotal evidence mentioned above. It should be easy to devise controlled experiments where test subjects must anticipate initially unknown but causally connected event sequences exhibiting more or less complex, learnable patterns or regularities. The subjects will be asked to quantify their intrinsic rewards in response to their improved predictions. Is the reward indeed strongest when the predictions are improving most rapidly? Does the intrinsic reward indeed vanish as the predictions become perfect or do not improve any more?

Finally, how to test our predictions through studies in neuroscience? Currently we hardly understand the human neural machinery. But it is well-known that certain neurons seem to predict others, and brain scans show how certain brain areas light up in response to reward. Therefore the psychological experiments suggested above should be accompanied by neurophysiological studies to localize the origins of intrinsic rewards, possibly linking them to improvements of neural predictors.

Success in this endeavor would provide additional motivation to implement our principle on robots.

A Appendix

This appendix is based in part on references [81,88].

The world can be explained to a degree by compressing it. Discoveries correspond to large data compression improvements (found by the given, application-dependent compressor improvement algorithm). How to build an adaptive agent that not only tries to achieve externally given rewards but also to discover, in an unsupervised and experiment-based fashion, explainable and compressible data? (The explanations gained through explorative behavior may eventually help to solve teacher-given tasks.)

Let us formally consider a learning agent whose single life consists of discrete cycles or time steps $t = 1, 2, \ldots, T$. Its complete lifetime T may or may not be known in advance. In what follows, the value of any time-varying variable Q at time t ($1 \leq t \leq T$) will be denoted by $Q(t)$, the ordered sequence of values $Q(1), \ldots, Q(t)$ by $Q(\leq t)$, and the (possibly empty) sequence $Q(1), \ldots, Q(t-1)$ by $Q(< t)$. At any given t the agent receives a real-valued input $x(t)$ from the environment and executes a real-valued action $y(t)$ which may affect future inputs. At times $t < T$ its goal is to maximize future success or *utility*

$$u(t) = E_\mu \left[\sum_{\tau=t+1}^{T} r(\tau) \;\middle|\; h(\leq t) \right], \tag{2}$$

where $r(t)$ is an additional real-valued reward input at time t, $h(t)$ the ordered triple $[x(t), y(t), r(t)]$ (hence $h(\leq t)$ is the known history up to t), and $E_\mu(\cdot \mid \cdot)$ denotes the conditional expectation operator with respect to some possibly unknown distribution μ from a set $\mathcal{M}$ of possible distributions. Here $\mathcal{M}$ reflects whatever is known about the possibly probabilistic reactions of the environment. For example, $\mathcal{M}$ may contain all computable distributions [106, 107, 37, 29]. There is just one life, no need for predefined repeatable trials, no restriction to Markovian interfaces between sensors and

environment, and the utility function implicitly takes into account the expected remaining lifespan $E_\mu(T \mid h(\leq t))$ and thus the possibility to extend it through appropriate actions [79,82,80,92].

Recent work has led to the first learning machines that are universal and optimal in various very general senses [29, 79, 82]. As mentioned in the introduction, such machines can in principle find out by themselves whether curiosity and world model construction are useful or useless in a given environment, and learn to behave accordingly. The present appendix, however, will assume *a priori* that compression / explanation of the history is good and should be done; here we shall not worry about the possibility that curiosity can be harmful and "kill the cat." Towards this end, in the spirit of our previous work since 1990 [57,58,61,59,60,108,68,72,76,81,88,87,89] we split the reward signal $r(t)$ into two scalar real-valued components: $r(t) = g(r_{ext}(t), r_{int}(t))$, where g maps pairs of real values to real values, e.g., $g(a, b) = a + b$. Here $r_{ext}(t)$ denotes traditional *external* reward provided by the environment, such as negative reward in response to bumping against a wall, or positive reward in response to reaching some teacher-given goal state. But for the purposes of this paper we are especially interested in $r_{int}(t)$, the internal or intrinsic or *curiosity* reward, which is provided whenever the data compressor / internal world model of the agent improves in some measurable sense. Our initial focus will be on the case $r_{ext}(t) = 0$ for all valid t. The basic principle is essentially the one we published before in various variants [57,58,61,59,60,108,68,72,76,81,88,87]:

Principle 1. *Generate curiosity reward for the controller in response to improvements of the predictor or history compressor.*

So we conceptually separate the goal (explaining / compressing the history) from the means of achieving the goal. Once the goal is formally specified in terms of an algorithm for computing curiosity rewards, let the controller's reinforcement learning (RL) mechanism figure out how to translate such rewards into action sequences that allow the given compressor improvement algorithm to find and exploit previously unknown types of compressibility.

A.1 Predictors vs. Compressors

Much of our previous work on artificial curiosity was prediction-oriented, e. g., [57, 58,61,59,60,108,68,72,76]. Prediction and compression are closely related though. A predictor that correctly predicts many $x(\tau)$, given history $h(< \tau)$, for $1 \leq \tau \leq t$, can be used to encode $h(\leq t)$ compactly. Given the predictor, only the wrongly predicted $x(\tau)$ plus information about the corresponding time steps τ are necessary to reconstruct history $h(\leq t)$, e.g., [63]. Similarly, a predictor that learns a probability distribution of the possible next events, given previous events, can be used to efficiently encode observations with high (respectively low) predicted probability by few (respectively many) bits [28, 95], thus achieving a compressed history representation. Generally speaking, we may view the predictor as the essential part of a program p that re-computes $h(\leq t)$. If this program is short in comparison to the raw data $h(\leq t)$, then $h(\leq t)$ is regular or non-random [106,34,37,73], presumably reflecting essential environmental laws. Then p may also be highly useful for predicting future, yet unseen $x(\tau)$ for $\tau > t$.

It should be mentioned, however, that the compressor-oriented approach to prediction based on the principle of Minimum Description Length (MDL) [34,112,113,54,37] does not necessarily converge to the correct predictions as quickly as Solomonoff's universal inductive inference [106,107,37], although both approaches converge in the limit under general conditions [52].

A.2 Which Predictor or History Compressor?

The complexity of evaluating some compressor p on history $h(\leq t)$ depends on both p and its performance measure C. Let us first focus on the former. Given t, one of the simplest p will just use a linear mapping to predict $x(t+1)$ from $x(t)$ and $y(t+1)$. More complex p such as adaptive recurrent neural networks (RNN) [115,120,55,62,47,26,93,77,78] will use a nonlinear mapping and possibly the entire history $h(\leq t)$ as a basis for the predictions. In fact, the first work on artificial curiosity [61] focused on online learning RNN of this type. A theoretically optimal predictor would be Solomonoff's above-mentioned universal induction scheme [106,107,37].

A.3 Compressor Performance Measures

At any time t ($1 \leq t < T$), given some compressor program p able to compress history $h(\leq t)$, let $C(p, h(\leq t))$ denote p's compression performance on $h(\leq t)$. An appropriate performance measure would be

$$C_l(p, h(\leq t)) = l(p), \tag{3}$$

where $l(p)$ denotes the length of p, measured in number of bits: the shorter p, the more algorithmic regularity and compressibility and predictability and lawfulness in the observations so far. The ultimate limit for $C_l(p, h(\leq t))$ would be $K^*(h(\leq t))$, a variant of the Kolmogorov complexity of $h(\leq t)$, namely, the length of the shortest program (for the given hardware) that computes an output starting with $h(\leq t)$ [106,34,37,73].

A.4 Compressor Performance Measures Taking Time into Account

$C_l(p, h(\leq t))$ does not take into account the time $\tau(p, h(\leq t))$ spent by p on computing $h(\leq t)$. An alternative performance measure inspired by concepts of optimal universal search [36,75] is

$$C_{l\tau}(p, h(\leq t)) = l(p) + \log\ \tau(p, h(\leq t)). \tag{4}$$

Here compression by one bit is worth as much as runtime reduction by a factor of $\frac{1}{2}$. From an asymptotic optimality-oriented point of view this is one of the best ways of trading off storage and computation time [36,75].

A.5 Measures of Compressor Progress / Learning Progress

The previous sections only discussed measures of compressor performance, but not of performance *improvement,* which is the essential issue in our curiosity-oriented context. To repeat the point made above: *The important thing are the improvements of the compressor, not its compression performance per se.* Our curiosity reward in response

to the compressor's progress (due to some application-dependent compressor improvement algorithm) between times t and $t+1$ should be

$$r_{int}(t+1) = f[C(p(t), h(\leq t+1)), C(p(t+1), h(\leq t+1))], \tag{5}$$

where f maps pairs of real values to real values. Various alternative progress measures are possible; most obvious is $f(a,b) = a-b$. This corresponds to a discrete time version of maximizing the first derivative of subjective data compressibility.

Note that both the old and the new compressor have to be tested on the same data, namely, the history so far.

A.6 Asynchronous Framework for Creating Curiosity Reward

Let $p(t)$ denote the agent's current compressor program at time t, $s(t)$ its current controller, and do:

Controller: At any time t ($1 \leq t < T$) do:

1. Let $s(t)$ use (parts of) history $h(\leq t)$ to select and execute $y(t+1)$.
2. Observe $x(t+1)$.
3. Check if there is non-zero curiosity reward $r_{int}(t+1)$ provided by the separate, asynchronously running compressor improvement algorithm (see below). If not, set $r_{int}(t+1) = 0$.
4. Let the controller's reinforcement learning (RL) algorithm use $h(\leq t+1)$ including $r_{int}(t+1)$ (and possibly also the latest available compressed version of the observed data—see below) to obtain a new controller $s(t+1)$, in line with objective (2).

Compressor: Set p_{new} equal to the initial data compressor. Starting at time 1, repeat forever until interrupted by death at time T:

1. Set $p_{old} = p_{new}$; get current time step t and set $h_{old} = h(\leq t)$.
2. Evaluate p_{old} on h_{old}, to obtain $C(p_{old}, h_{old})$ (Section A.3). This may take many time steps.
3. Let some (application-dependent) compressor improvement algorithm (such as a learning algorithm for an adaptive neural network predictor) use h_{old} to obtain a hopefully better compressor p_{new} (such as a neural net with the same size but improved prediction capability and therefore improved compression performance [95]). Although this may take many time steps (and could be partially performed during "sleep"), p_{new} may not be optimal, due to limitations of the learning algorithm, e.g., local maxima.
4. Evaluate p_{new} on h_{old}, to obtain $C(p_{new}, h_{old})$. This may take many time steps.
5. Get current time step τ and generate curiosity reward

$$r_{int}(\tau) = f[C(p_{old}, h_{old}), C(p_{new}, h_{old})], \tag{6}$$

e.g., $f(a,b) = a - b$; see Section A.5.

Obviously this asynchronuous scheme may cause long temporal delays between controller actions and corresponding curiosity rewards. This may impose a heavy burden on the controller's RL algorithm whose task is to assign credit to past actions (to inform the controller about beginnings of compressor evaluation processes etc., we may augment its input by unique representations of such events). Nevertheless, there are RL algorithms for this purpose which are theoretically optimal in various senses, to be discussed next.

A.7 Optimal Curiosity and Creativity and Focus of Attention

Our chosen compressor class typically will have certain computational limitations. In the absence of any external rewards, we may define *optimal pure curiosity behavior* relative to these limitations: At time t this behavior would select the action that maximizes

$$u(t) = E_\mu \left[\sum_{\tau=t+1}^{T} r_{int}(\tau) \;\middle|\; h(\leq t) \right]. \tag{7}$$

Since the true, world-governing probability distribution μ is unknown, the resulting task of the controller's RL algorithm may be a formidable one. As the system is revisiting previously incompressible parts of the environment, some of those will tend to become more subjectively compressible, and the corresponding curiosity rewards will decrease over time. A good RL algorithm must somehow detect and then *predict* this decrease, and act accordingly. Traditional RL algorithms [33], however, do not provide any theoretical guarantee of optimality for such situations. (This is not to say though that sub-optimal RL methods may not lead to success in certain applications; experimental studies might lead to interesting insights.)

Let us first make the natural assumption that the compressor is not super-complex such as Kolmogorov's, that is, its output and $r_{int}(t)$ are computable for all t. Is there a best possible RL algorithm that comes as close as any other to maximizing objective (7)? Indeed, there is. Its drawback, however, is that it is not computable in finite time. Nevertheless, it serves as a reference point for defining what is achievable at best.

A.8 Optimal but Incomputable Action Selector

There is an optimal way of selecting actions which makes use of Solomonoff's theoretically optimal universal predictors and their Bayesian learning algorithms [106, 107, 37, 29, 30]. The latter only assume that the reactions of the environment are sampled from an unknown probability distribution μ contained in a set $\mathcal{M}$ of all enumerable distributions—compare text after equation (2). More precisely, given an observation sequence $q(\leq t)$ we want to use the Bayes formula to predict the probability of the next possible $q(t+1)$. Our only assumption is that there exists a computer program that can take any $q(\leq t)$ as an input and compute its *a priori* probability according to the μ prior. In general we do not know this program, hence we predict using a mixture prior instead:

$$\xi(q(\leq t)) = \sum_i w_i \mu_i(q(\leq t)), \tag{8}$$

a weighted sum of *all* distributions $\mu_i \in \mathcal{M}$, $i = 1, 2, \ldots$, where the sum of the constant positive weights satisfies $\sum_i w_i \leq 1$. This is indeed the best one can possibly do, in a very general sense [107, 29]. The drawback of the scheme is its incomputability, since $\mathcal{M}$ contains infinitely many distributions. We may increase the theoretical power of the scheme by augmenting $\mathcal{M}$ by certain non-enumerable but limit-computable distributions [73], or restrict it such that it becomes computable, e.g., by assuming the world is computed by some unknown but deterministic computer program sampled from the Speed Prior [74] which assigns low probability to environments that are hard to compute by any method.

Once we have such an optimal predictor, we can extend it by formally including the effects of executed actions to define an optimal action selector maximizing future expected reward. At any time t, Hutter's theoretically optimal (yet uncomputable) RL algorithm AIXI [29] uses an extended version of Solomonoff's prediction scheme to select those action sequences that promise maximal future reward up to some horizon T, given the current data $h(\leq t)$. That is, in cycle $t + 1$, AIXI selects as its next action the first action of an action sequence maximizing ξ-predicted reward up to the given horizon, appropriately generalizing eq. (8). AIXI uses observations optimally [29]: the Bayes-optimal policy p^ξ based on the mixture ξ is self-optimizing in the sense that its average utility value converges asymptotically for all $\mu \in \mathcal{M}$ to the optimal value achieved by the Bayes-optimal policy p^μ which knows μ in advance. The necessary and sufficient condition is that $\mathcal{M}$ admits self-optimizing policies. The policy p^ξ is also Pareto-optimal in the sense that there is no other policy yielding higher or equal value in *all* environments $\nu \in \mathcal{M}$ and a strictly higher value in at least one [29].

A.9 A Computable Selector of Provably Optimal Actions

AIXI above needs unlimited computation time. Its computable variant AIXI*(t,l)* [29] has asymptotically optimal runtime but may suffer from a huge constant slowdown. To take the consumed computation time into account in a general, optimal way, we may use the recent Gödel machines [79, 82, 80, 92] instead. They represent the first class of mathematically rigorous, fully self-referential, self-improving, general, optimally efficient problem solvers. They are also applicable to the problem embodied by objective (7).

The initial software $\mathcal{S}$ of such a Gödel machine contains an initial problem solver, e.g., some typically sub-optimal method [33]. It also contains an asymptotically optimal initial proof searcher based on an online variant of Levin's *Universal Search* [36], which is used to run and test *proof techniques*. Proof techniques are programs written in a universal language implemented on the Gödel machine within $\mathcal{S}$. They are in principle able to compute proofs concerning the system's own future performance, based on an axiomatic system $\mathcal{A}$ encoded in $\mathcal{S}$. $\mathcal{A}$ describes the formal *utility* function, in our case eq. (7), the hardware properties, axioms of arithmetic and probability theory and data manipulation etc, and $\mathcal{S}$ itself, which is possible without introducing circularity [92].

Inspired by Kurt Gödel's celebrated self-referential formulas (1931), the Gödel machine rewrites any part of its own code (including the proof searcher) through a self-generated executable program as soon as its *Universal Search* variant has found a proof that the rewrite is *useful* according to objective (7). According to the Global Optimality Theorem [79, 82, 80, 92], such a self-rewrite is globally optimal—no local maxima

possible!—since the self-referential code first had to prove that it is not useful to continue the search for alternative self-rewrites.

If there is no provably useful optimal way of rewriting $\mathcal{S}$ at all, then humans will not find one either. But if there is one, then $\mathcal{S}$ itself can find and exploit it. Unlike the previous *non*-self-referential methods based on hardwired proof searchers [29], Gödel machines not only boast an optimal *order* of complexity but can optimally reduce (through self-changes) any slowdowns hidden by the $O()$-notation, provided the utility of such speed-ups is provable. Compare [83, 86, 85].

A.10 Non-universal But Still General and Practical RL Algorithms

Recently there has been substantial progress in RL algorithms that are not quite as universal as those above, but nevertheless capable of learning very general, program-like behavior. In particular, evolutionary methods [53, 99, 27] can be used for training Recurrent Neural Networks (RNN), which are general computers. Many approaches to evolving RNN have been proposed [40, 122, 121, 45, 39, 103, 42]. One particularly effective family of methods uses cooperative coevolution to search the space of network components (*neurons* or individual *synapses*) instead of complete networks. The components are *coevolved* by combining them into networks, and selecting those for reproduction that participated in the best performing networks [43, 20, 21, 19, 22, 24]. Other recent RL techniques for RNN are based on the concept of policy gradients [110, 119, 118, 56, 100, 117]. It will be of interest to evaluate variants of such control learning algorithms within the curiosity reward framework.

Acknowledgments

Thanks to Marcus Hutter, Andy Barto, Jonathan Lansey, Julian Togelius, Faustino J. Gomez, Giovanni Pezzulo, Gianluca Baldassarre, Martin Butz, for useful comments that helped to improve the first version of this paper.

References

1. Aleksander, I.: The World in My Mind, My Mind In The World: Key Mechanisms of Consciousness in Humans, Animals and Machines. Imprint Academic (2005)
2. Baars, B., Gage, N.M.: Cognition, Brain and Consciousness: An Introduction to Cognitive Neuroscience. Elsevier/Academic Press (2007)
3. Balter, M.: Seeking the key to music. Science 306, 1120–1122 (2004)
4. Barlow, H.B., Kaushal, T.P., Mitchison, G.J.: Finding minimum entropy codes. Neural Computation 1(3), 412–423 (1989)
5. Barto, A.G., Singh, S., Chentanez, N.: Intrinsically motivated learning of hierarchical collections of skills. In: Proceedings of International Conference on Developmental Learning (ICDL). MIT Press, Cambridge (2004)
6. Bense, M.: Einführung in die informationstheoretische Ästhetik. Grundlegung und Anwendung in der Texttheorie (Introduction to information-theoretical aesthetics. Foundation and application to text theory). Rowohlt Taschenbuch Verlag (1969)
7. Birkhoff, G.D.: Aesthetic Measure. Harvard University Press, Cambridge (1933)

8. Bishop, C.M.: Neural networks for pattern recognition. Oxford University Press, Oxford (1995)
9. Blank, D., Meeden, L.: Developmental Robotics AAAI Spring Symposium, Stanford, CA (2005), http://cs.brynmawr.edu/DevRob05/schedule/
10. Blank, D., Meeden, L.: Introduction to the special issue on developmental robotics. Connection Science 18(2) (2006)
11. Bongard, J.C., Lipson, H.: Nonlinear system identification using coevolution of models and tests. IEEE Transactions on Evolutionary Computation 9(4) (2005)
12. Butz, M.V.: How and why the brain lays the foundations for a conscious self. Constructivist Foundations 4(1), 1–14 (2008)
13. Cañamero, L.D.: Designing emotions for activity selection in autonomous agents. In: Trappl, R., Petta, P., Payr, S. (eds.) Emotions in Humans and Artifacts, pp. 115–148. The MIT Press, Cambridge (2003)
14. Cohn, D.A.: Neural network exploration using optimal experiment design. In: Cowan, J., Tesauro, G., Alspector, J. (eds.) Advances in Neural Information Processing Systems 6, pp. 679–686. Morgan Kaufmann, San Francisco (1994)
15. Cramer, N.L.: A representation for the adaptive generation of simple sequential programs. In: Grefenstette, J.J. (ed.) Proceedings of an International Conference on Genetic Algorithms and Their Applications, Carnegie-Mellon University, July 24-26. Lawrence Erlbaum Associates, Hillsdale (1985)
16. Fedorov, V.V.: Theory of optimal experiments. Academic Press, London (1972)
17. Galton, F.: Composite portraits made by combining those of many different persons into a single figure. Nature 18(9), 97–100 (1878)
18. Gödel, K.: Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I. Monatshefte für Mathematik und Physik 38, 173–198 (1931)
19. Gomez, F.J.: Robust Nonlinear Control through Neuroevolution. Ph.D thesis, Department of Computer Sciences, University of Texas at Austin (2003)
20. Gomez, F.J., Miikkulainen, R.: Incremental evolution of complex general behavior. Adaptive Behavior 5, 317–342 (1997)
21. Gomez, F.J., Miikkulainen, R.: Solving non-Markovian control tasks with neuroevolution. In: Proc. IJCAI 1999, Denver, CO. Morgan Kaufmann, San Francisco (1999)
22. Gomez, F.J., Miikkulainen, R.: Active guidance for a finless rocket using neuroevolution. In: Proc. GECCO 2003, Chicago (2003); Winner of Best Paper Award in Real World Applications. Gomez is working at IDSIA on a CSEM grant to Schmidhuber, J.
23. Gomez, F.J., Schmidhuber, J.: Co-evolving recurrent neurons learn deep memory POMDPs. In: Proc. of the 2005 conference on genetic and evolutionary computation (GECCO), Washington, D.C. ACM Press, New York (2005); Nominated for a best paper award
24. Gomez, F.J., Schmidhuber, J., Miikkulainen, R.: Efficient non-linear control through neuroevolution. Journal of Machine Learning Research JMLR 9, 937–965 (2008)
25. Haikonen, P.: The Cognitive Approach to Conscious Machines. Imprint Academic (2003)
26. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation 9(8), 1735–1780 (1997)
27. Holland, J.H.: Adaptation in Natural and Artificial Systems. University of Michigan Press, Ann Arbor (1975)
28. Huffman, D.A.: A method for construction of minimum-redundancy codes. Proceedings IRE 40, 1098–1101 (1952)
29. Hutter, M.: Universal Artificial Intelligence: Sequential Decisions based on Algorithmic Probability. Springer, Berlin (1847); On Schmidhuber's, J.: SNF grant 20-61847
30. Hutter, M.: On universal prediction and Bayesian confirmation. Theoretical Computer Science (2007)

31. Hwang, J., Choi, J., Oh, S., Marks II., R.J.: Query-based learning applied to partially trained multilayer perceptrons. IEEE Transactions on Neural Networks 2(1), 131–136 (1991)
32. Itti, L., Baldi, P.F.: Bayesian surprise attracts human attention. In: Advances in Neural Information Processing Systems 19, pp. 547–554. MIT Press, Cambridge (2005)
33. Kaelbling, L.P., Littman, M.L., Moore, A.W.: Reinforcement learning: a survey. Journal of AI research 4, 237–285 (1996)
34. Kolmogorov, A.N.: Three approaches to the quantitative definition of information. Problems of Information Transmission 1, 1–11 (1965)
35. Kullback, S.: Statistics and Information Theory. J. Wiley and Sons, New York (1959)
36. Levin, L.A.: Universal sequential search problems. Problems of Information Transmission 9(3), 265–266 (1973)
37. Li, M., Vitányi, P.M.B.: An Introduction to Kolmogorov Complexity and its Applications, 2nd edn. Springer, Heidelberg (1997)
38. MacKay, D.J.C.: Information-based objective functions for active data selection. Neural Computation 4(2), 550–604 (1992)
39. Miglino, O., Lund, H., Nolfi, S.: Evolving mobile robots in simulated and real environments. Artificial Life 2(4), 417–434 (1995)
40. Miller, G., Todd, P., Hedge, S.: Designing neural networks using genetic algorithms. In: Proceedings of the 3rd International Conference on Genetic Algorithms, pp. 379–384. Morgan Kaufmann, San Francisco (1989)
41. Moles, A.: Information Theory and Esthetic Perception. Univ. of Illinois Press (1968)
42. Moriarty, D.E., Langley, P.: Learning cooperative lane selection strategies for highways. In: Proceedings of the Fifteenth National Conference on Artificial Intelligence (AAAI 1998), Madison, WI, pp. 684–691. AAAI Press, Menlo Park (1998)
43. Moriarty, D.E., Miikkulainen, R.: Efficient reinforcement learning through symbiotic evolution. Machine Learning 22, 11–32 (1996)
44. Nake, F.: Ästhetik als Informationsverarbeitung. Springer, Heidelberg (1974)
45. Nolfi, S., Floreano, D., Miglino, O., Mondada, F.: How to evolve autonomous robots: Different approaches in evolutionary robotics. In: Brooks, R.A., Maes, P. (eds.) Fourth International Workshop on the Synthesis and Simulation of Living Systems (Artificial Life IV), pp. 190–197. MIT, Cambridge (1994)
46. Olsson, J.R.: Inductive functional programming using incremental program transformation. Artificial Intelligence 74(1), 55–83 (1995)
47. Pearlmutter, B.A.: Gradient calculations for dynamic recurrent neural networks: A survey. IEEE Transactions on Neural Networks 6(5), 1212–1228 (1995)
48. Perrett, D.I., May, K.A., Yoshikawa, S.: Facial shape and judgements of female attractiveness. Nature 368, 239–242 (1994)
49. Piaget, J.: The Child's Construction of Reality. Routledge and Kegan Paul., London (1955)
50. Pinker, S.: How the mind works. Norton, W. W. & Company, Inc. (1997)
51. Plutowski, M., Cottrell, G., White, H.: Learning Mackey-Glass from 25 examples, plus or minus 2. In: Cowan, J., Tesauro, G., Alspector, J. (eds.) Advances in Neural Information Processing Systems 6, pp. 1135–1142. Morgan Kaufmann, San Francisco (1994)
52. Poland, J., Hutter, M.: Strong asymptotic assertions for discrete MDL in regression and classification. In: Annual Machine Learning Conference of Belgium and the Netherlands (Benelearn 2005), Enschede (2005)
53. Rechenberg, I.: Evolutionsstrategie - Optimierung technischer Systeme nach Prinzipien der biologischen Evolution. In: Dissertation, 1971. Fromman-Holzboog (1973)
54. Rissanen, J.: Modeling by shortest data description. Automatica 14, 465–471 (1978)
55. Robinson, A.J., Fallside, F.: The utility driven dynamic error propagation network. Technical Report CUED/F-INFENG/TR.1, Cambridge University Engineering Department (1987)

56. Rückstieß, T., Felder, M., Schmidhuber, J.: State-Dependent Exploration for policy gradient methods. In: Daelemans, W., Goethals, B., Morik, K. (eds.) ECML PKDD 2008, Part II. LNCS(LNAI), vol. 5212, pp. 234–249. Springer, Heidelberg (2008)
57. Schmidhuber, J.: Dynamische neuronale Netze und das fundamentale raumzeitliche Lernproblem. Dissertation, Institut für Informatik, Technische Universität München (1990)
58. Schmidhuber, J.: Making the world differentiable: On using fully recurrent self-supervised neural networks for dynamic reinforcement learning and planning in non-stationary environments. Technical Report FKI-126-90, Institut für Informatik, Technische Universität München (1990)
59. Schmidhuber, J.: Adaptive curiosity and adaptive confidence. Technical Report FKI-149-91, Institut für Informatik, Technische Universität München (April 1991); See also [60]
60. Schmidhuber, J.: Curious model-building control systems. In: Proceedings of the International Joint Conference on Neural Networks, Singapore, vol. 2, pp. 1458–1463. IEEE Press, Los Alamitos (1991)
61. Schmidhuber, J.: A possibility for implementing curiosity and boredom in model-building neural controllers. In: Meyer, J.A., Wilson, S.W. (eds.) Proc. of the International Conference on Simulation of Adaptive Behavior: From Animals to Animats, pp. 222–227. MIT Press/Bradford Books (1991)
62. Schmidhuber, J.: A fixed size storage $O(n^3)$ time complexity learning algorithm for fully recurrent continually running networks. Neural Computation 4(2), 243–248 (1992)
63. Schmidhuber, J.: Learning complex, extended sequences using the principle of history compression. Neural Computation 4(2), 234–242 (1992)
64. Schmidhuber, J.: Learning factorial codes by predictability minimization. Neural Computation 4(6), 863–879 (1992)
65. Schmidhuber, J.: A computer scientist's view of life, the universe, and everything. In: Freksa, C., Jantzen, M., Valk, R. (eds.) Foundations of Computer Science. LNCS, vol. 1337, pp. 201–208. Springer, Heidelberg (1997)
66. Schmidhuber, J.: Femmes fractales (1997)
67. Schmidhuber, J.: Low-complexity art. Leonardo, Journal of the International Society for the Arts, Sciences, and Technology 30(2), 97–103 (1997)
68. Schmidhuber, J.: What's interesting? Technical Report IDSIA-35-97, IDSIA (1997), ftp://ftp.idsia.ch/pub/juergen/interest.ps.gz; extended abstract in Proc. Snowbird 1998, Utah (1998); see also [72]
69. Schmidhuber, J.: Facial beauty and fractal geometry. Technical Report TR IDSIA-28-98, IDSIA (1998), Published in the Cogprint Archive: http://cogprints.soton.ac.uk
70. Schmidhuber, J.: Algorithmic theories of everything. Technical Report IDSIA-20-00, quant-ph/0011122, IDSIA, Manno (Lugano), Switzerland, 2000. Sections 1-5: see [73]; Section 6: see [74]
71. Schmidhuber, J.: Sequential decision making based on direct search. In: Sun, R., Giles, C.L. (eds.) IJCAI-WS 1999. LNCS (LNAI), vol. 1828, p. 213. Springer, Heidelberg (2001)
72. Schmidhuber, J.: Exploring the predictable. In: Ghosh, A., Tsuitsui, S. (eds.) Advances in Evolutionary Computing, pp. 579–612. Springer, Heidelberg (2002)
73. Schmidhuber, J.: Hierarchies of generalized Kolmogorov complexities and nonenumerable universal measures computable in the limit. International Journal of Foundations of Computer Science 13(4), 587–612 (2002)
74. Schmidhuber, J.: The Speed Prior: a new simplicity measure yielding near-optimal computable predictions. In: Kivinen, J., Sloan, R.H. (eds.) COLT 2002. LNCS(LNAI), vol. 2375, pp. 216–228. Springer, Heidelberg (2002)
75. Schmidhuber, J.: Optimal ordered problem solver. Machine Learning 54, 211–254 (2004)

76. Schmidhuber, J.: Overview of artificial curiosity and active exploration, with links to publications since 1990 (2004), http://www.idsia.ch/~juergen/interest.html
77. Schmidhuber, J.: Overview of work on robot learning, with publications (2004), http://www.idsia.ch/~juergen/learningrobots.html
78. Schmidhuber, J.: RNN overview, with links to a dozen journal publications (2004), http://www.idsia.ch/~juergen/rnn.html
79. Schmidhuber, J.: Completely self-referential optimal reinforcement learners. In: Duch, W., Kacprzyk, J., Oja, E., Zadrożny, S. (eds.) ICANN 2005. LNCS, vol. 3697, pp. 223–233. Springer, Heidelberg (2005)
80. Schmidhuber, J.: Gödel machines: Towards a technical justification of consciousness. In: Kudenko, D., Kazakov, D., Alonso, E. (eds.) Adaptive Agents and Multi-Agent Systems III. LNCS, vol. 3394, pp. 1–23. Springer, Heidelberg (2005)
81. Schmidhuber, J.: Developmental robotics, optimal artificial curiosity, creativity, music, and the fine arts. Connection Science 18(2), 173–187 (2006)
82. Schmidhuber, J.: Gödel machines: Fully self-referential optimal universal self-improvers. In: Goertzel, B., Pennachin, C. (eds.) Artificial General Intelligence, pp. 199–226. Springer, Heidelberg (2006), arXiv:cs.LO/0309048
83. Schmidhuber, J.: The new AI: General & sound & relevant for physics. In: Goertzel, B., Pennachin, C. (eds.) Artificial General Intelligence, pp. 175–198. Springer, Heidelberg (2006), TR IDSIA-04-03, arXiv:cs.AI/0302012
84. Schmidhuber, J.: Randomness in physics. Nature 439(3), 392 (2006) (Correspondence)
85. Schmidhuber, J.: 2006: Celebrating 75 years of AI - history and outlook: the next 25 years. In: Lungarella, M., Iida, F., Bongard, J., Pfeifer, R. (eds.) 50 Years of Aritficial Intelligence. LNCS (LNAI), vol. 4850, pp. 29–41. Springer, Heidelberg (2007)
86. Schmidhuber, J.: New millennium AI and the convergence of history. In: Duch, W., Mandziuk, J. (eds.) Challenges to Computational Intelligence. Studies in Computational Intelligence, vol. 63, pp. 15–36. Springer, Heidelberg (2007), arXiv:cs.AI/0606081
87. Schmidhuber, J.: Simple algorithmic principles of discovery, subjective beauty, selective attention, curiosity & creativity. In: Hutter, M., Servedio, R.A., Takimoto, E. (eds.) ALT 2007. LNCS (LNAI), vol. 4754, pp. 32–33. Springer, Heidelberg (2007)
88. Schmidhuber, J.: Simple algorithmic principles of discovery, subjective beauty, selective attention, curiosity & creativity. In: Corruble, V., Takeda, M., Suzuki, E. (eds.) DS 2007. LNCS (LNAI), vol. 4755, pp. 26–38. Springer, Heidelberg (2007)
89. Schmidhuber, J.: Driven by compression progress. In: Lovrek, I., Howlett, R.J., Jain, L.C. (eds.) KES 2008, Part I. LNCS, vol. 5177, p. 11. Springer, Heidelberg (2008); Abstract of invited keynote
90. Schmidhuber, J.: Driven by compression progress: A simple principle explains essential aspects of subjective beauty, novelty, surprise, interestingness, attention, curiosity, creativity, art, science, music, jokes. In: Pezzulo, G., Butz, M.V., Sigaud, O., Baldassarre, G. (eds.) Anticipatory Behavior in Adaptive Learning Systems. LNCS (LNAI), vol. 5499, pp. 48–76. Springer, Heidelberg (2009) (in press)
91. Schmidhuber, J.: Simple algorithmic theory of subjective beauty, novelty, surprise, interestingness, attention, curiosity, creativity, art, science, music, jokes. Journal of SICE 48(1) (2009) (in press)
92. Schmidhuber, J.: Ultimate cognition à la Gödel. Cognitive Computation (2009) (in press)
93. Schmidhuber, J., Bakker, B.: NIPS, RNNaissance workshop on recurrent neural networks, Whistler, CA (2003), http://www.idsia.ch/~juergen/rnnaissance.html
94. Schmidhuber, J., Graves, A., Gomez, F.J., Fernandez, S., Hochreiter, S.: How to Learn Programs with Artificial Recurrent Neural Networks. Cambridge University Press, Cambridge (2009) (in preparation)

95. Schmidhuber, J., Heil, S.: Sequential neural text compression. IEEE Transactions on Neural Networks 7(1), 142–146 (1996)
96. Schmidhuber, J., Huber, R.: Learning to generate artificial fovea trajectories for target detection. International Journal of Neural Systems 2(1&2), 135–141 (1991)
97. Schmidhuber, J., Zhao, J., Schraudolph, N.: Reinforcement learning with self-modifying policies. In: Thrun, S., Pratt, L. (eds.) Learning to learn, pp. 293–309. Kluwer, Dordrecht (1997)
98. Schmidhuber, J., Zhao, J., Wiering, M.: Shifting inductive bias with success-story algorithm, adaptive Levin search, and incremental self-improvement. Machine Learning 28, 105–130 (1997)
99. Schwefel, H.P.: Numerische Optimierung von Computer-Modellen. Dissertation, 1974. Birkhäuser, Basel (1977)
100. Sehnke, F., Osendorfer, C., Rückstieß, T., Graves, A., Peters, J., Schmidhuber, J.: Policy gradients with parameter-based exploration for control. In: Proceedings of the International Conference on Artificial Neural Networks ICANN (2008)
101. Seth, A.K., Izhikevich, E., Reeke, G.N., Edelman, G.M.: Theories and measures of consciousness: An extended framework. Proc. Natl. Acad. Sciences USA 103, 10799–10804 (2006)
102. Shannon, C.E.: A mathematical theory of communication (parts I and II). Bell System Technical Journal XXVII, 379–423 (1948)
103. Sims, K.: Evolving virtual creatures. In: Glassner, A. (ed.) Proceedings of SIGGRAPH 1994, Computer Graphics Proceedings, Annual Conference, Orlando, Florida, July 1994, pp. 15–22. ACM SIGGRAPH, ACM Press, New York (1994) ISBN 0-89791-667-0
104. Singh, S., Barto, A.G., Chentanez, N.: Intrinsically motivated reinforcement learning. In: Advances in Neural Information Processing Systems 17 (NIPS). MIT Press, Cambridge (2005)
105. Sloman, A., Chrisley, R.L.: Virtual machines and consciousness. Journal of Consciousness Studies 10(4-5), 113–172 (2003)
106. Solomonoff, R.J.: A formal theory of inductive inference. Part I. Information and Control 7, 1–22 (1964)
107. Solomonoff, R.J.: Complexity-based induction systems. IEEE Transactions on Information Theory IT-24(5), 422–432 (1978)
108. Storck, J., Hochreiter, S., Schmidhuber, J.: Reinforcement driven information acquisition in non-deterministic environments. In: Proceedings of the International Conference on Artificial Neural Networks, Paris, vol. 2, pp. 159–164. EC2 & Cie (1995)
109. Sutton, R., Barto, A.: Reinforcement learning: An introduction. MIT Press, Cambridge (1998)
110. Sutton, R.S., McAllester, D.A., Singh, S.P., Mansour, Y.: Policy gradient methods for reinforcement learning with function approximation. In: Solla, S.A., Leen, T.K., Müller, K.-R. (eds.) Advances in Neural Information Processing Systems 12, NIPS Conference, Denver, Colorado, USA, November 29 - December 4, pp. 1057–1063. The MIT Press, Cambridge (1999)
111. Turing, A.M.: On computable numbers, with an application to the Entscheidungsproblem. Proceedings of the London Mathematical Society, Series 2(41), 230–267 (1936)
112. Wallace, C.S., Boulton, D.M.: An information theoretic measure for classification. Computer Journal 11(2), 185–194 (1968)
113. Wallace, C.S., Freeman, P.R.: Estimation and inference by compact coding. Journal of the Royal Statistical Society, Series "B" 49(3), 240–265 (1987)
114. Watkins, C.J.C.H.: Learning from Delayed Rewards. Ph.D thesis, King's College, Oxford (1989)

115. Werbos, P.J.: Generalization of backpropagation with application to a recurrent gas market model. Neural Networks 1 (1988)
116. Whitehead, S.D.: Reinforcement Learning for the adaptive control of perception and action. Ph.D thesis, University of Rochester (February 1992)
117. Wierstra, D., Schaul, T., Peters, J., Schmidhuber, J.: Fitness expectation maximization. In: Rudolph, G., Jansen, T., Lucas, S., Poloni, C., Beume, N. (eds.) PPSN 2008. LNCS, vol. 5199. Springer, Heidelberg (2008)
118. Wierstra, D., Schaul, T., Peters, J., Schmidhuber, J.: Natural evolution strategies. In: Congress of Evolutionary Computation, CEC 2008 (2008)
119. Wierstra, D., Schmidhuber, J.: Policy gradient critics. In: Kok, J.N., Koronacki, J., Lopez de Mantaras, R., Matwin, S., Mladenič, D., Skowron, A. (eds.) ECML 2007. LNCS, vol. 4701, pp. 466–477. Springer, Heidelberg (2007)
120. Williams, R.J., Zipser, D.: Gradient-based learning algorithms for recurrent networks and their computational complexity. In: Back-propagation: Theory, Architectures and Applications. Erlbaum, Hillsdale (1994)
121. Yamauchi, B.M., Beer, R.D.: Sequential behavior and learning in evolved dynamical neural networks. Adaptive Behavior 2(3), 219–246 (1994)
122. Yao, X.: A review of evolutionary artificial neural networks. International Journal of Intelligent Systems 4, 203–222 (1993)
123. Zuse, K.: Rechnender Raum. Elektronische Datenverarbeitung 8, 336–344 (1967)
124. Zuse, K.: Rechnender Raum. Friedrich Vieweg & Sohn, Braunschweig (1969); English translation: *Calculating Space,* MIT Technical Translation AZT-70-164-GEMIT, Massachusetts Institute of Technology (Proj. MAC), Cambridge, Mass. 02139 (Febuary 1970)

Steps to a Cyber-Physical Model of Networked Embodied Anticipatory Behavior

Fabio P. Bonsignorio

Heron Robots Srl, Via R.C.Ceccardi 1/18
16121 Genova, Italy
fabio.bonsignorio@heronrobots.com

Abstract. This paper proposes and discusses a modeling framework for embodied anticipatory behavior systems. This conceptual and theoretical framework is quite general and aims to be a, quite preliminary, step towards a general theory of cognitive adaptation to the environment of natural intelligent systems and to provide a possible approach to develop new more autonomous artificial systems. The main purpose of this discussion outline is to identify at least a few of the issues we have to cope with, and some of the possible methods to be used, if we aim to understand from a rigorous standpoint the dynamics of embodied adaptive learning systems both natural and artificial.

Keywords: anticipation, adaptive ,embodiment, intelligent agents, information, entropy, complexity, dynamical systems, network, emergence.

1 Introduction

According to many experimental results, [5,6,7,12,13,15], the human (and mammal) brain might be seen as a complex system which evolved mainly to control movement, in particular walking and what in the robotic domain is known as visual manipulation and grasping. To achieve that it minimizes uncertainty through Bayesian estimation, prediction of actions' consequence, controlling statistics of action effectiveness, comparing with expected outcomes and manage to smooth transition, from energy and information standpoint, from perception to action.

In the natural domain, to our knowledge, at least on our planet, the human brain is the most sophisticated cognitive machine, nevertheless the basic organizational principles are shared with more ancient living beings and are evolved on top of evolutionary earlier solutions.

There are several evidences suggesting that cognition might be an emerging adaptive (meta) process of loosely coupled networks of embodied and situated agents, [24,29]. In the natural domain the most widely used method of 'intelligence', computation and 'cognition' seems to be 'embodied' biological neural networks. Although, see [66], there are good reasons to exclude an 'intelligent design' of natural cognitive systems and although these systems have evolved not only according to 'cheap design', [22], but also 'good enough' principles, it is apparent that not only the more

G. Pezzulo et al. (Eds.): ABiALS 2008, LNAI 5499, pp. 77–94, 2009.

evolved human or mammal brains, but even the 'simple' 15-20000 neurons Aplysia nervous system shows much more robust than any current robotic application.

This justifies the search for biomimetic and bioinspired solutions for the co design of cognitive physical agent structure, processes and organizational principles.

Generally biological neural network are modeled by artificial neural networks, a simplified model of their natural counterpart. The original Rosenblatt's 'perceptron' [39], proposed in 1958, represents a neuron as a node of a graph where an output edge signal is triggered when a threshold of a sum of weighted connection values is reached. Although today most current neural network algorithms are more sophisticated as they do not use thresholds, but rather continuous valued squashing functions, they are still an approximation of their natural counterparts not considering plasticity and other characteristics of the biological neurons.

It is interesting to speculate on the system level characteristics which allow autonomous cognitive behavior in natural systems.

In the past years Pfeifer and other researchers, [22,23], have shown the importance of 'embodiment' and 'situatedness' in natural intelligent systems, 'passive walkers' are a clear example of that.

It is possible, anyhow, as it is pointed out by some researchers, that we still miss the quantitative framework to model the interplay between system dynamics and information processing in physical systems. In other terms we have a need to extend the theory of computation to the physical world, [65]. This (new) topic is called Cyber-Physical system theory. A 'cyber-physical' system is a physical system where there is a two ways relationship between its physical behavior and its control system. The study of the so called cyber-physical systems is a priority of US NSF (National Science Foundation).

In this paper we will show and discuss how anticipatory behaviors might emerge from a loosely network of embodied agents and which metrics might be used to develop such systems. The aim is to define a conceptual model capable of emulating the high level behaviors of natural intelligent and simple enough to be described in a quantitative way. We will describe such a conceptual and methodological framework aiming to be a first step towards a general theory on cognition in natural systems. We will derive a few preliminary relations and we will suggest some theoretical tools which in principle might allow to cope with these ambitious objectives.

This is one of the possible approaches to a quantitative representation of an intelligent physical system, equivalent others are possible.

In the next paragraph we will review the ABC (Anticipatory Behavioral Control) model that we believe captures at least some of the requisites that an intelligent embodied agent should have. In section 3 we will summarize the quantitative aspects of a quite general networked embodied intelligent system, following the discussions in [29,30] based on the findings in [25] and [34]. In section 4 we will highlight the requisites of networked embodied anticipatory systems and a possible system architecture coping with these needs.

Eventually it will be highlighted the work ahead and the open issues involved by such an approach.

2 The ABC Model

The observation of natural intelligent systems and the practice of robotics research and engineering lead us to think that 'intelligence' (and 'meaning' if not 'consciousness') are 'emerging' characteristics springing from the evolution of loosely coupled networks of intelligent 'embodied' and 'situated' agents, [23,24].

In robotics research this has led to develop some systems leveraging on the body dynamics, like in 'passive walker' approach exemplified by MIT biped and others, [35], and 'behavior based' control architecture, starting from the 'subsumption' architecture originally proposed by Brooks, [21].

The behavior based approach, in particular in the context of subsidiary architectures, has also proven capable of obtaining good performances, in tasks like navigation and obstacle avoidance and others, with a limited set of a priori hypotheses and programming effort.

This approach can be seen as a translation to the AI and robotics domain of the Stimulus-Reaction model of animal behavior dating back to Pavlov, [70], and quite popular for some time in behaviorist psychology.

It is also a natural application to robotics of the information processing model of cognition.

This approach shows some limits at least if we consider human psychology and mammals ethology as it does not consider the intentional behavior. It makes more sense to think that the function of cognitive processes is to enable the production of anticipated 'stimuli'.

The ABC theory (Anticipative Behavioral Control), [8,9,10], tries to go beyond those limits on the basis of theoretical considerations and experimental evidence, [8,5] coming from the investigation of animal and human associative learning processes and the impact of behavioral effects on the selection, initiation, and execution of simple voluntary acts.

It is shown that 'intentions', based on the anticipated outcomes of finalized actions, play a key role in the shaping of behaviors of natural cognitive systems and that this approach is different from both the 'behavior based' one and from the top down symbolic processing approach.

It is not surprising that in nature such kind of information structuring have evolved as in an open ended stochastic high dimensional, non linear and even fractional derivatives, environment the capability of generating 'cheap' and 'good enough' finalized actions strongly relies on the capability to generate 'reasonable' predictions at a low computational and energetic cost.

If we assume that the fit behavior generation model for an autonomous cognitive agent is given by something similar to the ABC model, it is interesting to understand how such a model of interaction might emerge from a loosely coupled network of embodied agents without a preset internal explicit representation and exploiting the body dynamics, according to the mentioned 'cheap design' principles. In particular it is interesting to speculate on a model of interaction with (or within?) the environment which makes possible a quantitative or semi quantitative description of the interaction.

Natural neural network themselves could be regarded as 'embodied' and 'situated' computing systems, as they are connected to a body.

So far we miss a quantitative comprehensive theory which allows us to model the interplay between the agents' 'morphology', in other words their mechanical structure, and the emerging of 'intelligence' and 'meaning'. Despite that some preliminary considerations are already possible.

3 Networks of Embodied Agents: Possible Models and Metrics

It is reasonable to think that in nature the biological neural networks are an adaptation of cell based organisms to the intelligent and cognitive tasks. This leaves open the question of which features of these systems are necessary and sufficient in order to achieve a robust cognitive adaptation to the environment (from the unstructured natural outdoor ones to the structured factory floors or human buildings).

If intelligence and cognition are emerging processes springing from loosely coupled networks of embodied physical agents, how can a 'fit' anticipatory behavior emerge from a system of this kind ? And which metrics is it possible to identify?

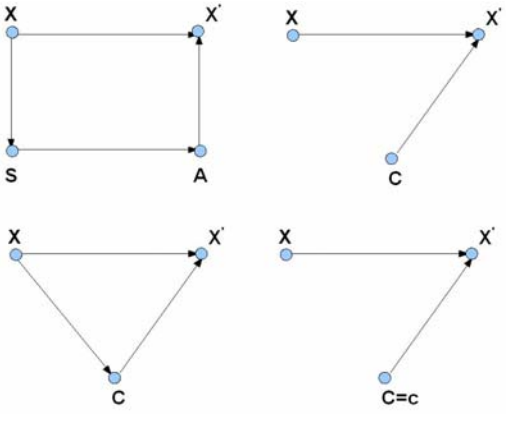


Fig. 1. Directed acyclic graphs representing a control process. (Upper left) Full control system with a sensor and an actuator. (Lower left) Shrinked Closed Loop diagram merging sensor and actuator, (Upper right) Reduced open loop diagram. (Lower right) Single actuation channel enacted by the controller's state C=c. The random variable X represents the initial state, X' the final state. Sensor is represented by state variable S and actuator is represented by state variable A.

After reviewing in this section some metrics of a network embodied system, in the following we will show how the nervous systems of natural intelligent systems might be regarded as huge networks of loosely structured 'resonators' and 'amplificators' of natural coupling processes, which actually, in simpler forms, for example in biped walking down a slope, occur without any specific and dedicated cognitive computing system.

We will first explain how, from a theoretical standpoint, a network of embodied agents can process information in the physical morphology of its agents and the

network relations between the agents of the network. This discussion is taken from [25, 29, 30]. If we see, for simplicity, an embodied agent as a controlled dynamical system, as in fig. 1, it is possible to show how the algorithmic complexity of the control program is related to its phase space 'footprint'.

$$\Delta H_{controller} \cong \Delta H_{closed} - \Delta H_{open}^{\max} \leq I(X;C) \tag{1}$$

This is possible starting from [25] where Shannon theory is applied to the modeling of controlled systems and statistical information metrics based definitions of controllability and observability are derived. In equation (1) we recall the most important result in [25] from our perspective. Equation (1) applies to a general control system. The meaning of the variables is given in Fig. 1. It links the variation of Shannon entropy in the controller to the variation of entropy in closed loop (with the controller in the loop) and the maximum variation of entropy in open loop (with feed forward control) to mutual information between the state variable and the controller state. Adding some reasonable hypotheses and exploiting the results described in [34] it is possible to derive:

$$K(X) \overset{+}{\leq} \log \frac{W_{closed}}{W_{open}^{\max}} \tag{2}$$

The equation (2) bounds the algorithmic complexity of the control program (the intelligence of the agent, in a simplified view) to the phase space volume, an estimate of the number of possible system state, of the controlled agent versus the phase space volume of the non controlled system.

From a qualitative standpoint (at least) this relations explains why a simpler walker like the MIT biped or the one described in [35] can be controlled with a 'short' program, while other walkers (like the Honda Asimo, [36], or the Sony Qrio) which don't have a limit cycle and show a larger phase space 'footprint' require more complex control systems.

The Shannon entropy related measures have been shown to be useful to quantitatively characterize sensory motor coordination, the evolution of sensory layouts and the complexity of the agent environment, [26,28,32].

In general the intelligent system is here assumed to be constituted of, and to be part of, a network of weakly coupled agents.

We assume (for simplicity) that the ‘cognitive network ‘ can be accessed by all the agents which are co evolving it and in fact share (constitute) it.

The idea that learning may actually emerge from some kind of evolutionary process was actually already proposed by Turing in a famous 1950’s paper, [57].

It must be noticed that the concept model described here is one in a large class of possible models, in particular one of most convincing is the semiotic dynamics approach, [58,60]. This idea is strongly influenced by Bateson’s concept of an ‘ecology of mind’, [69].

We assume that the model of the environment is distributed among all the agents constituting the network and depends on the (co) evolution of their interactions in time. We will see below how this can be explained and quantified on the basis of relations between some information measures.

In this perspective it is interesting to notice that in [59] the mathematical model of the collective behaviors of systems like that described in [60] are based on the theory of random acyclic graphs which is the basis of most network system physics formalizations.

In [59], the network of agents, where each word is initially represented by a subset of three or more nodes with all (possible) links present, evolves towards an equilibrium state represented by fully connected graph, with only single links.

The statistical distribution, necessary to determine the information managing capability of the network of physical agents and to link to equation (2) can be obtained from equations derived in the statistical physics of network domain.

From (2) it is possible to derive the relations recalled here below (these relations are demonstrated in the appendix).

$$K(X) \stackrel{+}{\leq} \log \frac{W_{closed}}{W_{open}^{\max}} \tag{I}$$

As told, relation (I) links the complexity ('the length') of the control program of a physical intelligent agent to the state available in closed loop and the non controlled condition. This shows the benefits of designing system structures whose 'basin of attractions' are close to the desired behaviors in the phase space.

$$\Delta HN + \sum_{i}^{n} \Delta H_i - \Delta I \leq I(X;C) \tag{II}$$

Relations (II) links the mutual information between the controlled variable and the controller to the information stored in the elements, the mutual information between them and the information stored in the network and accounts for the redundancies through the multi information term *ΔI*.

Relations (III) links the program complexity of the controller to the information stored in the elements, the mutual information between them and the information stored in the network.

$$K(X) = \Delta HN + \sum_{i}^{n} \Delta H_i - \Delta I \tag{III}$$

Relations (IV) links the program complexity of the controller to the information stored in the elements the mutual information between them and the information stored in the network.

$$\Delta HN = \log \frac{\Omega_{closed}}{\Omega_{open}^{\max}} + \Delta I \tag{IV}$$

These relations are quite preliminary, and perhaps need a more rigorous demonstration, but give an insight on how information is managed within a network of physical elements or agents interacting with a given environment in a finalized way. They suggest how the cognitive adaptation is at network level: in any environment niche it is possible with small networks of highly sophisticated individual agents, like in human societies, or with many limited autonomy individuals like in ant colonies, with a great variety of possibilities in the middle.

It is worth to observe that the relations reported above are quite general and can also be applied to a continuous intelligent material structure if you consider as physical

elements a suitable mesh of material finite elements, see [30]. In this case the information can be stored in the stress-deformation state itself.

On a different respect, these relations can be applied to the whole environment, meaning to the whole network of agents interacting among them over a specified threshold. The 'self' of the single cognitive agent might emerge by means of a process analogue to the mammal and human immune system, [55.56].

3.1 Example

A simple embodied agent is given by the oscillator given in fig. 2.

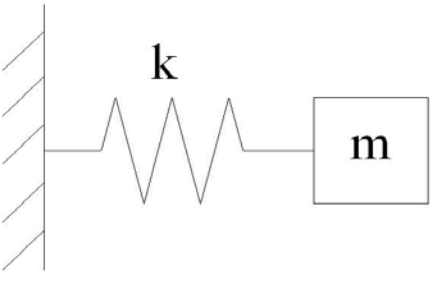


Fig. 2. A simple linear oscillator

If we apply equation (3) representing energy conservation:

$$E_{tot} = \frac{m\dot{x}^2}{2} + \frac{kx^2}{2} \tag{3}$$

We see that in phase space the system follows a closed curve. The shape of the curve depends on m and k and the initial values of x and its first derivative. If we assume an uniform distribution [0,X] for x and [0,XP] for the initial condition the phase space volume of equation (2) is given by the difference of the areas of the ellipses:

$$E_{tot}^{\max} = \frac{mXP^2}{2} + \frac{kX^2}{2} \tag{4}$$

The equation of the ellipses is:

$$\frac{m\dot{x}^2}{2E_{tot}^{\max}} + \frac{kx^2}{2E_{tot}^{\max}} = 1 \tag{5}$$

From which we derive the semi axis, a, b :

$$a = \sqrt{\frac{2E_{tot}^{\max}}{m}} \tag{6}$$

$$b = \sqrt{\frac{2E_{tot}^{\max}}{k}} \tag{7}$$

And we eventually derive the phase space volume:

$$W = \pi ab = \pi \sqrt{\frac{2E_{tot}^{\max}}{m}} \sqrt{\frac{2E_{tot}^{\max}}{k}} = \frac{2\pi}{\sqrt{km}} E_{tot}^{\max} \tag{8}$$

Assuming that the closed loop phase region is inside the open loop region, K becomes, in absolute value, close to 0 when these two area are close, while it tends (in absolute value) to infinite when we want to force the system in an phase space volume (area in this case) going to zero.

4 A possible Networked Embodied Cognitive System Model

In the previous section we have reviewed some of the metrics that a multi agent embodied intelligent system will comply to. Here we describe a conceptual model which on one side exhibit the capabilities that we think characterize the behavior of natural cognitive agents and on the other side allows the development of a a quantitative model. This quantitative model may eventually be compared to the reality and experimentally tested.

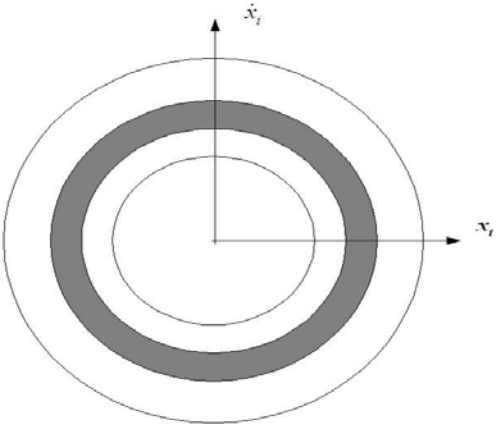


Fig. 3. Phase space portrait of the elementary oscillator

We define here a 'minimal set' networked embodied anticipatory behavior system architecture for an intelligent agent. We summarized above the requisites that an anticipatory networked embodied system should have, here we describe at functional level a possible system architecture believed to be capable of generating through self organization the required behaviors.

It seems reasonable to think that what we need is the capability to generate a wide set of coupled dynamical behaviors. Even in simpler and older cases like a moving target tracking by a rotating automated missile launch platform part of the anticipation is done by means of the inertial rotation of the platform.

A method to provide to a system a 'rich' internal dynamics is to model it as an n-dimensional set of oscillators, randomly oscillating, (modeled like in figure 3), which for simplicity can be represented as in the figure below. This set evolves into a directed acyclic graph where the links between the oscillators evolve dynamically according to homokinetics and others criteria, see [45,46,47,48]. The dynamical couplings of a chain of oscillators are described in [64].In this context 'modules' , whose 'economical' usefulness is discussed in [67], must be seen as hierarchies of basins of attraction (see genome activation schemes). Each oscillator is governed by its equations, simpler as in our example above, or more sophisticated like in [43]. The 'modules' are embodied into fractional distributed form and spring from a self organizing co evolution process extended to the environment network of relations. This constitutes an high dimensional system. Hierarchical modules are an useful way to structure data analysis as they allow to reduce uncertainty through iterated processing, [11,14,16].

In summary we connect a hierarchical modular system in the sense specified above from the sensors to a similar hierarchical modular system managing the actuators through a rich homokinetic massive loosely coupled network of chaotic self organizing oscillators. In principle a simplified version of the mathematics of Schwinger fields might be of some help here, although this has to be investigated.

As shown above in paragraph 3 the length of a control program is linked to the difference between the reachable phase space volume in open loop and the desired closed loop behaviors.

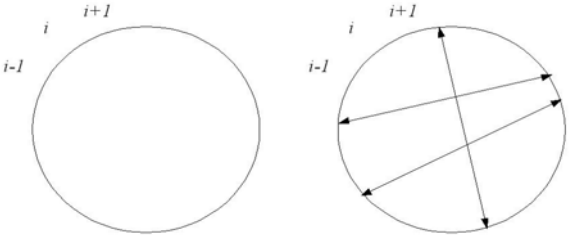


Fig. 4. Schematic representation of a network where coupling is only with the adjacent nodes, (Left). Schematic representation of a network with (weak) coupling with adjacent nodes, (Right).

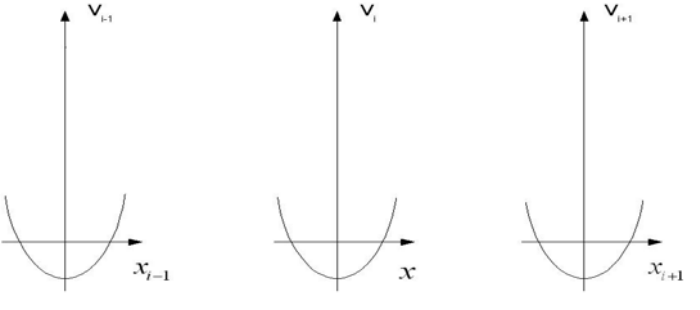


Fig. 5. Potential functions for three adjacent not coupled nodes

The Lie symmetry of the physical world, [27], as it is experienced by a mechanically extended body strongly reduces the reachable portion of the configuration and phase spaces making easier the control. The anticipation is based on the fixed point, limit cycles, attractors of the internal dynamics coupled to the 'external' environment. Consistency is guaranteed by homokinesys, energy minimization, complexity minimization criteria.

CPG will emerge naturally from a system like that described here, see [63].

A schematic representation is given in fig. 4 the links of the network allow to shape the information managed. Chaotic behaviors are induced from the outside environment noise.

There are chances that such a system might exhibit sensory substitution behaviors like those observed in the mammal and human brains and sensory motor systems, see [41,42], this has to be thoroughly investigated.

The most suitable tool to study such a system seems to be simulation as deducing closed form equations is challenging.

5 Discussion

Behavior based approaches in Robotics and AI have proven quite successful and might be considered a 'mapping ' of the S-R approach in psychology to the artificial domain. On the other end, as shown in section 2 there are hints that this might not be the best approach for the 'fit' interaction of an artificial or natural cognitive system with its environment. If we agree on that we must define an architectural framework capable to manage different anticipatory behavioral schemes.

Traditionally in GOFAI (Good Old Fashioned Artificial Intelligence), the model of the environment is explicitly mapped into the artificial system with a specifically designed symbolic structure superimposed and preimposed from the outside, by a supposedly 'omniscient' agent, the designer of the system. (and here we may observe that the knowledge of the environment of the designer is still incomplete and with an inherent probabilistic nature not necessarily capable of anticipating the real conditions with which the agent will have to cope). In control theory model based adaptive controls methods share the same inherent limitations. Predictive schemes based on stochastic identification methods like various kind of Kalman filters or less constraining polynomial observers have the advantage of doing very limited assumptions on the controlled system equations (linearity for Kalman), but lack of flexibility as the objective of the control actions must be defined in advance.

Under a certain respect, for a given physical system, the physical morphology and the natural dynamics force the possible combinations of sensor and actuator variables to a subset of all the values that in theory the system variables may assume while performing a specific task, leading to 'morphological computation', [22,23].

We need a complex adaptive system which, exploiting his embodiment and situatedness and its network relations within its environment, it is capable of interacting within its environment in a proactive and purposive way anticipating 'desired' sensor input.

A model of a typical environment should be a non linear (fractional derivatives?) stochastic many variable system exhibiting quite often itinerant chaos behavior with a

constant creation and disruption of new symmetries in a fractional (in the fractal sense) context.

Actually we should not 'carve' in advance into the agent or network of agents such model as this will in any case restrict the autonomy and behavioral flexibility of the agent and limit them to the advance knowledge of the designer, but, instead we should conceive a framework which allows the 'spontaneous' emerging of the embodied model into the agent itself.

We need a general behavioral structure generator exploiting its body dynamics and environment interaction patterns. The physics of the environment allows to limit the generality of the 'abstraction' that the system must be capable to generate.
The prediction machine will be in general given by a wide set of interleaved fractional dimension – due to the attractors' fractional dimension in the phase space - internal processes continuously evolving multiply coupled with the 'external' processes originating by the active interactions (patterns) of the agent.

There is a need to explain statistical learning as an emerging process from a network of embodied agents with their own natural dynamics, [31].

As simple as it is, the model described above in the example given in paragraph 3.1 allows to represent two essential aspects of our world: inertia (through the position second derivative and mass, and a basic (linear) force, or potential field (through the linear term in x), and energy conservation. The importance of 'time delay',i.e. phase relations, have proven to be important in the human and animal brains, [40]: they are a natural outcome of a physical oscillating system.

Thanks to equation (2) we have a substantial equivalence between the computing made by the controller and that 'embodied' into the system. While the relations recalled above, in section 3, show how the tasks can be split between the different agents. A system however implemented capable of representing these basic aspects is capable to have coupled oscillations with the external environment. Biological neurons themselves can be modeled as non linear oscillators, [23,24]. On an different respect, also groups, subnetworks and networks of artificial neurons can show oscillatory behaviors. We will see below some of the consequences we can (may) draw on the basis of these facts.

In the classical target tracking example quoted above the PID controller together with 'body morphology' and the sensors allow this coupling. In this case the coupling is possible thanks to the external off line 'design' of an intelligent cognitive embodied agent: the system engineer who designed the 'intelligent' weapon.

If dynamical coupling with 'external processes' is the basis of 'fit' interaction with the external environment, what we need is a system with a rich high dimensional dynamics, capable of establishing a wide set of multi scale recursive coupled oscillations with the environment. From what we have seen above in section 3 there is a substantial equivalence between the 'extensive' information managed by the body morphology and the 'intensive' information managed by a computer or by a biological neural network.

The nervous system function in natural intelligent system might be that of massively increasing the number of dimensions of the system phase space allowing richer internal trajectories and making possible a wider number of dynamical couplings with the exterior processes. The sensors and actuators translate from the 'extensive' dynamics of the external world.

The modeling framework discussed in this paper is not the only possible one.

For example, it has been shown that artificial neural networks may show attractors and limit cycles, so a possible alternative implementation can be by means of (a special class) neural networks. It makes sense to think that the economy of program length and power absorption are more likely in nature in emerging structures coming from the evolution optimization process.

Artificial autonomous systems have the same needs. We may think anyhow that any fit quantitative model of cognition should try to unify at deep level, information, control and non linear dynamics theory and general AI to be able to account for the behavioral complexities of what we observe in nature.

We hypothesize here that hierarchical Bayesian systems observed in natural systems might be implemented as small-world networks of non linear oscillators. A single neuron might be modeled as a chaotic non linear oscillator. From this perspective there is a continuous path from the the 'cognitive' processes in metabolic networks to the higher level behaviors in animals and humans.

The basys of information processing is seen in system dynamics: like in a dance the coupled synchronized movements of the dancers deeply rely on their body inertial dynamics and the sympathetic knowledge of the other dancer inertial dynamics and 'intentions'.

It is thought that the symmetries of the physical world must be represented and mimicked inside a cognitive system. The biological neuron networks do that in a compressed volume, with limited program complexity and reduced power consumption. This is possible thanks to the signal transduction operated by the sensor actuation systems: from and to mechanical/electromagnetic (distributed) measures to chemical electrical gradients. This gives a specific meaning to the interpretation of biological neural networks as embodied massive parallel cognitive systems.

6 Conclusions

Although the theoretical framework discussed above may show serious mathematical challenges it is thought that it exemplifies some of the features that a working quantitative general models of system of the kind we investigate and we aim to reproduce technically should have. An important characteristics of this conceptual model is the attempt to ground coordination of physical intelligent agents between them and with the environment on system dynamics and related information metrics, through the relations typical of stochastic control.

In general what we need is a high dimensional system model with a rich internal dynamics capable of evolving over time many complex adaptive internal sub dinamycs coupled with the 'external' environment dynamics.

This paper aims to suggest a methodology and to highlight a few of the challenges that the development of a working example of an embodied anticipatory cognitive system still presents.

Perhaps what we need is an integrated approach putting together concepts and methods from fields so far considered separated like non linear dynamics, information, computation and control theory as well as general AI and psychology.

A lot of work has still to be done.

References

1. Shannon, C.E.: The Mathematical Theory of Communication. Bell Sys. Tech. J. 27, 379, 623 (1948)
2. Kolmogorov, A.N.: Three approaches to the quantitative definition of information. Problems Inform. Transmission 1(1), 1–7 (1965)
3. Chaitin, G.J.: On the length of programs for computing finite binary sequences: statistical considerations. J. Assoc. Comput. Mach. 16, 145–159 (1969)
4. Wiener, N.: Cybernetics: or Control and Communication in the Animal and the Machine. MIT Press, Cambridge (1948)
5. Hommel, B.: Becoming an intentional agent: The emergence of voluntary action. In: 5th eu Cognition six montly meeting euCognition, Munchen (2008)
6. Biro, S., Hommel, B. (eds.): Becoming an intentional agent: Early development of action interpretation and action control. Special issue of Acta Psychologica (2007)
7. Biro, S., Hommel, B.: Becoming an intentional agent: Introduction to the special issue. Acta Psychologica 124, 1–7 (2007)
8. Hoffmann, J.: Anticipatory Behavioral Control. In: Butz, M.V., Sigaud, O., Gerard, P. (eds.) Anticipatory Behavior in Adaptive Learning Systems. LNCS (LNAI), vol. 2684, pp. 44–65. Springer, Heidelberg (2003)
9. Butz, M.V., Sigaud, O., Gerard, P.: Internal Models and anticipation in adaptive learning systems. In: Butz, M.V., Sigaud, O., Gérard, P. (eds.) Anticipatory Behavior in Adaptive Learning Systems. LNCS (LNAI), vol. 2684, pp. 86–109. Springer, Heidelberg (2003)
10. Pezzulo, G.: Anticipation and future oriented capabilities in natural and artificial cognition. In: Lungarella, M., Iida, F., Bongard, J., Pfeifer, R. (eds.) 50 Years of AI. Springer, Heidelberg (2007)
11. George, D., Hawkins, J.: A hierarchical Bayesian model of invariant pattern recognition in the visual cortex. In: Proceedings of the International Joint Conference on Neural Networks. IEEE, Los Alamitos (2005)
12. Van Essen, D.C., Anderson, C.H., Felleman, D.J.: Information processing in the primate visual system: an integrated systems perspective. Science 255(5043), 419–423 (1992)
13. Fukushima, K.: Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biological Cybernetics 36(4), 193–202 (1980)
14. Hawkins, J., Blakeslee, S.: On Intelligence. Times Books, Henry Holt and Company (2004)
15. Lee, T.S., Mumford, D.: Hierarchical Bayesian inference in the visual cortex. J. Opt. Soc. Am. A. Opt. Image Sci. Vis. 20(7), 1434–1448 (2003)
16. Pearl, J.: Probabilistic Reasoning in Intelligent Systems. Morgan Kaufman Publishers, San Francisco (1988)
17. Riesenhuber, M., Poggio, T.: Hierarchical models of object recognition in cortex. Nature Neuroscience 2(11), 1019–1025 (1999)
18. Stringer, S.M., Rolls, E.T.: Invariant object recognition in the visual system with novel views of 3D objects. Neural Computation 14(11), 2585–2596 (2002)
19. Bernardet, U., Bermúdez i Badia, S., Verschure, P.F.M.J.: A model for the neuronal substrate of dead reckoning and memory in arthropods: a comparative computational and behavioral study. Theory in Biosciences, 127 (2008)
20. Verschure, P.F.M.J.: Building a Cyborg: A Brain Based Architecture for Perception, Cognition and Action, Keynote talk, In: IROS 2008, Nice (2008)

21. Brooks, R.: A Robust Layered Control System for A Mobile Robot. IEEE Journal of Robotics and Automation (1986)
22. Pfeifer, R.: Cheap designs: exploiting the dynamics of the system-environment interaction. Three case studies on navigation. In: Conference on Prerational Intelligence — Phenomonology of Complexity Emerging in Systems of Agents Interacting Using Simple Rules, pp. 81–91. Center for Interdisciplinary Research, University of Bielefeld (1993)
23. Pfeifer, R., Iida, F.: Embodied artificial intelligence: Trends and challenges. In: Iida, F., Pfeifer, R., Steels, L., Kuniyoshi, Y. (eds.) Embodied Artificial Intelligence. LNCS (LNAI), vol. 3139, pp. 1–26. Springer, Heidelberg (2004)
24. Lungarella, M., Iida, F., Bongard, J., Pfeifer, R. (eds.): 50 Years of AI. Springer, Heidelberg (2007)
25. Touchette, H., Lloyd, S.: Information-theoretic approach to the study of control systems. Physica A 331, 140–172 (2003)
26. Gomez, G., Lungarella, M., Tarapore, D.: Information-theoretic approach to embodied category learning. In: Proc. of 10th Int. Conf. on Artificial Life and Robotics, pp. 332–337 (2005)
27. Philipona, D., O' Regan, J.K., Nadal, J.-P., Coenen, O.J.-M.D.: Perception of the structure of the physical world using unknown multimodal sensors and effectors. In: Advances in Neural Information Processing Systems (2004)
28. Olsson, L., Nehaiv, C.L., Polani, D.: Information Trade-Offs and the Evolution of Sensory Layouts. In: Proc. Artificial Life IX (2004)
29. Bonsignorio, F.P.: Preliminary Considerations for a Quantitative Theory of Networked Embodied Intelligence. In: Lungarella, M., Iida, F., Bongard, J., Pfeifer, R. (eds.) 50 Years of AI. Springer, Heidelberg (2007)
30. Bonsignorio, F.P.: On Some Information Metrics of Intelligent Material Systems. In: ASME ESDA 2008, Haifa (2008)
31. Burfoot, D., Lungarella, M., Kuniyoshi, Y.: Toward a Theory of Embodied Statistical Learning. In: Asada, M., Hallam, J.C.T., Meyer, J.-A., Tani, J. (eds.) SAB 2008. LNCS, vol. 5040, pp. 270–279. Springer, Heidelberg (2008)
32. Lampe, A., Chatila, R.: Performance measures for the evaluation of mobile robot autonomy. In: ICRA 2006, Orlando USA (2006)
33. Gacs, P.: The Boltzmann Entropy and Randomness Tests. In: Proc. 2nd IEEE Workshop on Physics and Computation (PhysComp 1994), pp. 209–216 (1994)
34. Gruenwald, P., Vitanyi, P.: Shannon Information and Kolmogorov Complexity. IEEE Transactions on Information Theory (2004)
35. Garcia, M., Chatterjee, A., Ruina, A., Coleman, M.: The Simplest Walking Model: Stability, Complexity, and Scaling, Transactions of the ASME. Journal of Biomechanical Engineering 120, 281–288 (1998)
36. `http://world.honda.com/ASIMO/technology/`
37. Lloyd, S.: Use of mutual information to decrease entropy: Implication for the second law of thermodynamics. Phys. Rev. A 39(10), 5378–5386 (1989)
38. Lloyd, S.: Measures of Complexity: A Non exhaustive List. IEEE Control Systems Magazine (2001)
39. Rosenblatt, F.: The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain, Cornell Aeronautical Laboratory. Psychological Review 65(6), 386–408 (1958)
40. Potter, S.: What can AI get from neuroscience? In: Lungarella, M., Iida, F., Bongard, J., Pfeifer, R. (eds.) 50 Years of AI. Springer, Heidelberg (2007)

41. Bach-y-Rita, P.: Sensory substitution and the human machine interface. Trends in Cognitive Sciences 7, 541–546 (2003)
42. Bach-y-Rita, P.: Brain Mechanisms in Sensory Substitution. Academic Press, New York (1972)
43. Aihara, K., Matsumoto, G., Ikegaya, Y.: Periodic and non–periodic responses of a periodically forced Hodgkin–Huxley oscillator. J. Theor. Biol. 109, 249–269 (1984)
44. Aihara, K., Takabe, T., Toyoda, M.: Chaotic neural networks. Phys. Lett. A 144(6/7), 333–340 (1990)
45. Der, R.: Self-organized acquisition of situated behavior. Theory in Biosciences 120, 179–187 (2001)
46. Der, R.: Artificial Life from the principle of homeokinesis. In: Proceedings of the German Workshop on Artificial Life (2008)
47. Ay, N., Bertschinger, N., Der, R., Güttler, F., Olbrich, E.: Predictive information and explorative behavior of autonomous robots. In: European Conference on Complex Systems, Dresden (2007)
48. Prokopenko, M., Gerasimov, V., Tanev, I.: Evolving Spatiotemporal Coordination in a Modular Robotic System. In: Nolfi, S., Baldassarre, G., Calabretta, R., Hallam, J.C.T., Marocco, D., Meyer, J.-A., Miglino, O., Parisi, D. (eds.) SAB 2006. LNCS, vol. 4095, pp. 558–569. Springer, Heidelberg (2006)
49. Hornik, K., Stinchcombe, M., White, H.: Multilayer feedforward networks are universal approximators. Neural Networks 2, 359–366 (1989)
50. Funahashi, K.: On the approximate realization of continuous mappings by neural networks. Neural Networks 2, 183–192 (1989)
51. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proc. of the IEEE Intl. Conf. On Neural Network, Washington DC, USA, vol. 4, pp. 1942–1948 (1995)
52. Millonas, M.M.: Swarms, Phase transitions, and Collective Intelligence. In: Langton, C.G. (ed.) Artificial Life III. Santa Fe Institute Studies in the Sciences of the Complexity, vol. XVII, pp. 417–445. Addison-Wesley, Reading (1994)
53. Albert, R., Barabasi, A.L.: Statistical physics of complex networks. Rev. Mod. Phys. 74, 47–97 (2002)
54. Bianconi, G., Barabasi, A.L.: Competition and multiscaling in evolving networks (2000) arXiv:cond-mat/0011029
55. Milutinovic, D.L., Lima, P.U.: Cells and Robots Modeling and Control of large size agent populations. Springer, Heidelberg (2007)
56. Morpurgo, D., Serenità, R., Seiden, P., Celada, F.: Modelling thymic functions in a cellular automation. International Immunology 7/4, 505–516 (1995)
57. Turing, A.M.: Computing machinery and intelligence. Mind 59, 433–460 (1950)
58. Steels, L.: Semiotic dynamics for embodied agents. IEEE Intelligent Systems, 32–38 (2006)
59. Baronchelli, A., Felici, M., Caglioti, E., Loreto, V., Steels, L.: Sharp Transitions towards Shared Vocabularies in Multi-Agent Systems (2005) arxiv.org/pdf/physics/0509075
60. Steels, L.: The Talking Heads Experiment, Laboratorium, Antwerpen. Words and Meanings, vol. 1 (1999)
61. Becker, S.: Implicit learning in 3D object recognition: The importance of temporal context. Neural Computation 11(2), 347–374 (1999)
62. Wiskot, L., Sejnowski, T.J.: Slow feature analysis:Unsupervised learning of invariances. Neural Computation 14(4), 715–770 (2002)

63. Kopell, N.: Toward a theory of modeling central pattern generators. In: Cohen, A., Rossignol, S., Grillner, S. (eds.) Neural control of rhytmic movements in vertebrates. Wiley, New York (1988)
64. Kopell, N., Ermentrout, G.: Phase transition and other phenomena in chains of coupled oscillators. SIAM J. Appl. Math. 50, 1014–1052 (1990)
65. Rus, D.L.: Robotics as Computation for Interaction with the Physical World. In: Special Session on CyberPhysical Systems, IEEE/RSJ 2008, Nice (2008)
66. Markus, G.F.: The Haphazard construction of the human mind. HoughtonMifflin, New York (2008)
67. Simon, H.: The architecture of complexity. Proc. Am. Phil. Soc. 106 (1962)
68. Ashby, W.R.: Design for a Brain. Chapman and Hill, London (1954)
69. Bateson, G.: Steps to an Ecology of Mind. University of Chicago Press, Chicago (1972)
70. Pavlov, I.P.: Conditioned Reflexes: An Investigation of the Physiological Activity of the Cerebral Cortex. Translated and Edited by Anrep, G.V. (ed.). Oxford University Press, London (1927)

Appendix: Information Metrics Relation Proofs

We will derive in the following the relations given in section 3.
In a network model like those adopted in this discussion, [53,54], the probability βi that a new node will connect to a node i already present in the network is a function of the connectivity k_i and on the fitness ηi of that node, such that

$$\Pi_i = \frac{\eta_j k_i}{\sum_j \eta_j k_j} \tag{A.1}$$

A node *i* will increase its connectivity *ki* at a rate that is proportional to the probability that a new node will attach to it, giving

$$\frac{\partial k_i}{\partial t} = m \frac{\eta_i k_i}{\sum_j k_j \eta_j} \tag{A.2}$$

The factor m accounts for the fact that each new node adds m links to the system.
In [26] it is shown that the connectivity distribution, i.e. the probability that a node has k links, is given by the integral

$$P(k) = \int_0^{\eta_{\max}} d\eta \frac{\partial P\left(k_\eta(t) > k\right)}{\partial t} \propto \int d\eta \rho(\eta) \left(\frac{m}{k}\right)^{\frac{C}{\eta}+1} \tag{A.3}$$

where $\rho(\eta)$ is the fitness distribution and *C* is given by:

$$C = d\eta \rho(\eta) \frac{\eta}{1-\beta(\eta)} \tag{A.4}$$

We define a proper *ηi* function which may basically be a performance index of the effectiveness of sensory motor coordination and which control the growth of the network.

The physical agents constituting the system are connected physically, but also from an information standpoint.

Equation (A.5) gives the expression for the Shannon entropy of the network of elements:

$$HN = -\sum_{k=1}^{\infty} P(k) \log P(k) \tag{A.5}$$

where *P(k)* represents the distribution of node connections and the 'infinite' in the summation is actually the big finite number of physical elements, considered, as a simplification, coinciding with the finite elements.

It is important to notice that this is only a part of the information 'stored' into the system: the information in a single neuron or body element is given by equation (2).

The aim of this short discussion is to show that a network of physical elements can actually manage information into the structure of its internal relations, as it can be shown starting from equation (A.5). The concept model described here actually represent a large class of similar models.

In this section the discussion is related to the one in section 3, as the networks of agents we are considering here are actually embodied and situated dynamical systems, which do have a phase space representation. This allows to derive a few further relations.

We can state, for a network of n physical elements, that:

$$\Delta H_{controller} = \Delta HN + \sum_{i}^{n} \Delta H_i - \Delta I \tag{A.6}$$

where $\Delta H_{controller}$ represents the information variation due to the controller, ΔHN is the information variation in the network itself, ΔH_i is the information variation for a single embodied agent, ΔI the multi information between the n agents of the network and the network itself, this last term account for redundancies in information measures between the individual 'intelligent elements' of the structure and the structure itself.
From equation (1), we have:

$$\Delta H_{closed} - \Delta H_{open}^{\max} = \Delta HN + \sum_{i}^{n} \Delta H_i - \Delta I \tag{A.7}$$

And:

$$\Delta HN + \sum_{i}^{n} \Delta H_i - \Delta I \leq I(X;C) \tag{A.8}$$

This is relation (II)
Furthermore:

$$K(X) = \Delta HN + \sum_{i}^{n} \Delta H_i - \Delta I \tag{A.9}$$

This is relation (III)
And from (2) and (A.6):

$$\Delta HN + \sum_{i}^{n} \Delta H_i - \Delta I = \log \frac{W_{closed}}{W_{open}^{\max}} \tag{A.10}$$

Applying again equation (2):

$$\Delta HN + \sum_{i}^{n} \log \frac{W_{closed\,(i)}}{W_{open(i)}^{\max}} - \Delta I = \log \frac{W_{closed}}{W_{open}^{\max}} \tag{A.11}$$

We derive:

$$\Delta HN - \Delta I = \log \frac{W_{closed}}{W_{open}^{\max}} - \log \prod_{i}^{n} \frac{W_{closed\,(i)}}{W_{open(i)}^{\max}} \tag{A.12}$$

If we define the quantities in (A.13), (A.14):

$$\Omega_{closed} = \frac{W_{closed}}{\prod_{i}^{n} W_{closed(i)}} \tag{A.13}$$

$$\Omega_{open}^{\max} = \frac{W_{open}^{\max}}{\prod_{i}^{n} W_{open}^{\max}} \tag{A.14}$$

We obtain equation (A.15):

$$\Delta HN = \log \frac{\Omega_{closed}}{\Omega_{open}^{\max}} + \Delta I \tag{A.15}$$

This is relation (IV)

Neural Pathways of Embodied Simulation

Henrik Svensson, Anthony F. Morse, and Tom Ziemke

Informatics Research Centre, Cognition & Interaction (COIN) Lab,
University of Skövde, P.O. Box 408, SE-541 28 Skövde, Sweden
{henrik.svensson,anthony.morse,tom.ziemke}@his.se

Abstract. Simulation theories have in recent years proposed that a cognitive agent's "inner world" can at least partly be constituted by internal emulations or simulations of its sensorimotor interaction with the world, i.e. covert perception and action. This paper further integrates simulation theory with the notion of the brain as a predictive machine. In particular, it outlines the neural pathways of covert simulations, which include implicit anticipation in cerebellar and basal gangliar circuits, bodily anticipation by means of forward models in the cerebellum, and environmental anticipation in the neocortex. The paper also discusses, to some extent, possible implications of the neural pathways of covert simulation for the frame problem, and the relation between procedural and declarative knowledge in covert simulations.

1 Introduction

According to simulation (or emulation) theories [e.g. 1 ch. 9, 2-5], thinking is, quite literally, rooted in perception and action. In line with empiricist and associationist ideas, thinking is the coupling of covert actions and perceptions. What we mean by *covert action* is the ability to reactivate some of the neural processes and structures used to plan (and execute) bodily movements, but without any actual movements. Similarly, *covert perception* refers to reactivation, in the absence of external stimulation to the sense organs, of some of the neural structures and mechanisms that process sensory input. Thus, simulation processes are off-line processes which can operate in the absence of sensory input and also without causing any movements (see Figure 1).

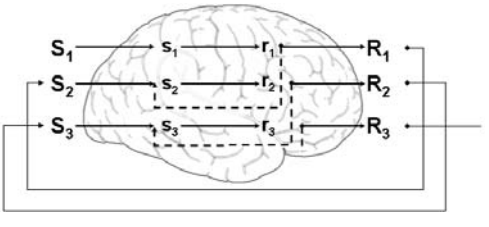


Fig. 1. Covert simulation. Instead of eliciting a new action, covert action r_1 generates a covert perception, s_2, which then generates a new covert action r_2 and so on. (Adapted from [3])

G. Pezzulo et al. (Eds.): ABiALS 2008, LNAI 5499, pp. 95–114, 2009.

Coupling covert actions and perceptions into covert or off-line simulations allows organisms to form, roughly speaking, an internal world. Several advantages come from having such an internal world, one being the ability to try out behavioral options in total safety; "letting our hypotheses die in our stead" to borrow a phrase from Popper [cf. 6]. These *Popperian creatures*, as Dennett calls them, are the opposite to the stupid "who lights the match to peer into the fuel tank, who saws off the limb he is sitting on, who locks his keys in his car and then spends the next hour wondering how on earth to get his family out of the car." [7]. Although not the main topic of the paper, it is worth pointing out that endowing agents with this kind of internal models might lead to something akin to the frame problem [7]. The frame problem was originally posed as a problem for traditional AI, but philosophers have applied it more generally and it may also be a problem that should be addressed by simulation theories, or any other theory, which aim to explain Popperian internal worlds [8]. Haselager and van Rappard [9] interpreted the general frame problem as follows:

> Psychologically speaking, people have an amazing ability to quickly see the relevant consequences of certain changes in a situation. They understand what is going on and are able to draw the right conclusions quickly … The problem is how to model this ability computationally. What are the computational mechanisms that enable people to make common-sense inferences? Especially, how can a computational model be prevented from fruitlessly engaging in time-consuming, irrelevant inferences? A rather straightforward suggestion is that seeing the relevant consequences of an event is made possible by an understanding of the situation. … Yet, human beings posses an enormous amount of information. The real difficulty underlying the frame problem is how the relevant pieces of knowledge are found and how they influence one's understanding of the situation. [9]

Covert simulations may be part of the process of understanding a situation by producing simulations of the possible future states that can be reached from the current [cf. 10, 11]. However, there must be mechanisms that constrain simulations to the relevant aspects of the situation if covert simulations are to be part of the complex task of understanding a situation.

Dennett [7] argued that a problem with many theories, for example associationist theories, was that they did not specify any real (physical) mechanisms to solve the frame problem [cf. 8, 9]:

> Hume explained this in terms of habits of expectation, in effect. But how do the habits work? Hume had a hand-waving answer - associationism - to the effect that certain transition paths between ideas grew more likely-to-be-followed as they became well worn, but since it was not Hume's job, surely, to explain in more detail the mechanics of these links, problems about how such paths could be put to good use - and not just turned into an impenetrable maze of untraversable alternatives - were not discovered. [7]

Simulation theories have argued to extend associationist ideas by paying close attention to how brains work [3], which may provide answers to how the covert simulations are organized and constrained. Looking at the neural mechanisms that produce

covert simulations, it seems that evolution have resulted in a rather general solution - anticipation. We argue in this paper that covert simulations to a large extent reuse the neural circuitry for so called procedural and declarative predictions [12, 13]. Downing [12] pointed out the possible compatibility of his work and simulation theories and also suggested that simulation theories need to explain the relationship between lower- and higher-level representations in covert simulations.

The paper further incorporates the notions of procedural prediction and declarative prediction, although somewhat redefined, with simulation theories, reviews other sources of evidence for its incorporation including neurophysiological mechanisms, and addresses the relationship between lower- and higher-level representations.

The paper is structured as follows. Section 2 provides a brief overview of simulation theories and introduces three notions of anticipation: implicit, bodily, and environmental anticipation. Section 3 is about implicit anticipation in simulations. The section first presents some empirical evidence for the presence of implicit predictions and then describes how implicit predictions might be implemented by the cerebellum and basal ganglia and their role in off-line simulations. Section 4 briefly reviews bodily anticipations in simulations and presents the view of the cerebellum as a forward model. Section 5 focuses on environmental predictions and relates this view to the ideo-motor view of cognition, as well as provides some empirical support for environmental predictions. Furthermore, it presents the view that environmental predictions based on efference copies at various levels of the neocortex plays a crucial role in simulated behavior. The paper ends with a discussion in Section 6.

2 Simulation Theories: What Are the Components?

Simulation or emulation theories explain many aspects of cognition ranging from perception to conceptualization. These theories share, although to various extent, the idea that cognition must be explained in terms of covert perceptions and actions, as defined above [14]. There are differences though. Some simulation theories argue that covert simulations are to be considered as reactivated perceptions and actions [3], inputs and outputs of emulators [5], or perceptual symbols [15]. Although these might be seen as merely minor semantical differences, they can, for example, imply different views on the extent to which simulations require additional theoretical and neural mechanisms beyond the sensorimotor systems. Related to this is whether simulations require the reactivation of the neural substrate closest to the sensory input and motor output terminals [cf. e.g. 11]. However, a more pragmatic perspective consistent with the empirical evidence is that covert simulations exist at many different levels of the sensorimotor hierarchy. There are also different views on the representational nature of covert simulations. Our view of simulation theory, suggests that covert simulations should not be equated with cognitivist notions of representation and internal models in cognitive science and AI. If covert simulations are to be seen as representations with epistemic functions, they cannot only be observer defined correspondences between aspects of the model and aspects of the world [16]. Rather, the covert simulations function as representations because they reactivate the neural activity present during embodied interaction. Covert simulations are representations in the Piagetian sense:

> [R]e-presentations in Piaget's sense are repetitions or reconstructions of items that were distinguished in previous experience. As Maturana explained … such representations are possible also in the autopoietic model [[e.g. 17]]. Maturana spoke there of re-living an experience, and from my perspective this coincides with the concept of representation as Vorstellung, without which there could be no reflection. From that angle, then, it becomes clear that, in the autopoietic organism also, "expectations" are nothing but re-presentations of experiences that are now projected into the direction of the not-yet-experienced. [18]

Covert simulations may result in these kinds of reactivated experiences, but they can also be present without resulting in conscious experiences. The paper makes no further attempt to explain what states or processes are likely to be conscious ones.

The starting point of our explanation of covert simulations is Hesslow's [3] "simulation hypothesis", which postulates the following elements of covert simulations:

> (1) Simulation of actions: we can activate motor structures of the brain in a way that resembles activity during a normal action but does not cause any overt movement. (2) Simulation of perception: imagining perceiving something is essentially the same as actually perceiving it, only the perceptual activity is generated by the brain itself rather than by external stimuli. (3) Anticipation: there exist associative mechanisms that enable both behavioral and perceptual activity to elicit perceptual activity in the sensory areas of the brain. [3]

The next section describes the empirical evidence for the existence of simulations of perception and action, or covert perceptions and actions as they are termed here. Section 2.2 outlines the second aspect of simulations, anticipation, and distinguishes three different forms of anticipation, which are then elaborated in the remainder of the paper.

2.1 Reactivation

A wide range of psychological and neuroscientific studies have shown that cognition to a considerable extent involves the reactivation of the neural processes active during perception and action in humans [for a detailed review see e.g. 19]. Reactivations might also be present in other animals as well. An indicative, but not conclusive, observation is the running movements and yapping of sleeping dogs, which suggests that something like a mental simulation might be present [20].

Reactivation has for a long time been a hypothesis in memory research, dating back to William James, which specifically states that sensory and motor brain regions that are active during encoding are also reactivated during retrieval of memories [21-24]. One of the first neuroscientists to adopt this reactivation hypothesis was Damasio [25] who explained procedural and declarative memory as "time-locked multiregional retroactivation". According to Damasio [25],

> perceptual experience depends on neural activity in multiple regions activated simultaneously … during free recall or recall generated by perception in a recognition task, the multiple region activity necessary for experience occurs near the sensory portals and motor output sites of the system rather than at the end of an integrative processing cascade removed from inputs and outputs. [25]

Both behavioral and, recent neuroimaging experiments have provided further support for the reactivation hypothesis in memory tasks [26]. One of the most spectacular behavioral demonstrations of the importance of the overlap of encoding and retrieval context comes from Godden and Baddeley's [1975, cited in 27] memory context experiment with divers. Using a free recall methodology, Gooden and Baddeley had divers learn lists of words either on land or submerged, and to recall the words either in the same context as during encoding or in the other context. When encoding and retrieval context matched memory performance was enhanced compared to non-matching contexts. These results indicating an interdependence of encoding and retrieval are consistent with the hypothesis that similar neural mechanisms are being used.

Recent neuroimaging experiments of memory have provided further support for the reactivation hypothesis and the assumption that the behavioral effects are due to the activation of the sensory and motor areas used to process the percept or associated action [cf. 21]. Using Positron Emission Tomography (PET), Nyberg et al. [24] found that remembering visual words that had been presented together with sounds at the encoding stage activated some of the auditory brain regions that were active during encoding. Moreover, this effect was present even when the subjects did not have to explicitly remember the sound, but only determine whether the word was part of the original list. This effect also transfers to other types of information, such as spatial location [Persson & Nyberg, 2000, cited in 24], and vivid visual information [23]. Furthermore, Nyberg et al. [22] found that both overt enactment and imaginary enactment of the to be remembered action phrase are accompanied by encoding-retrieval overlaps. However, it should be noted that the studies also show that encoding and retrieval are associated with different activity patterns [22]. However, they do show that sensory and motor regions participate in some cognitive processes that do not involve perception and action [cf. 22].

The reactivation hypothesis generally supports the reactivation of both perceptual and motor areas used during the encoding of the memory. Covert perceptions and covert actions are thus special cases of this general principle of memory and brain function. The two following sections focus on studies that emphasize the perceptual or motor aspect of the reactivation.

Covert Actions. Many experimental results suggest that, to some degree, the same neural substrate is used for action and covert action. Although reactivation of motor actions has been observed in other cognitive tasks such as language understanding, the most encompassing reactivation occurs in explicit or implicit motor imagery [cf. 28] leading some to suggest that covert actions are in fact actions, with the exception that no overt movement occurs [e.g. 29]. Motor imagery is usually defined as the recreation of an experience of actually performing an action, for example, the person should feel as if he or she was actually walking [30]. Motor imagery experiments have shown that mentally simulating an action is similar to overt action in the following aspects: execution time including the reproduction of Fitt's law and isochrony [5, 31-33], physiological effects [34, 35], PET, fMRI, and TMS [for reviews see 4, 36, 37].

In the case of motor imagery, the reactivation of actions is quite independent of the current input stimuli, i.e., independent in the sense that the reactivation is not caused by it. Covert motor activity has however also been found to be automatically elicited by various kinds of external sensory stimuli. The discovery of mirror neurons in the

macaque monkey [38, 39] and the possible existence of mirror systems in humans [40, 41] clearly illustrate this ability. Mirror neurons and canonical neurons have been found in the rostral region of the inferior premotor cortex (area F5) of the monkey brain which contains neurons that are known to discharge during goal directed hand movements, such as grasping, holding, tearing, or manipulating [42]. The special property of mirror neurons is that they are also activated by observation of the same goal-directed hand (and mouth) action being performed by someone else [38, 39].

The empirical evidence suggests that the brain has the ability to reactivate the brain areas responsible for action by means of internal or external stimuli. It also shows that it is possible to do this without causing overt actions. If the covert actions are sufficiently similar to the patterns normally producing movements and actions, the covert actions could internally drive activations in the sensory cortex to resemble the activation that would have occurred if the action had been executed.

Covert Perceptions. There is much empirical evidence, both behavioral and neuroscientific, that suggest that reactivation of covert perceptions is common in human cognition [43-45], but there are also animal learning studies that could suggest that even rats are able to reactivate a perception based on earlier cues [46]. Several studies have indicated that imagination evokes similar experiences to actual object interaction [e.g. 47], and are almost indistinguishable from the real perception [Perky, 1910 cited in 43, 48]. The seminal study by Shepard and Metzler [47], had subjects determine whether two three-dimensional forms had the same shape or not. The results showed that reaction times increased linearly with the angular difference, indicating that the imagined rotations were performed at a constant rate, as if a physical object were rotated [cf. 43]. Furthermore, they found reaction times not to be longer for depth rotations than for rotations in the picture plane. These two findings suggest that imagined rotations in some aspects correspond to actual physical rotations of objects [43]. Moreover, the subjects reported that they solved the task by mentally forming and rotating three-dimensional forms to "see" if they were the same, which might also be taken as support for the involvement of perceptual processes in mental imagery. There have also been findings that suggest a considerable overlap between the mechanisms of spatial attention and spatial working memory [49]. Furthermore, Lauwereyns et al. [50] found that their finding generalizes to non-spatial visual dimensions, such as color and shape.

A recurring issue in neuroscience is to what extent the sensorimotor loop and the off-line simulation overlap. As discussed above, some findings based on behavioral, physiological, brain imaging and single neuron recordings suggest that the overlap is almost complete, except for the overt execution. Other studies have observed small differences in some of the structures, such as a small shift in the rostral direction in the basal ganglia and dorsal premotor cortex for imagined as compared to real actions [51, 52]. The differential activation could perhaps be useful for thinking about an action, while performing another.

2.2 Anticipation

So far we have reviewed evidence concerning the existence of covert perceptions and actions and the extent to which they are similar to actual perceptions and actions. The next step is to address the mechanisms which enable the coupling of covert actions

and perceptions into extended covert simulations (cf. Figure 1). Based on Downing's [13] distinction between declarative and procedural prediction, we suggest that three forms of predictive processes are used to establish covert simulations, implicit, bodily, and environmental anticipations[1]. *Implicit anticipation*: Action selection mechanisms can be seen as anticipations, but of an implicit [53] or procedural [13] kind. This kind of prediction, formed by evolution or learning, allows an animal to act as if it has access to some future goal state, but without the need to produce a (sensory) state that correspond to that goal. In other words, implicit predictions generate actions, which mean that the only information about the external state is in the way that the animal coordinates its behavior with it. *Bodily anticipation*: Many models suggest that it is necessary to produce predictions of the (sensory) state of the body [54-56], because of the inherent time delays in the sensorimotor system. That means, since it is often not possible to successfully plan all motor commands in advance based on the current state and the time delays would prevent error correction during motion, predictions of the future states of the body have to be provided to update the motor planning process. *Environmental anticipation*: The ability to generate a prediction of a future perceptual state that is associated with a particular response in a given situation could be advantageous. For example, if this would lead an animal to reactivate the "image" of a predator, it could also automatically execute the associated action programs and might escape the predator (Hesslow, unpublished manuscript, cf. also [13]). Covert perceptions could initiate action selection mechanisms in similar ways as actual perceptions because of their similarity in terms of neural activity.

3 Implicit Anticipation

Although actions are situated in the sense that they are highly influenced by a particular bodily and environmental situation, prediction and the internal construction of simulated interactions are crucial aspects for the behaving animal. In AI one alternative has been to conceive of the internally constructed plans as prescriptions for actions [57]. An alternative view, the one favored here, is that internally constructed plans are but one of the causal influences (internal or external) on the resulting behavior and that the influence on actual behavior is less direct [cf. 58, 59]. Marques and Holland [60] extensively discussed the necessary and sufficient criteria for an embodied agent capable of planning by means of simulations or, in their terms, functional imagination. The neural mechanisms of implicit anticipation described in this section may also contribute to the understanding of such an agent by providing some hints about the neural mechanisms for goal-seeking behavior (i.e., approaching a goal without explicitly representing it) and action selection. These mechanisms also ensure that simulations are effective, i.e., only relevant simulation paths are considered, rather than simulating every possible action in a situation. However, this would also be the case for creatures not endowed with simulation abilities, since it is only possible to perform a few actions out of an almost infinite pool of possible actions simultaneously [cf. 61]. Thus, our hypothesis is that the non-simulating brain's solution to the action selection problem is reused by the simulating brain. These mechanisms do

[1] The terms prediction and anticipation are used synonymously.

not only speed up simulations by constraining them to a few simulation paths by means of implicit predictions, but also allow them to be directed towards future goals, without explicitly representing the goal.

3.1 Implicit Predictions in Humans and Animals

The establishment of stimulus-response (S-R) associations has been a major theory of animal learning. In the context of this paper, S-R associations can be seen as simple forms of implicit predictions. For example, eyeblink conditioning can be explained in terms of (implicitly anticipatory) S-R associations. A neutral conditioned stimulus (CS), e.g. a tone, is followed by an unconditioned stimulus (US), e.g. a puff of air, which elicits a conditioned response (CR), a blink [62, 63]. After training, the neutral stimulus directly elicits the conditioned response in anticipation of the unconditioned stimulus. The neural substrate of eyeblink conditioning is discussed further in the next section. Cisek and Kalaska [64] provided evidence for predictively activated (but not executed) motor representations in the dorsal premotor cortex of monkeys. More importantly, they also found that the predictive and performance related activity was strikingly similar. Thus, it implements a predictive relationship between the stimuli and the about to-be-activated action. A similar finding is that the perception of objects automatically activates motor representations of the action normally performed when using the object [65].

3.2 Neural Substrate of Implicit Predictions

Although many factors, processed in different parts of the brain, affect behavioral choices [66], basal-ganglia–cortex loops (including amygdala influence) [67] and cortico–cerebellar loops [68] are commonly considered crucial for action selection. These action selection mechanisms are in some respects anticipatory in nature since the agent's actions are directed towards a future situation. However, as described next, there are no explicit predictions involved in the anticipatory behaviors learnt by these action selection mechanisms [cf. 2, 13, 69].

Cerebellum. It has been suggested that the cerebellum learns sensory-motor contingencies through supervised learning [13, 70]. The cerebellum receives input from several different subcortical and cortical areas through mossy fibers to granular cells where the granular cell's axon forms parallel fibers (PF). Each PF synapses onto the dendrites of many Purkinje cells (PC) (~100000:1), whose firing ultimately inhibit a motor response via cells deep in the cerebellum. Each PC receives input from one climbing fiber (CF) (1:1) which gives feedback from afferents located nearby the muscles via the inferior olive [71, 72]. The supervised learning is dependent on the timing of the error feedback, which is explained in the form of eligibility traces that enables long term depression of PF-PC synapses that was active around 100-250msec prior to climbing fire activation [13, 73-75]. In other words, the error feedback from the muscles, affects the signals that was active some time ago, often around the time when those actions that caused the error signals were activated. Some studies have also found that motor imagery activates the spinal cord and muscle spindles [76]. In these cases, it might be possible to covertly generate (simulate) the error signals that the cerebellum needs for learning the correct actions [cf. 71]. Increased activity of the

cerebellum in motor imagery [e.g. 4] and the ability of motor imagery to improve later performance [e.g. 35] is in line with this hypothesis.

Eyeblink conditioning is also thought to be mediated by the cerebellum. In the case of eyeblink conditioning, the CS is presented via mossy fibers and the US via CFs [62]. During training, the PF-PC synapses are altered such that the PC response is decreased around the time of the tone (CS), which causes a disinhibition of the interpositus nucleus and the downstream motor pathways leading to a blink (CR) that coincides with the airpuff (US) [62].

The general conclusion is that the cerebellum can implement S-R relations; as soon as a particular sensory context is present, the cerebellum computes the correct signal to the motor system. The cerebellum could be part of extended simulation loops by helping to establish the S-R links (cf. Figure 1), i.e., to select the actions represented in the neocortex [cf. 77]. In that case, the cerebellum only implements an implicit model of the world, which means that the only criterion for being a model is that it generates correct actions. Other models suggest that the cerebellum functions as a forward model capable to generate predicted (sensory) states (discussed in Section 4).

Basal Ganglia. The basal ganglia have been suggested to play a major role in action selection [78]. For example, Humphries, Stewart and Gurney [79] suggested that "the BG are a critical neural substrate in the vertebrate action selection system, resolving conflict between multiple neural command centers trying to access the final common motor pathway" (p. 12921). The way the basal ganglia implements implicit predictions requires a longer explanation than is possible here [for full descriptions see 12, 13, 78], but in essence the input station of the basal ganglia, the striatum, learns to detect important (cortical) contexts which it maps to actions, represented in the cortex and the brain stem. The learning of a context-action pair is then guided by the emotional response that the action results in. As in the cerebellum, eligibility traces makes sure that contexts active roughly 100msec before an emotional response are the ones strengthened. Furthermore, earlier and earlier contexts can be made to predict the emotional response [13]. The prediction of emotional states allows the basal ganglia to learn context-action pairs that anticipate emotional states.

Several models argue that the basal ganglia together with associated cerebral and cerebellar structures are involved in off-line simulations [61, 80]. For example, Doya [80] suggested that a network consisting of the basal ganglia, parietal cortex and frontal cortex as well as the cerebellum could implement off-line simulations used for planning. However, in Doya's model, the cerebellum does not generate implicit predictions but provides predictions of the new (sensory) state (discussed in Section 4). A role more consistent with the view of the cerebellum as generating implicit predictions is that it contributes to covert simulations by fine-tuning the covert actions selected by the basal ganglia [81]. For example, Sears, Logue and Steinmetz [82] argued, in the context of eyeblink conditioning, that an efference copy of the CR may project to motor cortex, which serves to fine-tune movements and integrate simple responses with more complex movement sequences. A possible function of the basal ganglia (together with the cerebellum) in off-line simulations could be to direct and constrain the course of simulations by selecting some actions over others, but at the same time also prevent them from causing overt movements [2, 83]. In other words, just as the basal ganglia support action selection through reinforcement

learning they also might be able to select the action content of our thoughts [61]. The neuron populations in prefrontal, premotor and motor cortex activated by the basal ganglia can then serve as the input to cortical mechanisms, which predict the sensory consequences of that action (cf. Section 5).

In summary, the resulting sensori-motor associations formed by the cerebellum and basal ganglia may during simulations anticipatorily activate the various parts of the motor system, resulting in covert actions (some of which might be experienced as motor images [30]) or actions [cf. 2, 81]. A central aspect of the learning mechanisms is the eligibility trace, which ensures that the associative learning occurs on synapses that were active roughly 100-250msec prior to a teaching signal [13, 73, 74]. This is could in some cases be an example of co-evolution between the nervous system and body [cf. 84], since the neurochemical processes that allow for the modification of synapses are closely tied to the feedback delays of the sensorimotor system.

4 Bodily Anticipation

For most people it is very difficult to tickle one self. However, it might be possible if you use a feather or better yet if someone else tries to tickle you (using a feather). A possible explanation is that since we have had lots more practice with and can with some certainty know what actions we are about to perform we can predict the proprioceptive signals. Blakemore and colleagues [85-87] argued that the neural mechanisms that produce this phenomenon are based on efference copies feed to the cerebellum. The cerebellum both predicts the sensory consequences of that action and compares it with the resulting sensory feedback from touch sensors, which if there is no discrepancy attenuates the activity in somatosensory cortex. This is usually described as that the cerebellum implements a model of the world, a so called forward model [e.g. 68, 80]. This means that the cerebellum implements a prediction of the state of the body or the sensory afference from the proprioceptive (and proximal sense) organs based on efference copies [e.g. 68][2].

Motor control experiments have also suggested that forward models are necessary because the motor system needs to act on predictive knowledge of future states to, for example, compensate for feedback delays [e.g. 54, 56]. Many models argue that the forward models are found in the cerebellum, and that the forward models can be run off-line to generate covert simulations [e.g. 5, 71]. For example, if motor activity (generated by a cerebellar S-R association, cf. Section 3.2) does not lead to overt action, an efference copy might still be sent to the cerebellum to generate a sensory prediction. These kinds of covert simulations are likely to be closely tied to details of the execution and proximal consequences of bodily movement. Hence, the sensory predictions are related to proprioceptive signals and the proximal senses of touch (and perhaps taste) [88].

Bodily anticipation as we have chosen to call it is also declarative prediction in that it generates states that correlate to external states, i.e., external to the central nervous system, but at the same time the information is about events internal or at surface of the body.

[2] However, whether or not the cerebellum implements a forward model that predicts future sensory states is a matter of discussion [88].

5 Environmental Anticipation

Environmental anticipation differs from bodily anticipation both in its function and neural substrate. The function of environmental anticipation is to generate predictions of future sensory states relating to objects and situations in the world external to the animal's body. Environmental predictions are similar to declarative predictions in that they associate two neural states which each correlate to some environmental state. For example, if a particular perception is associated with the "image" of a predator then the animal might have a better chance of escaping its predator. This type of sensory-sensory associations is certainly present in many mental simulations [3]. Some simulation theories emphasize, as we do in this section, that motor patterns are often crucial in eliciting the sensory activity that is normally associated with the execution of the corresponding action [cf. 2, 3]. This type of environmental prediction might be crucial for implementing longer and more specific or goal directed covert simulations.

5.1 Environmental Predictions in Humans and Animals

The prediction of (sensory) effects was already a central tenet of William James's Ideo-Motor Principle (IMP), i.e., the idea that every action is preceded by a prediction of its effect.

> An anticipatory image, then, of the sensorial consequences of a movement, plus (on certain occasions) the fiat that these consequences shall become actual, is the only psychic state which introspection lets us discern as the forerunner of voluntary acts. [James, 1890/1981, p. 1112, quoted in 89]

The action-effect association is bi-directional [90], implying that it is both a prediction of the effects and a determinant of the behavior [89]. However, in this section the focus is on the prediction of sensory effects, rather than the action selection aspect. The predictive action-effect association has been demonstrated in several animal learning experiments. For example, Colwill and Rescorla [as described in 89] showed that rats do not only learn S-R relationships, but their behavior is determined by the response reinforcer association by devaluation of one of two previously learned response-reinforcement associations.

> Rats were first separately reinforced with food pellets after performing R1 and with a sucrose solution after R2. Once instrumental training had occurred, one of the two reinforcers (outcomes/effects) was devalued by associating it with a mild nausea. Finally, the rats were given the choice between the two responses, but with all outcomes omitted. In this test-phase rats showed a clear suppression of performing the response the outcome of which had been devalued. Obviously, the rats had not only associated the two responses with a situation wherein these were reinforced (S-R1 and S-R2), but they had also learned which response leads to which outcome (R1-food pellets, R2-sucrose solution). [89]

That means, the rats' behavior is guided by the effect associated with the response. Response-effect predictions have been found in several experiments with humans as well [reviewed in detail in 91]. Furthermore, some of these experiments suggest that

the effects are in the form of covert perceptions as suggested by simulation theories [Kiesel & Hoffmann, 2004; Kunde, 2003 in 91].

Furthermore, the experiments performed by Libet could be taken to demonstrate that the initiation of a movement reaches our awareness 50-80msec before the movement has actually started [56], which lends further support to the existence of predictive action-effect associations. Since the movement has not begun, the awareness cannot be generated by proprioceptive or sensory feedback but must be generated by other means, i.e., a prediction/simulation. The similarity of perceptions and covert perceptions, discussed earlier, indirectly also suggests the existence of predictive mechanisms for sensory activity.

5.2 Neural Substrate of Environmental Predictions

The ability to predict sensory states that correspond to various external states is the final functional aspect of simulation to be addressed. The sensory predictions close the simulation loop such that agent-environment interactions can be rehearsed internally in various cognitive processes. They might provide the means to generate chained simulations at various levels of abstraction. However, we shall not discuss the process of abstraction per se here. There are at least three different neural circuits that could implement environmental predictions, the neocortex, thalamo-cortical loops and the hippocampus [13]. The focus here is on the neocortex, but as described in the last subsection there is a common mechanism for environmental predictions.

Neocortex. Possible routes for predictions of sensory or perceptual consequences are located throughout the neocortex. The hierarchical structure of the motor and sensory cortices and the reciprocal connections between them at various levels [3, 92, 93] suggest the possibility of the cortex implementing both predictions from motor to sensory activity and the reverse. Cotterill [2] argued that the premotor areas send information back to the sensory cortex by way of axon collaterals. He further noted that "there are three such efference copy routes…One goes directly, another passes through the anterior cingulate, and the third goes via the thalamic ILN" (p. 22). Efference copy routes might indeed be a ubiquitous property throughout the sensorimotor hierarchy [Hesslow, personal communication cf. 92]. Gomez et al. [94] have, based on their own experiments with the contingent negative variation and other corroborating studies, suggested that there exists an attentional-anticipatory system that "include[s] not only the frequently described prefrontal, SMA, and primary motor cortices, but posterior parietal cortex, cingular cortex, and pulvinar thalamic nuclei too. The neural substrate of the perceptual domain is not so well-described, but, of course, the participation of primary sensory areas has been hypothesized" (p.67). Gomez et al.'s studies do not, however, show decisively how the preparatory activity of the sensory cortex is elicited, i.e., directly via the sensory cues or indirectly by preparatory activity of the motor related cortices. The study by Kastner et al. [95] showed influence from frontal and parietal areas on extrastriate cortex during covert attention shifts, suggesting the possibility of motor areas modulating the activity of sensory areas in an anticipatory manner.

The existence of predictive loops in the neocortex is also supported by research on the mirror neuron system. Canonical neurons are neurons whose response properties are somewhat more specific to particular visual (interaction) properties of objects (action affordances) rather than the action-object conjunction typical of mirror neurons

(India Morrison, personal communication). Iacoboni [as described in 88] postulated that the mirror neuron related areas can implement predictions of the consequences of actions. This would involve projections from area F5 of the ventral premotor cortex, through area PF, and to STS, essentially "converting the motor plan back into a predicted visual representation (a sensory outcome of the action)". However, it should also be pointed out that Miall [88] argued similar transformations might be implemented by pathways incorporating the cerebellum. In line with the distinction between bodily and environmental anticipation, Miall pointed out that mirror neuron related activity reflects more general aspects of actions, whereas forward models in motor control would be more detailed, suggesting that prediction of sensory effects might take place at several different levels of abstraction. It should also be noted that although the emphasis has been on the generation of sensory activity by motor activity, several associations form between perceptual stimuli, which do not include a motor aspect [3]. In other words, covert perceptions may elicit other covert perceptions.

Another type of covert simulations implicating the neocortex are the as-if loops of Damasio [96, 97]. He argued that feelings can occur in the absence of their normal bodily causes, by short circuiting the body loop. Instead, the feelings are simulated in loops involving the prefrontal cortex and the somatosensory cortex. One advantage according to Damasio [96], is that the connection between the prefrontal cortex and the somatosensory cortex, especially the insula, are very short, which means that the signaling can occur in hundreds of milliseconds as opposed to the body loop that takes up to 1 second to complete due to the long, often unmyelinated, axons. In effect, the as-if feelings can be seen as predictions of "bodily feelings".

Declarative Prediction Networks and Simulation. Downing [13] suggested a common model for how the kinds of declarative predictions are learnt in cortical, thalamocortical, and hippocampal circuits, which he called the general declarative predictive network (GDPN). Although his focus was the association of consequent sensory states, the neurophysiology behind this type of association might also explain the predictive association between a motor representation and its sensory consequence (at some or several levels of the sensorimotor hierarchy, cf. [92]). The declarative prediction networks that Downing postulates provide an unsupervised learning scheme. This would work in the neocortex as described briefly in the following text. The neocortex is organized horizontally into layers, and vertically into groups of cells linked synaptically across the horizontal layers called cortical columns or microcolumns [98]. As described by Swanson [99], the neocortex consists of the same number of layers throughout, six layers in both humans and rats while phylogenetically older parts of the cerebral cortex, such as the hippocampus only have 3 layers. In humans, as in rats, the first (outer) layer of the neocortex consists mainly of wiring and has relatively few cell bodies, layer 2 and 3 typically contain small pyramidal neurons which project to other cortical regions in the same and different hemispheres respectively. Layer 4 consists mainly of granule cells which form local circuits, while layers 5 and 6 contain larger pyramidal neurons typically projecting to the brainstem, thalamus, and spinal cord, as well as to the motor system broadly defined. The precise makeup of these layers in terms of the density of cell bodies in each layer varies considerably in different regions of cortex. Even though their function is not agreed upon, it has been suggested that they are essentially predictive elements [13, 100]. In brief, Hawkins [100] explained it as follows:

> Imagine you are a column of cells, and input form a lower region causes one of your layer 4 cells to fire. You are happy, and your layer 4 cell causes cells in layers 2 and 3, then 5 and the 6 also to fire. The entire column becomes active when driven from below. Your cells in layers 2, 3, and 4 each have thousands of synapses in layer 1. If some of these synapses are active when your layer 2, 4, and 5 cells fire, the synapses are strengthened. If this occurs often enough, these layer 1 synapses become strong enough to make the cells in layers 2, 3 and 5 fire even when a layer 4 cell hasn't fired[cf. [101]] - meaning parts of the column can become active without receiving input from a lower region of the cortex. [100]

Given that a large number of the connections onto a column come from other parts of the cortex, it is not to unlikely that some of the predictive associations are made between the motor areas of the cortex and sensory areas of the cortex via the different routes suggested above. Furthermore, as noted above it is possible that these associations form at different levels of the sensorimotor hierarchy. It is possible that the general declarative prediction network in the hippocampus is able to learn even more complex and abstracted sensory-motor and motor-sensory associations.

6 Discussion

In the introduction we argued that covert simulations might provide some answers to the human brain's solution to the general frame problem. One part of the answer lies in the way covert simulations are constructed to only focus on the relevant consequences of an action and are able to influence overt behavior in time. The neurochemical properties of the eligibility trace that closely matches the embodiment of the organism, or more specifically, time delays of the sensorimotor system ensure that the feedback signals that provide valuable information about the usefulness of an action is likely to be associated with the action that lead to the environmental state which the feedback is about. Furthermore, the learnt implicit predictions make the covert simulations effective by constraining the number of simulation paths that could otherwise be explored. At a higher level of abstraction, the general declarative prediction networks are biased toward only creating predictions that have been supported by environmental evidence to emerge. Covert simulations may then provide the kind of intrinsic representations thought to be necessary to be able to represent the world without describing everything about it [cf. 9]. The ability to focus on relevant consequences is, even though only briefly discussed in this paper, also crucially dependent on the existence of special brain circuitry for affect and emotion and their close relationship to action selection mechanisms and off-line simulations [e.g. 102] constitute mechanisms for connecting additional meaning to sensorimotor associations. The view of covert simulations as implicit, bodily, and environmental anticipations is to some extent already implemented in computational models [11, 103, 104], which is where the actual frame problems arise [9]. For example, Möller and Schenck [11] showed how covert simulations could support the understanding of space and shape in object recognition. However, it might be argued that these models are still too simple for frame problems to be an issue as it is often thought to be a problem of common sense reasoning in humans [cf. 9]. Future work aiming to

achieve more advanced forms of planning [cf. 60] may need to consider the implications of the frame problem in more depth and especially to what extent the neural mechanisms proposed in this paper are able to resolve the problems.

An important property of the neocortex that has largely been ignored in the paper, but may prove important to covert simulations is its hierarchical organization. Information flows up and down within the sensory and the motor hierarchies and not just between them, as emphasized above, which can explain several aspects of off-line simulations. This can perhaps provide useful insights about how covert simulations are established at different levels of abstraction [100]. Furthermore, it can explain why brain damage closer in the lower parts of the hierarchy, such as primary motor and sensory areas sometimes (although not always) leaves the capacity for mental imagery intact [44]. Farah [44] argued that can be explained by the hierarchical structure of the neocortex and considering mental imagery mainly as a top-down process.

> Assume the damaged parts are among those shared by imagery and perception, not purely perceptual afferents, and consider the impact of interrupting processing at this stage: When the flow of processing is bottom-up or afferent, as in perception, the impact will be large because the majority of visual representations cannot be accessed. In contrast, when the flow of processing is top down or efferent, as in imagery, the impact will be smaller because just a minority of the representations normally activated in imagery is unavailable. [44]

Similarly, but in the context of motor imagery, Jeannerod [4] speculated that lesions higher-up in the motor hierarchy, including the supplementary motor area (SMA) and premotor cortex, would cause more impairment to the imagery process. This is consistent with brain imaging experiments of motor imagery which do not always find activations of the primary motor cortex [105].

A final question to be addressed is the one posed by Downing [12]. He argued, on neuroscientific grounds, that declarative knowledge could not be created from procedural knowledge and asked how this distinction could be explained by simulation theories. Our answer is that the two types of knowledge complement each other in covert simulations via multiple neural simulation pathways. As discussed earlier, a typical example of a task that involves off-line simulations is motor imagery (MI). MI involves both procedural and declarative properties, according to both neural and psychological definitions. Procedurally, MI is associated with unconscious effects, such as increased respiratory and heart rates with increased imagined effort, and has been shown to activate the cerebellum, basal ganglia and primary motor cortex. Declaratively, MI is more or less defined as the conscious feeling of performing an action, and it involves higher motor areas, and perhaps also sensory areas [5]. This is not surprising as many real agent-environment interactions would involve both procedural and declarative elements. For example, Downing [12] argued that although each word or phrase of a song is stored in the cortex, the extraction of a particular word or phrase is "mediated by the preceding cortical context (declarative) and basal gangliar wiring (procedural)" (p. 97). In other words, you access the declarative structures, the words and phrases, by performing a skill, in this case singing. In accordance with simulation theories, the extraction can be made either by singing or by rehearsing it internally without producing any actual sounds. It would seem that simulation theories that aim to explain conceptualization based on the reactivation of sensorimotor structures [15, 106], would not have to cross the gap between the procedural and declarative

either. Simulations are in those theories thought to enact the concept, which could then consist of both declarative knowledge and procedural or skill-based knowledge.

Acknowledgements. This work has been partly supported by a European Commission grant to the FP6 project *"Integrating Cognition, Emotion and Autonomy"* (ICEA, IST-027819, www.iceaproject.eu).

References

1. Balkenius, C.: Natural Intelligence in Artificial Creatures. Doctoral Dissertation. University of Lund (1995)
2. Cotterill, R.M.J.: Cooperation of the Basal Ganglia, Cerebellum, Sensory Cerebrum and Hippocampus: Possible Implications for Cognition, Consciousness, Intelligence and Creativity. Progress in Neurobiology 64, 1–33 (2001)
3. Hesslow, G.: Conscious Thought as Simulation of Behaviour and Perception. Trends in Cognitive Sciences 6, 242–247 (2002)
4. Jeannerod, M.: Neural Simulation of Action: A Unifying Mechanism for Motor Cognition. NeuroImage 14, 103–109 (2001)
5. Grush, R.: The Emulation Theory of Representation: Motor Control, Imagery, and Perception. Behavioral and Brain Sciences 27, 377–396 (2004)
6. Dennett, D.C.: Kinds of Minds: Toward an Understanding of Consciousness. Basic Books, New York (1996)
7. Dennett, D.C.: Cognitive Wheels: The Frame Problem in Artificial Intelligence. In: Hookway, C. (ed.) Minds, Machines and Evolution, pp. 129–151. Cambridge University Press, Cambridge (1984)
8. Shanahan, M., Baars, B.: Applying Global Workspace Theory to the Frame Problem. Cognition 98, 157–176 (2005)
9. Haselager, W.F.G., van Rappard, J.F.H.: Connectionism, Systematicity, and the Frame Problem. Minds and Machines 8, 161–179 (1998)
10. Chrisley, R.: Cognitive Map Construction and Use: A Parallel Distributed Processing Approach. In: Touretzky, D., Elman, J., Hinton, G., Sejnowski, T. (eds.) Connectionist Models: Proceedings of the 1990 Summer School, pp. 287–302. Morgan Kaufman, San Mateo (1990)
11. Möller, R., Schenck, W.: Bootstrapping Cognition from Behavior—A Computerized Thought Experiment. Cognitive Science 32, 504–542 (2008)
12. Downing, K.L.: Neuroscientific Implications for Situated and Embodied Artificial Intelligence. Connection Science 19, 75–104 (2007)
13. Downing, K.L.: Predictive Models in the Brain. Connection Science (in press)
14. Svensson, H.: Embodied Simulation as Off-Line Representation. University of Skövde, Licentiate Dissertation, Linköping (2007)
15. Barsalou, L.W.: Perceptual Symbol Systems. Behavioral and Brain Sciences 22, 577–660 (1999)
16. Bickhard, M.H.: Language as an Interaction System. New Ideas in Psychology 25, 171–187 (2007)
17. Maturana, H.R., Varela, F.J.: The Tree of Knowledge: The Biological Roots of Human Understanding. Shambhala, Boston (1987)
18. von Glasersfeld, E.: Distinguishing the Observer: An Attempt at Interpreting Maturana, Transl. (1990). Originally appeared as von Glasersfeld, E (1990) Die Unterscheidung des Beobachters: Versuch einer Auslegung. In: Riegas, V., Vetter, C. (eds.) Zur Biologie der Kognition, pp. 281–295. Suhrkamp, Frankfurt, Germany (1990)

19. Svensson, H., Lindblom, J., Ziemke, T.: Making Sense of Embodiment: Simulation Theories of Shared Neural Mechanisms for Sensorimotor and Cognitive Processes. In: Ziemke, T., Zlatev, J., Frank, R. (eds.) Body, Language and Mind. Embodiment, vol. 1, pp. 241–270. Mouton de Gruyter, Berlin (2007)
20. Sjölander, S.: Some Cognitive Breakthroughs in the Evolution of Cognition and Consciousness, and Their Impact on the Biology of Language. Evolution and Cognition 3, 3–11 (1995)
21. Gandhi, S.P.: Memory Retrieval: Reactivating Sensory Cortex. Current Biology 11, R32–R34 (2001)
22. Nyberg, L., Petersson, K.M., Nilsson, L.G., Sandblom, J., Aberg, C., Ingvar, M.: Reactivation of Motor Brain Areas During Explicit Memory for Actions. NeuroImage 14, 521–528 (2001)
23. Wheeler, M.E., Petersen, S.E., Buckner, R.L.: Memory's Echo: Vivid Remembering Reactivates Sensory-Specific Cortex. Proc. Natl. Acad. Sci. USA 97, 11125–11129 (2000)
24. Nyberg, L., Habib, R., McIntosh, A.R., Tulving, E.: Reactivation of Encoding-Related Brain Activity During Memory Retrieval. Proc. Natl. Acad. Sci. USA 97, 11120–11124 (2000)
25. Damasio, A.R.: Time-Locked Multiregional Retroactivation: A Systems-Level Proposal for the Neural Substrates of Recall and Recognition. Cognition 33, 25–62 (1989)
26. Nyberg, L., Forkstam, C., Petersson, K.M., Cabeza, R., Ingvar, M.: Brain Imaging of Human Memory Systems: Between-Systems Similarities and within-System Differences. Cognitive Brain Research 13, 281–292 (2002)
27. Baddeley, A.D.: Essentials of Human Memory. Psychology Press, Hove (1999)
28. de Lange, F.P., Roelofs, K., Toni, I.: Motor Imagery: A Window into the Mechanisms and Alterations of the Motor System. Cortex 44, 494–506 (2008)
29. Gentili, R., Papaxanthis, C., Pozzo, T.: Improvement and Generalization of Arm Motor Performance through Motor Imagery Practice. Neuroscience 137, 761–772 (2006)
30. Jeannerod, M.: The Representing Brain: Neural Correlates of Motor Intention and Imagery. Behavioral and Brain Sciences 17, 187–245 (1994)
31. Jeannerod, M., Frak, V.: Mental Imaging of Motor Activity in Humans. Current Opinion in Neurobiology 9, 735–739 (1999)
32. Papaxanthis, C., Schieppati, M., Gentili, R., Pozzo, T.: Imagined and Actual Arm Movements Have Similar Durations When Performed under Different Conditions of Direction and Mass. Experimental Brain Research 143, 447–452 (2002)
33. Guillot, A., Collet, C.: Duration of Mentally Simulated Movement: A Review. Journal of Motor Behavior 37, 10–20 (2005)
34. Decety, J., Jeannerod, M., Durozard, D., Baverel, G.: Central Activation of Autonomic Effectors During Mental Simulation of Motor Actions in Man. The Journal of Physiology 461, 549–563 (1993)
35. Yue, G., Cole, K.J.: Strength Increases from the Motor Program: Comparison of Training with Maximal Voluntary and Imagined Muscle Contractions. Journal of Neurophysiology 67, 1114–1123 (1992)
36. Grèzes, J., Decety, J.: Functional Anatomy of Execution, Mental Simulation, Observation, and Verb Generation of Actions: A Meta-Analysis. Human Brain Mapping 12, 1–19 (2001)
37. Fadiga, L., Craighero, L.: Electrophysiology of Action Representation. Journal of Clinical Neurophysiology 21, 157–169 (2004)
38. Rizzolatti, G., Fadiga, L., Gallese, V., Fogassi, L.: Premotor Cortex and the Recognition of Motor Actions. Cognitive Brain Research 3, 131–141 (1996)
39. di Pellegrino, G., Fadiga, L., Fogassi, L., Gallese, V., Rizzolatti, G.: Understanding Motor Events: A Neurophysiological Study. Experimental Brain Research 91, 176–180 (1992)

40. Rizzolatti, G., Arbib, M.A.: Language within Our Grasp. Trends in Neurosciences 21, 188–194 (1998)
41. Rizzolatti, G.: The Mirror Neuron System and Its Function in Humans. Anatomy and Embryology 210, 419–421 (2005)
42. Rizzolatti, G., Camarda, R., Fogassi, L., Gentilucci, M., Luppino, G., Matelli, M.: Functional Organization of Inferior Area 6 in the Macaque Monkey. Experimental Brain Research 71, 491–507 (1988)
43. Finke, R.A.: Principles of Mental Imagery. MIT Press, Cambridge (1989)
44. Farah, M.J.: The Neural Bases of Mental Imagery. In: Gazzaniga, M.S. (ed.) The New Cognitive Neurosciences, pp. 965–974. MIT Press, Cambridge (2000)
45. Kosslyn, S.M.: Image and Brain: The Resolution of the Imagery Debate. MIT Press, Cambridge (1994)
46. Holland, P.C.: Event Representation in Pavlovian Conditioning: Image and Action. Cognition 37, 105–131 (1990)
47. Shepard, R.N., Metzler, J.: Mental Rotation of Three-Dimensional Objects. Science 171, 701–703 (1971)
48. Cotterill, R.M.J.: Enchanted Looms: Conscious Networks in Brains and Computers. Cambridge University Press, Cambridge (1998)
49. Awh, E., Jonides, J.: Overlapping Mechanisms of Attention and Spatial Working Memory. Trends in Cognitive Sciences 5, 119–126 (2001)
50. Lauwereyns, J., Wisnewski, R., Keown, K., Govan, S.: Crosstalk between On-Line and Off-Line Processing of Visual Features. Psychological Research 70, 170–179 (2006)
51. Gerardin, E., Sirigu, A., Lehericy, S., Poline, J.-B., Gaymard, B., Marsault, C., Agid, Y., Le Bihan, D.: Partially Overlapping Neural Networks for Real and Imagined Hand Movements. Cerebral Cortex 10, 1093–1104 (2000)
52. Lamm, C., Windischberger, C., Moser, E., Bauer, H.: The Functional Role of Dorso-Lateral Premotor Cortex During Mental Rotation: An Event-Related fMRI Study Separating Cognitive Processing Steps Using a Novel Task Paradigm. NeuroImage 36, 1374–1386 (2007)
53. Butz, M.V., Sigaud, O., Gérard, P.: Internal Models and Anticipations in Adaptive Learning Systems. In: Butz, M.V., Sigaud, O., Gérard, P. (eds.) Anticipatory Behavior in Adaptive Learning Systems. LNCS, vol. 2684, pp. 86–109. Springer, Heidelberg (2003)
54. Wolpert, D.M., Miall, R.C., Kawato, M.: Internal Models in the Cerebellum. Trends in Cognitive Sciences 2, 338–347 (1998)
55. Miall, R.C., Wolpert, D.M.: Forward Models for Physiological Motor Control. Neural Networks 9, 1265–1279 (1996)
56. Frith, C.D., Blakemore, S.J., Wolpert, D.M.: Abnormalities in the Awareness and Control of Action. Phil. Trans. R. Soc. Lond. B 355, 1771–1788 (2000)
57. Bryson, J.J.: Mechanisms of Action Selection: Introduction to the Special Issue. Adaptive Behavior 15, 5–8 (2007)
58. Baldassarre, G.: A Biologically Plausible Model of Human Planning Based on Neural Networks and Dyna-Pi Models. In: Proceedings of the Workshop on Adaptive Behaviour in Anticipatory Learning Systems (ABiALS 2002), pp. 40–60 (2002)
59. Baldassarre, G.: Forward and Bidirectional Planning Based on Reinforcement Learning and Neural Networks in a Simulated Robot. In: Butz, M.V., Sigaud, O., Gérard, P. (eds.) Anticipatory Behavior in Adaptive Learning Systems. LNCS, vol. 2684, pp. 179–200. Springer, Heidelberg (2003)
60. Marques, H.G., Holland, O.: Architectures for Functional Imagination. Neurocomputing 72, 743–759 (2009)

61. Humphries, M.D., Gurney, K.N.: The Role of Intra-Thalamic and Thalamocortical Circuits in Action Selection. Network: Computation in Neural Systems 13, 131–156 (2002)
62. Jirenhed, D.-A., Bengtsson, F., Hesslow, G.: Acquisition, Extinction, and Reacquisition of a Cerebellar Cortical Memory Trace. Journal of Neuroscience 27, 2493–2502 (2007)
63. Mauk, M.D., Medina, J.F., Nores, W.L., Ohyama, T.: Cerebellar Function: Coordination, Learning or Timing? Current Biology 10, 522–525 (2000)
64. Cisek, P., Kalaska, J.F.: Neural Correlates of Mental Rehearsal in Dorsal Premotor Cortex. Nature 431, 993–996 (2004)
65. Tucker, M., Ellis, R.: On the Relations between Seen Objects and Components of Potential Actions. Journal of Experimental Psychology: Human Perception and Performance 24, 830–846 (1998)
66. Cisek, P.: Cortical Mechanisms of Action Selection: The Affordance Competition Hypothesis. Phil. Trans. Roy. Soc. B 362, 1585–1600 (2007)
67. Prescott, T.J., Redgrave, P., Gurney, K.: Layered Control Architectures in Robots and Vertebrates. Adaptive Behavior 7, 99–127 (1999)
68. Wolpert, D., Miall, R.C., Kawato, M.: Internal Models in the Cerebellum. Trends in Cognitive Sciences 2, 338–347 (1998)
69. Riegler, A.: Whose Anticipations? In: Butz, M.V., Sigaud, O., Gérard, P. (eds.) Anticipatory Behavior in Adaptive Learning Systems. LNCS, vol. 2684, pp. 11–22. Springer, Heidelberg (2003)
70. Doya, K., Uchibe, E.: The Cyber Rodent Project: Exploration of Adaptive Mechanisms for Self-Preservation and Self-Reproduction. Adaptive Behavior 13, 149–160 (2005)
71. Doya, K.: What Are the Computations of the Cerebellum, the Basal Ganglia and the Cerebral Cortex? Neural Networks 12, 961–974 (1999)
72. Houk, J.C., Buckingham, J.T., Barto, A.G.: Models of the Cerebellum and Motor Learning. Behavioral and Brain Sciences 19, 368–383 (1996)
73. Houk, J.C., Alford, S.: Computational Significance of the Cellular Mechanisms for Synaptic Plasticity in Purkinje Cells. Behavioral and Brain Sciences 19, 457–460 (1996)
74. Kettner, R.E., Mahamud, S., Leung, H.C., Sitkoff, N., Houk, J.C., Peterson, B.W., Barto, A.G.: Prediction of Complex Two-Dimensional Trajectories by a Cerebellar Model of Smooth Pursuit Eye Movement. Journal of Neurophysiology 77, 2115–2130 (1997)
75. Medina, J.F., Carey, M.R., Lisberger, S.G.: The Representation of Time for Motor Learning. Neuron 45, 157–167 (2005)
76. Lethin, A.: Covert Agency with Proprioceptive Feedback. Journal of Consciousness Studies 12, 96–115 (2005)
77. Yeo, C.H., Hesslow, G.: Cerebellum and Conditioned Reflexes. Trends in Cognitive Sciences 2, 322–330 (1998)
78. Prescott, T.J., Gurney, K., Redgrave, P.: Basal Ganglia. In: Arbib, M.A. (ed.) The Handbook of Brain Theory and Neural Networks, pp. 147–151. MIT Press, Cambridge (2002)
79. Humphries, M.D., Stewart, R.D., Gurney, K.N.: A Physiologically Plausible Model of Action Selection and Oscillatory Activity in the Basal Ganglia. Journal of Neuroscience 26, 12921–12942 (2006)
80. Doya, K.: Reinforcement Learning: Computational Theory and Biological Mechanisms. HFSP Journal 1, 30–40 (2007)
81. Houk, J.C., Bastianen, C., Fansler, D., Fishbach, A., Fraser, D., Reber, P.J., Roy, S.A., Simo, L.S.: Action Selection and Refinement in Subcortical Loops through Basal Ganglia and Cerebellum. Phil. Trans. R. Soc. Lond. B 362, 1573–1584 (2007)
82. Sears, L.L., Logue, S.F., Steinmetz, J.E.: Involvement of the Ventrolateral Thalamic Nucleus in Rabbit Classical Eyeblink Conditioning. Behavioural Brain Research 74, 105–117 (1996)

83. Shanahan, M.: A Cognitive Architecture That Combines Internal Simulation with a Global Workspace. Consciousness and Cognition 15, 433–449 (2006)
84. Chiel, H.J., Beer, R.D.: The Brain Has a Body: Adaptive Behavior Emerges from Interactions of Nervous System, Body and Environment. Trends in Neurosciences 20, 553–557 (1997)
85. Blakemore, S.J., Frith, C.D., Wolpert, D.M.: Spatio-Temporal Prediction Modulates the Perception of Self-Produced Stimuli. Journal of Cognitive Neuroscience 11, 551–559 (1999)
86. Blakemore, S.J., Wolpert, D., Frith, C.: Why Can't You Tickle Yourself? NeuroReport 11, R11–R16 (2000)
87. Blakemore, S.J., Frith, C.D., Wolpert, D.M.: The Cerebellum Is Involved in Predicting the Sensory Consequences of Action. NeuroReport 12, 1879–1884 (2001)
88. Miall, R.C.: Connecting Mirror Neurons and Forward Models. NeuroReport 14, 2135 (2003)
89. Hoffmann, J.: Anticipatory Behavioral Control. In: Butz, M.V., Sigaud, O., Gérard, P. (eds.) Anticipatory Behavior in Adaptive Learning Systems. LNCS, vol. 2684, pp. 44–65. Springer, Heidelberg (2003)
90. Elsner, B., Hommel, B., Mentschel, C., Drzezga, A., Prinz, W., Conrad, B., Siebner, H.: Linking Actions and Their Perceivable Consequences in the Human Brain. NeuroImage 17, 364–372 (2002)
91. Kunde, W., Elsner, K., Kiesel, A.: No Anticipation–No Action: The Role of Anticipation in Action and Perception. Cognitive Processing 8, 71–78 (2007)
92. Fuster, J.M.: Upper Processing Stages of the Perception-Action Cycle. Trends in Cognitive Sciences 8, 143–145 (2004)
93. Fuster, J.M.: Network Memory. Trends in Neurosciences 20, 451–459 (1997)
94. Gomez, C.M., Fernandez, A., Maestu, F., Amo, C., Gonzalez-Rosa, J.J., Vaquero, E., Ortiz, T.: Task-Specific Sensory and Motor Preparatory Activation Revealed by Contingent Magnetic Variation. Cognitive Brain Research 21, 59–68 (2004)
95. Kastner, S., Pinsk, M.A., De Weerd, P., Desimone, R., Ungerleider, L.G.: Increased Activity in Human Visual Cortex During Directed Attention in the Absence of Visual Stimulation. Neuron 22, 751–761 (1999)
96. Damasio, A.R.: Looking for Spinoza: Joy, Sorrow, and the Human Brain. Harcourt, Orlando (2003)
97. Damasio, A.R.: Descartes'error: Emotion, Reason, and the Human Brain. Penguin, New York (1994)
98. Mountcastle, V.B.: The Columnar Organization of the Neocortex. Brain 120, 701–722 (1997)
99. Swanson, L.W.: Brain Architecture: Understanding the Basic Plan. Oxford University Press, Oxford (2003)
100. Hawkins, J., Blakeslee, S.: On Intelligence. Henry Holt, New York (2004)
101. Hansel, C., Artola, A., Singer, W.: Different Threshold Levels of Postsynaptic [Ca2+]I Have to Be Reached to Induce LTP and LTD in Neocortical Pyramidal Cells. Journal of Physiology-Paris 90, 317–319 (1996)
102. Damasio, A.R.: The Feeling of What Happens: Body and Emotion in the Making of Consciousness. Harcourt Brace, New York (1999)
103. Ziemke, T., Jirenhed, D.A., Hesslow, G.: Internal Simulation of Perception: A Minimal Neuro-Robotic Model. Neurocomputing 68, 85–104 (2005)
104. Svensson, H.: Representation as Internal Simulation: A Robotic Model. In: CogSci 2009: 31st Annual Meeting of the Cognitive Science Society (submitted)
105. Dechent, P., Merboldt, K.-D., Frahm, J.: Is the Human Primary Motor Cortex Involved in Motor Imagery? Cognitive Brain Research 19, 138–144 (2004)
106. Zwaan, R.A.: The Immersed Experiencer: Toward an Embodied Theory of Language Comprehension. The Psychology of Learning and Motivation 44, 35–62 (2004)

The Autopoietic Nature of the "Inner World"
A Study with Evolved "Blind" Robots

Michela Ponticorvo[1], Domenico Parisi[2], and Orazio Miglino[1,2]

[1] Natural and Artificial Cognition Laboratory,
University of Naples "Federico II", Italy
[2] Laboratory of Autonomous Robotics and Artificial Life, Institute of Cognitive Sciences and Technologies, National Research Council, Rome, Italy

Abstract. In this paper we propose a model of anticipatory behavior in robots which lack any sort of external stimulation. It would seem that in order to foresee an event and produce an anticipatory action an organism should receive some input from the external environment as a basis to predict what comes next. We ask if, even in absence of external stimulation, the organism can derive this knowledge from an "inner" world which "resonates" with the external world and is built up by an autopoietic process.

We describe a number of computer simulations that show how the behavior of living organisms can reflect the particular characteristics of the environment in which they live and can be adaptive with respect to that environment even if the organism obtains extremely little information from the environment through its sensors, or no information at all. We use the Evorobot simulator to evolve a population of artificial organisms (software robots) with the ability to explore a square arena. Results indicate that sensor-less robots are able to accomplish this exploration task by exploiting three mechanisms: (1) they rely on the internal dynamics produced by recurrent connections; (2) they diversify their behavior by employing a larger number of micro-behaviors; (3) they self-generate an internal rhythm which is coupled to the external environment constraints. These mechanisms are all mediated by the robot's actions.

1 Introduction

From a psychological point of view past events and future events are essentially the same: they do not actually exist in the environment experienced by the organism but, this notwithstanding, they influence the organism's behavior.

Past and future become real exclusively in the organism's mind or brain, as the organism recalls a past experience or foresees something which is going to happen. Recalling and foreseeing are two functions of memory, the neuro-cognitive function that allows organisms to keep trace of what has happened in order to decide what to do next (von Foerster, 1969). In recent years some psychologists have proposed to add to the classical memory typologies (short-term memory, long-term memory, episodic memory, etc.) another kind of memory defined as prospective memory (Brandimonte et al., 1996) which is at work in forecasting

G. Pezzulo et al. (Eds.): ABiALS 2008, LNAI 5499, pp. 115–131, 2009.

and planning behavior. An organism's mind/brain contains a structure/function which encodes, at the same time, what the organism has experienced previously (past), what the organism will potentially experience (future), and what the organisms is currently experiencing (present). The present, under the form of every sensation, thought, expectation, movement which the organism is now experiencing, must be "digested" (elaborated) in the framework of this neuro-cognitive structure/function. In this sense, cognition may be the result of an autopoietic process as defined by Maturana and Varela (1980). According to these authors, a genotype becomes an organism trough an active interaction with the external environment: it extracts primary resources from the external environment (water, food, etc.) and it transforms them into tissues, bones, organs, systems, etc. The organism is the factory of itself. This view of the organism can be extended to the genesis of cognitive structures. Cognition is an internal world that is linked with the external environment but is not a direct copy of it. Living systems are self-reproducing systems and cognition is one of the processes that characterize their self-reproduction. These systems are self-referential, operationally closed, and they compensate for the perturbations arriving from the external world to save their organization. However they also transform as a consequence of environmental stimulation.

This view can be related to Piaget's (1971) conception of cognitive development as an interaction between assimilation and accommodation. Assimilation and accommodation are two complementary processes of adaptation. Assimilation changes the external word to adapt it to the internal world. Accommodation changes the internal world to adapt it to the external world. Cognition is the result of both processes.

In the context of robotics research Tom Ziemke (Ziemke, 2005; 2007; 2008; Jirenhed et al., 2001) has studied intensively this "nonphysical space" that holds past, present, and future together in an "inner world", a notion which has been first introduced by Hesslow (2002) and developed by Grush (2004). The metaphor is powerful and useful because it makes clear the crucial split between the external, physical world the organism is immersed in and its "internal", private world which is hidden to other organisms but is fundamental in determining the organism's behavior.

What is emerging in the new field of artificial adaptive systems is an issue which has played a fundamental role in the philosophical reflection and psychological investigation concerning the behavior of organisms: the delicate balance that exists in organisms between "external" (physical, concrete, public) reality and "internal" (psychological, immaterial, private) reality. Consider two pioneers of psychological research, Wundt (1874) and Watson (1913; 1914). Wundt investigated what happens inside a person's mind by interrogating well-trained subjects whereas Watson used animal models to study observable behavior, each choosing one side of the split between "external" and "internal". If in order to understand the behavior of organisms it is necessary to consider both their inner world and their external environment, constructing artificial organisms might make this possible.

The simultaneous consideration of both the internal and external side of behavior is particular relevant if we wish to study anticipatory behavior. In order to produce an anticipatory action, organisms should possess some knowledge concerning the environment in which they live. It is usually assumed that this knowledge is founded upon integrating innate schemes with sensory experience, where sensory experience is provided by environmental stimuli (light, sound, smell, etc.) and is received by the organism's sensory apparatus, including eyes, ears, nose. However, in addition to the external environment the organism's body itself is a precious source of stimuli (Parisi, 2004). Examples are internal clocks, proprioception, signals from the gastroenteric apparatus, the hormonal system, etc. This stimulation is considered relevant to regulate the organism's behavior but not to build up a "knowledge" of the external environment. For example, hunger can motivate an organism to choose a certain action to satisfy this need but it is not useful to construct a representation of the environment in which the organism lives.

Some authors have underlined that knowledge is not a registration of what happens in the external world. For example Maturana and Varela (1992) have stressed the fundamental importance of the "inner world" to build neuro-cognitive functions such as memory that allow organisms to survive in an unpredictable world. Human experience is fallacious, as it is shown by the blind spot of the optic nerve in the retina. If one looks at a fixed point and then he or she moves the gaze, the point disappears from the visual field, showing a blind spot in the retina in the area of departure of the optic nerve.

Beyond the physiological relevance of this phenomenon, it is interesting that humans *do not see that they do not see*: human beings are tempted by an illusion of certitude, by the inclination to think that they live in a world in which things are what they seem to be, without considering alternatives. On the contrary, neuronal activations which are primed by external stimulation are determined by what is inside the person and not only by the perturbing agent, and are therefore different from person to person. For this reason we do not "see the world" but we "live our visual field". We do not see the colors of the world but we live our chromatic space: every experience involves the experiencer and it is deeply rooted in his/her individual biological structure. An organism's knowledge of the world is not a representation of the world "out there" but is the production of "sense for action".

These challenging issues raise numerous and important questions and we have tried to answer some of them using an Artificial Life approach. We aim at investigating the possible role of internal stimulation in building a knowledge of the environment in which an organism lives and from which it receives no stimulation at all. We will explain in detail our approach with a concrete example. Let's imagine an organism with a motor apparatus and an internal sensory apparatus, but totally without sensory organs that directly inform the organism concerning the current state of the external environment. The organism is completely closed inside itself at the sensory level. It can interact with the external environment but it cannot get any direct information from the external environ-

ment. The organism is forced to create its own inner world on the basis of purely self-generated stimuli. But the organism cannot be said to be isolated from the external environment in that its actions have effects that modify the physical relation of the organism to the external environment and that the organism can exploit to behave adaptively in the environment. In nature there may exist no such organisms but the tools of Artificial Life can be used to explore this type of questions because Artificial Life is the study of both real and *possible* organisms (Langton, 1989). In the next paragraph we will describe our approach, summarize some similar studied that have been already done, and make our research hypothesis explicit.

1.1 The Agent/Environment Dynamic in the Animat Approach

In the last 15 years a research methodology that uses simulation of artificial organisms to understand cognition in real organisms has carved out a space for itself inside Cognitive Science. Using this methodology, often called the Animat approach (Todd, 1992; Guillot and Meyer, 1994), it is possible to study how cognition emerges in the interaction between the agent, that is to say the artificial animal, and the environment, as represented in Figure 1.

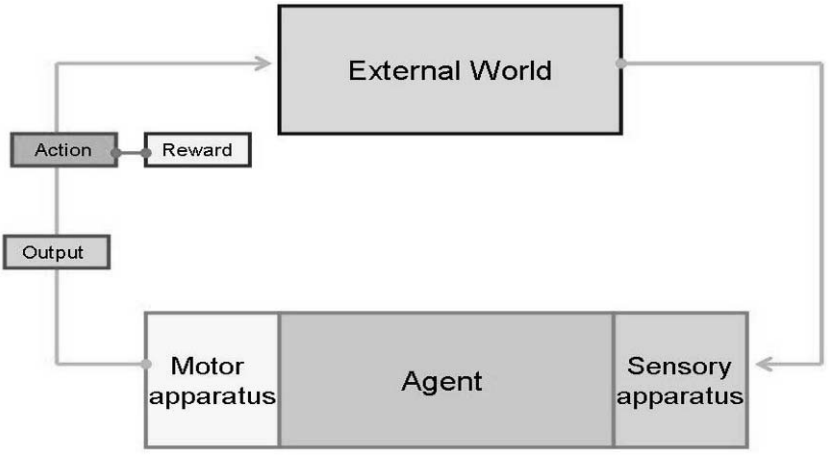


Fig. 1. Environment/Agent dynamic in the Animat approach

The agent receives stimulation from the environment and reacts consequently. The environment, in turn, reinforces the agent, as stressed by Wilson (1991). The reinforcement can be given at different timescales. In embedded systems that use the back-propagation learning algorithm, for any input from the environment the experimenter provides the correct motor output to the artificial neural network controlling the organism's behavior. In this case the reinforcement is immediate and provided step by step. In the framework of artificial evolution (Nolfi and Floreano, 2000; Harvey et al., 1996), the reinforcement is given at the end of

an individual's lifetime as a fitness score which decides whether the individual will or will not reproduce. The two systems can be combined together with both artificial evolution at the population level and back-propagation learning at the individual level (Parisi et al., 1990).

What happens if the agent doesn't receive any stimulation from the environment? This is interesting because natural organisms receive information from the external as well from the internal world whereas Animat models usually focus on environmental stimulation.

In the present paper we use a modified version of the environment/agent dynamic in the Animat approach. We eliminate any stimulation that the agent may receive from the environment in order to address explicitly the issue of how our artificial organisms can *produce their own world* by just relying on its their "inner world", as shown in Figure 2.

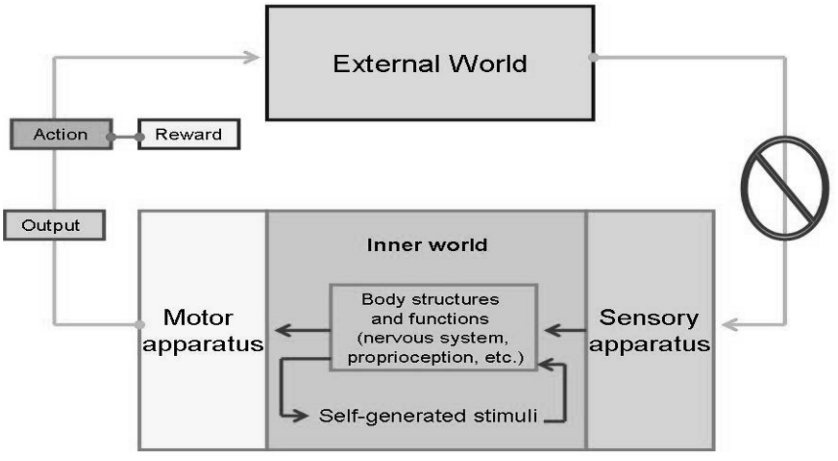


Fig. 2. Environment/Agent dynamic in the Animat approach: the inner world

As shown in the above Figure we modify the agent/environment interaction by removing the sensory link between the agent and the environment. The agent cannot rely on its sensory apparatus to decide which action to take but it can still use its motor apparatus to produce actions that are not rewarded step by step but at the end of the evolutionary process.

1.2 Some Seminal Studies

The research presented here draws inspiration from some seminal studies that we quickly review here in order to underlie what is different in our study.

In 1994 Todd and colleagues published an experiment in which an adaptive, survival-enhancing behavior emerged in simple simulated creatures which had no direct sensory contact with their environment. They described the evolution

of the behavioral repertoires of these sensor-less creatures in response to environments with different spatial and temporal distributions of food. The main difference with our study lies in the theoretical goals. They explored the level of adaptiveness in these blind creatures to establish a baseline with which the adaptive behavior of Animats with sensors and internal states could be compared, whereas we wish to understand how the internal world can come in resonance with the external world to produce adaptation.

In this respect it is appropriate to report Ziemke and colleagues' work (cfr. Ziemke et al., 2005; Hesslow and Jirenhed, 2007, Jirenhed et al., 2001) who explored the possibility of *providing robots with an "inner world" based on internal simulation of perception rather than an explicit representation world model*. Starting from a neuroscientific hypothesis they studied how internal simulation of perception can be used by mobile robots to respond appropriately to their environment, presenting various experiments with a simulated robot controlled by a recurrent neural network shaped by an evolutionary algorithm. Their work suggests that internal simulation of perception may be sufficient to adapt. We start from a very similar research hypothesis and we use a very similar setup but our agents do not have to explicitly simulate their perception.

The most recent paper we cite here is by Lungarella and Sporns (2006). In their work they start from the idea that organisms continuously "select and sample information used by their neural structures for perception and action, and for creating coherent cognitive states guiding their autonomous behavior" (ibidem). They stress that information processing is not solely an internal function of the nervous system, but instead sensorimotor interaction and body morphology can act as constraints that create statistical regularities in sensory input which allow the emergence of adaptive behavior. Their paper is important for interpreting our results since we also wish to understand how internal and external worlds can coordinate and generate regularities that can result in adaptive behaviors.

In the next section we will formulate more explicitly our research questions.

Our research hypothesis. In the present paper we continue to explore the issues that have been raised by the studies reviewed above, trying to answer the following questions: Can internal stimulation be sufficient to solve a spatial task? Is there any difference with respects to Animats that rely on stimulation? Which mechanism do our organisms use to adapt?

We have simulated a spatial behavior with two questions in mind:

1. Can robots exhibit adaptive spatial behaviors that rely only on internal stimulation?
2. If yes, how is that possible?

Asking these questions may be important because the possible answers may clarify how at an evolutionary scale anticipation can lead to adaptation even in the extreme case of an organism with no direct sensory feedback from the external world.

Our agents, even if they lack stimuli, can rely on two channels to get in touch with the environment: action and "evolutionary" reinforcement, that is,

the selective reproduction of the best individuals in a succession of generations. In the simulations that we will describe the robots are able to move adaptively in their environment by implicitly predicting the sensory input and by just relying on their predictions to generate their behavior, and not on actual sensory input coming from the outside environment.

2 Method

In our experiments we use the Evorobot simulator (Nolfi, 2000) to evolve populations of artificial organisms (software robots) with the ability to solve a spatial task: exploring a square arena by visiting as many portions of the arena as possible. The simulator, developed by Stefano Nolfi at ISTC-Cnr, makes it possible to run Evolutionary Robotics simulations that can then be transferred on real robots.

2.1 Artificial Organisms

Each artificial organism consists of a physically accurate simulation of a robot with a circular body of 5.5 cm of diameter, which is a model of the E-puck robot developed at EPFL, Switzerland (www.e-puck.org)(Fig. 3).

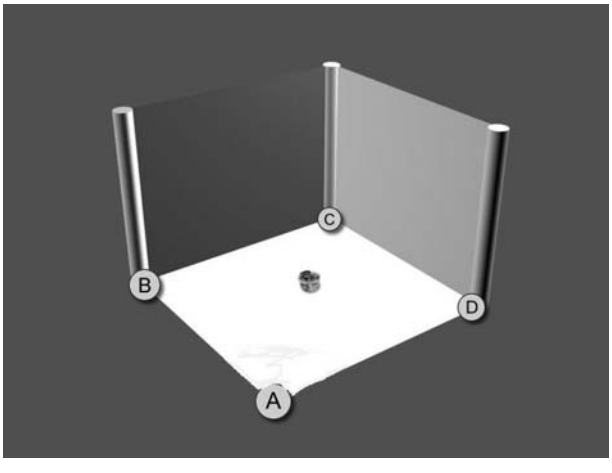


Fig. 3. The E-puck robot model in the 3D simulated environment

Each robot is equipped with 8 infrared proximity sensors (that can detect objects within 3 cm of the sensor) and a black/white linear camera with a receptive field of 100 degrees whose content is encoded in 8 input units. The robot displaces itself by using 2 wheels (one on each side of the robot) powered by separate, independently controlled motors. The control system is an Artificial Neural Network. We use neural architectures with two properties: the existence of recursive connections and the nature of the sensory input. All neural

networks have an output layer with two units that control the robot's two wheels. In all neural networks, furthermore, there are five internal units which are all connected to both output units. However, in different simulations the robots have neural architectures that can be different with respect to sensory input and internal structure. There are four conditions of sensory input: no sensory input, infrared sensory input, visual (camera) input, both infrared and visual input. There are also two internal architectures: no recursive connections and recursive connections (from the output units to the internal units and Elman "memories" for the internal units), for eight conditions in total which are shown in Figure 4. At time t0, neural networks with no sensory input have an input pattern of 1 for each hidden unit that has a different, evolvable threshold.

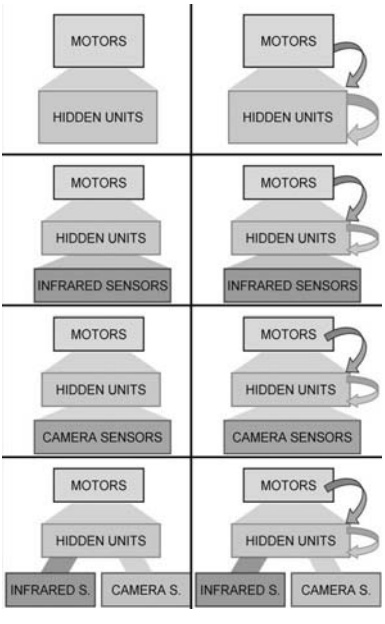


Fig. 4. The neural architectures. In the graph are shown the eight different internal architectures. In four of these architectures there are no recursive connections while the recursive connections are present in the remaining four architectures. Each of these internal architectures is associated with one of four possible sensory input conditions: no sensory input, infrared sensory input, visual (camera) input, both infrared and visual input. Therefore we have a total of eight different experimental conditions.

2.2 The Task and Training Procedure

A Genetic Algorithm is used to train the connection weights of all network architectures. At the beginning of each simulation, we create 100 neural networks with random connection weights that are assigned to 100 robots. We then test each robot's ability to solve the exploration task. Each robot is positioned at

the centre of a square arena with peripheral walls and four lights placed at the four corners of the arena. If the robot happens to hit the walls, the robot dies. At the beginning of each trial the robot is positioned in the arena with a randomly chosen face-direction and is allowed to move around for 500 computation cycles (1 ms per cycle). For analysis's sake we consider the square arena, which is actually continuous, as made up of cells that cover the entire square area (40x40 cells). Each time a robot visits a cell it has not visited before its fitness is increased by one unit. At the end of life the 80 robots with the lowest fitness are eliminated (truncation selection) while the remaining 20 robots are cloned (asexual reproduction). Each parent generates five offspring, and a value randomly chosen from the uniform distribution [-1, +1] is added to 2 per cent of the offspring's connection weights. We run eight different experiments with "recursive" vs. "non recursive" conditions and four "sensory" conditions (see Figure 4). Each experiment is repeated 10 times with different initial conditions (different randomly generated connection weights for the neural networks of the individuals of the first generation).

3 Results

3.1 Fitness Values

The next graph (Fig. 5) shows the fitness values, that is, the number of cells that are visited for the first time by the simulated robot, for each of our eight

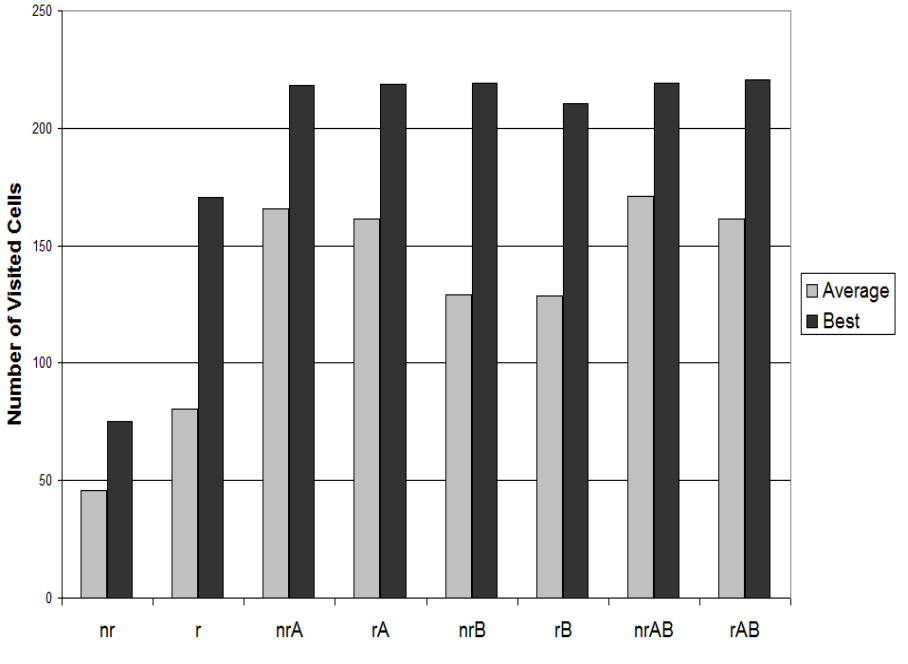


Fig. 5. Fitness values (visited cells) for each simulation. See text for explanation

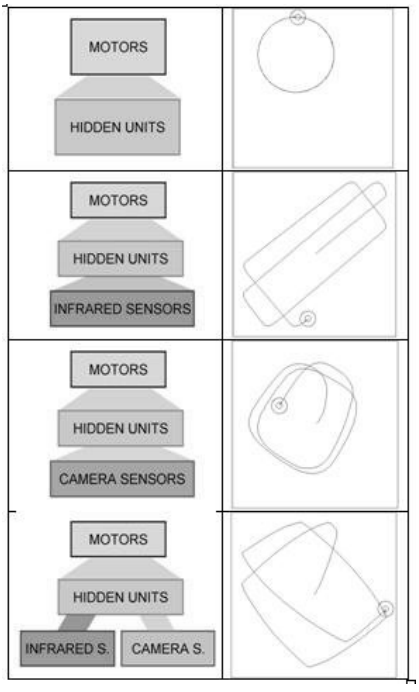


Fig. 6. Behavioral strategy of the best robots in nonrecursive condition and their neural architectures (on the left)

experiments. Data refer to the fitness values calculated on the last 10 generations (on 100 total generations) for 10 repetitions (these are therefore aggregated data calculated on 100 values). Each bar in the histogram represents the average for a single simulation. Grey bars refer to average values while black bars refer to best values. To identify each simulation we use this code: r means recursive condition while nr means non recursive condition. Neural architectures *without* recursive condition are shown on the left columns in Figure 4 while neural architectures *with* recursive condition are shown on the right columns. Letters indicate sensory input conditions where no letter means no external stimuli, A means 8 infrared sensors, B means 8 units for the camera, and AB means 8 infrared sensors plus 8 units for the camera.

As can be seen from the graph, the fitness values of the best subjects of the last 10 generations suggest that recursive connections are beneficial for robots with no sensory information. In the graph, in fact, considering the best values for each simulation we can see that in the experimental conditions without sensory input and with recurrent connections as many cells are visited by the organisms as in the conditions with sensory input, while in absence of recurrent connections the performance of the organisms without sensory input is very bad.

Considering the best robots for r (no sensory input and recurrent connections) and related condition rA, rB, and rAB and applying a one-way ANOVA, we see

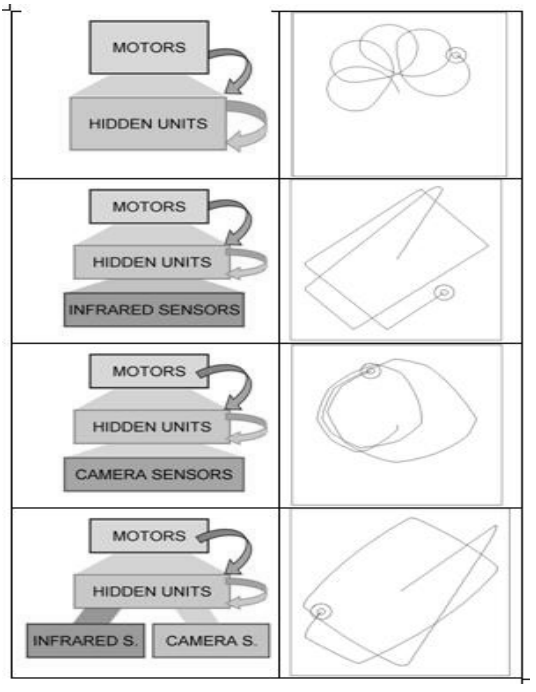


Fig. 7. Behavioral strategy of the best robots in recursive condition and their neural architectures (on the left)

that there are no significant differences in the number of visited cells with sensory condition as factor: $F(3,36)= 2.667$; $p= 0.062$.

The important role played by recursive connections depends on the fact that they allow to overcome stereotypical behavior in favor of more variable behaviors, as we will see in the next section. Therefore we have an interesting answer to our first question: yes, it is possible to observe adaptive behaviors in robots with only internal stimulation under certain conditions, namely the presence of recursive connections that allow the robot to build an internal dynamic, an "inner world" in Ziemke and colleagues' words, which is coupled with the environment. They succeed in coordinating endogenous stimuli with the constraints of the external environment and to use this coordination to generate a motor behavior which is adaptive.

3.2 Behaviors

In Figures 6 and 7 behavioral strategies of the best robot of each simulation with and without sensory input are represented together with their neural architecture. In absence of recurrent connections, robots can produce only stereotypical behaviors: they draw a circular trajectory because a circular trajectory of the appropriate radius allows them to avoid hitting the wall and die.

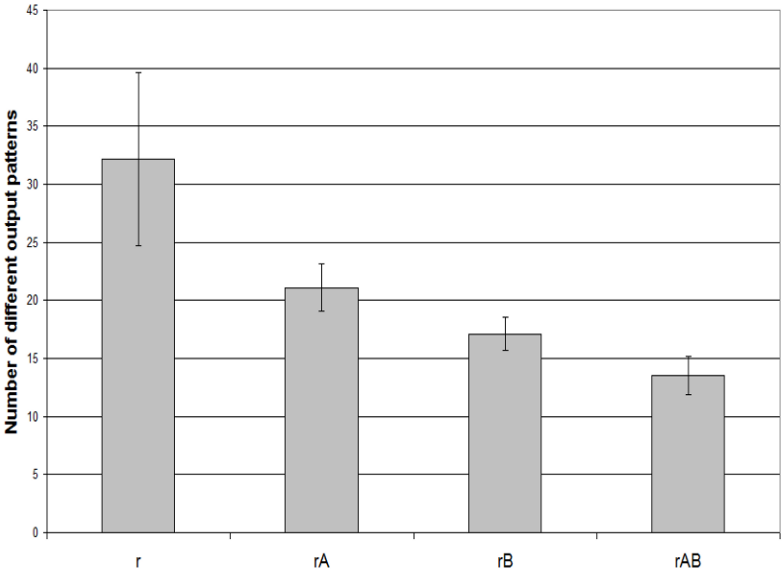


Fig. 8. Number of output patterns in different sensory conditions. Bars represent means, lines represent standard deviation. We use the same code as before (r means recursive condition while nr means non recursive condition). Letters indicate sensory input conditions where no letter means no external stimuli, A means 8 infrared sensors, B means 8 units for the camera and AB means 8 infrared sensors plus 8 units for the camera.

In fact it is worth noting that this circle has a radius that depends on the size of the square arena. This is an efficient strategy, considering that the robots has no access to external information and must avoid bumping into walls which they cannot perceive. However, when the robots have recurrent connections their performance increases dramatically: their recurrent connections generate an internal dynamic that, under evolutionary pressure, tends to be tuned with the constraints of the external environment. We observe more variable trajectories which are always curved but which lead the robots to visit many more cells of the arena.

This behavioral strategy is quite different from what we observe in robots with sensory input, because, in this case, the best thing to do is to go straight until they perceive a wall and to turn before bumping into it. In absence of stimulation walls become a constraint, because they must be avoided, and this leads to the emergence of behavioral strategies which explore the cells that are far enough from the walls.

3.3 Output Patterns

In order to answer our second question and investigate the role of the inner world in producing appropriate behavioral sequences we examined the activation

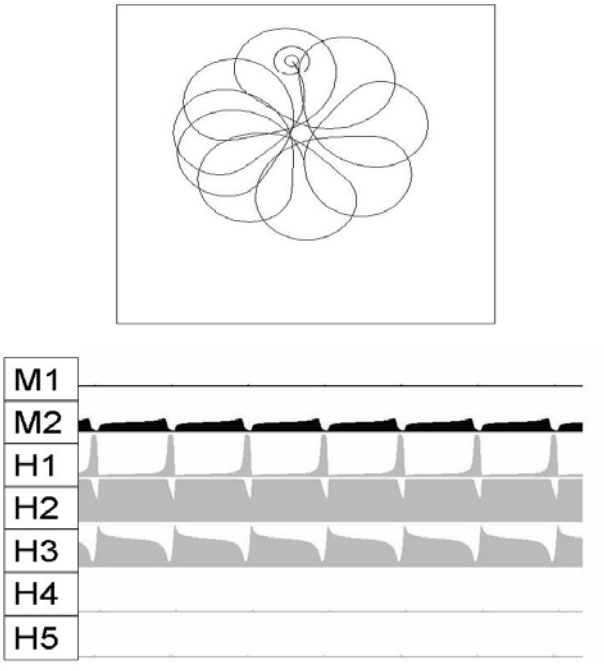


Fig. 9. The behavior (up) and the corresponding neural activation (down) of one of the best performing sensor-less robot

patterns of the output layer for the single best agent of the last generation in each replication of the simulations. The continuous activation value of the two output units is made discrete when it is transferred to the two wheels, with values going from -20 to 20 for each motor. Therefore we can count the number of different output patterns, which is a measure of the variability or variety of the robots' micro-behaviors. We focus on the recursive condition, as in absence of recursion performance without external stimuli is very poor. The results indicate that a larger number of different micro-behaviors (number of different output patterns) are necessary to solve the task with self-generated stimulation. There are significant differences between the experimental Conditions, with a decreasing mean number of output patterns going from the condition with IR sensors + camera to the condition without stimulation, as shown in Figure 8.

Comparing for example the best robots of the last generation for condition r (no sensory input and recurrent connection) and related condition rA, rB and rAB and applying a one-way ANOVA, we see that there are significant differences in the number of output patterns with sensory condition as factor: $F(3,36)= 40.606$; $p= 0.00$. In absence of variable external stimulation, the robots create their own internal variable stimulation. Robots without any external stimulation are able to accomplish the task to explore effectively the square arena if they are provided with recursive connections that create an internal dynamic which in turn produces a more varied behavior.

3.4 Hidden Units Activation

To further answer our questions, we have also analyzed what kind of internal dynamic emerges in robots with no access to external information. We have observed that, when robots display an efficient behavior, a kind of oscillator emerges in their hidden unit activation. Let us consider, for example, one of the most interesting behavioral strategy and the corresponding neural activation of motors and hidden units that was displayed by the best robot in the recursive condition r without stimulation. Both behavior and neural activation are shown in Figure 9.

As can be seen from the Figure, this very efficient sensor-less strategy results from the emergence of an internal oscillator that provides an internal temporal dynamic. It is important to underline that this temporal dynamic is strictly coupled with the spatial constraints so as to make it possible to avoid bumping on the wall while still visiting as many cells as possible. The spatial constraints that are present in the environment are translated into a time rhythm in the "inner world".

4 Conclusions

The results of our simulations suggest that, at least for the simple artificial organisms studied in our research, an adaptive behavior can indeed emerge even in absence of direct sensory information from the external environment. Even if they are closed in their own self-generated internal world, the simulated robots establish a useful relation with the external environment through their action. In fact, by realizing and exploiting a precise coordination between produced output and self-generated internal input, i.e., between the external and the internal worlds, the robots are able to successfully adapt to their environment. This is possible because action is accurately selected under evolutionary pressure, and the evolutionary pressure causes the emergence of a kind of resonance between inner world and external world. Through the physical interactions between the organism and the environment, after a demanding search the possibility emerges to utilize action to know the environment, even if there is no sensory input from the environment. In other words, the organism's actions become the vehicle for developing a representation of the environment.

Our sensor-less organisms, which cannot sense directly the external environment, are in fact not isolated from the external environment because action is able to establish a link with the external world. This is an "operationally closed system" in Maturana and Varela's sense. Our organisms, in fact, are provided with a motor apparatus and an internal dynamic but they completely lack sensors that can collect information on what is "out there". They can interact with the external world but they cannot receive direct information from the external world.

A system like this, since it cannot react to external stimulation, is forced to build an internal model of the world on the basis of self-generated, internal, private stimulation. The system is not actually isolated from the external world

because it has an opportunity to act on the environment in such a way that it becomes possible to develop a relation between the system and the environment, a relation which is not sensory but is behavior-based. In other words, the organism cannot receive external stimuli but it can collect clues about the external environment through action and these clues give the organism the possibility to establish an useful system/environment interaction.

This interaction allows the simulated robot to anticipate what is going to happen. Even in absence of sensory stimulation the robot is able to avoid bumping into the walls, anticipating the inauspicious event that would put an end to its life. This anticipation ability emerges in the neural network in that the network's hidden units autonomously develop an oscillator that provides an internal temporal dynamic. This primordial form of anticipation resides in the neural network's rhythm which is strictly coupled with the spatial properties of the environment and the robot's current location and orientation in the environment.

Under evolutionary pressures the agents' neural control architecture extracts and incorporates the statistical regularities and information structure underlying their interactions with the environment, and in this way the external constraints and the internal dynamic come in resonance. The flow of information between the hidden units and the robot's effectors is actively shaped by the robot's interactions with the environment on an evolutionary scale. These results confirm the fundamental importance of embodiment and situatedness in the behavior of organisms.

Our computer simulations demonstrate how the behavior of artificial organisms can reflect the particular characteristics of the environment in which the organisms live. The behavior that emerges can be adaptive with respect to the environment even if the organisms obtain extremely little information from the environment through their sensors, or no information at all. Of course we observe very peculiar behavioral strategies in our cognitively challenged, sensor-less creatures, including the use of looping movements as time-keepers. Our simulations show how the ability to explore the environment can emerge from the interaction, made possible by action, between two coupled processes: the agent's internal dynamic and the agent/environment dynamic.

From the point of view of autopoiesis the results of our simulations suggest that at least for our simple artificial organisms internal stimulation can by itself generate adaptive behaviours and can be the building block of an internal world that produces adaptation to the external world.

One possible objection to our conclusion might be that real organisms hugely rely on external stimulation to adapt to their environment. This is undoubtedly true but we think that our simulations demonstrate that external stimulation is only one of the information sources that make it possible to build a representation of the world. In principle, and in extreme cases, internal self-generated stimulation may be sufficient. The "in" is as important as the "out"; they work together in the process of "producing of world".

This implies that real organisms, endowed with cognitive systems with very complex dynamics, may exploit their internal dynamics to anticipate the future

and on the basis of these anticipations generate useful behaviors. These results also confirm that the brain is a self-referential recursive machine whose organization is maintained in spite of environmental perturbations, even if it is triggered by them.

Our simulated agent "knows" the external world in that its control system is able to map the spatial structure of the environment (obstacle position, reinforcement area, etc.) in self-produced temporal structures (internal rhythms). Spatial regularities in the outside world and temporal regularities in the "inner" world come into resonance and this happens thanks to the agent's actions in the environment.

As already discussed in the introduction our work is inspirited by previous studies and it tries to extends some of their results. With respect to Todd's simulations (1994) our work proposes a mechanism to explain how the internal world can come into resonance with the external world to produce adaptation. What we have found is that one type of spatial regularity (the square walled arena) is translated into an internal representation based on time. In relation to Ziemke and colleagues' work (cfr. Ziemke et al., 2005; Hesslow and Jirenhed, 2007, Jirenhed et al., 2001) we start from a very similar research hypothesis and use a very similar set-up with a minimal animal model, but our agents do not have to explicitly simulate their perception. Rather than simulating the percepts that they cannot obtain from the external world, our agents are forced to build an independent internal dynamic that is coupled with the external constraints.

In our future research our goal is to extend the types of tasks that our organisms have to accomplish and to use other neural architectures with and without recurrence in order to understand what are the best architectures, and why.

Acknowledgement

Funding was provided by CNR in the framework of the programme "Cooperation in Corvids" (COCOR, which forms part of the ESF-EUROCORES programme "The Evolution of Cooperation and Trading" (TECT).

References

1. Brandimonte, M., Einstein, G.O., McDaniel, M.A.: Prospective memory: Theory and application. Erlbaum, Mahwah (1996)
2. Guillot, A., Meyer, J.A.: Computer simulations of adaptive behavior in animats. In: Proceedings Computer Animation 1994. IEEE Computer Society, Los Alamitos (1994)
3. Grush, R.: The emulation theory of representation: Motor control, imagery, and perception. Behavioral and Brain Sciences 27, 377–442 (2004)
4. Harvey, I., Husbands, P., Cliff, D., Thompson, A., Jakobi, N.: Evolutionary Robotics at Sussex. In: Proceedings of ISRAM 1996, International Symposium on Robotics and Manufacturing, Montpellier, France, May 27-30 (1996)
5. Hesslow, G.: Conscious thought as simulation of behaviour and perception. Trends in Cognitive Sciences 6(6), 242–247 (2002)

6. Hesslow, G., Jirenhed, D.-A.: The Inner World of a Simple Robot. Journal of Consciousness Studies 14-7, 85-96(12) (2007)
7. Jirenhed, D.-A., Hesslow, G., Ziemke, T.: Exploring internal simulation of perception in a mobile robot. Lund Univ. Cogn. Stud. 86, 107–113 (2001)
8. Langton, C.: Artificial Life. In: Langton, C. (ed.) Artificial Life, pp. 1–47. Addison-Wesley, Reading (1989)
9. Lungarella, M., Sporns, O.: Mapping information flow in sensorimotor networks. PLoS Comput. Biol. 2(10), e144 (2006)
10. Maturana, H.R., Varela, F.J.: Autopoiesis and Cognition. Reidel, Boston (1980)
11. Maturana, H.R., Varela, F.J.: The tree of knowledge. In: The biological roots of human understanding. Shambhala, Boston (1992)
12. Nolfi, S.: Evorobot 1.1 User Manual. Rome: Institute of Psychology, CNR (2000)
13. Nolfi, S., Floreano, D.: Evolutionary Robotics: The Biology, Intelligence, and Technology of Self-Organizing Machines. MIT Press/Bradford Books, Cambridge (2000)
14. Parisi, D.: Internal Robotics. Connection Science 16(4), 325–338 (2004)
15. Parisi, D., Cecconi, F., Nolfi, S.: ECONETS: neural networks that learn in an environment. Network 1, 149–168 (1990)
16. Piaget, J.: Biology and Knowledge. University of Chicago Press, Chicago (1971)
17. Todd, P.M.: The animat approach to intelligent behavior. Computer 25(11), 78–81 (1992)
18. Todd, P., Wilson, S., Somayaji, A., Yanco, H.: The blind breeding the blind: Adaptive behavior without looking. In: Cliff, D., Husbands, P., Meyer, J.-A., Wilson, S.W. (eds.) From Animals to Animats:The Third International Conference on Simulation of Adaptive Behavior, pp. 228–237. MIT Press, Cambridge (1994)
19. von Foerster, H.: What is Memory that It May Have Hindsight and Foresight as well? In: Bogoch (Hg.) The Future of the Brain Sciences, Proceedings of a Conference held at the New York Academy, pp. 19–64. Plenum Press, New York (1969)
20. Wundt, W.: Grundzuge der Physiologicischen Psychologie. Verlag von Engelmann, Leipzig (1874)
21. Watson, J.B.: Psychology as the behaviorist views it. Psychological Review 20, 158–177 (1913)
22. Watson, J.B.: Behavior: An introduction to comparative psychology. Holt, New York (1914)
23. Wilson, S.W.: The animat path to AI. In: Meyer, J.-A., Wilson, S.W. (eds.) From animal to animats: Proceedings of the First International Conference on Simulation of Adaptive Behavior. MIT Press/Bradford Books, Cambridge (1991)
24. Ziemke, T.: Cybernetics and embodied cognition: on the construction of realities in organisms and robots. Kybernetes 34(1/2), 118–128 (2005)
25. Ziemke, T., Jirenhed, D.-A., Hesslow, G.: Internal simulation of perception: a minimal neuro-robotic model. Neurocomputing 68, 85–104 (2005)
26. Ziemke, T.: The Embodied Self - Theories, hunches and robot models. Journal of Consciousness Studies 14(7), 167–179 (2007)
27. Ziemke, T.: On the role of emotion in biological and robotic autonomy. BioSystems 91, 401–408 (2008)

The Cognitive Body: From Dynamic Modulation to Anticipation

Alberto Montebelli, Robert Lowe, and Tom Ziemke

University of Skövde
School of Humanities and Informatics
SE-541 28 Skövde, Sweden
{alberto.montebelli,robert.lowe,tom.ziemke}@his.se

Abstract. Starting from the situated and embodied perspective on the study of cognition as a source of inspiration, this paper programmatically outlines a path towards an experimental exploration of the role of the body in a minimal anticipatory cognitive architecture. Cognition is here conceived and synthetically analyzed as a broadly extended and distributed dynamic process emerging from the interplay between a body, a nervous system and their environment. Firstly, we show how a non-neural internal state, crucially characterized by slowly changing dynamics, can modulate the activity of a simple neurocontroller. The result, emergent from the use of a standard evolutionary robotic simulation, is a self-organized, dynamic action selection mechanism, effectively operating in a context dependent way. Secondly, we show how these characteristics can be exploited by a novel minimalist anticipatory cognitive architecture. Rather than a direct causal connection between the anticipation process and the selection of the appropriate behavior, it implements a model for dynamic anticipation that operates via bodily mediation (*bodily-anticipation hypothesis*). This allows the system to swiftly scale up to more complex tasks never experienced before, achieving flexible and robust behavior with minimal adaptive cost.

1 Introduction: A Cognitive Body

While there is much agreement that cognition is *embodied*, it remains less clear exactly what role(s) the body plays in cognitive processes. The obvious common-sense answer would highlight the role of the body in dictating the physical relation between an agent and its environment. Of course the fundamental function that the body of any organism plays - the appropriate adaptive situating of its available set of sensors and actuators in a spatio-temporal frame of reference - cannot be missed or ignored. However, this common-sense interpretation is broadened in embodied and situated approaches to the study of cognition, both at theoretical and experimental levels [1,2,3,4,5]. The body shapes the cognitive potential of the agent by completely specifying the nature and range of all possible interactions with its environment. A self-organized agent typically depends on and deeply exploits such constraints [6].

G. Pezzulo et al. (Eds.): ABiALS 2008, LNAI 5499, pp. 132–151, 2009.

The basic idea of a highly systemic approach to the study of cognition was already centrally present in the work of early cyberneticists (i.e. [7,8]), gestalt psychology [9] and ecological psychology [10]. The sudden rise of cognitive science cast a shadow on such historically prominent intellectual work. This should not come as a surprise: apparently, large bodies of sometimes outstanding scientific knowledge are destined to be reconsidered, or even completely rediscovered, over and over, whenever there is an intellectual need for them. Presently massive research efforts investigate the problem of understanding cognition by a systematic decomposition. In the scientific tradition, reductionism proved powerfully effective in producing sound explanations (with predictive power) of natural phenomena. Nevertheless, it is wrong to infer that all explanations are reductionistic. Such a misconception might be particularly pernicious in an epoch where the scientific community masters, and has large availability of, the necessary technology to engage in the exploration of the problem of non-linear complexity, and is intellectually committed to the development of the appropriate mathematics to start addressing it. That is, dynamic systems theory offers a natural language to a systemic approach to the study of adaptive behavior and cognition [11]. Much work has already clarified the need for a consistent deployment of the existing mathematical tools and for their further development [12,2,13,14].

Nowadays, a more systemic view of the mind pervades at least a few major theoretical frameworks in the study of cognitive processes. Several authors are currently committed to the underpinning of a theoretical background, in which the specific embodiment of an organism has non-trivial cognitive consequences. The body massively pre-/post-processes the information flow to and from the nervous system, and the common evolutionary history and ontogenesis of body and nervous system provides a deep, distributed integration of bodily and nervous functions (e.g. see [15]). Perception and action are not causally sequential activities, but can be seen as closely interrelated and in fact inseparable, one supporting the other [16,6].

Nevertheless, we have reason to think that this perspective does not go far enough. Rather than treating the body as a mere interface to the world, we should also take into account what happens inside the body of an organism, and its potential cognitive consequences [17,18,19,20,21]. We find that the hidden, bio-regulatory dynamics developing under the surface of the body are largely neglected in the study of cognitive phenomena. As some authors put it, the interaction between bio-regulatory events that take place inside the body of an agent and what is traditionally interpreted as its control system, might be a crucial component of its ongoing cognitive processes [17]. In line with this thread, in this paper we describe our current experimental work in cognitive robotics, focusing on the role that the intrinsic non-neural bodily dynamics might play in supporting and boosting cognition. In Section 2, we discuss some preliminary results showing how a non-neural internal state can modulate the activity of a simple neurocontroller. We then formulate the programmatic foundations for an extension of our work towards a bodily-mediated anticipatory cognitive architecture. Firstly, on theoretical grounds, we advocate that non-neural bodily

dynamics might play a fundamental role in a new anticipatory cognitive architecture (Section 4). Secondly and more concretely, we report on the initial experimental analysis of this idea (Section 5). Then, we briefly comment on the theoretical implications of our work and on necessary future developments (Sections 6 and 7).

2 The Dynamic Role of the Cognitive Body: A Minimalist Case Study

In a recent study [22,23], we have shown how even very simple non-neural bodily states can play a crucial role in the modulation of the activity of an artificial nervous system, i.e. on the behavior generated by an artificial neural network (ANN) implementing the neurocontroller of our simulated robotic agent. We used standard evolutionary algorithms to set the weights and biases for a simple reactive ANN with no hidden layers, driving the motoneurons of a simulated Khepera robot (see Figure 1). The system self-organized in order to find a recharger for its energy level (i.e. each instantiation of ANN during the evolution was simply rewarded for the maintenance of a positive level of energy, punished otherwise), thus overcoming its temporal linear energy decay. The invisible recharger was placed in a circular area centered under one of the two visually identical light sources, randomly selected for each replication. An energy level sensor, together with a battery of light and infra-red sensors constituted the sensory inputs to the ANN.

As part of the analysis of the successfully evolved system[1], we manipulated the energy level as the control parameter for the whole system [12,24]. By systematically clamping[2] it to a discrete set of possible values, ranging from zero to 'full', we observed and classified a number of possible behaviors, exemplified in Figure 2- left. After exhaustion of the behavioral transients, we found three general classes of qualitative behavioral attractors. We observed: exploratory behaviors at the lowest levels of energy, i.e. the agent engaged in loops between potential energy sources and also in external loops broadening its explorations to the rest of the environment (i.e. see trajectory in panel 'A' of Figure 2); more local behaviors at higher levels of energy (i.e. the agent was closely looping in the neighborhood of a single source as in panel 'C'; hybrid behaviors, embedding characteristics from both previous classes (as in panel 'B') for intermediate levels of energy. The relative frequency of the three groups of behaviors was reliably dependent on the current energy level (Figure 2- right).

[1] By the term 'system', here and in what follows, we refer not just to the evolved ANN, but to the global system constituted also by the agent, its environment and its non-neural dynamic mechanism of the energy level. Therefore, cognition is here conceived and analyzed as a broadly distributed process; a cognitive aggregate, rather than a localized and proprietary process.

[2] The term 'clamping' here refers to the injection of a constant energy level as input for the ANN during the whole duration of the replication of the experiment. The agent is free to behave in its environment for a period of time sufficient to exhaust all behavioral transients and permit observations of satisfactory duration.

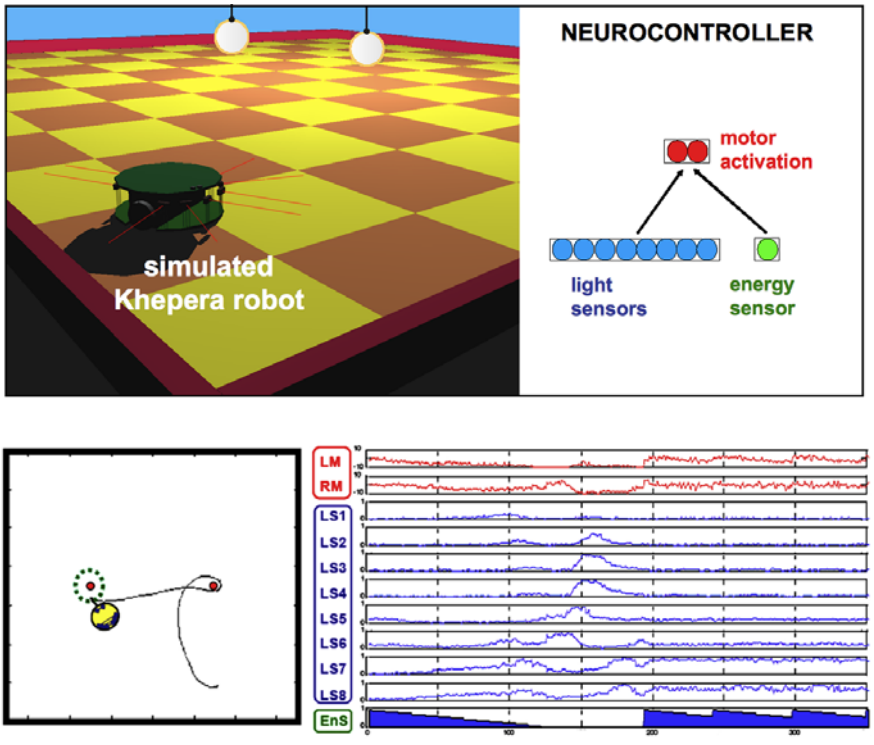


Fig. 1. top - Experimental setup. A simulated Khepera robot moves in a square environment containing two visually identical light sources (suspended white light bulbs). Its neurocontroller is a simple reactive ANN with no hidden layers, directly controlling the two motors of the robot and receiving information from the light sensors and from an energy level mechanism. The choice of such a simple scenario aimed to facilitate our analysis and emphasize the object of study. **bottom -** Example of the evolved behavior. As its energy level sensor measures the temporal linear decay (graph labeled EnS), the simulated agent (large cylinder) approaches the light to the right (filled circle). The neutral effect on its energy level determines the approach of the next light. The recharging area (dashed circle) is invisible to the robot, that can sense it only by virtue of its effect on the energy level sensor. As its energy reservoir is instantaneously refilled to the maximum level, the agent is engaged in a stable behavior in the proximity of the rewarding light source. The signals labeled LM and RM show, as a function of time, the activation of the left and right motors; LS1-8 represent the activation of the light sensors.

Regarding the evolutionary task, we then examined the implications of the behaviors that we observed in clamped conditions. As the energy level is left free to follow its natural dynamics, it constitutes an effective self-organized and dynamic action selection mechanism. Different classes of behaviors are locally available to the agent as a function of its current energy level. Apparently, high energy

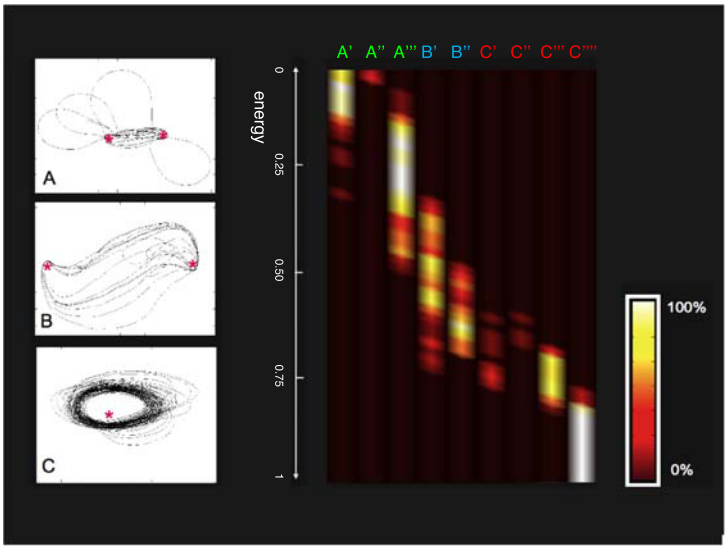


Fig. 2. left- Sample spatial trajectories for the three classes of behaviors observed in clamped conditions after transient exhaustion. Exploratory behaviors (panel A), local behaviors (panel C) and hybrid forms (panel B). Potential energy rechargers (i.e. the position of the light sources) are indicated by red stars. For a better resolution of details, the icons representing each class of trajectories zoom on the area of main interest surrounding the light sources. **right-** The intensity of the pixels for each column (corresponding to attractors belonging to classes A-C, as specified by their labels on the top row) represents the relative frequency of the behavioral attractor as a function of the energy level. For example, an energy level of 0.7 leads to the expression of attractor C"' (in 70% of the replications), C' (20%) or B' (10%). For energy levels in the interval [0.0, 0.4] we can observe a clear dominance of attractors in class A. A similar dominance in the energy interval [0.7, 1.0] is shown by attractors in class C. The hybrid forms in class B characterize intermediate energy levels. Adapted from [22].

levels imply that a source of energy was recently visited. Given the obvious physical constraints on the agent's speed, it follows that it must be still in the proximity of the agent, consistent with the selection of local behaviors. On the other hand, low energy levels imply that the recent search for an energy source was unsuccessful. This effectively correlates with broader exploratory behaviors. The solution of this minimalist cognitive task relies on the self-organized dynamics of the whole system. In the traditional cognitivist approach however, a similar mechanism would be modeled in terms of explicit representations and memory.

3 The Dynamics of Anticipation

In the current paper we intend to present our work towards anticipatory cognitive architectures, with an emphasis on the role of non-neural internal states. Thus

far, we have discussed a fundamental premise demonstrating how the dynamics of the body, and in particular its bio-regulatory processes, might be partially constitutive of cognition.

Operative definitions of anticipatory behavior stress the effect that an estimation of the future state of the system has on its current behavior [25,26]. Anticipation endows a cognitive agent with the capacity for faster and smoother action execution, facilitate action initiation, improve information seeking, decision making, predictive attention and social interaction [27,28,29,30]. In a recent paper, Butz argues how an anticipatory tension might influence both the development of neural structures and bias the agent to anticipatory behavior [31].

We suggest that settling on a dynamic attractor (e.g. see [24]) constitutes an implicit form of anticipation in at least one important sense. Once engaged with an attractor, the system enters a stable and fully determined regime. Our capacity to predict the trajectory of a strange attractor might well be limited by the confidence in the prediction that we can draw, as the system's non-linearities amplify our error. Nevertheless, once settled on an attractor that currently satisfies specific functional requirements, the whole dynamic of the system is attuned to a specific flow of events. An example of this attunement and its anticipatory role is Pavlov's dog, that salivates when food is made potentially available, thus effectively preparing its body for the digestive process. Some authors consider this kind of anticipation so important for an agent that conditioning, the prototypical basic form of learning in organisms, can be interpreted as mainly functional to its potentiation for originally neutral stimuli become suitable for the elicitation of anticipatory responses [32].

This observation constitutes a second important premise for what follows. To summarize, the body (in an extended sense that includes its non-neural internal mechanisms) constitutes a critical component of the potential dynamical richness of an agent attuned to its environment. Such richness, when autonomously viable, is intrinsically endowed with anticipatory power.

4 The Bodily Path of Anticipation

A brief example of a prototypical situation will shed some light on our proposal. Let us consider a cognitive agent engaged in some activity, for example lightheartedly roaming on a soft lawn, enjoying the sight of colorful flowers and picking up wild berries. Suddenly something unexpected pops up from the bushes, something potentially noxious and maybe never experienced before (e.g., depending on the agent's particular sensitivity it could be a spider, a coral snake, or even a carnivorous dinosaur). We can be quite confident that a viable evolved agent would find a way to inhibit or redirect its current activity towards a more conservative behavior.

With reference to Fig.3, we have to state a few preliminary assumptions:

– the global activity determining the current behavioral engagement between the agent and its environment (namely, a behavioral attractor, similar

to [22,33,34]) is described by a few global variables that compress the specific relevant information for the current sensory-motor activity out of the enormous number of degrees of freedom of the system [12];
- the box labeled SENSORY-MOTOR FLOW represents the neural activity associated with the sensory-motor flow of the behavioral attractor;
- the corresponding non-neural bodily dynamics are summarized by the box labeled NON-NEURAL INTERNAL DYNAMICS;
- embedded within the global current dynamic we recognize a neural sensory-motor emulator[3] (box ANTICIPATION), whose evolution over time is dynamically correlated with the actual sensory-motor flow (similar to [33]), although not necessarily identical to it (as in [35]);
- the dynamics of the emulator (adapted during the evolutionary history and/or during the agent's ontogenesis) can anticipate, in the dynamic sense illustrated above, the sensory-motor consequences of the engagement with a potentially noxious activity, as they follow a faster time scale.

Crucially, the capacity to predict the potential negative outcome endows the agent with a massive advantage: it attains the possibility to prepare itself before confronting the consequences of its current behavior, or to inhibit its behavior altogether. If we assume the possibility of a direct interaction between anticipatory and actual sensory-motor dynamics (i.e. a direct path between the boxes ANTICIPATION and SENSORY-MOTOR FLOW in Fig.3), we immediately recognize a critical problem. Which kind of dynamics would eventually emerge after the current action is inhibited? Obviously, the dynamic structure emerging in the emulator should elicit a viable alternative behavioral attractor. How would that be selected?

Generalizing our example to other situations critical for the agent's viability, our *bodily-anticipation hypothesis* is that, rather than a direct influence on the current behavior, the effect of the prediction of the emulator is actually mediated by the body. The outcome of the emulator affects the actual bodily dynamics (path a-b in Fig.3), and altered bodily quantities transiently act as control parameters for the actual sensory-motor flow (path b-sm). Hence, the problem of the determination of the next behavioral attractor is off-loaded onto the bio-regulatory dynamics of the body. Destabilized by the input from the sensory-motor emulator, the body viscerally reacts *as-if* actually engaged in such sensory-motor experience, eliciting behaviors that pull back the system towards viable regions. That implies that the body can achieve homeostatic balance not only in virtue of isolated non-neural internal dynamics, but also by triggering the selection of an appropriate behavior (path sm-b). This mechanism exploits the knowledge 'accumulated' by the body during a long and complex process of evolutionary and ontogenetic adaptation, functional to the viability of the agent. Equivalent knowledge, in case of a (theoretically possible) neural path directly

[3] We follow here the terminology introduced by Grush [28], to denote an explicit subsystem that dynamically generates a prediction of the agent-environment sensory-motor interaction.

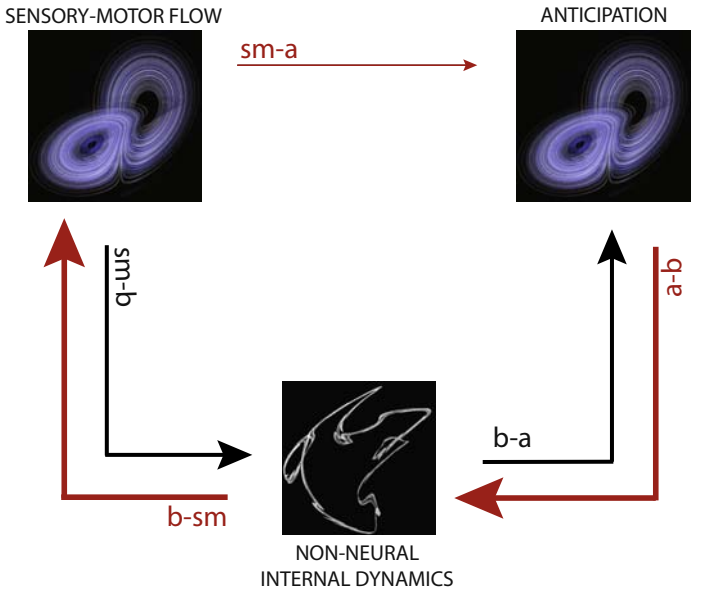


Fig. 3. Illustration of the bodily-anticipation hypothesis. In its roaming, our agent gets engaged with a potentially noxious interaction. Neural sensory-motor anticipatory dynamics, here conveniently isolated within the global coupled system (box labeled ANTICIPATION), predict the risk by determining a change in the current non-neural bodily dynamics (box NON-NEURAL INTERNAL DYNAMICS) through path a-b. This induces the agent to a visceral reaction, *as-if* actually engaged in the noxious sensory-motor experience. From here, indirectly through a further path b-sm, the anticipatory dynamics modulate the actual sensory-motor dynamics (box SENSORY-MOTOR FLOW). Following a quick reorganization of its behavioral attractor, our agent is attuned to escape the danger thanks to the mediation of its body, as there is no direct neural coupling path between anticipatory and sensory-motor dynamics.

coupling anticipatory and sensory-motor dynamics (through the missing path a-sm), should be somehow achieved by the nervous system.

5 The Bodily Path Hypothesis Put to the Test

5.1 Implementation

The present section describes the first experimental steps toward a minimal implementation of the architectural plan outlined in Section 4. The experimental task takes place in the same simulated square arena and with the same agents as described in Section 2. The experimental task is extended to a scenario that can be abstractly likened to a *go-no go* task (loosely inspired by e.g. [36]). The light sources in our simulated setup emit according to two different patterns[4]. The

[4] In a more natural metaphor, this might model the case of a succulent berry whose external pigmentation is different when unripe (and toxic) or ripe (and energizing).

first matches exactly the characteristics used in the experiment described above (continuous sensory regime), i.e. each light source emits a continuous steady level of luminance. The sensory consequences for the agents have already been demonstrated in Figure 2. Under this regime, nothing differs in the task with respect to the experiment described in Section 2. The agent, whose energy level is subject to a linear time decay (-0.008 per time step), is rewarded with an instantaneous full energy recharge upon invasion of the recharging area. The second sensory pattern is different in that the sensory input is rhythmically set to zero every third time step (intermittent sensory regime). This implies that during this modality the agent is subject to regular intervals of blindness. As pointed out elsewhere [22], the agent evolved in the previous experiment is robust enough to cope with massive perturbations, even of this nature, with no significant alteration of its behavior. Under intermittent regime, entering in the recharging area determines a punishment (-0.08 per time step) that speeds up the linear time decay. Each individual, whose lifetime lasts 1200 time steps, experiences the continuous sensory regime during time intervals [1, 200], [501, 700] and [1000, 1200]; intermittence occurs in the intervals [201, 500] and [701, 1000]. Severe punishment (-1000) was integrated in the fitness score in case of crashing against the walls.

A neurocontroller, assembled as a simple implementation of the general architecture introduced in Section 4, is sketched in Figure 4. We deployed simple

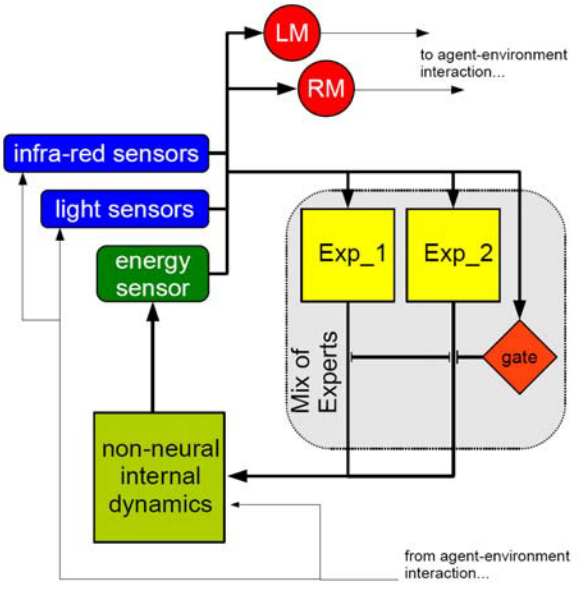


Fig. 4. Sketch of a minimal implementation of our anticipatory cognitive architecture. Infra-red, light and energy sensors drive the two motorneurons through a feedforward ANN with no hidden layers. They also constitute a sensory flow that is processed by a mixture of recurrent experts. Each expert specializes on a specific sensory regime, and the gating signal perturbates the non-neural internal dynamics of the agent.

feedforward artificial neural networks with no hidden layers, extracted from the population of the best individual in the previous experiment. Each ANN works in parallel with a Mixture of Recurrent Experts [37,33,38] whose role is to discriminate between the two different sensory regimes (continuous and intermittent). As each expert tries to outperform the other by generating the most accurate prediction on the sensor's activity at the next time steps, they are actually chunking the sensory-motor flow according to its basic dynamical characteristics, as illustrated in [33].

In other words, each expert, by suppressing the output of the other, signals the engagement of the system with a specific sensory-motor flow, i.e. a specific stream of coupled perception and action. The gating sequence of the mixture of experts (that is, the description of which expert is currently active) can be mapped onto a binary variable. When the continuous sensory regime is detected, nothing differs with respect to the dynamics used in the previously described experimental scenario. On the other hand, during the intermittent sensory regime, the energy level mechanism is overridden by a different mechanism, where the decay rate is freely evolved under the conditions specified above.

5.2 Results

A standard evolutionary algorithm was run in this new scenario in order to select appropriate parameters for the neurocontrollers. Each agent was tested on its capacity to maximize the integral of its energy level over its lifetime (averaged over 10 epochs per individual), starting from random positions in the square environment.[5] Nevertheless, during continuous regime, stationary behaviors within the recharging area are discouraged, as the energy level is not integrated in the fitness until the agent leaves the rewarding space under this condition.

In this paper we compare the results for three different neurocontrollers:

1. The basic feedforward architecture described in Section 2, whose weights and biases have been evolved from scratch on the new task.
2. As above, with evolution starting from the population of the best individual resulting from the previous experiment.
3. The minimal implementation of the general anticipatory cognitive architecture just introduced. The decay rate of the overriding mechanism for the energy level during intermittent regime is the only parameter modified by the evolutionary algorithm, as the rest of the networks remain frozen.

Figure 5 reports the fitness of the best individual (averaged over ten epochs of 1200 time steps each) for the best replication of the experiments for each of the three architectures described above (the parameters used in the evolutionary algorithm are shown in Table 1). Evolving weights and biases for the whole feedforward architectures (arch1 and arch2) produces similar results in terms of final performances. Nevertheless, the evolutionary process is much quicker when

[5] In order to partly make up for more advantageous starting positions, the first 100 time steps were non computed in the fitness function.

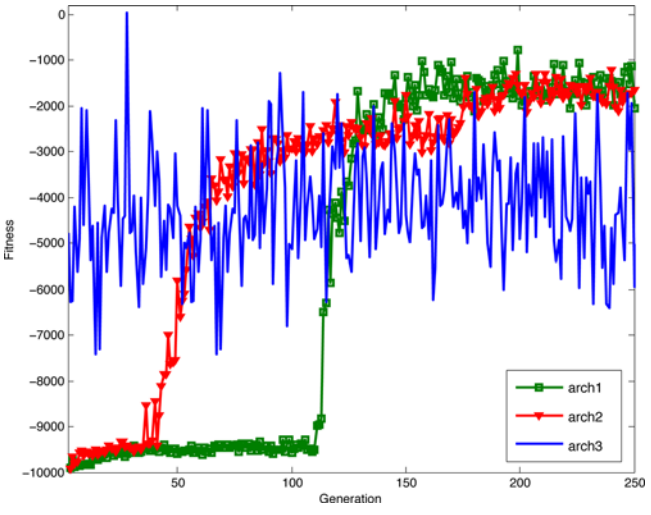


Fig. 5. Performances of the best individuals of the three neurocontrollers during the evolutionary process. Although facing a problematic exploration in its parameter space, the minimal anticipatory cognitive architecture (arch3, continuous line) achieves satisfactory performance without any bootstrap phase.

Table 1. Evolutionary Parameters

Number of generations:	1000
Number of individuals/generation:	100
Number of test epochs/generation:	10
Duration of one epoch (time steps):	1200 (= 120 s)
Starting position:	random
Probability of mutation:	0.02
Probability of crossover:	0
Sensory noise:	0.05

it can develop on the basis of the best population evolved on the simplified task presented in Section 2, although there is no initial advantage in this condition (the two curves basically overlap during the first 40 generations). The evolution of the new architecture (arch3) produces the best absolute performance and the bootstrapping of its performance is immediate. The evolutionary algorithm tends to select values for the single evolved parameter so that the energy level sensed during the intermittent regime simulates high energy. Therefore, consistently with the previous experiment, a tendency towards a photophobic behavior is triggered. Nevertheless, the fitness curve in this condition shows a very high variance, and in the long run the best individuals of the other architectures tend to outperform it.

A qualitative behavioral analysis emphasizes the different strategies deployed by the two classes of architecture, the simple feedforward ANNs with no hidden

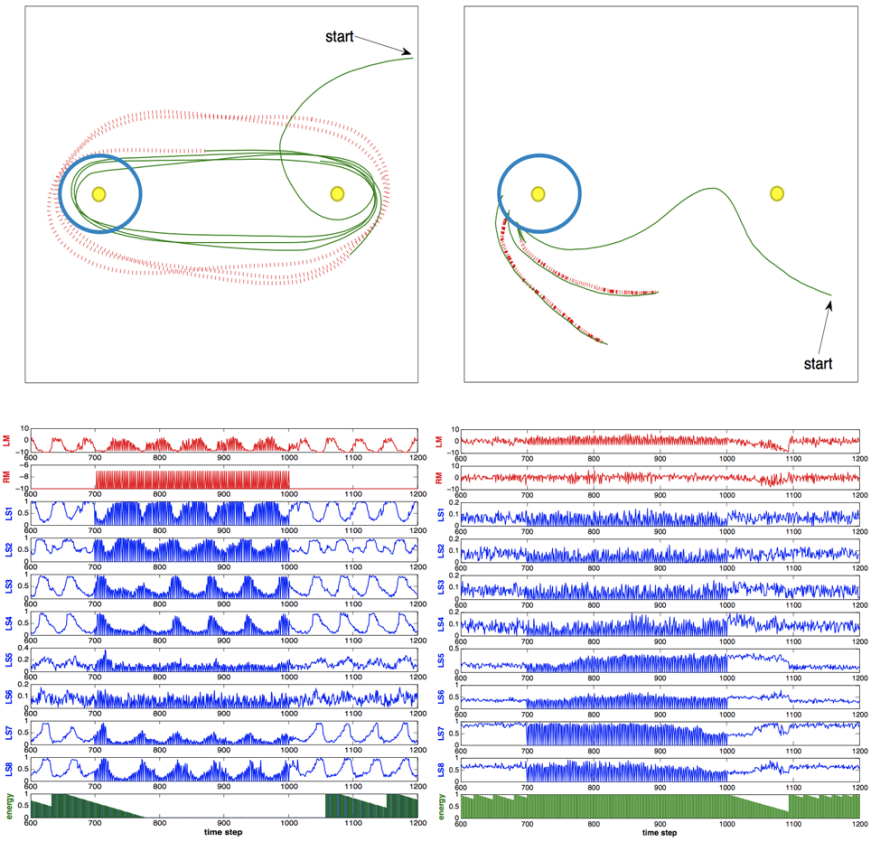


Fig. 6. Typical spatial trajectories developed by the different architectures during evolutionary adaptation. **top left -** Simple feedforward ANNs tend to deploy a stereotypical strategy, i.e. their trajectories systematically engage in exploratory loops between the two light sources, entering the recharging area during the continuous regime (continuous line) and avoiding it during the intermittent regime (dashed line). **top right -** On the other hand, the behaviors that tend to emerge from our minimal anticipatory architectures dynamically engage and disengage with the rewarding/punishing area according to the different sensory regime (continuous/dashed lines represent the trajectories during continuous/intermittent sensory regimes). For a better resolution of details in the trajectory, the two pictures zoom on the area of main interest surrounding the light sources. **bottom -** The left and right panels exemplify, respectively for a feedforward and an anticipatory architecture, the activation of the two motoneurons (LM and RM), of the light sensors (LS1-8) and of the energy level sensor, during 600 time steps that include a double regime transition (continuous-intermittent-continuous) occurring at time steps 700 and 1000.

layers on one hand and the anticipatory architecture on the other. In the experiment described in section 2 we observed two main classes of strategy. The first, briefly described above and extensively reported in previous work [22,23],

is highly dynamic and determines, under the modulation of the non-neural internal control parameter, an overt engagement with a specific object of interest, i.e. the rewarding light source (dynamic engagement). A second strategy relies on the geometrical constraints of the environment: the agent draws ad hoc spatial trajectories in order to achieve the appropriate timing required for the task (stereotypical engagement). Interestingly, the two classes of architecture specialize in producing the two different strategies.

Figure 6 demonstrates the typical behaviors of fit individuals in the two classes. Architectures belonging to the first class (arch 1 and arch2) tend to produce stereotypical attractors. Under the modulation of the different regimes, tighter loops invading the rewarding area will collect frequent rewards during continuous sensory regime, whereas slightly wider loops will stay clear of the recharging area in order to escape the punishment during intermittence. Therefore, the agent ignores the local effect of the rewarding area on its energy level. These behaviors depend on spatio-temporal constraints, as changes in the geometrical characteristics of the environment or in the timing of the different regimes might induce a dramatic drop in terms of performance. Our minimal anticipatory architectures, on the other hand, tend to develop a dynamic engagement with the light source (i.e. moving towards it - continuous trajectory) during the continuous sensory regime and a similarly straightforward disengagement during the intermittent regime (moving to safe distance from the punishing light - dashed line). In our viable anticipatory architectures, photophobic and phototactic behaviors are constantly balanced in order to take the agent either sufficiently close to, or far from, the recharging area, according to the current sensory regime.

6 Discussion

The original task described in Section 2 (a stationary recharging area located in the proximity of a light source) and the extended task (alternate regime of reward and punishment on the same area) are obviously related. Nevertheless, it is interesting to notice that although even simple feedforward ANNs with no hidden layers can cope quite effectively with the new task, they achieve their skills after several generations of evolutionary adaptation. We obtain slightly better results starting the adaptation process from a population that already masters the original task. Nevertheless, in both cases we observe a slow bootstrapping, beginning with remarkably low performance (see Figure 5, arch1 and arch2). On the other hand, the minimal anticipatory architecture introduced in this paper, starting from the same population as used for arch2, demonstrates the instantaneous capacity to achieve satisfactory performance. Interestingly, the viable emerging behaviors with the anticipatory architecture are typically characterized by dynamical engagement and disengagement from the light sources, according to the current sensory regime. This results in flexible and robust behaviors, that contrast with the stereotypical behaviors achieved during the evolution of simple feedforward networks.

It might be observed that high variability of the data plotted in Figure 5 for our anticipatory architecture suggests a problematic evolution of its single free parameter. This is not surprising, since we maintained the exact same parameters for the evolutionary algorithm for all three architectures, and did not pay any attention to an optimal tweaking in this particular condition. In fact, the evolutionary process performs only slightly better on the anticipatory architecture than a random search (result not shown in detail). Nevertheless the best evolved neurocontroller achieves a fairly high (albeit isolated) performance, and for that particular value of the parameter even the average fitness of the population rises to the level of the best individuals (result not shown in detail). Rather, what should be emphasized is that the performance is achieved in parallel with a drastic dimensional reduction of the search space, and in these conditions even a random search can produce a number of individuals that immediately exhibit satisfactory behaviors in a dynamic scenario never experienced before. This is reminiscent of Ashby's proposal of a 'dumb' mechanism that, in the need of maintaining a homeostatic balance for a set of *essential variables* critical to the agent's survival, produce adaptive behaviors [8].

Obviously, our bodily-anticipation hypothesis does not rule out the feasibility of a totally disembodied and direct influence of the sensory-motor emulation on sensory-motor flow (the missing path a-sm in Fig.3). Nevertheless, our approach drastically reduces the complexity of the problem of synthesis and adaptation. The search for viable parameters in the (potentially) massive dimensionality of the system's degrees of freedom is reduced to a search in the subspace of the bodily parameters (in this case, the mere energy level). In our preliminary experiment, the search of the appropriate decay rate that is necessary to cope with the new task proves an effortless procedure for the minimalist anticipatory architecture. On the other hand, readaptation to the new task is cumbersome when we evolve the whole set of weights and biases in the ANNs. In a more naturalistic perspective, a basic organism constituted of a body coupled with a simple nervous system learns to survive first, by deploying a set of elementary sensory-motor reflexes in order to establish a basic form of viable coupling with its environment. This involves the evolutionary and ontogenetic adaptation of the interaction paths b-sm and sm-b in Fig.3. Then the agent adaptively extends its viability by governing predictions. Incidentally, this is in accord with the design principle of *holistic reductionism* [39], where the cognitive capacity of a minimal realization of a whole and viable autonomous system is incrementally extended.

The adaptation of the emulator takes place on the basis of the sensory-motor information provided along paths sm-a and b-a; paths ontogenetically adaptable, whose role could be adaptively weighted during the agent's life (as in [28]) and that might be transiently wiped out as the emulator proves its ability to produce effective predictions. In this extreme situation the system would express the capacity for 'blind navigation', i.e. navigation achieved by relying on its own sensory-motor predictions rather than on actual sensory information [35]. Therefore, the boxes ANTICIPATION and SENSORY-MOTOR FLOW in Fig.3 act as informationally semi-permeable subsystems. Their coupling, from the latter

to the former, should be modulated in a context-dependent way. The opposite coupling, far from absent, is indirectly realized via the body, according to our bodily-anticipation hypothesis.

In Section 2, we have described our model using the intuitive metaphor of an energy level mechanism, thus evoking biologically plausible dynamics of food intake and metabolism. Nevertheless, our intentionally simple scenario aimed to facilitate the abstraction to general principles. Metaphor aside, the fundamental aspect to consider is the coupling of different dynamic systems characterized by time scales that differ by several orders of magnitude (in particular we refer to the dynamics of the sensory-motor and the energy level systems). The availability of the slower dynamic of the energy level is exploited during the evolutionary adaptation of the system. In fact, the neurocontroller receives input vectors which are organized as dynamically related events in a continuous sensory-motor flow (i.e. contexts with a similar, although continuously varying, level of energy). The outcome of the adaptation process allows the system to integrate information over time. Although the sensory-motor mapping as such is purely reactive, this is not valid for the motor-sensory mapping and thereby for the behavior of the system as a whole. On the basis of these observations, we formulate the *hypothesis* that the access to a collection of attuned dynamic sub-systems characterized by intrinsic dynamics at different time scales and the exploitation of such differences, constitutes a powerful mechanism of embodied cognition, widely operating at the different levels of organization of biological cognition. A mechanism providing the cognitive system with the capacity to structure information on events which are relevant to its survival, with no need for explicit representations, memory or consciousness.

The focus on the role of multiple time scales, thus remapping the interpretation of our system in more abstract terms, dissolves the problematic distinction between non-neural and neural dynamics. We are advocating a mechanism where intrinsic time scales, characterizing mechanical, chemical and electrical phenomena in the body, might be coherently integrated into the cognitive process [17]. The dynamical richness of non-neural bodily processes might support the characteristic time scales of regular sensory-motor dynamics. The interest for the role of multiple time scales is currently growing in the neuroscientific community (e.g. [40,41]) as well as in cognitive robotics (e.g. [42,34,43,44]). A parallel might be drawn with other nurocomputational architectures that deploy rich potential dynamics at different time scales, like Echo State Networks and Liquid State Machines [45,46]. Nevertheless, in the case of our architecture the bodily dynamics that inspire the non-neural internal mechanism are homeostatically and evolutionary relevant, i.e. they have a crucial effect on the body of the agent and on its behavior, independent on whether or not any cognitive process makes use of them (e.g. see [47]). Reservoir dynamics in ESN and LSM, on the other hand, can be completely random and irrelevant, and they have no effect whatsoever unless they are actually read out.

Our own and related experimental evidence in cognitive robotics supports our assumptions on paths b-sm and sm-b in Fig.3, as examined in the previous

Section 2 (e.g. [22,48,49,34]). This is to say that bodily states can modulate cognitive dynamics (e.g., think of the effects of particular chemical substances injected in your body) and particular behaviors can critically affect our body (e.g. in eating disorders). The capacity of the brain to anticipate sensory-motor correlates (path sm-a) is also supported by experiments in cognitive robotics, as in [33,34,35], and object of neuroscientific investigation (e.g., see [50]). In addition, the effect of mental imagery on non-neural bodily states is also rooted in neuroscientific evidence of biological cognitive processes (e.g. see [20]). The same author inspired the seemingly arbitrary choice to implement an overriding energy mechanism that takes over during intermittent regime. False bodily information can sometimes substitute for the correspondent actual state, for example when a contingent urge induces us to ignore pain [21]. Damasio seems to bring forth a somewhat opposite hypothesis, as he advocates *as-if body loops* [20,21]. In Damasio's theory, the emotional machinery, deeply integrated in the homeostatic mechanisms, plays a crucial role even in the case of highly logical functions, as in decision making. During the process of decision making it continuously supports the mental activity (*body loop*). After multiple exposure the brain builds appropriate neural causal associations that completely obliterate bodily information from the process (*as-if body loops*). Nevertheless, Bechara reports experimental results suggesting that *as-if body loops* are more plausible during choices made in highly predictable conditions (choice under certainty). As the decision process takes place in less predictable scenarios (full uncertainty) the *body loop* mode of operation becomes prominent [51]. We find this observation in perfect agreement with the intuition deployed in our model.

The architecture sketched in Figure 3 evokes a dynamic complexity that is drastically simplified in our initial implementation. The balance between the two subsystems ANTICIPATION and SENSORY-MOTOR FLOW is of course of the most delicate nature. In fact, via their effect on the non-neural internal dynamics, each of the two systems might simultaneously try to drive the system towards different dynamic attractors. This apparent contradiction should not necessarily be interpreted as a flaw in our proposed architecture. The dynamic tension between two competing requests maintaining the system in a regime of metastability[6], rather than (more traditionally) of stability, might also be exploited as a potential opportunity. Some authors (e.g. [12]) consider metastability the fundamental state for a complex dynamic system like the brain, for it allows flexible and fast engagement and disengagement with contingent environmental requirements and constraints. Kelso, for example, offers an inspiring dynamic image of biological brains [ibid., p. 26]:

> The human brain is essentially a pattern-forming self-organized system governed by nonlinear dynamic laws. Rather than compute, our brain "dwells" (at least transiently) in metastable states: it is poised on the brink of instability where it can switch flexibly and quickly. By living

[6] The concept of *metastability* can be intuitively introduced as a dynamical situation where the system does not express stable states, but a mere tendency towards them [12].

> near critically, the brain is able to anticipate the future, not simply react to the present. All this involves the new physics of self-organization in which, incidentally, no single level is any more or less important than any other.

This intriguing scenario deserves further experimental investigation. Along a related line, we advocate the intrinsic unity of the general anticipatory architecture sketched in Fig.3. We graphically split a system, that is actually meaningful only as a whole, into three different compartments just for the sake of clarity. The system should be conceived as a unity, where no component has a dominant role over the others, consistent with the final statement in Kelso's quote. Incidentally, under this perspective the traditional dichotomy between controlled and controller should be re-considered, as the different subsystems, through their coupling, mutually influence and regulate each other. Such interactions are graphically represented by the arrows in Fig.3. Nevertheless, we might be interested in how a specific unbalance in one of the subsystems influences the others, and accordingly, as observers, choose the most convenient perspective. This step is legitimate and often even necessary to illuminate our analysis, albeit it does not modify the unitary nature of the system. As much as we emphasize the constitutional unity of our system, crucially linked with its environment, a more traditional symbolic approach would draw a sharp distinction between the agent and its environment.

7 Conclusions and Future Work

Non-neural internal states, in virtue of their different time scales, prove powerful potential props in support of cognitive processes. With this paper we hope to have contributed in some measure to highlighting their potential role, both in synthetic cognitive systems and, by extension, in biological ones. Preliminarily, we showed how a non-neural internal state, crucially characterized by a time scale that is orders of magnitude slower than ordinary dynamics of the sensory-motor interactions, can modulate the activity of a simple neurocontroller. What we achieved is the implementation of a self-organized, dynamic action selection mechanism, effectively operating in a context dependent way. Then we showed how these characteristics can be exploited by a minimal anticipatory cognitive architecture, using an explicit model for dynamic anticipation that operates via bodily mediation (*bodily-anticipation hypothesis*). This allows the system to scale up to more complex tasks never experienced before, achieving flexible and robust behavior with minimal adaptive cost.

Clearly, our hypotheses presented in Section 4 and 6 require more experimental investigation and validation, which is currently under development in our lab. The work presented in this paper is still in progress and far from maturity. In order to facilitate the analysis and the extraction of general principles, our starting point is the synthesis of simple systems. A first extension will be the deployment of the full dynamic of the general anticipatory architecture sketched

in Figure 3. The implementation of a more realistic internal dynamic, inspired by natural or artificial metabolic systems such as *microbial fuel cells* [47], represents the necessary step in order to systematically assess the potential of the architecture that we present here.

Acknowledgments

The authors thank Silvia Coradeschi and Serge Thill for their comments on an early version of this paper, and Malin Aktius, Anthony Morse, Pierre Philippe and Henrik Svensson. We also thank the three anonymous reviewers for their suggestions. This work has been supported by a European Commission grant to the project *Integrating Cognition, Emotion and Autonomy* (ICEA, www.iceaproject.eu IST-027819,) as part of the *European Cognitive Systems* initiative.

References

1. Varela, F.J., Thompson, E.T., Rosch, E.: The Embodied Mind: Cognitive Science and Human Experience. MIT Press, Cambridge (1992)
2. Thelen, E., Smith, L.B.: A Dynamic Systems Approach to the Development of Cognition and Action. MIT Press, Cambridge (1996)
3. Clark, A.: Being There: Putting Brain, Body, and World Together Again. MIT Press, Cambridge (1997)
4. Chrisley, R., Ziemke, T.: Embodiment. In: Encyclopedia of Cognitive Science, pp. 1102–1108. McMillan, London (2002)
5. Ziemke, T., Zlatev, J., Frank, R.M. (eds.): Body, Language and Mind: Embodiment, vol. 1. Mouton de Gruyter, Berlin (2007)
6. Nolfi, S.: Power and limits of reactive agents. Neurocomputing 42(1-4), 119–145 (2002)
7. Wiener, N.: Cybernetics, or Control and Communication in the Animal and the Machine. MIT Press, Cambridge (1965)
8. Ashby, W.R.: Design for a Brain: The Origin of Adaptive Behavior. Chapman '&' Hall, London (1952)
9. Köhler, W.: Gestalt Psychology. Liveright (1947)
10. Gibson, J.J.: The Ecological Approach To Visual Perception. Houghton Mifflin (1979)
11. Van Gelder, T.: The dynamical hypothesis in cognitive science. Behavioral and Brain Sciences 21, 615–628 (2000)
12. Kelso, J.A.S.: Dynamic Patterns: The Self-organization of Brain and Behavior. MIT Press, Cambridge (1995)
13. Beer, R.D.: Parameter space structure of continuous-time recurrent neural networks neural networks. Neural Computation 18, 3009–3051 (2006)
14. Beer, R.D.: Dynamical approaches to cognitive science. Trends in Cognitive Sciences 4(3), 91–99 (2000)
15. Chiel, H., Beer, R.D.: The brain has a body: Adaptive behavior emerges from interactions of nervous system, body and environment. Trends in Neurosciences 20(12), 553–557 (1997)
16. Suzuki, M., Floreano, D.: Enactive robot vision. Adaptive Behavior 16, 122–128 (2008)

17. Parisi, D.: Internal robotics. Connection Science 16(4), 325–338 (2004)
18. Ziemke, T.: On the role of emotion in biological and robotic autonomy. BioSystems 91, 401–408 (2008)
19. Ziemke, T., Lowe, R.: On the role of emotion in embodied cognitive architectures: From organisms to robots. In: Cognitive computation (2009) (accepted)
20. Damasio, A.: The Feeling of What Happens: Body and Emotion in the Making of Consciousness. Harvest Books (2000)
21. Damasio, A.: Looking for Spinoza: Joy, Sorrow, and the Feeling Brain. Harcourt (2003)
22. Montebelli, A., Herrera, C., Ziemke, T.: On cognition as dynamical coupling: An analysis of behavioral attractor dynamics. Adaptive Behavior 16(2-3), 182–195 (2008)
23. Montebelli, A., Herrera, C., Ziemke, T.: An analysis of behavioral attractor dynamics. In: Almeida e Costa, F. (ed.) Advances in Artificial Life: Proceedings of the 9th European Conference on Artificial Life, pp. 213–222. Springer, Berlin (2007)
24. Strogatz, S.H.: Nonlinear Dynamics and Chaos. Westview Press, Cambridge (1994)
25. Rosen, R.: Anticipatory Systems. Pergamon Press, Oxford (1985)
26. Butz, M.V., Sigaud, O., Gérard, P.: Anticipatory behavior: Exploiting knowledge about the future to improve current behavior. In: Butz, M.V., Sigaud, O., Gérard, P. (eds.) Anticipatory Behavior in Adaptive Learning Systems. LNCS, vol. 2684, pp. 1–10. Springer, Heidelberg (2003)
27. Butz, M.V., Pezzulo, G.: Benefits of anticipation in cognitive agents. In: Pezzulo, G., Butz, M.V., Castelfranchi, C., Falcone, R. (eds.) The Challenge of Anticipation: A Unifying Framework for the Analysis and Design of Artificial Cognitive Systems, pp. 45–62. Springer, Heidelberg (2008)
28. Grush, R.: The emulation theory of representation: motor control, imagery, and perception. Behavioral and Brain Sciences 27, 377–442 (2004)
29. Barsalou, L.W.: Perceptual symbol systems. Behavioral and Brain Sciences 22, 577–660 (1999)
30. Barsalou, L.W.: Social embodiment. In: Ross, B.H. (ed.) The Psychology of Learning and Motivation, pp. 43–92. Academic Press, London (2003)
31. Butz, M.V.: How and why the brain lays the foundations for a conscious self. Constructivist Foundations 4(1), 1–42 (2008)
32. Parisi, D.: Mente: i nuovi modelli della Vita Artificiale. Il Mulino, Bologna (1999)
33. Tani, J., Nolfi, S.: Learning to perceive the world as articulated: an approach for hierarchical learning in sensory-motor systems. Neural Networks 12(7-8), 1131–1141 (1999)
34. Ito, M., Noda, K., Hoshino, Y., Tani, J.: Dynamic and interactive generation of object handling behaviors by a small humanoid robot using a dynamic neural network model. Neural Networks 19(3), 323–337 (2006)
35. Ziemke, T., Hesslow, G., Jirenhed, D.A.: Internal simulation of perception: a minimal neuro-robotic model. Neurocomputing 68, 85–104 (2005)
36. Schoenbaum, G., Chiba, A.A., Gallagher, M.: Orbitofrontal cortex and basolateral amygdala encode expected outcomes during learning. Nature Neuroscience 1(2), 155–159 (1998)
37. Haykin, S.: Neural networks: a comprehensive foundation. Prentice Hall, Englewood Cliffs (1999)
38. Jacobs, R.A., Jordan, M.I., Nowlan, S.J., Hinton, Geoffrey, E.: Adaptive mixtures of local experts. Neural Computation 3(1), 79–87 (1991)
39. Morse, A., Lowe, R., Ziemke, T.: Towards an enactive cognitive architecture. In: Proceedings of the 2008 International Conference on Cognitive Systems (2008)

40. Kiebel, S.J., Daunizeau, J., Friston, K.J.: A hierarchy of time-scales and the brain. PLoS Computational Biology 4(11) (2008)
41. Fusi, S., Asaad, W.F., Miller, E.K., Wang, X.J.: A neural circuit model of flexible sensorimotor mapping: Learning and forgetting on multiple timescales. Neuron 54(2), 319–333 (2007)
42. Yamashita, Y., Tani, J.: Emergence of functional hierarchy in a multiple timescale neural network model: A humanoid robot experiment. PLoS Computational Biology 4(11) (2008)
43. Paine, R.W., Tani, J.: How hierarchical control self-organizes in artificial adaptive systems. Adaptive Behavior 13(3), 211–225 (2005)
44. Maniadakis, M., Tani, J.: Dynamical systems account for meta-level cognition. In: Asada, M., Hallam, J.C.T., Meyer, J.-A., Tani, J. (eds.) SAB 2008. LNCS, vol. 5040, pp. 311–320. Springer, Heidelberg (2008)
45. Jaeger, H., Haas, H.: Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication. Science 304(5667), 78–80 (2004)
46. Maass, W., Natschlager, T., Markram, H.: Real-time computing without stable states: A new framework for neural computation based on perturbations. Neural Computation 14(11), 2531–2560 (2002)
47. Melhuish, C., Ieropoulos, I., Greenman, J., Horsfield, I.: Energetically autonomous robots: food for thought. Autonomous Robots 21, 187–198 (2006)
48. Tani, J., Ito, M.: Self-organization of behavioral primitives as multiple attractor dynamics: A robot experiment. IEEE Trans. on Systems, Man, and Cybernetics. Part B 33(4), 481–488 (2003)
49. Tani, J.: Learning to generate articulated behavior through the bottom-up and the top-down interaction processes. Neural Networks 16, 11–23 (2003)
50. Hesslow, G.: Conscious thought as simulation of behaviour and perception. Trends in Cognitive Sciences 6(6), 242–247 (2002)
51. Bechara, A.: The role of emotion in decision-making: Evidence from neurological patients with orbitofrontal damage. Brain and Cognition (55), 30–40 (2004)

A Neurocomputational Model of Anticipation and Sustained Inattentional Blindness in Hierarchies

Anthony F. Morse, Robert Lowe, and Tom Ziemke

COIN Lab, Informatics Research Centre, University of Skövde
SE-541 28 Skövde, Sweden
{anthony.morse,robert.lowe,tom.ziemke}@his.se

Abstract. Anticipation and prediction have been identified as key functions of many brain areas facilitating recognition, perception, and planning. In this chapter we present a hierarchical neurocomputational model in which feedback, effectively predicting or anticipating task-relevant features, leads to sustained inattentional blindness. A psychological experiment on sustained inattentional blindness in human subjects is simulated to provide visual input to a hierarchy of Echo State Networks. Other parts of the model receive input relevant to tracking the attended object and also detecting the unexpected object, feedback from which is then used to simulate engagement in the task and compared to results obtained without feedback, simulating passive observation. We find a significant effect of anticipation enhancing performance at the task and simultaneously degrading detection of unexpected features, thereby modelling the sustained inattentional blindness effect. We therefore suggest that anticipatory / predictive mechanisms are responsible for sustained inattentional blindness.

Keywords: Enaction, Anticipation, Prediction, Neurocomputation, Reservoir Systems, Association, Sustained Inattentional Blindness, Neural Modelling, Cortical Hierarchies.

1 Anticipation and Prediction; From Neuroscience to Cognition

In work published elsewhere [1, 2] we presented a model of sustained inattentional blindness in which predictive feedback enhanced performance at the feedback relevant task while degrading performance at other tasks. Furthermore by systematically varying the similarity between target and unexpected stimuli we were able to replicate human data showing that the size of the sustained inattentional blindness effect is reduced as similarity is increased. Somewhat surprisingly we also found that in our model, the size of the sustained inattentional blindness effect is also reduced as similarity decreases beyond a half way point. This prediction has yet to be confirmed in human data but provides an example of the classic U shaped curve where detection of the unexpected object is most likely if it is very similar or very dissimilar to the tracked object. Central to this model was the use of a predictive feedback signal which was artificially provided. It was therefore not clear where this predictive feedback would originate from, just that given this kind of feedback the inattentional blindness effect was present. In this chapter we extend that original work providing a

G. Pezzulo et al. (Eds.): ABiALS 2008, LNAI 5499, pp. 152–169, 2009.

minimal hierarchy of simulated cortical micro-columns making explicit how and why such predictive feedback signals can be generated and used.

Neural mechanisms for prediction and anticipation are thought to be pervasive throughout much of the brain and are thought to vary greatly in instantiation in different anatomical structures. For example, Downing and others [3-5] provide a review of neurocomputational models of the cerebellum, the basal ganglia, the neocortex, the hippocampus, and the thalamocortical loop. In each of these models, and reflecting the underlying neuroscience, the anticipatory systems differ both in their mechanics and functional significance, which ranges from the anticipation of relational and sequential data in neocortical models, to the prediction of reward stimuli in the cerebellum and basal ganglia. Hawkins and Blakeslee [6] have suggested that "the cortex's core function is to make predictions" [p. 113] and that "all predictions are learned by experience… if there are consistent patterns among the inputs flowing into your brain, your cortex will use them to predict future events" [p. 120]. This places prediction centrally in our understanding of brain function. To understand how this predictive process works, Hawkins depicts the neocortex as a macro hierarchy of homogeneous micro-circuits or structures, meaning that the same basic unit is repeated throughout the neocortex but the way they are wired together and thus their influence on each other varies from region to region. The hierarchical view of the cortex highlights the interplay between low levels (i.e. regions of cortex close to sensory input such as area V1 in the visual cortex) responding to fast changing 'features' and higher levels (such as the inferotemporal or prefrontal cortex) responding to more invariant, larger scale, and slower changing things such as faces and objects. Here, according to Hawkins, top-down signals indicating expectations or predictions, of which low level features will be active, not only fill in sensory gaps, but facilitate the higher level response and differentiate expected from unexpected bottom-up activity. In this chapter we develop such a model but highlight a different aspect of top-down predictive signals, specifically that they can be use to 'tune' input filters facilitating recognition of the anticipated features with consequences for the recognition of non-anticipated features.

Accounts of cognition rooted in underlying predictive or anticipatory neuroscience are becoming increasingly common. Gallese and Lakoff [7], for example, site extensive neuroscientific evidence for predictive / anticipatory circuitry in the sensorimotor systems of the brain and go on to propose a sensorimotor theory of conceptual knowledge based therein. The point to note here is that prediction and anticipation are not simply ancillary functions of the brain but are central to its proper functioning and their existence is heavily supported by neuroscientific data. Such theories all have in common the idea that feedback or spreading activation from active regions, serves the role to anticipate, predict, and prime other related structures as a significant (and by some theories necessary) part of the cognitive perceptual process. In many ways this is not a new idea and has much in common with spreading activation models in early connectionism [8] and also psychological theories of associationism [9]. Unlike those early models and theories, which were largely disembodied, the current focus on embodied and situated cognition or enactive perception provides a different perspective from which to view these problems.

"...Human evolution has indeed led to increasingly complex forms of behaviour. However, these behaviours are not simply triggered from genetically determined mechanisms. Rather they are the outcome of the gradual formation of internal representations during the lengthy process of ontogenetic development." [10][p. 144].

While the adult neocortex contains many function or behaviour specific areas, as Karmiloff-Smith [10] shows with extensive fMRI data, these functions are not genetically pre-specified but are globally processed in children and gradually formed into these areas by a process of active re-structuring according to relationships in external stimuli. Taking Hawkins [6] perspective this re-structuring is a matter of modifying the connectivity and thus influence of the already hierarchically connected cortical micro-columns. By comparison genetic disorders such as Williams Syndrome (WS), "appear to follow a deviant developmental pathway" [10][p. 151] whereby genetic alteration changes the adaptive mechanisms in the brain that would otherwise have led to the formation of these modules, as a result of which brain structuring by experience follows a different path. Thus what is different is not some genetic predisposition to develop such and such a capacity but rather the mechanics of the re-structuring process itself.

While the plasticity of the cortex is well recognized, the extent of this re-structuring is brought into focus by the work of Sharma et al [11] who performed a number of experiments in which the auditory nerve and the optic nerve were cut and crossed over in infant ferrets. The ferrets subsequently developed visual orientation modules in the auditory cortex (these are structures only ever previously discovered in the visual cortex, and never normally present in the auditory cortex). This suggests that rather than the visual and auditory cortex being specialized for one type of processing, they are actively structured around the input streams they receive. This is a radical view to take but one for which there is a great deal of evidence. Clearly evolution has played a part in specialising regions of cortex but rather than providing innate structures, we suggest rather that evolution has refined the relative sizes of the various unimodal and polymodal regions involved in processing different sensory streams and their innate hierarchical structure. As an example, in primates the visual areas of the cortex are significantly larger than the auditory areas and so the auditory cortex is more limited in the structures it can construct than the visual cortex is. To some extent this can be seen as division of resources so that in our case greater resources are given to vision than to audition.

Complementing this perspective is Mountcastle's [12] view of the cortex as performing the same operation everywhere. While most neuroscientists are highlighting the differences (functional or anatomical) between different regions of cortex, Mountcastles' analysis suggested that the cortex consists of the same basic unit, the cortical micro-column, everywhere one looks. While there may be some variation in micro-columns in different regions they are to a large extent very similar throughout the cortex. This is the basic unit of the hierarchy that we model herein and we shall return to the cortical micro-column in section 5.

This aim of this chapter is to investigate and model some of the less obvious consequences of prediction or anticipation in hierarchical structures and their effect on an account of cognitive processes. To provide an example we further develop a cortical microcircuit model of sustained inattentional blindness already reported elsewhere [1, 2] implementing a hierarchical structure based on the literature just introduced. The resulting models are demonstrated to preserve the sustained inattentional blindness effect. In the next section we introduce a cognitive theory of perception and work back to expose possible mechanisms able to implement such an account, in the hope that these match to some extent those neuroscientific mechanisms just introduced.

We then, in Section 3, highlight some of the computational difficulties in implementing these cognitive theories in an embodied and situated agent. Section 4 departs somewhat from this discussion and introduces psychological literature on the phenomena of sustained inattentional blindness, which is the target of the modelling experiments provided in Section 5. Finally Section 6 provides some discussion and conclusions to this work.

2 Perception and Action; From Cognition to Neuroscience

The central claim of Noë's [13] enactive theory of perception "is that our ability to perceive not only depends on, but is constituted by, our possession of… sensorimotor knowledge." [p. 2], where sensorimotor knowledge "is implicit practical knowledge of the ways movement gives rise to changes in stimulation." [p. 8]. This means that, sensorimotor knowledge is not simply factual knowledge about a domain but is intimately about the dynamic relationship between an agent and its environment. Mastering this dynamic relationship is manifest in the ability to predict the sensory consequences of actions, which in turn, according to Noë is constitutive of perception. Under this view the distinction between skills and knowledge is collapsed [14], as Maturana and Varela [15] put it, "all doing is knowing, and all knowing is doing". For example, to perceive something as a round plate is to exercise a particular skill predicting how sensory contact with the plate will change as one moves a little this way or a little that way. Such perception can be mistaken and such mistakes 'pop out' when these predictions are invalidated by further experience (consider the Ames distorted room illusion [16]). Such predictive ability would seem essential to any embodied agent interacting with a complex environment; for example, simulating or anticipating the effects of possible actions for evaluation in an agent-centred way (e.g. [17-20]). Here theories such as the simulation hypothesis suggest the re-use of, primarily, sensory pathways. The idea is that sensory data is processed as normal but rather than producing an overt motor response, the response is used to generate a prediction of what the next sensory input would be had that response been overt. By projecting this predicted sensory state back into the sensory areas the process can re-use the existing circuitry to iterate the prediction process further and further ahead in time. For more detail on the biological basis of the simulation hypothesis see the chapter on this subject by Svensson et. al. also in this book.

For Noë [13], the ability to make predictions is simply the application of sensorimotor knowledge which comes from finding "pattern[s] in the structure of sensorimotor contingency" [p. 103], i.e. patterns in the relationships between our actions and sensations. While this may seem to the uninitiated trivial to implement there are significant problems in the application of such pattern recognition, as we will discuss in Section 3. Going beyond Noë's formulation we would draw a distinction between 'shallow' perception of experience, which could result from merely recognising the relationship between optic flow and turning the head, and 'deep' perception of a world consisting of objects and affordances. 'Deep' sensorimotor knowledge requires the recognition of profiles of change and provides a means to recognize Gibsonian [21] affordances, i.e. "to perceive… is to perceive structure in sensorimotor contingencies. To see that something is flat is precisely to see it as giving rise to certain

possibilities of sensorimotor contingency [*to see its affordances*]. To feel a surface as flat is precisely to perceive it as impeding or shaping one's possibilities of movement" [13] (italics added) [p. 105]. For any agent then, skilled action is practical knowledge, the mastery of which can be achieved through identifying the contextual regularities between action and sensory perception. Here we can see a strong link between Noë's account of perception, Gallese and Lakoff's [7] theory of conceptual knowledge, and Hesslow's [19] simulation hypothesis.

This formulation of the enactive approach not only suggests that robots or agents could learn to "perceive an idiom of possibilities for movement" [13] [p. 105], but also suggests that such a capacity is both particularly amenable to the integration of other cognitive phenomena, such as planning (for which prediction would seem necessary [18, 19, 22]), and significantly different from mainstream representational theories (e.g. [23, 24]). Although evolution obviously has a role in this, it seems reasonably clear that much of our own world knowledge and skills are either derived from, or heavily shaped by, our life-time experiences. Learning sensorimotor relationships from experience and then using these learnt relationships for prediction is the target of our neurocomputational model.

While Gallese and Lakoff's [7] theory of conceptual knowledge differs from Noë's, the general form of both theories of cognition based on sensorimotor prediction have much in common. Having now briefly reviewed some of the biological (section 1) and the philosophical (section 2) theories of sensorimotor cognition, we turn in the next section to the computational problems any implementation of these perspectives faces.

3 Circularity, Regularity, and Time, but Not in That Order…

Acquiring the ability to predict changes in sensory streams that result from either motor actions (simulated or real), or from temporal aspects of our environment and embodiment requires an ability to learn temporal sequences and relational information. The latter would seem to be explained by known plasticity in biological nervous systems, roughly approximated by Hebbian plasticity [9] 'what fires together wires together'. Such plasticity is highly suggestive of the idea that co-occurring features, presumably indicated by the activity of sub-sets of neurons would lead to those neurons 'wiring' together with the result that activity in one set (caused by the presence of the relevant feature in sensory data) would lead to a spreading activation via the new wiring resulting in activation of the other set of neurons. Such ideas are well developed in localist connectionist models which are able to replicate a great deal of psychological data from a variety of phenomena such as; semantic and associative priming [25], cued recognition, name processing and lexical processing [26], semantic and repetition priming [27-29], face recognition and visual prosopagnosia [30], classical and operant conditioning [29, 31], and many many more. Though such connectionist modelling is often far abstracted from computational neuroscience the basic principles of spreading activation (rather than localism) would seem to be highly plausible biological mechanisms capable of accounting for a great deal of psychological phenomena. As Page states: "I make no claim to be the first to note each of these properties; nonetheless, I believe the power they have in combination has either gone unnoticed or has been widely underappreciated." [32][p. 450]

The power of such models to explain embodied cognition is highly limited by their assumption of localist representation which bypasses important questions such as where these representations come from and what from they take. Indeed for many the idea of an internal representation is a Cartesian regression with no explanatory power whatsoever, as Harvey puts it: "The gun I reach for whenever I hear the word 'representation' has this engraved on it: 'When ***P*** is used by ***Q*** to represent ***R*** to ***S***, *who is Q and who is S?* If others have different criteria for what constitutes a representation, it is incumbent on them to make this explicit" [33]. While we do not wish to delve into the representation debate here it is clear that localist representation is neither present in neurobiology nor forthcoming in modelling without introducing significant design bias. Detectors can often be achieved by supervised training or complex statistical analysis, however the design choice of which features to detect will always be a limiting factor on such approaches. Restricting modelling methods to the more biologically plausible varieties of plasticity raises a serious problem concerning the separation of features in data streams. This is *the problem of marginal regularity*, as Kirsh [34] puts it; "what if so little of regularity is present in the data that for all intents and purposes it would be totally serendipitous to strike upon it? It seems … that such a demonstration would constitute a form of the poverty of stimulus argument" [p. 317]. As anyone working in the fields of robotics, machine learning, or pattern recognition will tell you, this 'what if' is in fact the case for the vast majority of data and especially for the kinds of things that we humans seem to perceive so effortlessly.

The *problem of marginal regularity* results from the fact that most interesting features of a data stream, be it sensory or whatever, are not explicitly represented in that data stream, they are in fact relational by which we mean that their presence is indicated by specific relationships between the 'bits' of the data streams. For example the image of a cup is distributed over many pixels or retinal cells, the majority of which take very similar values for a range of different images. What changes between each image is the relationships between the pixels; however, these relationships can change also for the same cup seen from different angles or different distances or in different parts of the image. Under traditional static conceptions of vision and data analysis such problems often seem insurmountable even given implausible supervision (c.f. [35]).

Enactive sensorimotor theories such as that proposed by Alva Noë [13, 36] and introduced in section 2 explicitly reject such a static view of perception and propose that rather than looking for statistical regularities in snapshot images, that instead we search for patterns of contingency between actions and sensory streams. That is to say we no longer identify environmental features such as objects and affordances by the sensory regularities they provide, as we know these are not fixed anyway, instead we look for regularities in the ways that sensory streams change over time relative to our actions. Learnt or otherwise acquired profiles of such changes is what Noë refers to as sensorimotor knowledge, and the application of this, in prediction, Noë argues is the basis of perception. Similarly Gallese and Lakoff [7] argue that the association of sensory and motor areas allows for the simulation of actions leading to the prediction of sensory consequences. Such stimulation of sensory and motor areas with predicted rather than actual sensory data also conforms with the simulation hypothesis [19] and various other theories of cognition.

While sensorimotor theories would seem to provide a way out of the problem of marginal regularity, as we have argued elsewhere [37], the profiling of such changes still requires consistent tracking over time before the profiles can be learned from which consistent tracking is supposed to follow. Thus the proposed solution can easily become circular. To clarify this point with an example, if I know this sub section of sensory data right now is a cup (or whatever) then I need to be able to track the cup over some changes in order to make a profile of those changes in order to be able to track the cup. In part this circularity comes from not completely letting go of the static snapshot perspective, the way out is to recognise from experience that 'this' set of changes over time happens sometimes in coincidence with performing action X, AND that 'this other' set of changes consistently coincides with performing action Y, when the original set coincide with action X. Thus there are consistent clusters of relationships and actions that happen together, when they do occur we can infer the presence of some external object or more abstractly an external situation or event.

The enactive solution we propose then is to find patterns in the dynamics of sensory and motor streams over time; however, time introduces another important problem to the modelling of sensorimotor perception. *The temporal problem* as a variation of the *credit assignment problem* results from the fact that the result of an action may not be immediate, and may in fact result from a sequence of actions. The problem then is how to discover which subset of the actions performed is actually responsible for this sensory change. As a minimal requirement then *the temporal problem* necessitates the inclusion of some form of memory such that past events in the sequence are available to take part in the formation of profiles of change and the marginal regularity problem requires some consistent tracking of features as our sensory contact with them changes over time. Thus we argue for the inclusion of memory and the transformation into warped high dimensional spaces to maximise the availability of features from which to construct temporal profiles leading to perception of features and affordances in the environment. In Section 5 we will develop a model that does exactly this, but first, in the next section we introduce sustained inattentional blindness as this will be a bi-product of our predictive models.

4 Sustained Inattentional Blindness

Our perception of the world around us is subject to manipulation, even to the extent that we can be experientially blind to highly salient and temporally extended events. We can even be unaware of things we are looking directly at. Simons and Chabris [38] note that "we perceive and remember only those objects and details that receive focused attention." p. 1059. Though it is not entirely clear in this context what attention is, similar claims have been made by many researchers e.g. [13, 36, 39]. This effect is demonstrated most startlingly in an experiment on sustained inattentional blindness by Simons and Chabris [40] in which human subjects watch a video showing two intermingled groups of people, one dressed in white and the other in black, each passing a basketball between members of their own group. Subjects are asked to count how many times the ball is passed by one particular group (either those dressed in white or those dressed in black depending on which condition the subject is in). Somewhat surprisingly, many "observers fail to notice an ongoing and highly salient

but unexpected event…[a] Gorilla walked from right to left into the live basketball passing event, stopped in the middle of the players as the action continued all around it, turned to face the camera, thumped its chest, and then resumed walking across the screen" p. 1069. Observers in this study "were consistently surprised when they viewed the display a second time, some even exclaiming, 'I missed *that*!?' " p. 1072. In this, and other psychological experiments, the effect of similarity between the attended (the team the subject is watching), distracter (the other team) and unexpected objects (the gorilla) has been systematically varied showing that close similarity between the attended and unexpected objects reduces the occurrence of inattentional blindness [40-43]. For example subjects attending to the team dressed in white were more likely to miss the black-haired gorilla than subjects attending to the team dressed in black. Most et al [44] vary the luminance of the attended and unexpected objects showing that increasing similarity (in terms of luminance) decreases the likelihood of failing to detect the unexpected object. In a different task Koivisto and Revonsuo [43] ask subjects to count how many times balls of one colour bounce of the edge of a computer screen, while balls of a different colour also bounce around the screen (see figure 1 below). In this task the unexpected object appears on the left of the screen and travels across it until it exits on the right. Subjects engaged in the counting task often miss the unexpected object and thereby display sustained inattentional blindness. In a number of experiments Koivisto and Revonsuo [43] systematically vary the number of distracter objects and their similarity to the attended objects showing that (a) distracter objects have little or no effect and that (b) sustained inattentional blindness can occur even in the absence of any distracters. For simplicity sake it is this scenario, with no distracters that we will focus on here.

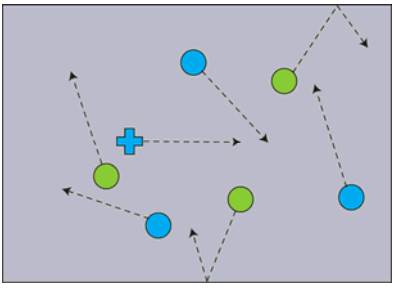


Fig. 1. Illustration of Koivisto & Revonsuo's task. Human subjects count how many times the green (lighter) balls bounce, while ignoring the blue (darker) balls. The unexpected object, here a blue cross moves across the screen, often undetected.

5 Modelling a Cortical Hierarchy

Following discussion of the connectivity between cortical microcolumns in different regions of cortex in section 1, and noting that others model cortical microcolumns in far more detail than they are treated herein, we are not the first to suggest that dynamic reservoirs such as those found in Liquid State Machines or Echo State Networks, capture many of their properties. In fact the Liquid State Machine originates

from attempts to model the neuroscientific data produced by Markram et al. [45] and Gupta et al. [46] on the cortical micro-columns of rat somatosensory cortex. While the number of cells in a microcolumn and density of each kind vary between regions, certain features of microcolumns remain constant. Of particular interest here is that micro-columns are sparsely internally interconnected, that they are observed not to be chaotic, and not to implement stable attractor dymanics. Shepherd [47] and Douglas & Martin [48] provide neuroscientific examples highlighting the stereotypical circuitry found in these structures and the wide range of tasks they seem to be involved in. While there is a great deal more structural detail to the cortical micro-column, some of which is currently being modelled elsewhere, Maass et al. [49-51] provide an abstract model using random connectivity between 'leaky integrate and fire' neurons with both static and dynamic synapse models, the parameters of which are based on Gaussian distributions of the data from Markram et al. and Gupta et al.'s experiments. The resulting model, the Liquid State Machine (LSM) has been shown to possess a number of important computational properties, acting as an analogue memory, and as a recursive kernel function [49-51]. Simpler models preserving these computational aspects of the cortical micro-column have been developed such as the Echo State Network (ESN) [52-55] which we implement here. While the ESN model is somewhat removed from the anatomical detail of the cortical micro-column, we view it as a useful abstraction preserving particular properties of the underlying neuroscience while providing significant advantages in terms of lower computational cost.

To implement an ESN we generate a random valued and sparse (30%) randomly connected weight matrix for 100 neurons. The randomly achieved weight matrix $\boldsymbol{W}$ of the resulting network is restricted to have a spectral radius of less than one, i.e. $| \lambda_{max} | < 1$, where λ_{max} is the eigenvalue of $\boldsymbol{W}$ which has the largest absolute value, which guarantees a null state attractor. Similarly $| \lambda_{min} | > -1$ where λ_{min} is the smallest absolute eigenvalue of $\boldsymbol{W}$. As the separation property is also preserved the ESN is here viewed as a computational simplification of the biologically derived LSM architecture. Unlike Jaeger [52-55], who uses a CTRNN, we update the neurons according to simple discrete time dynamics using the following standard equations.

$$a_i = \Sigma y_j w_{ij}$$

$$y_i = \frac{1}{1 + e^{(-ai)}}$$

Next we turn to the uncontroversial and far from new idea that the cortex is loosely structured in a hierarchical manner, at least with respect to major pathways of inter-micro-column connectivity. For example micro-columns in area V1 of the visual cortex are heavily connected to area V2 (about half the size of V1), and then to V4 (again about half the size) and from there to the Inferotemporal-cortex (a larger area receiving input from several different modalities). The computational properties of the ESN rely in part on the input being significantly smaller than the size of the ESN, thereby forcing an expansion of dimensions in the data stream and fostering the linear separation of relational features in that input stream. Stacking such systems into a hierarchy presents a problem in that the output from a reservoir is the size of that

reservoir, and feeding the output from several into one would require increasingly large reservoirs at each level of the hierarchy in order to preserve a dimension expansion at each step. The solution we adopt here is to autonomously compress the state of each ESN using a Self-Organizing Map (SOM) [56]. Each dimension of the map can be used as an output so that we effectively output the address of the winning SOM unit.

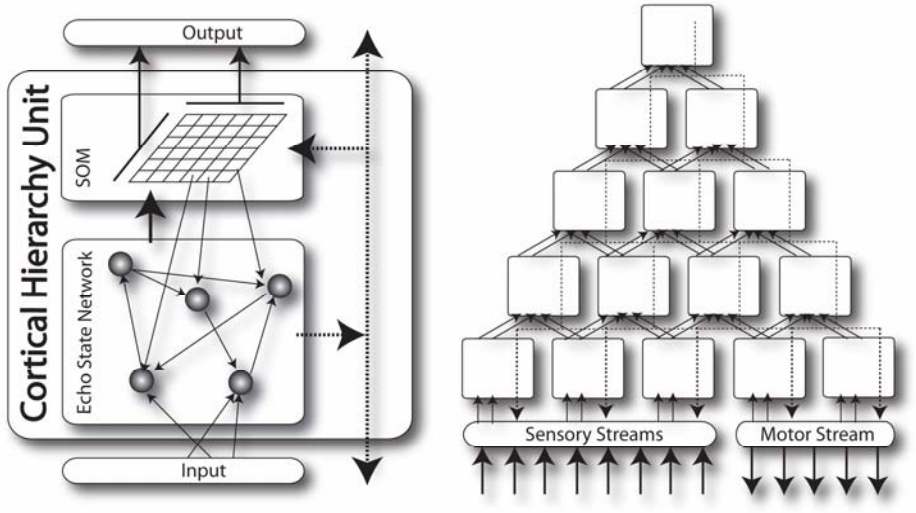


Fig. 2. (Left) showing the internal structure of each unit in the cortical hierarchy. Input flows into an ESN which is then classified by a SOM. The SOM projects back into the ESN and the address of the winning SOM unit is passed on as output. Additionally associative connections are learned between the ESN and the SOMs of other units. **(Right)** the overall structure of the whole hierarchy. Neighbouring units pass their outputs to the same unit in the next layer of the hierarchy. Associative connections are between the ESN of one unit and the SOMs of the units that that unit connects to.

This approximates a principle component analysis passing variance information on to the next layer. While we fully recognize that this dimension reduction throws away a great deal of information, the reduced output then combines with the output from neighbouring units as input to an ESN at the next level of the hierarchy where the same process is repeated again (see fig 2). This allows for the discovery of new input features, specifically those relying on relational properties between the parts of the input stream kept separate at all previous levels. SOM units then also provide normal input via sparse random connectivity back into the ESN driving that SOM.

As discussed in section 1 the development of functional regions of cortex results from plasticity refining and altering the connectivity between microcolumns within connected regions of cortex. This plasticity is modelled here using perceptrons autonomously trained to predict, from the activity of one ESN, what state connected

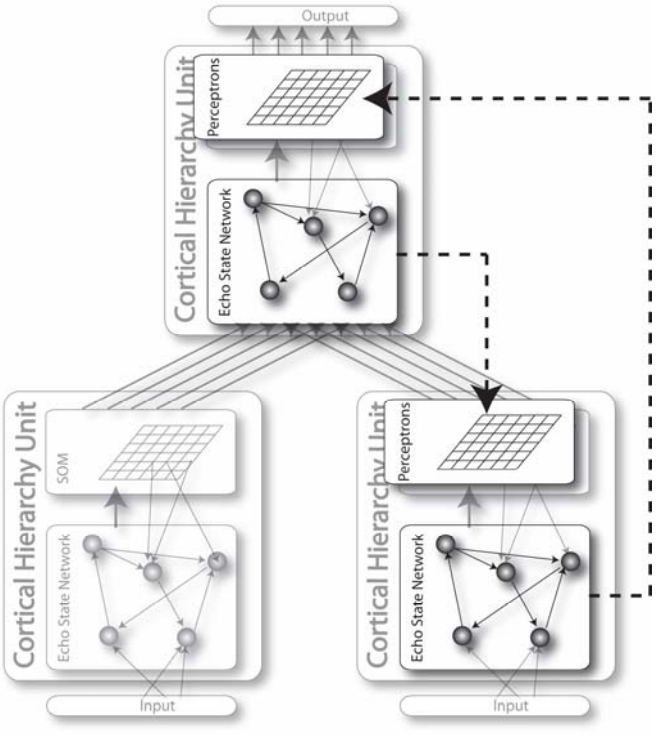


Fig. 3. Highlighting the plasticity between connected columns. Perceptrons are trained to predict the state (SOM) of one column, from the ESN activity of other connected columns.

columns are in. Thus based on the information available in one column, predictions are made as to what the likely state of other columns, having other information available to them, is likely to be (see Figure 3). The output and weight changes to each perceptron were calculated using the following standard formulas.

$$y_i = \Sigma\, y_j\, w_{ij}$$

$$\Delta\, w_{ij} = \alpha\, (y_i - target_i)\, y_j$$

5.1 Experimental Setup

Following a model based account of sustained inattentional blindness presented in [1, 2], we here provide an extension of the same experiment using the hierarchy model just described. We constructed a small hierarchy with three cortical units in the first level, two in the second level, and one in the third level. In the experiments detailed herein we simplified Koivisto and Revonsuo's task in the following ways. Firstly we removed all distracter objects as the number or presence of distracter objects was found not to significantly alter the extent of sustained inattentional blindness in experiments carried out in [43]. Secondly we reduced the number of attended objects to 1 so as to simplify the modelling task. The visual area was then divided up

into a 4 x 4 grid and the average green or blue pixel values of each cell provided two inputs respectively from each cell. This provided a total of 32 inputs to an ESN at every time step. The task here is to constantly track the vertical aspect of the direction of the attended simulated object, i.e. is the blue ball moving upward or downward (ignoring left and right velocity).

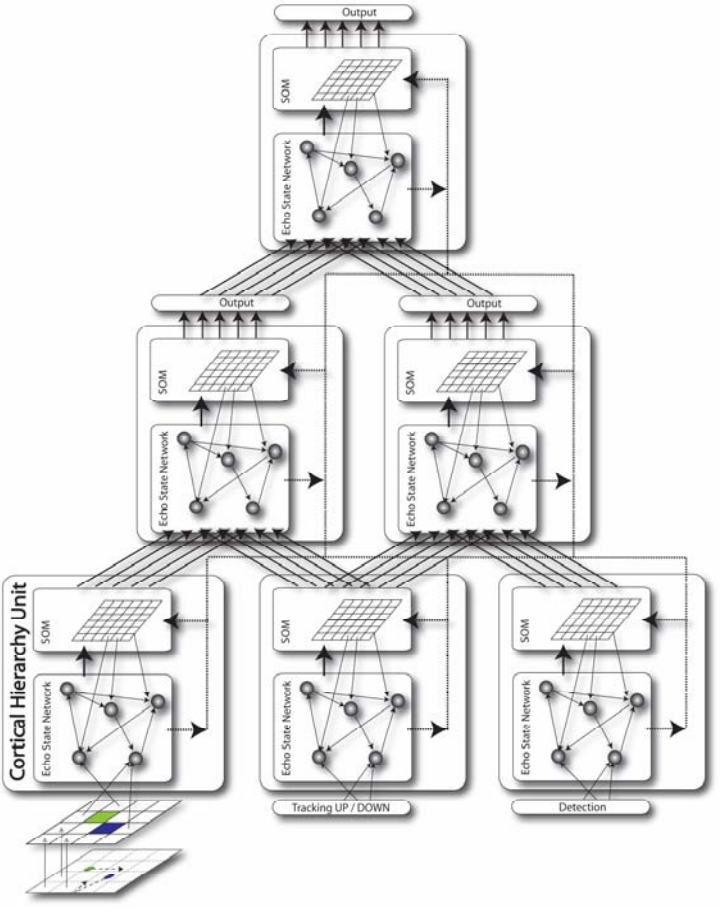


Fig. 4. The 6 unit hierarchy. Visual input is averaged over a 4 by 4 grid and then passed as input to the first cortical hierarchy unit. Tracking information and detection of the unexpected object are fed as input to the 2nd and 3rd units. Activity flows up the hierarchy driven by this input. Activity also flows down the hierarchy via the learnt associative connections, see text for a full explanation.

Our two conditions are then whether we allow top down activity to reach the unit receiving the image as input. In the first stage of the experiment we provide input for a total of 10000 time steps during which the model is 'conditioned', in that the model indirectly learns the association between the different inputs. During this stage we

also record the SOM activity of the tracking and detection units and note a strong correlation between the input state and the SOM output. In stage 2 we enter a testing phase in which the tracking and detection input is removed but the visual input continues. We record the SOM activity of the tracking and detection units, recording a correct output at every time step that the observed SOM activity corresponds with the noted correlation of what the actual tracking and detecting input should be. Thus if the association has been learnt then the hierarchy should reproduce the same activity in the tracking and detecting unit SOMs even in the absence of that input. During stage 2 the perceptron learning was also disabled. We repeated the whole experiment with feedback to the first unit enabled and disabled.

5.2 Results

As can be seen from figure 5 below, performance at tracking was considerably improved by feedback from the hierarchy. This shows that top down predictive information aids discrimination in correlated tasks. As can be seen from figure 6, performance at detection is hindered by the same feedback that improves tracking performance. We conducted a repeated measures ANOVA on this data and found significant main and interaction effects. Where the effect of feedback on tracking had a probability of $p < 0.001$, the effect of feedback on detection had a probability of $p < 0.05$ all interaction effects had a probability of $p < 0.001$. As can be seen from fig 5 and 6, feedback improved performance at tracking while degrading performance at detection, thus we have the sustained inattentional blindness phenomena. During engagement in a difficult task, predictive feedback is necessary to produce adequate performance at that task but has the effect of making detection of an otherwise detectable object less likely.

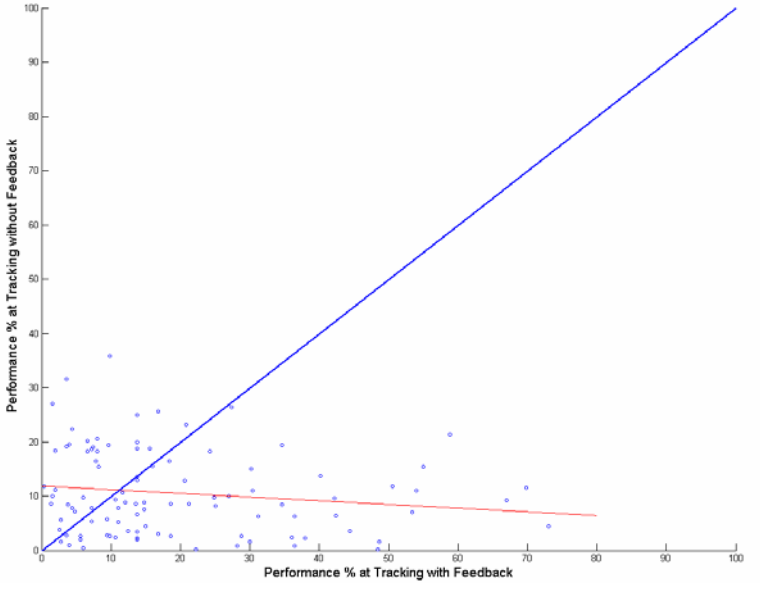


Fig. 5. Scatterplot showing the hierarchies performance at tracking with and without feedback. Note that the scale here is adjusted so that 0% indicates chance levels of performance.

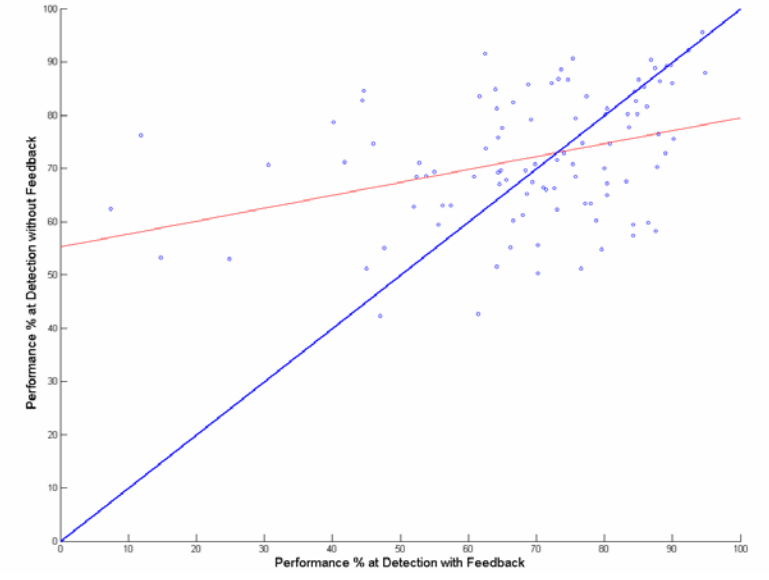


Fig. 6. Scatterplot showing the hierarchies performance at detecting the unexpected object with and without feedback. Note that the scale here is adjusted so that 0% indicates chance levels of performance.

6 Discussion, Whats Going on?

The model presented here draws on biological and philosophical theories of the relation between sensormotor knowledge discussed in sections 1 and 2 to provide a scalable model of sensorimotor learning, one by-product of which is the inattentional blindness phenomenon. In this model, the role of each micro-column, is to identify relational patterns in and over time, from the activity of the input streams or columns in the layer below. Having identified these patterns, a spreading activation between layers provides anticipatory input from above and classification from below. The hierarchical model presented learns correlations between its inputs, without introducing supervision bias, allowing for the top down spreading of activation to aid, or hinder, the identification of relational features patterns and sequences. This is closely related to the simpler model of inattentional blindness presented in [1, 2] however we here provide a minimal hierarchical implementation to clarify and provide a more plausible account of where the feedback comes from. This model is very closely linked with the enactive account of sensorimotor perception, in as much as experienced correlations between input streams are learned and then provide the basis for a spreading activation providing prediction or anticipation of what the unobserved or missing input could be. This is however, not done directly to the input but rather manifests in the form of tuning input filters (ESN's) to improve separation of the anticipated features at the cost of less separation of other features.

As a tentative explanation of why the sustained inattentional blindness phenomena is observed in this model we can analyse the effect of feedback on a single ESN.

Clearly the addition of new inputs to an ESN correlated with some feature, changes the attractor landscape of the ESN and moves the trajectory in state space (for a fuller analysis of this see [2]). However the Euclidean distance between input with that feature and input without that feature is enlarged by the presence of the correlated input. This means that the pointwise separation between ESN states following input with or without those inputs is also greater. This typically leads to enhanced separation of the streams and facilitates linear separation by a perceptron. This accounts for the improved performance at tracking (see figure 7 upper graph). As for degrading performance at detection we can see that an input image with one ball on it provides a different level of input than an image with two balls on it (one being the unexpected stimulus). This difference in input magnitude provides further separation of the following ESN states facilitating normal detection of the unexpected object. With the presence of an uncorrelated feedback input, however, this separation is distorted leading to loss in performance at detecting (see figure 7 lower graph).

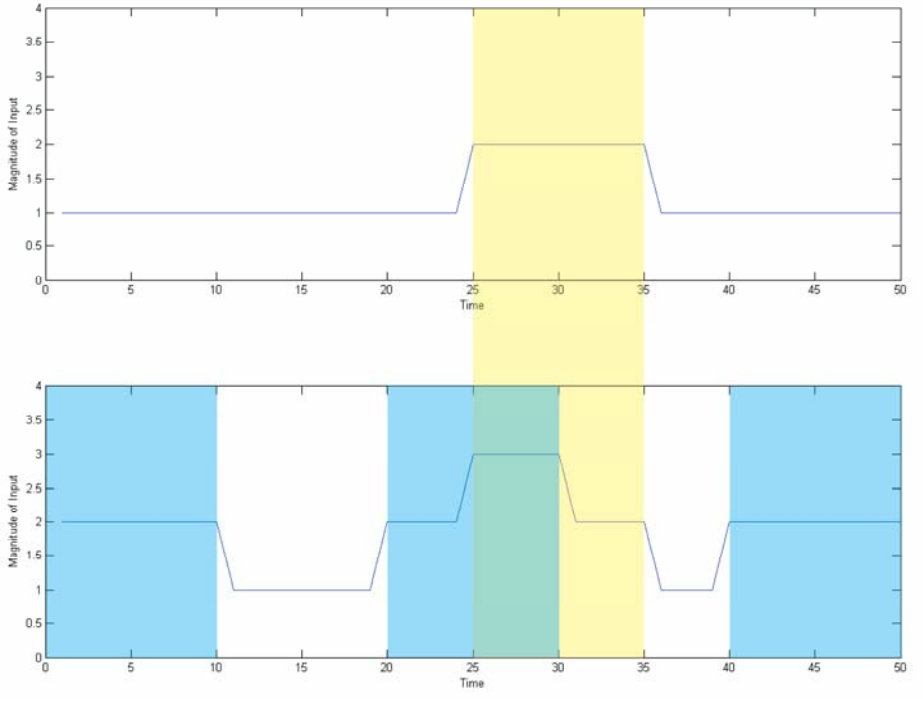


Fig. 7. The magnitude of the input to the visual ESN. In the case of tracking, the magnitude of input remains constant whether the ball is moving up or down, however, when the unexpected object appears the magnitude of input goes up thus the point-wise separation of these states is increased (**Top figure**). When feedback from tracking is made available, the magnitude (assuming correct feedback) as the ball moves up or down is different (e.g. 0-10 vs 10-20 in the **lower figure**), however the separation of the uncorrelated unexpected input is reduced (25-35 in the lower figure).

Acknowledgements

This work was supported by a European Commission grant to the project "*Integrating Cognition, Emotion and Autonomy*" (IST-027819, www.iceaproject.eu), as part of the European *Cognitive Systems* initiative.

References

1. Morse, A.F.: Neural Models of Prediction and Sustained Inattentional Blindness. In: NCPW11 the Neural Computation and Psychology Workshop. World Scientific, Oxford (in press)
2. Morse, A., Lowe, R., Ziemke, T.: Manipulating Space: Modelling the Role of Transient Dynamics in Inattentional Blindness. Connection Science (accepted)
3. Downing, K.L.: Neuroscientific implications for situated and embodied artificial intelligence. Connection Science 19, 75–104 (2007)
4. Downing, K.L.: Predictive models in the brain. Connection Science (in press)
5. Fleischer, J.G.: Neural correlates of anticipation in cerebellum, basal ganglia, and hippocampus. In: Butz, M.V., Sigaud, O., Pezzulo, G., Baldassarre, G. (eds.) ABiALS 2006. LNCS, vol. 4520, pp. 19–34. Springer, Heidelberg (2007)
6. Hawkins, J., Blakeslee, S.: On Intelligence. Times Books (2004)
7. Gallese, V., Lakoff, G.: The brain's concepts: The role of the sensory-motor system in reason and language. Cognitive Neuropsychology 22, 455–479 (2005)
8. Rumelhart, D.E., McClelland, J.L.: The PDP Research Group, vol. 2 (1986)
9. Hebb, D.O.: The Organization of Behavior: A Neuropsychological Theory. John Wiley & Sons, Chichester (1949)
10. Karmiloff-Smith, A.: Why Babies' Brains Are Not Swiss Army Knives. In: Rose, H., Rose, S. (eds.) Alas, poor Darwin, pp. 144–156. Jonathan Cape, London (2000)
11. Sharma, J., Angelucci, A., Sur, M.: Induction of visual orientation modules in auditory cortex. Nature 404, 841–847 (2000)
12. Mountcastle, V.B.: An Organizing Principle for Cerebral Function: The Unit Model and the Distributed System. In: Edelman, Mountcastle (eds.) The Mindful Brain. MIT Press, Cambridge (1978)
13. Noë, A.: Action in Perception. MIT Press, Cambridge (2004)
14. Morse, A., Ziemke, T.: Cognitive Robotics, Enactive Perception, and Learning in the Real World. In: CogSci 2007 - The 29th Annual Conference of the Cognitive Science Society, pp. 485–490. Erlbaum, New York (2007)
15. Maturana, H., Varela, F.: The tree of knowledge, revised edition. Shambhala, Boston (1992)
16. Gehringer, W.L., Engel, E.: Effect of ecological viewing conditions on the Ames' distorted room illusion. Journal of Experimental Psychology: Human Perception and Performance 12, 181–185 (1986)
17. Clark, A., Grush, R.: Towards a Cognitive Robotics. Adaptive Behavior 7, 5 (1999)
18. Grush, R.: The emulation theory of representation: Motor control, imagery, and perception. Behavioral and Brain Sciences 27, 377–396 (2004)
19. Hesslow, G.: Conscious thought as simulation of behaviour and perception. Trends in Cognitive Sciences 6, 242–247 (2002)
20. Kunde, W., Elsner, K., Kiesel, A.: No anticipation–no action: the role of anticipation in action and perception. Cognitive Processing 8, 71–78 (2007)

21. Gibson, J.J.: The Ecological Approach to Visual Perception. Houghton Mifflin, Boston (1979)
22. Barsalou, L., Breazeal, C., Smith, L.: Cognition as coordinated non-cognition. Cognitive Processing 8, 79–91 (2007)
23. Fodor, J.: The language of thought. Harvard University Press, Cambridge (1975)
24. Fodor, J., Pylyshyn, Z.: Connectionism and cognitive architecture: A critical analysis. Connections and Symbols, 3–71 (1988)
25. Bruce, V., Burton, A.M., Craw, I.: Modelling face recognition. Philosophical Transactions of the Royal Society, B 335, 121–128 (1992)
26. Burton, A.M., Bruce, V.: Naming Faces and Naming Names: Exploring an Interactive Activation Model of Person Recognition. Memory for Proper Names (1993)
27. Bruce, V., Burton, M., Carson, D., Hanna, E., Mason, O.: Repetition Priming of Face Recognition. Attention and Performance XV: Conscious and Nonconscious Information Processing (1994)
28. Morse, A.F.: Psychological ALife: Bridging The Gap Between Mind And Brain; Enactive Distributed Associationism & Transient Localism. In: Cangelosi, A., Bugmann, G., Borisyuk, R. (eds.) Modeling Language, Cognition, and Action: Proceedings of the ninth conference on neural computation and psychology, pp. 403–407. World Scientific, Singapore (2005)
29. Morse, A.F.: Cortical Cognition: Associative Learning in the Real World. D Phil. Thesis, Department of Informatics, University of Sussex, UK (2006)
30. Burton, A.M., Bruce, V., Hancock, P.J.B.: From pixels to people: A model of familiar face recognition. Cognitive Science 23, 1–31 (1999)
31. Morse, A.F.: Autonomous Generation of Burton's IAC Cognitive Models. In: Schmalhofer, Young, Katz (eds.) EuroCogSci 2003, The European Cognitive Science Conference. LEA Press (2003)
32. Page, M.: Connectionist modelling in psychology: A localist manifesto. Behavioral and Brain Sciences 23, 443–467 (2001)
33. Harvey, I.: Untimed and Misrepresented: Connectionism and the Computer Metaphor. AISB Quarterly 96, 20–27 (1996)
34. Kirsh, D.: From connectionist theory to practice. In: Connectionism: Theory and practice. Oxford Univ. Press, New York (1992)
35. Clark, A., Thornton, C.: Trading spaces: Computation, representation, and the limits of uninformed learning. Behavioral and Brain Sciences 20, 57–66 (1997)
36. O'Regan, K., Noë, A.: A sensorimotor account of visual perception and consciousness. Behavioral and Brain Sciences 24, 939–1011 (2001)
37. Morse, A.F., Ziemke, T.: Cognitive Robotics, Enactive Perception, and Learning in the Real World. In: CogSci 2007 - The 29th Annual Conference of the Cognitive Science Society. Erlbaum, New York (2007)
38. Simons, D.J., Chabris, C.F.: Gorillas in our midst: Sustained inattentional blindness for dynamic events. Perception 28, 1059–1074 (1999)
39. Rensink, R.A., O'Regan, J.K., Clark, J.: To see or not to see: the need for attention to perceive changes in scenes. Psychological Science 8(5), 368–373 (1997)
40. Simons, D., Chabris, C.: Gorillas in our midst: Sustained inattentional blindness for dynamic events. Perception 28, 1059–1074 (1999)
41. Most, S., Simons, D., Scholl, B., Jimenez, R., Clifford, E., Chabris, C.: How Not to Be Seen: The Contribution of Similarity and Selective Ignoring to Sustained Inattentional Blindness. Psychological Science 12, 9–17 (2001)

42. Most, S., Scholl, B., Clifford, E., Simons, D.: What you see is what you set: Sustained inattentional blindness and the capture of awareness. Psychological Review 112, 217–242 (2005)
43. Koivisto, M., Revonsuo, A.: The role of unattended distractors in sustained inattentional blindness. Psychological Research 72, 39–48 (2008)
44. Most, S.B., Simons, D.J., Scholl, B.J., Jimenez, R., Clifford, E., Chabris, C.F.: How Not to Be Seen: The Contribution of Similarity and Selective Ignoring to Sustained Inattentional Blindness. Psychological Science 12, 9–17 (2001)
45. Markram, H., Wang, Y., Tsodyks, M.: Differential signaling via the same axon of neocortical pyramidal neurons. National Acad. Sciences 95, 5323–5328 (1998)
46. Gupta, A., Silberber, G., Toledo-Rodriguez, M., Wu, C.Z., Wang, Y., Markram, H.: Organizing principles of neocortical microcircuits. Cellular and Molecular Life Sciences (2002)
47. Shepherd, G.M.: A basic circuit for cortical organization. Perspectives in Memory Research, 93–134 (1988)
48. Douglas, R.J., Martin, K.A.C.: Neuronal Circuits of the Neocortex. Annu. Rev. Neurosci. 27, 419–451 (2004)
49. Maass, W., Natschlager, T., Markram, H.: Real-Time Computing Without Stable States: A New Framework for Neural Computation Based on Perturbations. Neural Computation 14, 2531–2560 (2002)
50. Maass, W., Natschlager, T., Markram, H.: A Model for Real-Time Computation in Generic Neural Microcircuits. Advances in Neural Information Processing Systems 15 (2003)
51. Maass, W., Natschlager, T., Markram, H.: Computational models for generic cortical microcircuits. Computational Neuroscience: A Comprehensive Approach (2003)
52. Jaeger, H.: The echo state approach to analysing and training recurrent neural networks. German National Institute for Computer Science (2001)
53. Jaeger, H.: Short term memory in echo state networks. German National Institute for Computer Science (2001)
54. Jaeger, H.: Tutorial on Training Recurrent Neural Networks, Covering BPTT, RTRL, EKF and the echo State Network Approach. GMD-Forschungszentrum Informationstechnik (2002)
55. Jaeger, H.: Adaptive Nonlinear System Identification with Echo State Networks. Neural Information Processing Systems, NIPS (2002)
56. Kohonen, T.: The self-organizing map. Neurocomputing 21, 1–6 (1998)

Anticipation of Time Spans: New Data from the Foreperiod Paradigm and the Adaptation of a Computational Model

Johannes Lohmann, Oliver Herbort, Annika Wagener, and Andrea Kiesel

University of Würzburg
Department of Psychology
Roentgenring 11
97070 Würzburg
johannes.lohmann@stud-mail.uni-wuerzburg.de,
{oliver.herbort,wagener,kiesel}@psychologie.uni-wuerzburg.de

Abstract. To act successfully, it is necessary to adjust the timing of one's behavior to events in the environment. One way to examine human timing is the foreperiod paradigm. It requires experimental participants to react to events that occur at more or less unpredictable time points after a warning stimulus (foreperiod). In the current article, we first review the empirical and theoretical literature on the foreperiod paradigm briefly. Second, we examine how behavior depends on either a uniform or peaked (at 500ms) probability distribution of many (15) possible foreperiods. We report adaptation to different probability distribution with a pronounced adaptation for the peaked (more predictable) distribution. Third, we show that Los and colleagues' [1] computational model accounts for our results. A discussion of specific findings and general implications concludes the paper.

1 Introduction

To act successfully it is not only necessary to behave in a skillful way, but also the timing of the behavior is crucial. On a macroscopic timescale, timing of buys and sells on the real estate market can make a considerable difference. On a smaller timescale, waving down a bus or catching a ball requires us to initiate movements in time. On an ever smaller timescale, in sports, like tennis or baseball, the precise timing of a stroke is of paramount importance. And finally, even the eye blink reflex may be adjusted by mere milliseconds.

Acting successfully is even more complex because the events on which we need to react are not always fully predictable. Often, we have to learn which warning signals precede critical events and the duration of the time interval between both. Learning helps to anticipate the onset of critical events and to prepare or adjust behavior to react quickly and adequately. Thus, to understand how humans excel at a broad range of tasks and skills, we need to understand how humans adapt their behavior in time, when events occur more or less predictable.

G. Pezzulo et al. (Eds.): ABiALS 2008, LNAI 5499, pp. 170–187, 2009.

1.1 Investigation of Behavior in Time

In the lab, behavior in time has been systematically studied with the foreperiod paradigm [2,3], for a review see [4]. In these experiments, a human participant has to react as quickly as possible to a target stimulus, like the onset of a visual stimulus or a sound. To enable the formation of temporal anticipations, a warning stimulus (WS) appears at a certain point in time before the target stimulus. The interval between the WS and the target stimulus is called the foreperiod (FP).

If the FP stays constant for a while, participants are able to form expectations about the time of the appearance of the target stimulus and thus react faster upon it. Interestingly, they react the faster the shorter the FP, as the temporal resolution is higher for shorter FPs [5,6]. Only if FPs are very short, between 0 and 100ms, RTs rise again [2].

If the FP varies unpredictably from trial to trial, it is not possible to anticipate the exact time point of stimulus onset. Nevertheless, participants still have some information to prepare their response. For example, as time passes by and the target stimulus has not yet occured, the time window in which the target stimulus might appear shrinks. Thus, in the case of unpredictable FPs, participants react the faster the longer the FP [7,2,8,9].

In addition, the data reveal strong sequential effects. Compared to repetition trials (subsequent trials with identical FPs) reaction times (RT) increase if the preceding FP was longer than the current FP. The effect seems to exist only in one direction, because the preceding FP does not affect RTs if it was shorter than the current FP. Hence, RT increase if the current FP is unexpectedly shorter while RTs are unaffected if the current FP is longer than expected [10,11,12] (but see [13] for contradictory results).

The FP paradigm is well established and there is a substantial body of data that describes how humans anticipate upcoming events and adapt to the predictability of those events [14,15,16,17,18,19,20]. The aim of the current article is threefold. First, we review current theories of timing. Second, we provide new experimental data that reveals how human behavior adapts to different probability distributions of FPs. Third, we test if the data can be explained by a current model of timing [1].

The remainder of the article is structured as follows. The next section reviews different theories of timing. Then, the behavioral experiment and the novel empirical data will be described. After that, a mathematical formulation of a computational model of timing will be given and it will be compared to empirical data. A short discussion concludes the paper.

2 Theories of Timing

By now, mainly two theories of timing emerged to account for behavior in the FP paradigm: Gibbon's "scalar expectancy theory" (SET) [21] and the "behavioral theory of timing" (BeT) by Killen and Fetterman [22]. The SET is a cognitive

approach that explains temporal regularities of learned behavior by a number of information processing devices. An internal pacemaker generates variable pulses with a high frequency [23]. An accumulator accumulates the pulses up to a critical event. The number of accumulated pulses is stored in longterm–memory. To recall or reproduce a certain duration, the memorized number of pulses is compared to the currently accumulated number of pulses. The relative discrepancy of these two values determines behavior, mediated by adjustable thresholds.

In contrast, the BeT conceives the organism as moving across an invariant series of "behavioral classes" between a WS and a target stimulus. Like in SET an internal pacemaker generates pulses that cause the organism to cycle through the behavioral classes. When the target stimulus appears, the currently active class is reinforced. When the organism perceives a WS later on, it again starts to cycle through the behavioral classes. As its behavioral intensity is partially determined by the activity of the currently active behavioral class, it is then able to adjust its behavior to the experienced FP. The development of BeT as well as SET stimulated the quantitative description of behavior in time.

However, crucial aspects of timing remained unexplained. The adjustable thresholds in SET and the discriminative function of behavioral classes in BeT imply a learning process, but both theories fail to specify one. Hence, Machado [24] reformulated the BeT as a mathematical model, specifying two mechanisms of adaptation: reinforcement and exinction (for other approaches see [25,26,27]). Finally, Los and colleagues [1,28] reinterpreted the output of the model to account for human RTs. Their formulation of the BeT makes four assumptions:

1. Peaks of activation develop around the possible moments, at which the target stimulus may appear. The more FPs are used in a experimental design, the more peaks can be expected.
2. However, the temporal resolution is limited and degrades for longer intervals. Activation peaks become broader and flatter as they are more remote from the WS.
3. Reinforcement only occurs if the peak coincides with the time point of the occurrence of the target stimulus.
4. Extinction occurs at any peak that is associated with a moment prior to the relevant moment. Peaks of time points after the appearance of the target stimulus remain unchanged.

The model conceives RTs as inverse proportional to the activation at the moment the target stimulus occurs. The four assumptions explain most of the observed effects. If FPs are predictable a single activation peak is reinforced and will quickly reach its maximal amplitude. This results in faster reactions if the target stimulus appears at the expected time point. If FPs vary unpredictably from trial to trial, all FPs are reinforced or subject to extinction from time to time. Because peaks associated to shorter FPs are activated more frequently, they are also more often discounted than reinforced. This results in higher RTs for shorter FPs.

Assumption three and four explain sequential effects on RTs. All peaks that are associated with time points in the FP of one trial are subject to extinction and only the peak associated to the actual FP is reinforced in the respective trial. If the subsequent trial requires a reaction at one of these early time points, the RT tends to be higher because the respective peaks have just recently been decreased. If the FP of a subsequent trial is longer then RTs are only slightly affected.

To conclude, our understanding of human timing is based on FP experiments and models that assume that different time intervals are represented by discrete peaks of activity. This raises further questions regarding the experimental methodology and the theoretical models. First, in many FP experiments, FPs were either completely predictable or unpredictable (e.g. [1,18]). However, in everyday life, timing intervals are usually distributed around a specific duration. In most cases an interval has a certain duration but also shorter or longer intervals might be experienced from time to time. Thus, we are interested in how humans adapt their anticipation to a single-peaked probability distribution of possible FPs. Second, in most experiments, participants are exposed to a limited number of more or less distinguishable FPs. However, in many situations not only some but a continuum of FPs may be expected. Thus, we examine if the same effects can be observed if 15 possible FPs are applied in an experiment. Finally, we want to test if the computational model of Los and Agter [18] is also valid for peaked, quasi-continous distribution of FPs, or if the model needs to be further refined.

In the next section, the experimental protocol and results are described. We then give a mathematical description of the computational model we used and test the model on our data.

3 Foreperiod Experiment

In the following section, we report the protocol and results of two experiments. According to the FP paradigm, participants had to press a key upon the appearance of a a target stimulus. A WS preceded the target stimulus at random and thus unpredictable FPs, which ranged between 100ms and 1500ms. Both experiments differed in the probability distribution of the different possible FPs. In the first experiment, each FP had the same probability (uniform distribution), while in the second experiment, the 500ms FP was much more frequent than any other (peaked distribution). To measure the degree to which participants adapted their behavior, we recorded RTs, with shorter RTs indicating better timing. To evaluate the general adaptation to the different probability distributions, we compared RTs at different FPs. To further evaluate the short-term adaptation based on single trials, we analyzed sequential effects, that is, we compared RTs dependent on the FP of the current and the preceding trial.

3.1 Experimental Method

Participants. In each experiment, ten participants (uniform: 8 women and 2 men, age 19-22; peaked: 7 women and 3 men, age 19-25) volunteered to either

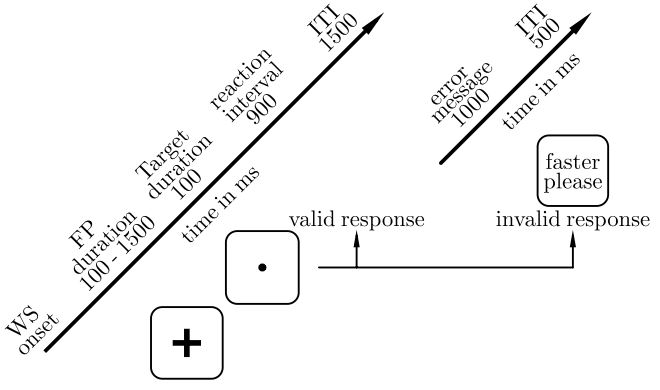


Fig. 1. The schematic time course of one trial (ITI: intertrial interval)

satisfy course requirements or in exchange for pay. All participants reported having normal or corrected-to-normal vision and were not familiar with the purpose of the experiment.

Apparatus and Stimuli. Stimuli were displayed on a 17 inch CRT monitor and RTs were recorded with an IBM-compatible computer (Pentium IV with 2.6 GHz) running E-Prime [29]. Figure 1 illustrates the trial procedure. Each trial started with the presentation of a fixation cross. The onset of the fixation cross also marked the onset of the FP. We used fifteen different FPs: 100ms, 200ms, ... , and 1500ms. After the FP, a circle (approximately 2 cm x 2 cm) was displayed as target stimulus for 100ms, followed by a blank interval of 900ms. All stimuli appeared in white on dark-grey background. Participants had to press a key with the right index finger upon appearance of the target. If participants responded within 1000ms to the target stimulus, the screen stayed blank for another 1500ms, then the next trial began. If participants failed to respond, the German words “bitte schneller” (faster, please) were displayed in red letters for 1000 ms and the next trial was initiated another 500ms later. Both experiments consisted of ten blocks with 120 trials each. Table 1 lists the distribution of foreperiods in each block for both experiments. In the uniform distribution experiment, all FP appeared with the same probability. In contrast, in the peak distribution experiment, the FP of 500ms was 46 times as likely as any other FP. The presentation order of all trials were randomized for each block and were thus unpredictable for the participants.

3.2 Experimental Results

Uniform Distribution Experiment. We aggregated data from every three FPs, resulting in the five different FP ranges (see left panel of Table 1) to enable

Table 1. Frequencies of FPs, FP ranges, and data points in each FP range (Freq) in each block of the peak and the uniform distribution experiment

	Uniform Distribution			Peaked Distribution		
FP (ms)	Freq_{FP}	FP_{Range}	$\text{Freq}_{\text{FPrange}}$	Freq_{FP}	FP_{Range}	$\text{Freq}_{\text{FPrange}}$
100	8	100 - 300		2	100 - 400	
200	8	↕	24	2	↕	8
300	8	100 - 300		2		
400	8	400 - 600		2	100 - 400	
500	8	↕	24	92	500	92
600	8	400 - 600		2	600 - 1000	
700	8	700 - 900		2		
800	8	↕	24	2	↕	10
900	8	700 - 900		2		
1000	8	1000 - 1200		2	600 - 1000	
1100	8	↕	24	2	1100 - 1500	
1200	8	1000 - 1200		2		
1300	8	1300 - 1500		2	↕	10
1400	8	↕	24	2		
1500	8	1300 - 1500		2	1100 - 1500	

better illustration and statistical analysis. We analyzed the RT and response validity[1] data using ANOVAs[2] with two within-subject factors: the FP range of the trial (FP range_{trial}) and the FP range of the preceding trial (FP $\text{range}_{trial-1}$).

Figure 2 shows RT (A, B) and response validity data (C, D). The single means for the different FP ranges are typical for unpredictable FPs: the shorter the FP, the higher the RTs, $F(4, 36) = 57.9,\ p < .001$. Also, the FP range of the preceding trial (FP $\text{range}_{trial-1}$) has a significant influence on RT, resulting in generally shorter RTs in a trial if the preceding trial also had a short FP, $F(4, 36) = 21.7,\ p < .001$. Both factors are not independent but interact: The impact of FP $\text{range}_{trial-1}$ on RT data decreases with increasing FP range_{trial}, $F(16, 144) = 4.2,\ p < .01$.

Response validity data roughly follows the same pattern (Fig. 2C, D). Error rates increase with the increasing FP range_{trial}, $F(4, 36) = 4.8,\ p < .05$. Additional, there is a significant influence of FP $\text{range}_{trial-1}$, $F(4, 36) = 0.1,\ p < .01$, but the interaction failed to reach significane, $F(16, 144) = 2.2,\ p = 0.11$.

In general, the results fit nicely to existing data and model assumptions. Participants respond the faster, the longer the FP in each trial. In addition, the data reveals asymmetric sequential effects. If the current FP is short (FP

[1] A trial response was invalid if the participant pressed the key either before the onset of the target stimulus, more than 1000ms after its onset, or not at all. As most invalid responses were due to premature key presses, invalid responses mostly reflect a higher behavioral activation.

[2] We applied the Greenhouse–Geisser correction because the assumption of sphericity was violated in our data. For clarity, we report F-values with unadjusted degrees of freedom.

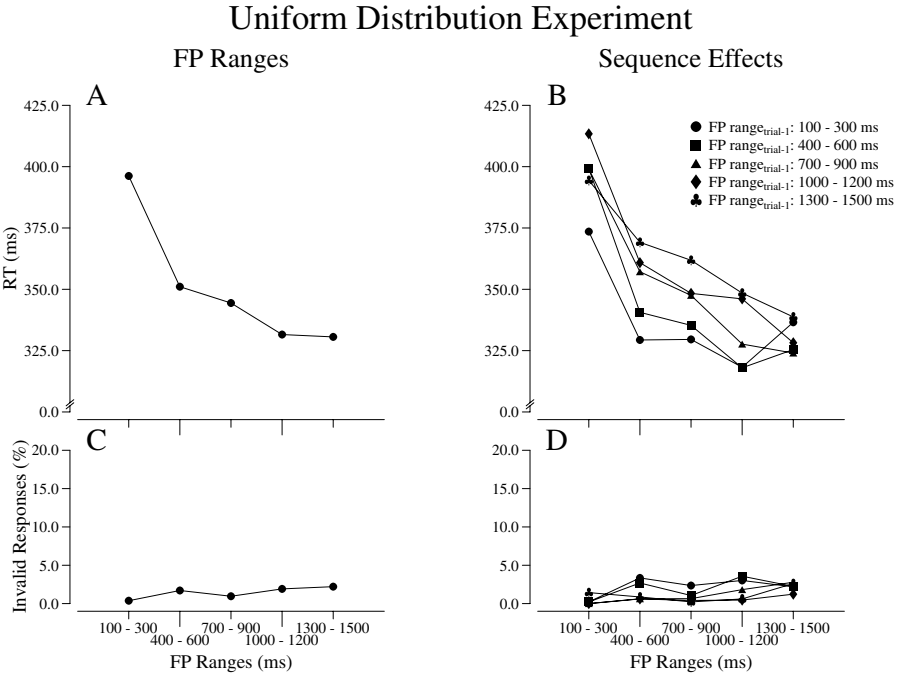


Fig. 2. The charts show the results of the uniform distribution experiment: the impact of the FP range on RT (A) and response validity (C) and the impact of specific sequences of FPs on RT (B) and response validity (D)

range 100-300) the impact of the previous FP seems to be much more severe than when the current FP is long. However, our data does not show an ordered influence of the preceding FP, as would be theoretically expected. Finally, in this experiment, the repetition of the same FP did not necessarily cause the greatest benefits for RT. Especially higher FP ranges showed the steepest increase in RT if the preceeding trial's FP was slightly longer.

Peaked Distribution Experiments. Again, we analyzed the data on the level of aggregated FP ranges as displayed in Table 1 using ANOVAs with FP range$_{trial}$ and FP range$_{trial-1}$ as within subject factors. Note that due to the non-uniform distribution of FPs (most of the trials had a FP of 500ms) the different FP ranges consist of different numbers of data points. Figure 3A, B show that RTs decrease with increasing FP range$_{trial}$, $F(3, 27) = 31.0$, $p < .001$ and depended on the preceding trials FP (FP range$_{trial-1}$), $F(3, 27) = 3.8$, $p < .05$. There is no interaction between FP range$_{trial}$ and FP range$_{trial-1}$ $F(9, 81) = 0.6$, $p = 0.67$. Interestingly, RTs seem to be generally much faster than in the uniform distribution experiment. Especially, the decrease from the shortest FP range to the next one is much more pronounced in the peaked distribution experiment than for the uniform distribution experiment, but RT decrease further

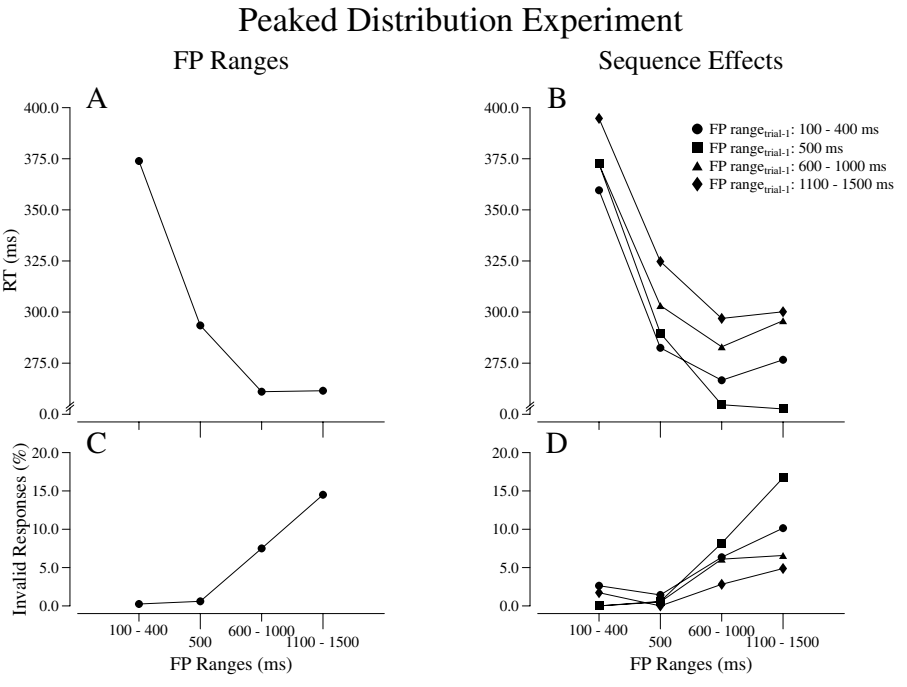


Fig. 3. The charts show the results of the peaked distribution experiment: the impact of the FP range on RT (A) and response validity (C) and the impact of specific sequences of FPs on RT (B) and response validity (D)

for the higher FP ranges. This may be due the high behavioral activation for the frequent FP of 500ms, which seems to be maintained for higher FP. The same trend can also be found in the response validity data, $F(3, 27) = 8.3,\ p < .01$. However, there was neither a significant main effect for FP range$_{trial-1}$ nor a significant interaction for response validity data, $F(3, 27) = 0.9,\ p = .48$, $F(9, 81) = 1.5,\ p = .23$. The analysis of sequence effects revealed an RT advantage if the preceding trial contained one of the shorter FP ranges. Participants responded especially fast in trials that followed a trial with the frequent FP of 500ms. We assume that the comparatively slow reactions following trials that did not contain the frequent FP 500ms may be attributed to RT costs caused by expectancy violations after trials with uncommon FPs.

4 Simulation

The following section mathematically formulates Machado's / Los and colleagues' model [1,24]. The model has a serial structure with interconnected timing nodes. Every node has two connections, one to the subsequent node and one to an output node. The links to the output node, which determines RT, are weighted, the weights are adjustable through learning processes. After the occurrence of a

WS activation is propagated through the nodes in the structure. Depending on the time elapsed since the WS occurred the single nodes of the serial structure contain a different amount of activation. Hence their contribution to the output differs over time, with a characteristic activation peak for every node. The basic structure of the model is shown in Figure 4. In this section, we now describe the propagation of activity through the nodes, the learning rules, and the response rule.

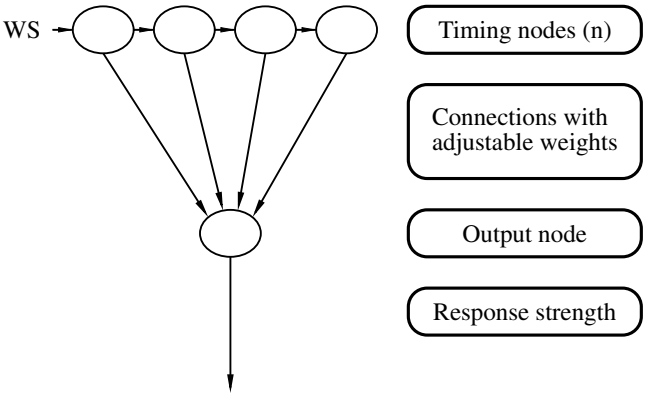


Fig. 4. The basic structure of the model adapted from Machado [24]

4.1 Formal Outline of the Model

Node Activations. When the WS appears, all activity is contained in the first node $X_0(t)$ and the remaining nodes with an n greater than 0 are not activated at all: $X_n(t) = 0 \quad n = 1, \ldots, N$, (Fig. 5A). This activation is then propagated through the system as time passes. The current activation of each of the remaining nodes depends on the activation that a given node receives from its predecessor and the activation it passes on to its successor. Figure 6A illustrates this process of a constant flow of activation by Machado's cascade analogy. The flow of activation is modeled by the following differential equations:

$$\frac{\delta}{\delta t} X_0(t) = -\lambda X_0(t) \tag{1}$$

$$\frac{\delta}{\delta t} X_n(t) = \lambda X_{n-1}(t) - \lambda X_n(t) \quad \text{for } n = 1, \ldots, N \tag{2}$$

where λ describes the time range and the speed of the activity propagation. The solution of (1) and (2) leads to the poisson density function

$$X_n(t) = \frac{e^{-\lambda t}(\lambda t)^n}{n!} \quad . \tag{3}$$

The activation of one state over time can be described as a poisson process. With the exception of the first node, the activity in each node $X_n(t)$ rises continuously,

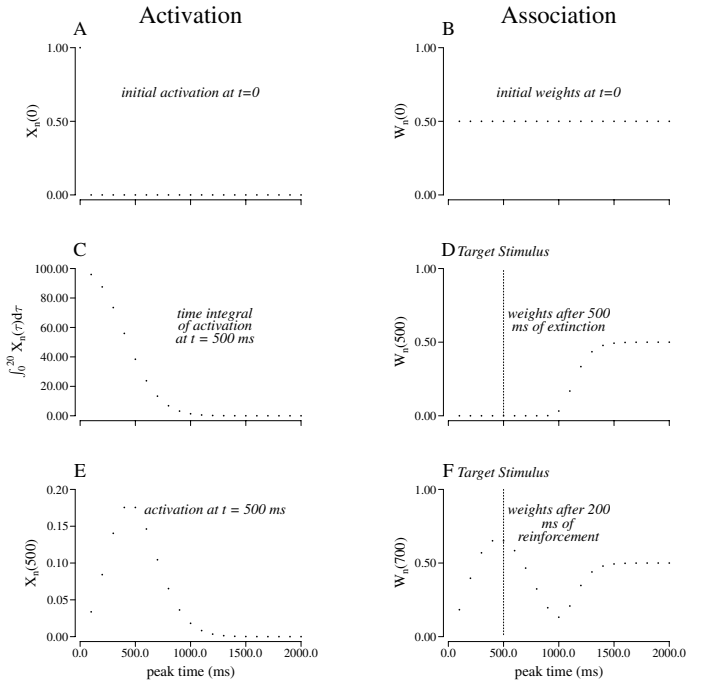


Fig. 5. The charts show different aspects of the model, occurring in a single trial with an FP of 500 ms. A: activation at $t = 0$; B: initial weight distribution; C: cumulative activation of nodes at the onset of the target stimulus ($t = 500ms$); D: Decreased weights (extinction) for nodes associated to time points before the onset of the target stimulus; E: activation of nodes at the onset of the target stimulus; F: Increased weights (reinforcement) after the end of the reinforcement period ($\lambda = 0.01,\ n = 20,\ \alpha = 2,\ \beta = 0.03,\ K = 1,\ d = 200ms$).

peaks at $t = \frac{n}{\lambda}$, and decreases afterward. The sum of activation is constant in the system, because the area of the poisson density function is $\frac{1}{\lambda}$, independent of the value of n . However, mean and variance of the activation curves are proportional to n yielding flatter and broader peaks for larger values of n. Consequently, the temporal resolution is high for short FPs and decreases for longer FPs. Figure 6B shows some activation curves for different values of n and $\lambda = 0.01$.

Extinction and Reinforcement. The weights of the links between the timing nodes and the output node are adjustable. The following section introduces the mathematical formulation of the learning rules.

The weight $W_n(t)$ of the connection between node n and the output node is subject to extinction and reinforcement during each trial. Initially, before the onset of the first trial of the experiment, no specific FP distribution can be expected and all weights are set to $W_n(0) = 0.5,\ n = 1, \ldots, N$ (Fig. 5B). All later trials start with the weight distribution that resulted from the previous trial.

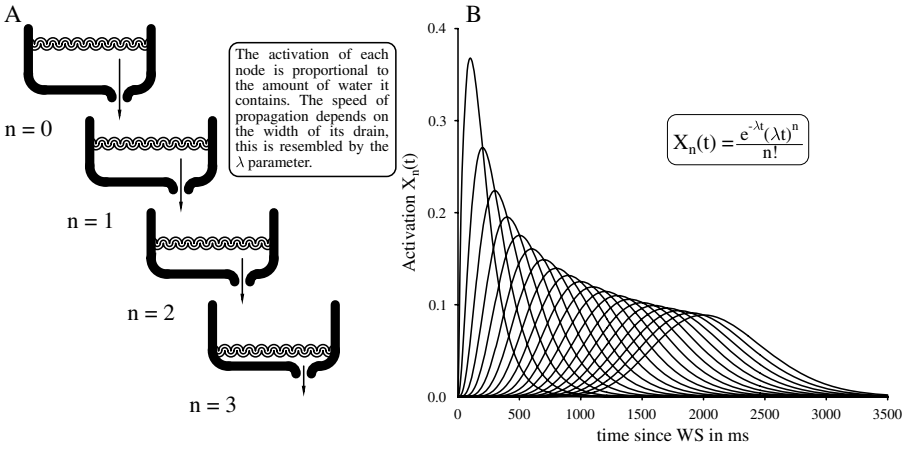


Fig. 6. A: The cascade analogy illustrates the propagation of activation through the series of nodes. B: The chart shows the poisson density distribution for different values of n and $\lambda = 0.01$.

Extinction takes place from the onset of the WS until the onset of the target stimulus. The decrease of the weights depends on the activation a node received during a trial, the activation of the node at the time the target stimulus occurs, and the initial weight of the connection. The adaption takes place dynamically, the changes are asymptotic, hence weights equal to zero are not possible, as well as weights equal to the specified upper bound. The following differential equation shows the dynamic extinction:

$$\frac{\delta}{\delta t} W_n(t) = -\alpha X_n(t) W_n(t) \quad \text{with } \alpha > 0 \text{ and } 0 \leq t \leq FP \tag{4}$$

with the following closed solution:

$$W_n(t) = W_n(0) e^{-\alpha \int_0^t X_n(\tau)\delta\tau} \tag{5}$$

where α is a learning rate parameter for the extinction process. The actual weight change is proportional to the initial weight $W_n(0)$. Extinction has a stronger effect on strong connections and only mildly affects weak connections. The weight change also depends on the cumulative activation of the respective state, that is, the whole activation that was propagated through this node in the time between WS and target stimulus (Fig. 5C). As shown in Fig. 5D, this results in a depression of all weights of nodes that are associated to time points before the appearance of the target stimulus.

Reinforcement is restricted to a fixed interval of duration d following the onset of the target stimulus at $t = FP$ and can be described through the differential equation:

$$\frac{\delta}{\delta t} W_n(t) = \beta X_n(FP)[1 - W_n(t)] \quad \text{with } \beta > 0 \text{ and } FP \leq t \leq FP + d \tag{6}$$

with the closed solution:

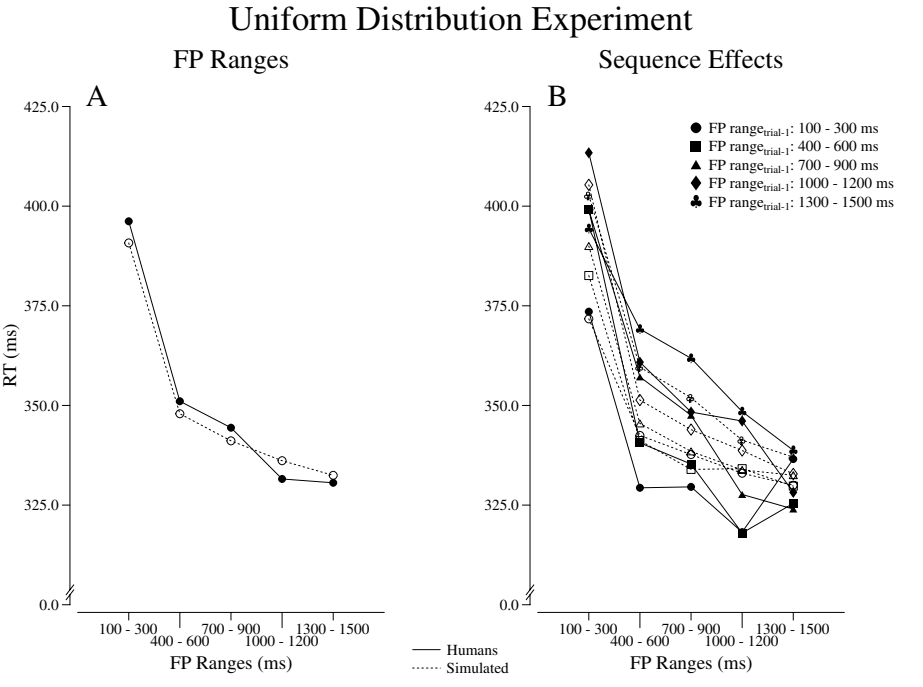


Fig. 7. The charts show simulated and empirical RTs depending on FP ranges (A) and depending on different FP ranges of the preceding and current trial (B) of the uniform distribution experiment

$$W_n(t) = K - (K - W_n(FP))e^{-\beta d X_n(FP)} \tag{7}$$

where β is a reinforcement learning rate parameter, t is the time that elapsed between WS and target stimulus, and K is the upper bound for every single weight. During reinforcement the weights at the beginning of the reinforcement period $W_n(t)$ are strengthened depending on the initial weight, the activation of the node at the time of reinforcement $X_n(t)$, and the reinforcement duration d. Figure 5E shows the activation of the nodes at the FP (i.e. at the appearance of the target stimulus) and Fig. 5F shows the resulting weights.

Response Rule. After the description of the time sensitive structure and the learning principles, we now turn to the response rule, which translates activations and weights into RTs. In our adaptation of the model[3], RT(t) is the RT that would result in a given trial if the target stimulus is displayed at time t:

[3] Note, that the response rule in [1] includes an additive term in the divisor, which was introduced to study tonic and phasic activation levels. As we do not deal with this topic here and to reduce the degrees of freedom of the model, we removed this term from the response rule.

Table 2. Uniform distribution experiment: R^2 of the prediction of the RT, averaged over sequences of FP ranges (sequential means, 25 predicted mean RTs per participant) and FP ranges (FP means, 5 predicted mean RTs per participant), for each particiant

participant	$R^2_{\text{sequential means}}$	$R^2_{\text{FP means}}$
9	0.90	0.98
3	0.87	0.98
4	0.87	0.98
6	0.82	0.96
7	0.69	0.91
5	0.68	0.81
2	0.63	0.85
8	0.58	0.99
10	0.58	0.93
1	0.57	0.99
M	0.72	0.94

$$RT(t) = RT_0 + \frac{A}{\sum_{n=1}^{N} X_n(t) W_n(t)} \tag{8}$$

where $X_n(t)$ is the activation of node n at time t, $W_n(t)$ the corresponding weight, RT_0 is an intercept, and A is a scaling coefficient. RT_0 represents the time taken by other processes contributing to the RT, like sensory stimulus processing or motor signal transmission. A is a necessary scaling factor, because the temporal regulation given by the sum in the divisor is bound between zero and K.

4.2 Simulation Method

To test if this adaptation of Los and colleagues' model [1] accounts for the behavioral data we estimated the model parameters and analyzed the overall fit to the empirical RTs. Following [1] and [24] we set the number of nodes to $N = 60$ and the reinforcement interval to $d = 200ms$. The remaining five parameters (RT_0, λ, A, α, and β) were fitted with the downhill simplex algorithm [30]. As each experimental participant received a different order of FPs and the model is sensitive to the order of FPs, we optimized the parameters of the model to predict each individual trial's RT as closely as possible (1-norm). We estimated individual parameter values for every participant. The resulting simulated RTs were aggregated similar to the empirical data to enable a direct comparison.

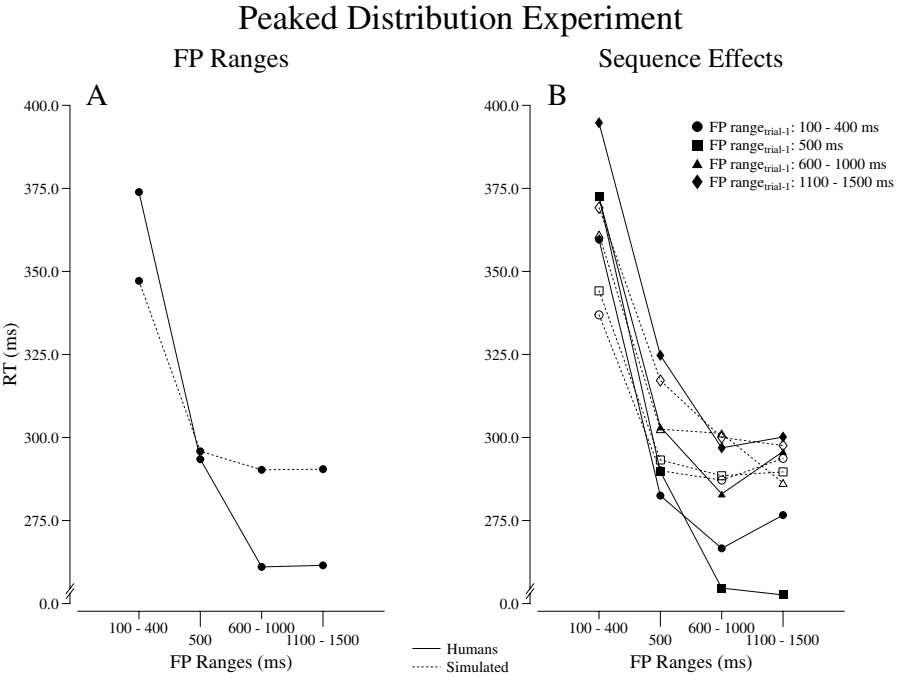


Fig. 8. The charts show simulated and empirical RTs depending on FP ranges (A) and depending on different FP ranges of the preceding and current trial (B) of the uniform distribution experiment

4.3 Simulation Results

Model Fit to Uniform Distribution Experiment. Figures 7A displays the results of our simulation of the uniform distribution experiment. The mean RTs for the different FP ranges simulated by the model correspond very well to the empirical data. Figure 7B shows the sequential effects of FP ranges for simulated and empirical RTs. Most of the qualitative features of the empirical data were reproduced by the model. However, the order of the impacts of previous trials on the short FP range 100-300ms could not be reproduced. Additionally, the impact of previous trial's FP seems to be somewhat reduced in the simulated data. Table 2 displays the amount of variance the simulated RTs can account for (indicated by R^2)[4] when considering RTs dependent on the FP and RTs dependent on the sequence of FPs (FP in trial n and trial n-1). Given the highly noisy individual data, the model acceptably reproduces the empirical RTs.

Model Fit to Peaked Distribution Experiment. Figure 8A displays the results of the simulation for the peaked distribution experiment. The empirical

[4] We used the coefficient of determination to estimate the goodness of fit of the model. $R^2 = \frac{SS_{\text{regression}}}{SS_{\text{total}}}$ is a measure for the amount of variance explained by the model.

Table 3. Peaked distribution experiment: R^2 of the prediction of the RT, averaged over sequences of FP ranges (sequential means, 25 predicted mean RTs per participant) and FP ranges (FP means, 5 predicted mean RTs per participant), for each participant

participant	$R^2_{\text{sequential means}}$	$R^2_{\text{FP means}}$
5	0.87	0.99
4	0.79	0.97
3	0.70	0.95
7	0.70	0.90
8	0.67	0.76
9	0.64	0.98
10	0.40	0.94
1	0.34	0.99
6	0.28	0.89
2	0.26	0.99
M	0.56	0.94

and simulated RTs corresponded for the different FP ranges. However, the between FP variability is smaller in the simulated data. This might be due to the high number of data points contributing to the 500ms FP. Figure 8B shows the empirical and simulated RTs as a function of FP range and FP range in the preceding trial. Again, the simulated RTs exhibit less variability than the empirical data. There are also some qualitative aspects of the empirical data that were not reproduced. Especially, the highly reduced RTs for trials following a trial with a 500ms FP could not be reproduced. We assume that this is caused by an effect which is systematically produced by the experimental design but not reflected by the model. The slower RTs for trials following a trial with a FP different from 500ms might be caused by cognitive processes, which result from the rather unexpected foreperiod in the previous trial. Probably, the model were suitable to reproduce the data if we put more weight on rare FP ranges for the fitting algorithm.

Table 3 displays the amount of variance the simulated RTs can account for (indicated by R^2) on different levels of aggregation. Similar to the results in the uniform distribution experiment, the accounted variance on the level of individual RTs is acceptable. However, the variance of the R^2 for sequential effects, which ranges between .26 to .87, is rather high.

5 Discussion

The purpose of the present study was to examine the adaptation of behavior in time to different distributions of many possible FPs. Our experimental results

show that humans are able to adjust their behavior to different predictability conditions. A comparison of both experiments reveals that RTs are much faster in the peaked distribution experiment than in the uniform distribution experiment. This implies that humans preactivate their behavioral system according to the predictability of a stimulus and that they are then able to quickly process that stimulus. Moreover, the described computational model accounts for human behavior in time under the conditions of our experiments.

5.1 Experimental Results

In detail, the conducted experiments replicated and extended typical findings. In general the RTs are the shorter the longer the FPs are. The typical sequential effects were also reproduced. RT increase with the length of the preceding FP relative to the current FP. These findings were in line with the common results reported for unpredictable FPs [7,2,8,9]. An interesting aspect of the sequential data is that participants did not respond fastest to direct repetitions of FPs. This was also true in the peaked distribution experiment where one FP was much more frequent than any other FP. Currently, we can only speculate how to interpret these findings. The shape of the different FP–RT functions might be the result of higher order sequential effects, also the FP distributions may be involved as well as the participants' ability to distinguish the FPs.

5.2 Computational Model

The computational model developed by Los and Agter [18] using the formal outlines of Machado [24] proved its ability to account for most of our experimental results. Please note that we fitted the model based on individual RTs of single trials and thus the fitting algorithm had to cope with very noisy data. We would expect even better fits if we ran several participants through identical sequences of FPs to average out RT variance that is caused by other than the preparatory mechanisms we want to study. Interestingly, the shifted minimum of the FP–RT functions was clearly reproduced. This was caused both by the model structure and the underlying learning mechanisms. The qualitative fit of the model was very good, even if the quantitative features of the empirical data could not be fully reproduced. Hence, in future research it might be beneficial to adjust the response rule to allow for a tighter replication of the data. In sum, both adjustments may improve fitting in future studies.

The supposed poisson process in connection with the applied learning rules seems to be able to account for a lot of qualitative aspects of the human ability of temporal anticipation. The quantitative fit may be improved by reformulating or extending the applied learning rules, as well as the response rule. For instance it seems to be quite simple to derive an expectancy of the length of the next FP after a single trial. The adjustment of the different weights cause the structure to be more or less "prepared" to react at different time points. The time point with the greatest product of activation and association strength could be conceived as a temporal expectancy or anticipation. The match or mismatch of

this "anticipation" with the subsequent FP could be used to predict erroneous behavior like premature responses or misses.

Please note, the presented model has five free parameters, which were used to predict five RT averages with high and 25 RT averages with acceptable accuracy. Due to the amount of free parameters, one may assume that the model can be fitted to a broad range of data. Indeed, the purpose of this paper is not to provide the most efficient model that accounts for the results but to present a model that is based on neurophysiological and psychological considerations. Given the neurologically derived architecture and the biologically plausible learning algorithms, the model may yield more explanational value than sparser, purely descriptive models.

5.3 Outlook

Another remarkable feature of the model is the possibility to adapt it to many experimental settings, which could differ from the FP paradigm. Machado proved the validity of the model in nearly all designs used to investigate the effects of temporal manipulations on the behavior of animals [24]. In all cases the structure of the model remained unchanged, only the response rule was adapted. Thus, the model reflects basic properties of the processing of temporal information in a wide range of species and behaviors [31].

References

1. Los, S., Knol, D., Boers, R.: The foreperiod effect revisited: conditioning as a basis for nonspecific preparation. Acta Psychologica 106, 121–145 (2001)
2. Bertelson, P., Tisseyre, F.: The time-course of preparation with regular and irregular foreperiods. Quarterly Journal of Experimental Psychology 20, 297–300 (1968)
3. Woodrow, H.: The measurement of attention. Psychological Monographs 5(76), 1–158 (1914)
4. Niemi, P., Näätänen, R.: Foreperiod and simple reaction time. Psychological Bulletin 89, 133–162 (1981)
5. Allan, L.G., Gibbon, J.: Human bisection at the geometric mean. Learning and Motivation 22, 39–58 (1991)
6. Wearden, J.H., Lejeune, H.: Scalar properties in human timing: conformity and violations. Quarterly Journal of Experimental Psychology 4, 569–587 (2008)
7. Elithorn, A., Lawrence, C.: Central inhibition - some refractory observations. Quarterly Journal of Experimental Psychology 11, 211–220 (1955)
8. Mattes, S., Ulrich, R.: Response force is sensitive to the temporal uncertainty of response stimuli. Perception and Psychophysics 59, 1089–1097 (1997)
9. Näätänen, R.: The diminishing time-uncertainty with the lapse of time after the warning-signal in reaction-time experiments with varying fore-periods. Acta Psycologica 34, 399–419 (1970)
10. Alegria, J.: Sequential effects of foreperiod duration: Some strategical factors in tasks involving time uncertainty. In: Alegria, J. (ed.) Attention and Performance, pp. 1–10. Academic Press, London (1975)

11. Karlin, L.: Reaction time as a function of foreperiod duration and variability. Journal of Experimental Psychology 16, 185–191 (1959)
12. Thomas, E.A.: Reaction-time studies: The anticipation and interaction of response. British Journal of Mathematical and Statistical Psychology 20, 1–29 (1967)
13. Drazin, B.: Effects of foreperiod, foreperiod variability, and probability of stimulus occurence on simple reaction time. Journal of Experimental Psychology 62, 43–50 (1961)
14. Bausenhart, K.M., Rolke, B., Hackley, S.A., Ulrich, R.: The locus of temporal preparation effects: Evidence from the psychological refractory period paradigm. Psychonomic Bulletin and Review 13, 536–542 (2006)
15. Fischer, R., Schubert, T., Liepelt, R.: Accessory stimuli modulate effects of nonconscious priming. Perception and Psychophysics 69(1), 9–22 (2007)
16. Hackley, S.A., Valle-Inclán, F.: Which stages of processing are speeded by a warning signal? Biological Psychology 64, 27–45 (2003)
17. Kiesel, A., Miller, J.: Impact of contingency manipulations on accessory stimulus effects. Perception and Psychophysics 69, 1117–1125 (2007)
18. Los, S.A., Agter, F.: Reweighting sequential effects across different distributions of foreperiods: Segregating elementary contributions to nonspecific preparation. Perception & Psychophysics 67(7), 1161–1170 (2005)
19. Miller, J., Franz, V., Ulrich, R.: Effects of auditory stimulus intensity on response force in simple, go/no-go, and choice rt tasks. Perception and Psychophysics 61, 107–119 (1999)
20. Müller-Gethmann, H., Ulrich, R., Rinkenauer, G.: Locus of the effect of temporal preparation: Evidence from the lateral readiness potential. Psychophysiology 40, 597–611 (2003)
21. Gibbon, J.: Scalar expectancy theory and weber's law in animal timing. Psychological Review 84, 279–325 (1977)
22. Killeen, P.R., Fetterman, J.G.: A behavioral theory of timing. Psychological Review 95(2), 274–295 (1988)
23. Treisman, M.: Temporal discrimination and the indifference interval: Implications for a model of the ïnternal clock: Psychological Monographs 77, 1–31 (1963)
24. Machado, A.: Learning the temporal dynamics of behavior. Psychological Review 104(2), 241–265 (1997)
25. Balkenius, C., Morén, J.: Dynamics of a classical conditioning model. Autonomous Robots 7, 41–56 (1999)
26. Grossberg, S., Schmajuk, N.: Neural dynamics of adaptive timing and temporal discrimination during associative learning. Neural Networks 2, 79–102 (1989)
27. Staddon, J.E.R., Higa, J.J.: Multiple time scales in simple habituation. Psychological Review 103(4), 720–733 (1996)
28. Los, S., Van den Heuvel, C.E.: Intentional and unintentional contributions of nonspecific preparation during reaction time foreperiods. Journal of Experimental Psychology: Human Perception and Performance 27, 370–386 (2001)
29. Schneider, W., Eschman, A., Zuccolotto, A.: E-prime user's guide. Psychology Software Tools Inc., Pittsburgh (2002)
30. Nelder, J.A., Mead, R.: A simplex method for function minimization. Computer Journal 7, 308–313 (1965)
31. Nobre, A.C., Correa, A., Coull, J.T.: The hazards of time. Current Opinion in Neurobiology 17, 465–470 (2007)

Collision-Avoidance Characteristics of Grasping
Early Signs in Hand and Arm Kinematics

Janneke Lommertzen[1], Eliana Costa e Silva[2], Raymond H. Cuijpers[1], and Ruud G.J. Meulenbroek[1]

[1] Nijmegen Institute for Cognition and Information, Radboud University Nijmegen, Nijmegen, The Netherlands
[2] Department of Industrial Electronics, University of Minho, Guimarães, Portugal

Abstract. Grasping an object successfully implies avoiding colliding into it before the hand is closed around the object. The present study focuses on prehension kinematics that typically reflect collision-avoidance characteristics of grasping movements. Twelve participants repeatedly grasped vertically-oriented cylinders of various heights, starting from two starting positions and performing the task at two different speeds. Movements of trunk, arm and hand were recorded by means of a 3D motion-tracking system. The results show that cylinder-height moderated the approach phase as expected: small cylinders induced grasps from above whereas large cylinders elicited grasps from the side. The collision-avoidance constraint proved not only to be accommodated by aperture overshoots but its effects already showed up early on as differential adaptations of the distal upper limb parameters. We discuss some implications of the present analysis of grasping movements for designing anthropomorphic robots.

1 Introduction

Grasping objects is a task that people perform almost on a continuous basis. Such a seemingly simple task proves extremely complex when it comes to computationally describing the mechanisms that allow us to do so. For example, when developing anthropomorphic robots. Numerous studies have scrutinised the kinematics of this basic human motor skill, often quantifying typical kinematic landmarks such as peak velocities of the wrist trajectory that vary systematically as a function of (I) the distance between the starting location of the hand and the position of the to-be-grasped object, and (II) the evolution of the grip aperture, of which the size and timing vary systematically as a function of the size of the to-be-grasped object (Jeannerod 1981; Jeannerod 1984; Paulignan, Frak, Toni and Jeannerod 1997; Smeets and Brenner 1999; Meulenbroek et al. 2001; Smeets and Brenner 2001; Cuijpers, Smeets and Brenner 2004).

First we will give some background on research that focusses on collision avoidance behaviour in human prehension. Next, we describe collision avoidance techniques that are used in robotic manipulators including the **ARoS**

G. Pezzulo et al. (Eds.): ABiALS 2008, LNAI 5499, pp. 188–208, 2009.

(**A**nthropomorphic **Ro**botic **S**ystem), which is an anthropomorphic robotic system that was built on the Mobile and Anthropomorphic Robotics Laboratory group at University of Minho, Portugal (ARoS, Silva, Bicho, Erlhagen 2008). We will conclude our paper with a discussion of the implications of our experimental results for robotics, we show that our anthropomorphic robot (ARoS) is capable of reproducing human movement characteristics, thus facilitating interactions with humans, and we discuss some implications.

1.1 Obstacle Avoidance in Humans

Only few prehension studies take into account the ways in which grasping movements are tuned to avoid collisions with the target object or any intermediate object (see e.g. Vaughan, Rosenbaum et al. 2001, Butz, Herbort and Hoffmann 2007). The present study was conducted to fill this gap. Additionally, the collision-avoidance component of grasping forms an essential ingredient of the posture-based motion planning theory developed by Rosenbaum, Meulenbroek, Vaughan and Jansen (2001). This theory states that the aperture overshoots that are commonly observed when the hand shapes around to-be-grasped objects, or any other biphasic component of the movement pattern, are due to the collision-avoidance constraint inherent in grasping. This claim also prompted the present study.

It is still not fully understood how the human prehension system copes with collision avoidance. Some studies focussed on reach-to-grasp movements in the presence of distractor objects that may have acted as obstacles (Meegan and Tipper 1998; Kritikos, Bennett, Dunai and Castiello 2000). In these studies it was observed that the hand trajectory veered away from intermediate distractors. This tendency was regarded as an interference effect related to the inhibition of a planned movement towards the distractor. Humans smoothly adjust movements of their effector system to circumvent obstacles by planning a movement through a 'via point'(Edelman and Flash, 1987), or 'via posture' (e.g. Rosenbaum et al., 2001). Meulenbroek et al. (2001), emphasised, that in order to avoid collisions with intermediate objects while grasping a target object, moving around the obstacle requires a biphasic component that, when superimposed on the default movement plan that will bring the hand to the target in the absence of the obstacle, ensures that the obstacle is avoided with an acceptable spatial tolerance zone (see also: Vaughan, Rosenbaum et al. 2001). It should be noted that these models ignore the fact that an end posture depends both on start point and trajectory, a recent paper by Butz et al. (2007), describes a model (SURE_REACH), which adds a neural-based, unsupervised learning architecture that grounds distance measures in experienced sensorimotor contingencies. In this model, an obstacle representation can inhibit parts of the hand space, causing the arm to generate alternative movement trajectories when the inhibition is propagated through to posture space.

In the present study, we focus on how grasping movements of which the collision-avoidance characteristics were varied, are executed. To manipulate the degree with which target objects itself acted as obstacles, we chose two starting

positions at equal distances from the target location (see Figure 1). One from which a straight hand movement would suffice for a safe and successful grasp, and one from which additional arm-configuration adjustments were needed in order to prevent a collision of the hand with the target. In line with Meulenbroek et al. (2001), we expected movements with the right hand, starting to the right from the target (S2) to elicit a smaller effect of the collision avoidance constraint on the grasp (i.e. less or no additional arm-configuration adjustments) than movements starting from the left of the body midline (S1).

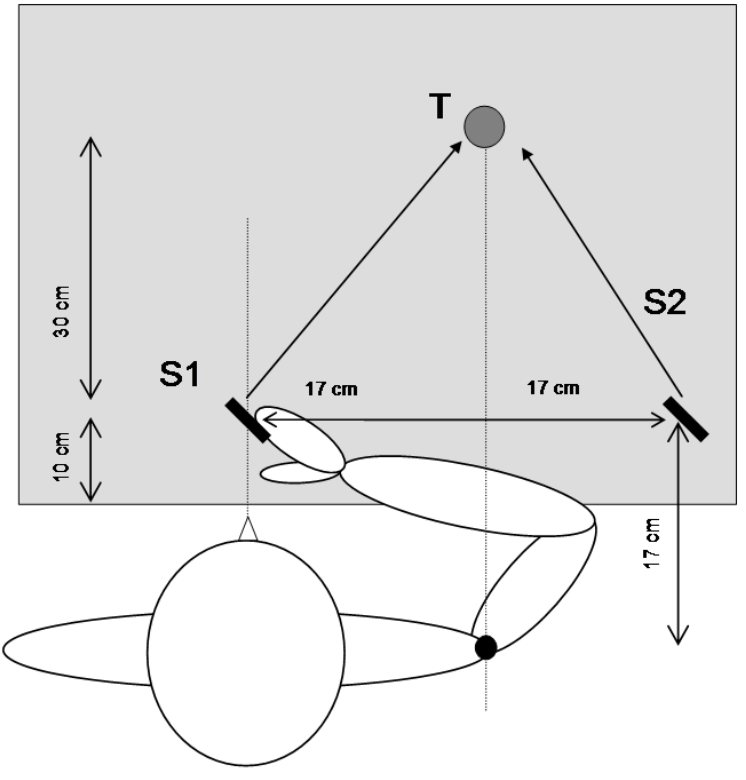


Fig. 1. Top view of experimental setup. 'S1' and 'S2' indicate the starting positions, and 'T' the target location.

Another way we manipulated the risk of collision was to vary the height of the target cylinders between 1 to 15 cm. Conceivably, collision with the shallowest cylinders is easily avoided by moving the fingers over the target cylinder before grasping it, whereas such a grasping strategy would probably be inefficient for the tallest cylinders. For the tallest cylinders, a lateral approach of the hand was expected since lifting the arm upwards against gravity to manoeuvre the hand above the cylinder top, was considered energetically suboptimal.

1.2 Collision Avoidance in Robotics

The simplest way to model a grasping movement would be to compute the required hand trajectory or joint rotations necessary to move from the starting posture to the final posture. This can result in successful grasps in some cases but will often result in a movement during which part of the effector system will collide, or even virtually move through the target object. Knowledge about how humans adjust their movements when avoiding collisions with obstacles is necessary if one attempts to develop robots that can safely interact with humans (Erlhagen et al. 2006). The latter challenge for roboticists formed the applied context that inspired the present study.

Typically, the human arm is modelled as a rigid stick-figure of seven degrees of freedom (DoF): three DoFs in the shoulder, one in the elbow, and three in the wrist. Only six DoF are needed to describe the position and orientation of the hand in Cartesian space (x, y, z, and Rx, Ry, Rz). Thus one degree of freedom remains that enables multiple joint configurations to result in the same hand position and orientation. This allows us to smoothly avoid obstacles or to choose the most efficient movement path out of numerous possibilities. It should be noted that the hand itself also has many degrees of freedom and that stretching and flexing of the fingers also plays a role in obstacle avoidance behaviour. But for this paper we focus on the upper limb with only two fingers as 'gripper'. We use an anthropomorphic robotics system (ARoS, Silva et al., in press), with a similar configuration for simulating reaching and grasping cylindrical objects in 3D space, as described below.

1.3 Antropomorphic Robotic System (ARoS)

The ARoS model is based on observations from experiments studying the human upper limb: (I) movement planning is done in joint-space (Osheron, Kosslyn and Hollerbach 1990; Rosenbaum 1990), (II) joints move in synchrony (Klein Breteler and Meulenbroek 2006); (III) planning of a reaching and grasping movement in joint space is divided into two sub-problems: (a) end posture selection and (b) trajectory selection (Meulenbroek, et al. 2001; Rosenbaum et al. 2001; Elsinger and Rosenbaum 2003) [1], (IV) end posture is computed prior to trajectory (Gréa, Desmurget and Prablanc 2000; Elsinger and Rosenbaum 2003), (V) end posture varies as a function of initial posture (Soechting, Buneo, Herrmann and Flanders 1995; Fischer, Rosenbaum and Vaughan 1997), and (VI) obstacle avoidance is incorporated by a mechanism that superposes two movements: a direct movement from the initial to the end posture and a via movement from the initial posture to the via posture and back (Rosenbaum, Meulenbroek et al. 1999; Meulenbroek, Rosenbaum et al. 2001; Vaughan, Rosenbaum and Meulenbroek 2006). First, the most adequate end posture is determined by choosing the posture that can be obtained such that the object is successfully grasped without collisions with any

[1] By posture we mean the set of joint angles of the arm and hand. Posture is represented using the well known, and widely used in robotics, Denavit-Hartenberg (proximal) convention (Craig 1998).

obstacle or the target itself at the moment of grasp, with a minimum displacement of the joints from begin to the end of the movement. Different joints may have different expense factors that contribute differently to the selection of end posture and trajectory.

Next, the trajectory of the joints is computed. We applied the minimum jerk principle to the joints of the arm and hand, such that the default movement of the joints follows a bell-shaped unimodal velocity profile, resulting in a smooth straight-line movement in joint space.

If this direct movement does not lead to collisions with obstacles, the movement is performed, otherwise a 'via movement' is added to the default movement by finding a detour through joint space that is collision-free. This via movement is a back-and-forth movement from the initial posture to a promising via posture and back again to the initial posture. The 'via movement' is superimposed on the direct movement and both are performed simultaneously.

2 Method

2.1 Participants

Twelve participants, (4 male, 8 female), ranging in age between 20 and 34 years (mean = 28 years) were included in the analyses of this study. All participants participated for course credit or remuneration after giving their informed consent.

2.2 Procedure

Participants sat comfortably at a table on a height adjustable chair and they were asked to make prehension movements from one of two starting positions and to grasp a target cylinder that could vary in height. The table was mounted with a board, on which two small strips of sandpaper were stuck to indicate the starting positions, and a circular hole was sawn out to indicate the target position (see Figure 1).

Participants started each trial with the index finger of their right hand aligned with one of the two strips of sandpaper that indicated the start locations. Upon hearing the auditory 'go'-signal, participants moved their right hand from one of two start locations to the target cylinder, grasped the target between thumb and index finger, and, as soon as a second auditory cue sounded, lifted the target briefly put it back on the table, and returned their hand to the start location (see Figure 2). During the response sequence we recorded the 3D movements of the index finger, thumb, hand, wrist, upper arm, and trunk, and we evaluated various kinematic variables normalised in time.

Movements were recorded by means of two Optotrak camera units (Optotrak 3020, Northern Digital). Recordings were made for 5 s with a sampling frequency of 100 Hz, of the trunk, upper arm, wrist, hand, and thumb and index finger. The thumb and index finger trajectories were recorded using single markers that were attached to the tips of the nails of these digits. All other movements were

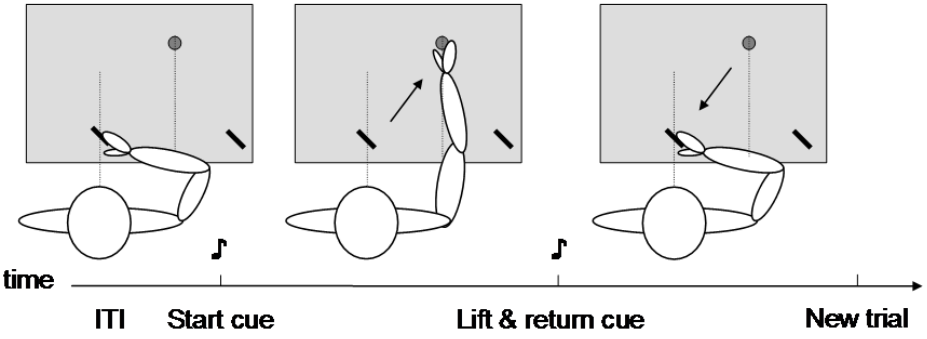


Fig. 2. Timing of trial events. A top-view example of a trial starting from S1 (left panel). As soon as the start cue sounds, the participant reaches out and grasps the target cylinder (central panel). Then the participant waits until the second auditory cue is sounded, lifts the cylinder, puts it back on the table, and returns to the starting position to prepare for the next trial (right panel). (ITI = Inter-Trial Interval).

recorded by means of rigid bodies (RB). RBs consist of minimally three IRED markers (the wrist and hand RB had four IREDs) at fixed positions relative to each other. This enables recording of not only the spatial location, but also the spatial orientation (Euclidean rotations around the x, y, and z- axes, see: Bouwhuisen, Meulenbroek et al. 2002). A specific calibration procedure allowed us to look at the relative orientations of the body segments making up the kinematic chain of the arm, hand and fingers. The orientation of the upper arm RB was recorded relative to the orientation of the trunk RB, the orientation of the wrist RB was recorded relative to the upper arm RB, and the orientaton of the hand RB relative to the wrist RB. This way, the upper arm RB rotations give an estimate of the rotations in the shoulder joint around three axes in Cartesian space (Bouwhuisen, Meulenbroek and Thomassen 2002). The displacements of the individual markers and the trunk RB were recorded relative to an external reference frame with the x-axis aligned with the horizontal, frontoparallel line, the y-axis with the horizontal, midsaggital line, and the z-axis with the vertical.

In order to induce obstacle avoidance behaviour during this task, we manipulated two factors, the Starting Position and the Target Height. We used eight different target cylinders with a diameter of 4.5 cm that were 1, 3, 5, 7, 9, 11, 13 and 15 cm tall. The central Starting Position (S1) was about 17 cm directly in front of the body midline on the tabletop and the lateral starting position was about 17 cm laterally in front of the shoulder. The target location was at about 47 cm distance from the trunk and at equal distance from the two start locations (see Figure 1). In order to induce different movement speeds, we also manipulated the response window, i.e. the interval in which participants had to perform their grasping movements. We did this because we assumed that extra time stress would force/stimulate the participants to use the most efficient movement plans (Rosenbaum et al. 2001).

2.3 Design

In every alternating block the Starting Positions changed, and eight Cylinder Heights were quasi-randomly repeated twice. Every participant performed 196 trials, run in twelve blocks of 16. The response window, i.e. the time interval between the starting cue and the lifting cue changed after half of the trials. Half the participants started with the fast condition (i.e. 1.5 s prehension interval), and the other half of the participants started with the slow condition (2 s prehension interval).

2.4 Analyses

Position and rotation data were linearly interpolated in case of missing data (which occurred infrequently and never more than 10 successive samples), and filtered by means of a Butterworth filter with a cut-off frequency of 12 Hz. Computed velocities were filtered with a cut-off frequency of 8 Hz. Trials with too many missing samples were excluded from further analyses, in total (2.7%) of all trials.

We only analysed the first part of the response sequences, i.e. the movements from start to the end of the grasp. Begin and end of this movement phase were deduced from the tangential velocity of the grip, defined as the magnitude of the first derivative of the mean position of thumb and index finger (as measured by the respective IREDs). The start of the movement was defined as the last local minimum in the tangential grip speed profile before it exceeded the threshold of 5% of the maximum tangential grip speed, and the end of the movement was defined as the first local minimum in the speed profile after it dropped below this threshold again (after the maximum velocity was reached). After the beginning and end of the prehension phase were determined, all displacements, rotations, and derived variables were normalised to time and resampled to 50 samples.

Because we were interested in different grasping strategies, we looked at the ways participants approached the cylinders, i.e. whether they approached the cylinder with their hand from the side, or whether they moved their hand over the top of the cylinder before completing their grasping. To this aim, we analysed the locations of the fingertips relative to the centre of the hand -as defined by the rigid body of IREDs attached to it- in the horizontal plane. The cylinder is defined as a circle with a radius of 2.25 cm centered at the origin. The finger trajectories were translated such that the final locations of thumb and index finger were positioned on the cylinder. We also determined the amount with which the line connecting the location of the index finger and the centre of the hand swept across the circle that defined the cylinder's top. Trials in which this occurred, were labelled as Overlap (OL) (see Figure 3A for an example) and all other trials, in which the cylinder was approached and grasped from the side, were labeled as No Overlap (NoOL).

The most important variable we manipulated to induce obstacle avoidance behaviour was the cylinder height. Because we expected the grip height to depend on the target height, we first looked at the grip height in time. Grip height is the mean z-coordinate of thumb and index markers, and therefore a good indicator of the behaviour of the most distal part of the effector system.

Because we aimed at inducing obstacle avoidance behaviour, which we also expected to be reflected in biphasic velocity profiles, we also studied the tangential velocity profiles of the grip (i.e. the average position of the thumb and index markers).

Since participants were instructed to start their responses with their hand flat on the table top, it is also interesting to study the change in hand orientation in time. To this aim, we computed the hand plane angle (HPA). HPA was defined as the angle between the horizontal plane and the plane that is defined by the marker of the hand RB closest to the MCP-II joint, the index marker, and the thumb marker. A horizontal position (palm down) is defined as 0 deg, and a vertical orientation with the thumb down is defined to be 90 deg. We expected the HPA to start near horizontal, to become more vertical during the prehension phase, and rotate back to a more horizontal posture towards the end of the grasp.

Because we expected differences in obstacle avoidance to be reflected in the relation between proximal and distal parts of the effector system, we contrasted the HPA with the arm plane angle, which is defined as the angle between the horizontal plane and the plane that is spanned by the vectors denoting the upper arm RB and the wrist RB.

To compare the proximal and distal involvement (i.e. the shoulder and wrist) in the grasping movements at joint level, we computed the net shoulder (Rs) and wrist (Rw) rotations as the square root of the sum of the squared rotations around the x, y, and z-axes of the upper arm RB relative to the trunk RB (Rs), and of the hand RB, relative to the wrist RB (Rw), as described in Eq. 1.

$$Rw, s = \sqrt{Rx^2 + Ry^2 + Rz^2} \tag{1}$$

where for the shoulder rotation (Rs), Rx, Ry, and Rz are the rotation angles of the upper arm RB relative to the trunk RB, and for the wrist rotation (Rw), Rx, Ry, and Rz are the rotation angles of the hand RB relative to the wrist RB. These rotation measures are independent of rotation direction, and give an estimate of the degree of rotation in the specific joint.

After deriving all these variables, every time series of these variables was normalised in time to 50 samples. This way we were able to compare trials with different durations.

3 Results

First we established that our experimental manipulations were effective in causing different types of obstacle avoidance behaviour, as reflected by different ways

Table 1. Incidence (number of trials) of the two distinguished Grip Types (NoOL = No Overlap grip; OL = Overlap grip) as a function of Starting Position (Start=1 and Start=2) and Cylinder Height (in cm)

	Start = 1	Start = 1	Start = 2	Start = 2
Cylinder Height (cm)	No OL	OL	NoOL	OL
1	49	94	117	26
3	87	56	131	13
5	116	28	133	10
7	123	20	140	4
9	129	12	142	2
11	134	8	141	3
13	137	5	141	2
15	138	4	139	5
Grand Total	913	227	1084	65

to approach and grasp the target cylinder. Figure 3 shows examples of grasping responses to the shallowest and highest cylinders from both starting locations. Participants moved their hand over the top of the target cylinder in some trials, and approached the cylinders sideways in other trials. As mentioned before, we labeled the trials in which participants moved their index finger over the cylinder as "Overlap trials" (OL) and all other trials as "No Overlap trials" (NoOL). Figure 4 and Table 1 show the overall number of OL trials per starting position plotted against cylinder height. Note that the shallowest cylinders are most often grasped with an overlap grip, and that this occurs most often in responses starting from S1, as we expected.

3.1 Grip Height

After having established that varying start location and cylinder height yielded different grasp types, we focused on how our main variables of interest varied as function of cylinder height and start location. As expected, the grip height increases and decreases in time, and differentiates between different cylinder heights (see Figure 5). The moment at which the grip height starts to differentiate between different cylinder heights was captured by analysing the time-normalised standard deviations (SD) across cylinder heights (see Figure 5B). 'Kick-in' was defined as the moment at which the SD reached the threshold of 1% of the range of grip heights. This analysis clearly shows that the effect kicked in early on in the movements, in particular already at 10% of the movement time.

3.2 Arm-Plane Angle and Hand-Plane Angle

Now we know that the kinematic variable grip height, that characterises the most distal part of the upper limb is affected by the target height, it is interesting to look at two other variables that -together- incorporate the whole movement of the upper limb. The Arm Plane Angle (APA) and Hand Plane Angle (HPA) are

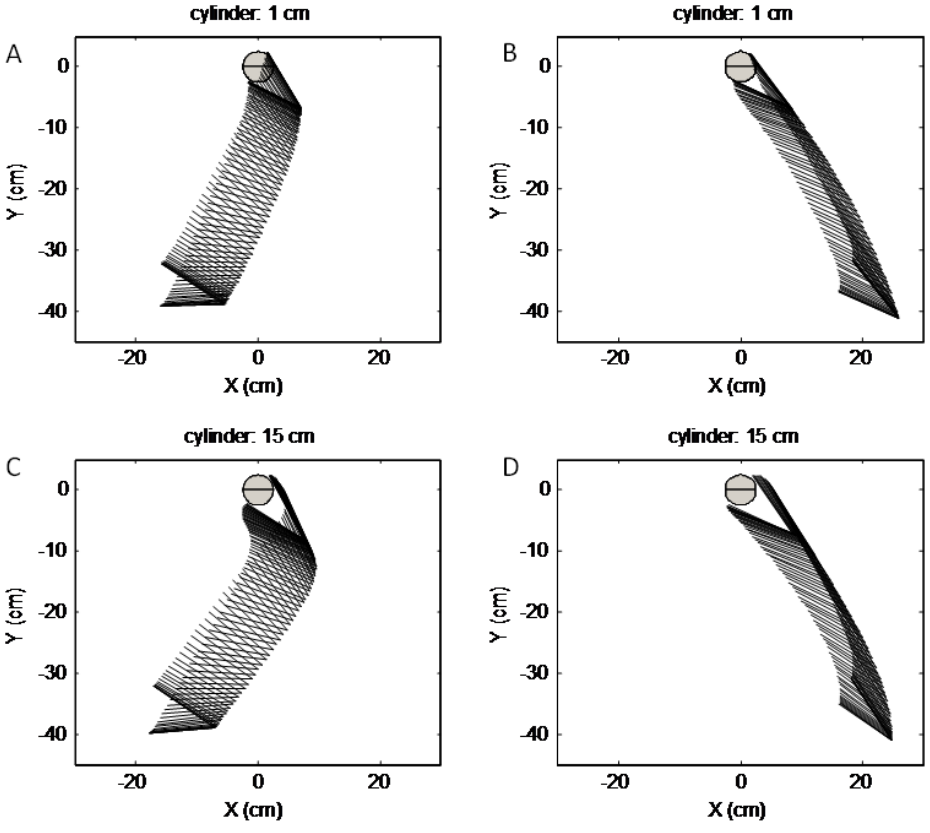


Fig. 3. Top view of the position changes of the hand in the horizontal plane during individual grasps. The bottom black line of the V-shapes connects the centre of the hand RB with the IRED on the tip of the thumb; the top black line of the V-shapes connects the centre of the hand RB with the IRED on the tip of the index finger. (A) from S1 to the shallowest cylinder with an Overlap grip, (B) from S2 to the shallowest cylinder with a No Overlap grip, (C) from S1 to the tallest cylinder with a No Overlap grip, and (D) from S2 to the tallest cylinder with a No Overlap grip. See also Figure 1.

shown as a function of normalised time in Figure 6, revealing that HPA varies with cylinder height, whereas APA shows a very stable pattern across cylinders (see Figure 6A, C). The final APA and HPA are shown in Figure 6C and D.

To find the moment at which the effect of cylinder height on the HPA kicked in, we used the same method as earlier described for the grip height:

The moment at which the standard deviations of APA and HPA started to differentiate was defined as the moment the difference between the SDs of APA and HPA reached the threshold of 1% of the mean range of SD(HPA) and SD(APA). This occurred at 12% of movement time for S1 and at 20% of movement time for S2 (see Figure 6B).

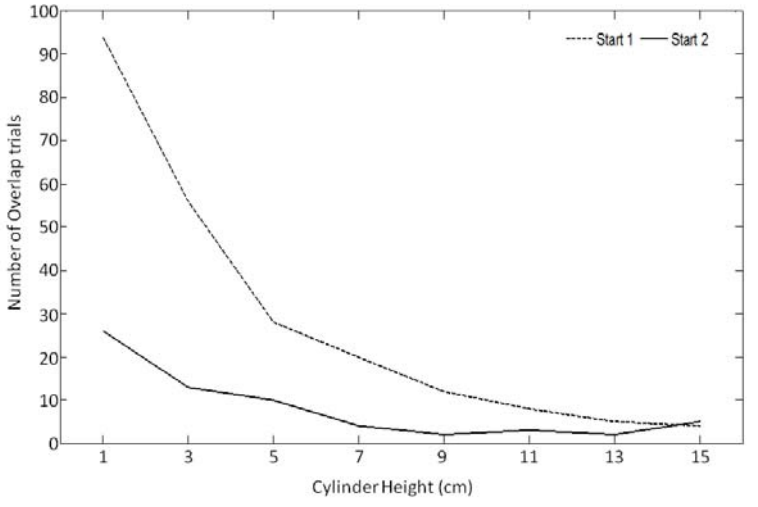


Fig. 4. Number of trials with an overlap grip counted across all participants, for every cylinder height. The dashed line represents Start 1 and the solid line represents Start 2.

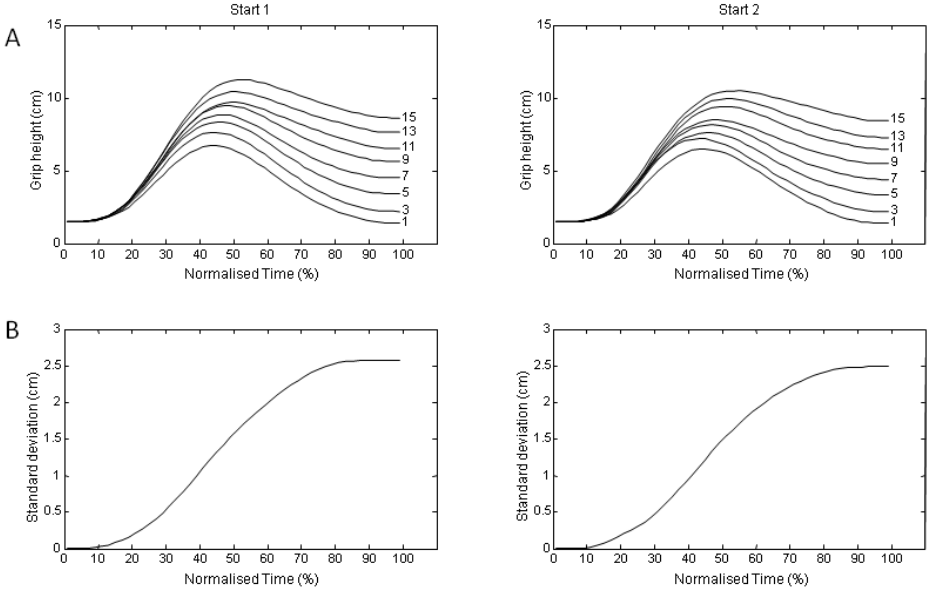


Fig. 5. (A) Time-normalised Grip Height changes for Start 1 and Start 2, averaged across participants (N=12). Different lines represent different cylinder heights, as indicated by the numbers at the righthand-side of the curves. (B) Standard deviations across cylinder heights as a function of time for Start 1 (bottom left) and Start 2 (bottom right).

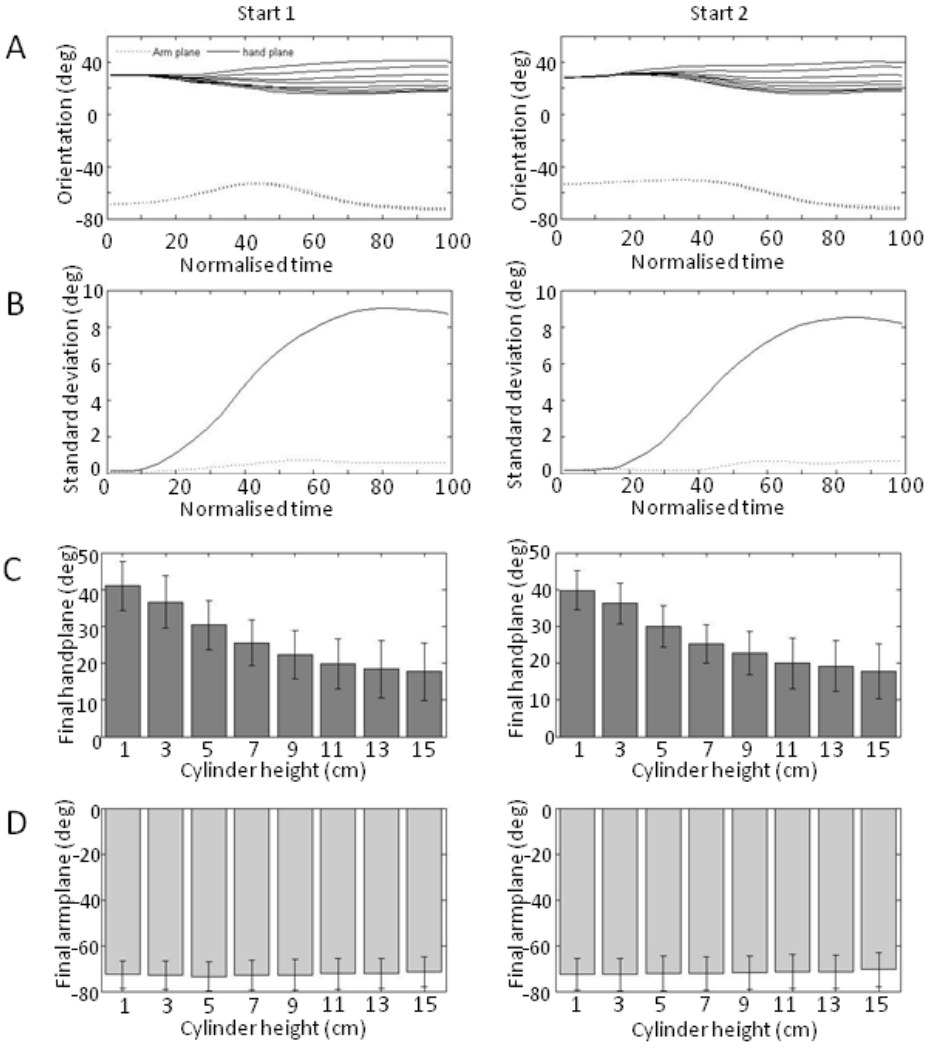


Fig. 6. (A) Arm-plane (solid lines) and hand-plane (dashed lines) angles (deg) in time for Start 1 and Start 2. Different lines represent different cylinder heights. (B) Standard deviation across cylinder heights in time for Start 1 and 2. (C) Final hand plane per cylinder height, averaged across cylinders for Start 1 and Start 2, (D) Final arm plane per cylinder height, averaged across cylinders for Start 1 and Start 2.

3.3 Wrist Rotation and Shoulder Rotation

A similar approach can also be applied to a comparison of the net shoulder and wrist rotations (Rs and Rw). Figure 7A shows that wrist-rotation patterns overlap in some cases, but still differentiate more between cylinder heights than the shoulder rotation patterns do (see also the final Rw and Rs, as plotted in Figure 7C,D).

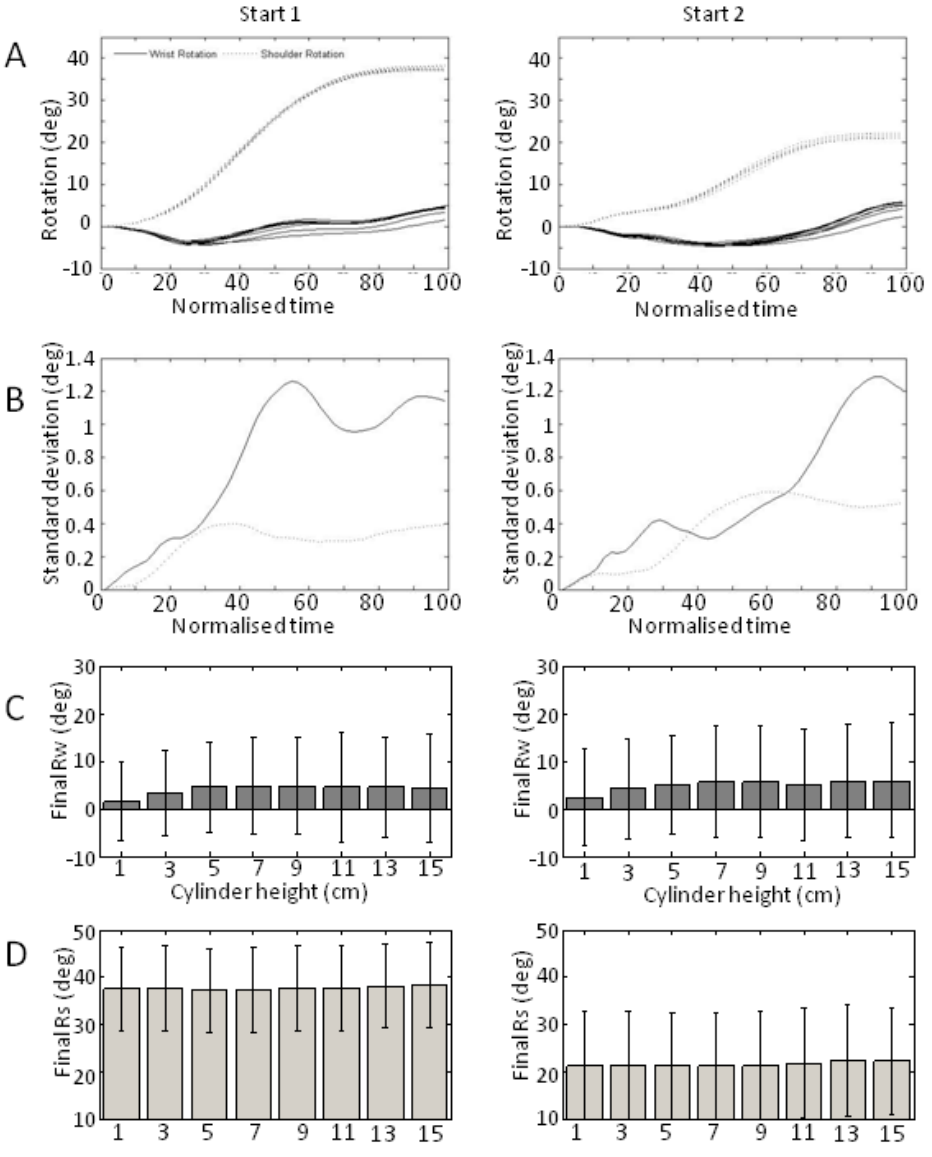


Fig. 7. Wrist rotation (solid lines) and shoulder rotation (dashed lines) in time for Start 1 and Start 2. Different lines represent different cylinder heights. (B) Standard deviation for wrist- (solid line) and shoulder rotation (dashed line) across cylinder heights in time for Start 1 and 2. (C) Final wrist rotation per cylinder height, averaged across cylinders for Start 1 and Start 2, (D) Final shoulder rotation per cylinder height, averaged across cylinders for Start 1 and Start 2.

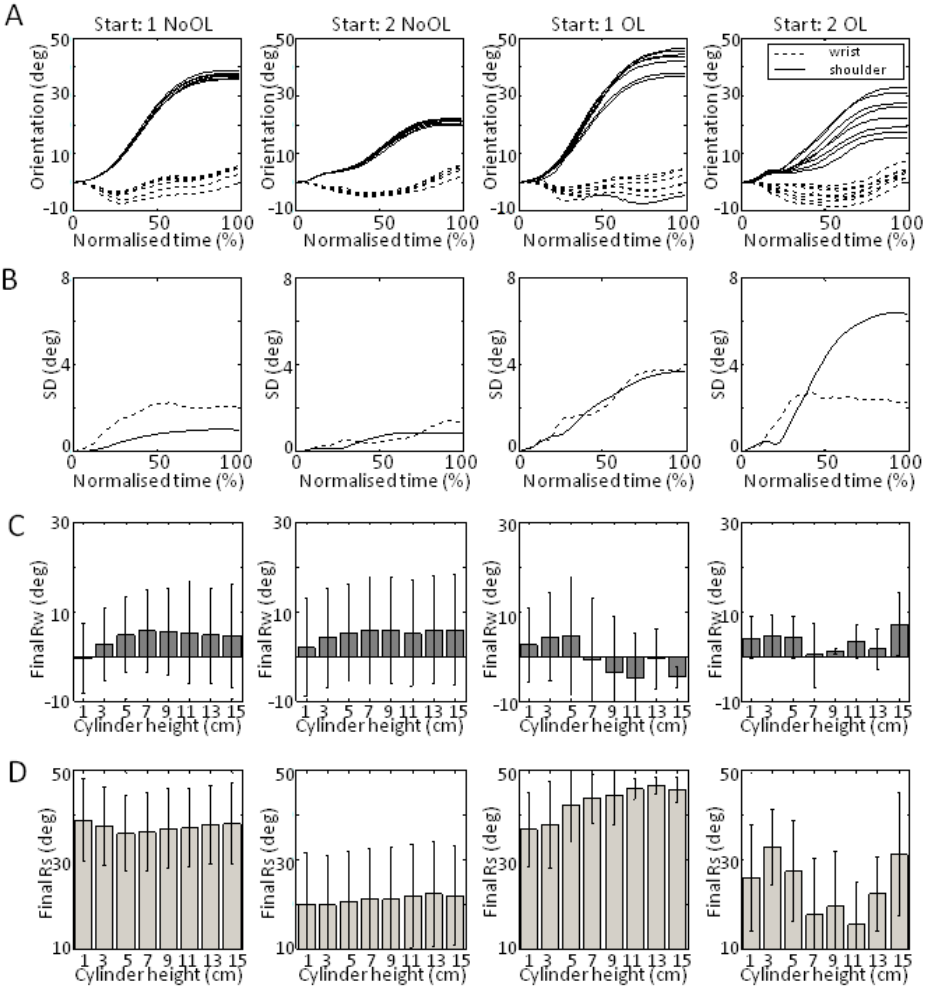


Fig. 8. Wrist rotation and shoulder rotation. Columns represent the two starting positions and grasp types (Overlap and No Overlap) (A) Wrist rotation (solid lines) and Shoulder rotation (dashed lines) in time. Different lines represent different cylinder heights. (B) Standard deviation for wrist- (solid line) and shoulder rotation (dashed line) across cylinder heights in time. (C) Mean final wrist rotation per cylinder height (D) Mean final shoulder rotation per cylinder height.

SDs computed across cylinder heights, are larger for the wrist rotation than for the shoulder rotation (see Figure 7B). Furthermore, the SD patterns seem to differ for the two start locations, suggesting that the effects of the Cylinder Height kick in later during the response in responses starting from S2 than from S1. The difference between Rw and Rs is also evident in Figures 7C,D: the final wrist rotation angle (Rw) differs slightly between the shortest cylinders, whereas final shoulder rotation angle (Rs) is stable across cylinder height.

The difference we observed between the rotation patterns for the two start locations might be related to the effects of Overlap and No Overlap grasps. To evaluate this aspect, we also compared the development of these variables between the two grip types (See Figure 8). Both the wrist and shoulder rotation were most strongly affected by the cylinder height in the OL trials, presumably because participants grasped the cylinders near the top. The development of SDs across cylinder heights, as shown in Figure 8B showed steeper SD curves for the OL trials, and this was most pronounced in the wrist rotation. It should be noted that OL and NoOL trials were not equally distributed across participants and conditions (see Table 1), therefore it is hard to statistically test these data patterns.

3.4 Speed Instructions

Movement time was compared between the two Speed Instruction conditions and the two Starting Positions by means of a 2 x 2 repeated measures ANOVA. The participants followed the speed instructions; movements in the high-speed instruction condition took less time than in the low-speed condition (914 ms and 967 ms, respectively; $F(1, 13) = 6.228; p < .05$). Movements starting from S2 lasted longer than from S1 (910 and 972 ms, respectively; $F(1, 13) = 45.542; p < .001$). There was no interaction between Speed Instruction and Starting Position.

4 Discussion

One of the main purposes of our experimental study was to gain more insights in the ways upper-limb movements are altered in order to prevent collisions with a target to-be-grasped. The paradigm we used showed that participants roughly used two collision-avoidance strategies: circumnavigating the cylinder and approaching from the side (NoOL grip), or approaching it from above (OL grip). As expected, the OL grip types occurred more frequently in the trials starting from S1 than S2, because the shortest trajectory from start to target would collide with the cylinder starting from S1 but not from S2. OL grip types occurred also more frequently when shallow cylinders had to be grasped, because lifting the hand over the cylinder requires less effort for shallow cylinders than taller ones.

We analysed the time-normalised grip height, hand-plane angle (HPA), arm-plane angle (APA), wrist- and shoulder rotations (Rw and Rs) for the two starting positions and all cylinder heights. APA and Rs did not show any target height-dependent patterns. There seems to be some differentiation in Rw for the shortest cylinders, but the strongest effects of target height were reflected in the time-normalised Grip height and HPA -patterns. These height-effects were present immediately at the start of the response for the HPA, and at 10% of movement time for the grip height. Showing that the more distal parts of the effector system are more sensitive to slight alterations in task requirements than proximal parts.

Because Rw and Rs showed different SD patterns for the two start locations (see Figure 7B), and knowing that start location has a strong effect on the grasp type, we zoomed in on the difference between the development of wrist and shoulder rotations, in relation to the grasp type (see Figure 8). The effect of cylinder height is stronger in the overlap-trials, as reflected in the steeper increasing SD patterns. This difference between grip strategy is strongest reflected in the shoulder rotation data.

4.1 Implications for Robotics

As stated in the Introduction, Obstacle avoidance is generally reflected in biphasic velocity profiles (Rosenbaum et al. 2001), The top panel of Figure 9 shows such a biphasic tangential grip velocity profile of a trial in which a 15-cm tall cylinder had to be grasped from S1. The lower panels show the whole trajectory of the lines connecting the centre of the handRB with the thumb and the index finger (like in Figure 3), and snapshots of the movement at 1, 20, 40, 50, 60, 70, 80, 90 and 100% of movement time. Although such biphasic tangential velocity profile is not recognisable in every trial, the idea of planning and executing movements with a bouncing posture, or through a via-point can be a valuable addition to the present anthropomorphic robot models [2]. A simulator can perform a similar task, and the tangential grip velocity is a variable that is easy to compare qualitatively. Figure 10 shows an example of a simulation of a trial (Start location 1, cylinder height = 15 cm) with obstacle avoidance characteristics. The panels show snapshots of a top view of the simulator at successive moments in time. The progress in time is indicated with a star shape on the biphasic tangential velocity profiles plotted in the left top of every panel.

In order to build anthropomorphic robots that have to interact with humans, it is convenient if these robots move like humans. Since humans are well trained in interpreting gestures and other movements of other humans, the intentions of a robot that behaves more humanlike, are recognised more easily and its human collaborator can smoothly adapt to this, which is safer, and facilitates the collaboration. But for safety reasons it is also a prerequisite that robots are able to avoid colliding into their human collaborators.

Our behavioural data show that the distal parts of the human prehension system are more flexible in adjusting to different target heights and starting positions or directions than the proximal parts. The snapshots of the ARoS robot simulator (Silva et al. in press) in Figure 10 also show the largest rotations in the distal joints while the simulator successfully approaches the cylinder while avoiding collisions with itself, the tabletop and the target. The present findings further illustrate that grasping an object from the side is not always the preferred

[2] It should be noted that not all trials showed such a biphasic pattern, and that the mean pattern of the tangential velocity shows a positively skewed bell shape. This can be explained by the relative difference in peak height between the first and second velocity peak and the fact that the second peak occurs at different moments across trials.

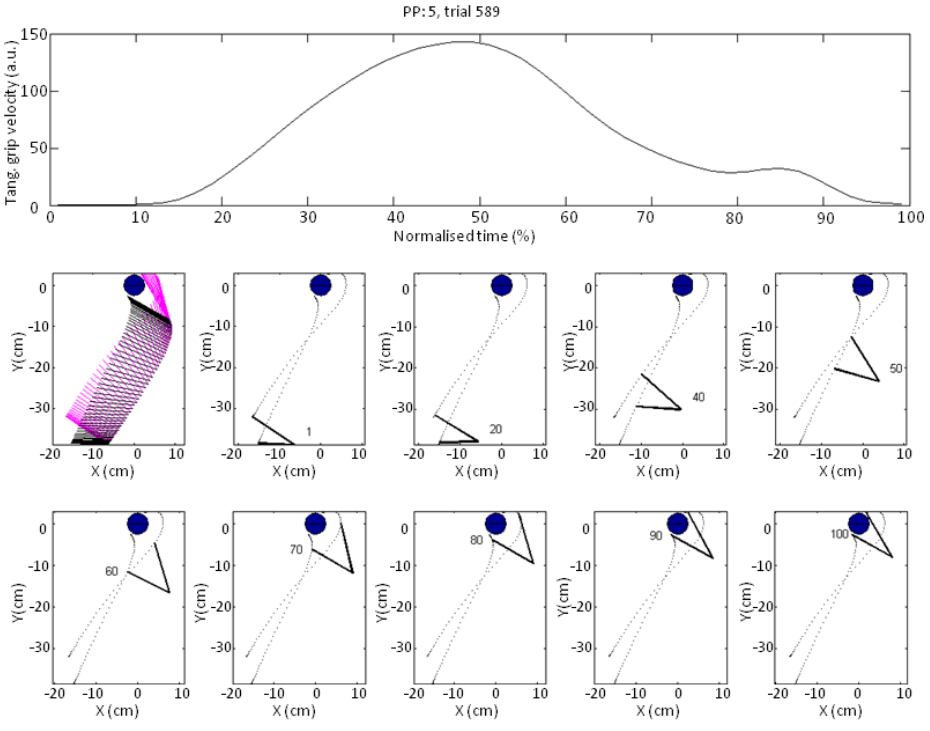


Fig. 9. Example of a trial (Start location 1, cylinder height = 15 cm) with obstacle avoidance characteristics. The top panel shows a biphasic tangential velocity profile and the other panels show estimates of the thumb and index finger movements in the horizontal plane at successive moments in time, the dashed lines show the trajectories of the thumb and index finger and the numbers indicate the relative moment during the response (in %).

option, therefore anthropomorphic robot models should have the flexibility to be able to approach and grasp target objects from above.

4.2 Alternative Approaches to Obstacle Avoidance in Robotics

The research on collision avoidance for movement of robotic manipulators can be divided into global and local methods. In global methods the collision avoidance is carried out by on-line algorithms before movement starts. On the other hand, in local methods, on-line algorithms are used in which possible collisions are tested during the motion, and the robot reacts by activating strategies to avoid obstacles when necessary.

Global methods include approaches where the motion planning is performed by searching for collision-free paths from start to goal configuration, in the robot's configuration space. Obstacles are mapped into this space as forbidden regions (for a review see Latombe 1999). Other methods treat the motion planning problem as an optimisation problem, where obstacles and joint limits are

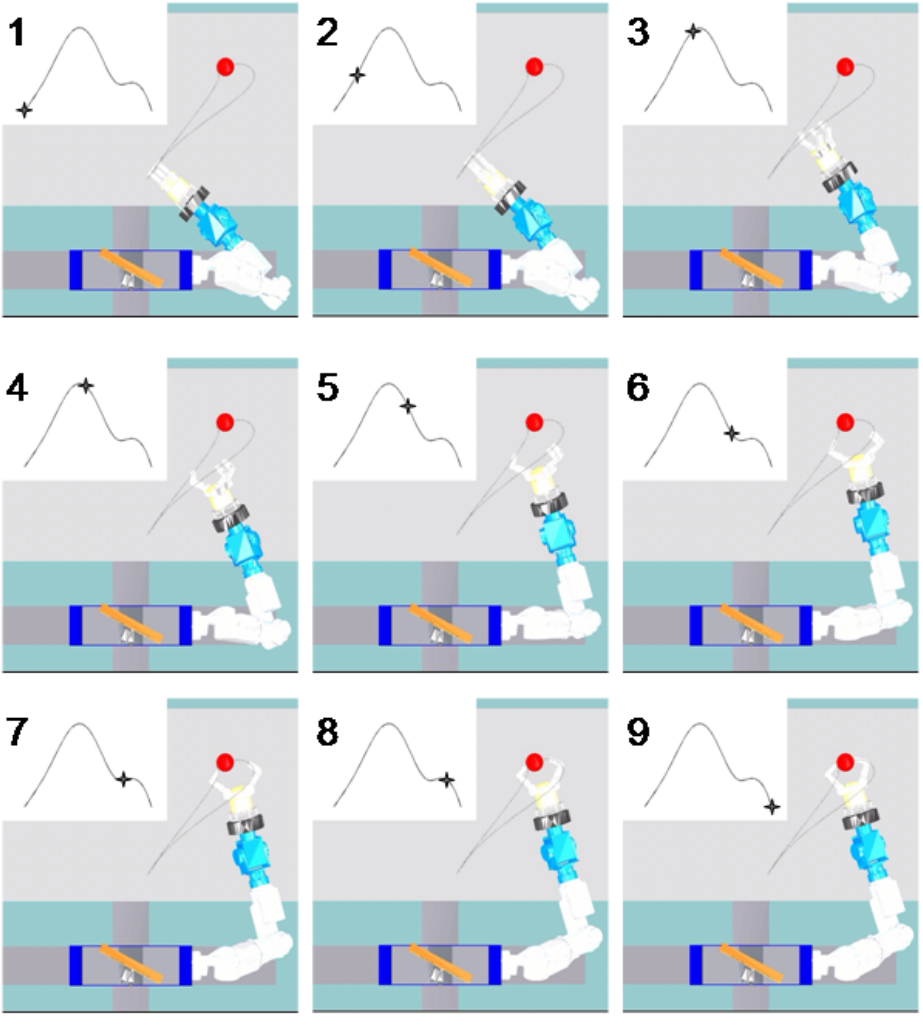


Fig. 10. Top view snapshots of the ARoS robot grasping a cylinder while avoiding collisions with itself, the tabletop and the target. Every panel shows the tangential velocity of the gripper in time, the star indicates the tangential velocity at the moment the snapshot was taken, the dashed lines show the trajectories of both sides of the gripper and the configuration of the robot's upper limb is shown at t=1, 20, 40, 50, 60, 70, 80, 90 and 100% of movement time in the successive panels from top left to bottom right.

the problem's constraints, and techniques like optimal control theory (Galicki 1998), nonlinear programming (Park 2006) and dynamic programming (Fiorini and Shiller 1996) are used.

Potential field methods are quite popular on-line collision avoidance methods for robot manipulators. These methods were first introduced by Khatib (1986),

and there is a large variety of these methods. At each step, the robot moves by following the gradient of a potential field consisting of attractive potentials, due to goal positions, repulsive potential due to obstacles and also repulsive potentials due to joint limits. Another local method is the attractor dynamics approach initially introduced for mobile robots (see e.g. Bicho 2000) and more recently to anthropomorphic robotic arms (Iossifidis and Schoner 2006). Here the time courses of the heading direction of the end effector, elevation and azimuth angles, and elbow motion were obtained from an attractor dynamics, into which obstacles contributed repulsive force-lets and joint limit constraints were coupled as repulsive force-lets as well.

In general alternative methods cannot produce human-like movements. However, experimental studies on human behaviour show that human-like movement facilitate interactions between robot and human. The movements of the anthropomorphic robot we have described before, are qualitatively similar to human movements, as is evident from our experimental and simulation results.

4.3 General Conclusion

We have found that humans avoid collisions while reaching to, and grasping cylinders by adjusting the movements of their distal joints. We successfully mimicked the resulting biphasic velocity profile in the ARoS simulation of a prehension and grasping movement like one of the conditions of our behavioural experiment. This validates the human-like movement characteristics of ARoS and, consequently, facilitates human-robot interaction.

Acknowledgements. This study was supported by the EU-funded project Joint-Action Science and Technology (JAST) (ref. IST-FP6-003747) and by FCT and UM through project "Anthropomorphic robotic systems: control based on the processing principles of the human and other primates motor system and potential applications in service robotics and biomedical engineering" (ref. CONC-REEQ/17/2001). Eliana Costa e Silva was supported by the Portuguese Foundation for Science and Technology (grant: FRH/BD/23821/2005). We would further like to acknowledge Wolfram Erlhagen and Estela Bicho (both at DEI, University of Minho, Portugal), Martin Butz and two anonymous reviewers for their comments on this paper.

References

1. Butz, M.V., Herbort, O., Hoffmann, J.: Exploiting redundancy for flexible behavior: unsupervised learning in a modular sensorimotor control architecture. Psychological Review 114(4), 1015–1046 (2007)
2. Bicho, E.: Dynamic Approach to Behavior-Based Robotics. Shaker-Verlag (2000)
3. Bouwhuisen, C.F., Meulenbroek, R.G.J., Thomassen, A.J.W.M.: A 3D motion-tracking method in graphonomic research: possible applications in future handwriting recognition studies. Pattern Recognition 35, 1039–1047 (2002)
4. Craig, J.J.: Introduction to robotics: mechanics and control. Addison-Wesley, Reading (1998)

5. Cuijpers, R.H., Smeets, J.B.J., Brenner, E.: On the relation between object shape and grasping kinematics. Journal of Neurophysiology 91(6), 2598–2606 (2004)
6. Edelman, S., Flash, T.: A model of handwriting. Biological Cybernetics 57(1-2), 25–36 (1987)
7. Elsinger, C.L., Rosenbaum, D.A.: End Posture selection in manual positioning: Evidence for feedforward modelling based on a movement choice methos. Experimental Brain Research 152, 499–509 (2003)
8. Erlhagen, W., Mukovskiy, A., Bicho, E., Panin, G., Kiss, C., Knoll, A., Van Schie, H.T., Bekkering, H.: Goal-directed imitation for robots: A bio-inspired appraoch to action understanding and skill learning. Robotics and autonomous systems 54(4), 353–360 (2006)
9. Fiorini, P., Shiller, Z.: Time Optimal Trajectory Planning in Dynamic Environments. In: Proc. of the IEEE Int. Conf. on Robotics and Automation, pp. 1553–1558 (1996)
10. Fischer, M.H., Rosenbaum, D.A., Vaughan, J.: Speed and sequential effects in reaching. Journal of Experimental Psychology-Human Perception and Performance 23(2), 404–428 (1997)
11. Galicki, M.: Robotics and Automation. In: Proc. of the IEEE Int. Conf. on Robotics and Automation, vol. 1, pp. 101–106 (1998)
12. Gréa, H., Desmurget, M., Prablanc, C.: Postural invariance in three-dimensional reaching and grasping movements. Experimental Brain Research 134, 155–162 (2000)
13. Iossifidis, I., Schöner, G.J.: Dynamical Systems Approach for the Autonomous Avoidance of Obstacles and Joint-limits for an Redundant Robot Arm. In: Proc. of the IEEE Int. Conf. on Intelligent Robots and Systems, pp. 508–585 (2006)
14. Jeannerod, M. (ed.): Intersegmental coordination during reaching at natural visual objects. Attention and Performace. Erlbaum, Hillsdale (1981)
15. Jeannerod, M.: The Timing of Natural Prehension Movements. Journal of Motor Behavior 16, 235–254 (1984)
16. Khatib, O.: Real-Time Obstacle Avoidance for Manipulators and Mobile Robots. The International Journal of Robotics Research 5(1), 90–98 (1986)
17. Klein Breteler, M.D., Meulenbroek, R.G.J.: Modeling 3D object manipulation: synchronous single-axis joint rotations? Experimental Brain Research 168(3), 395–409 (2006)
18. Kritikos, A., Bennett, K.M.B., Dunai, J., Castiello, U.: Interference from distractors in reach-to-grasp movements. Quarterly Journal of Experimental Psychology Section a-Human Experimental Psychology 53(1), 131–151 (2000)
19. Latombe, J.: Motion Planning: A journey of robots. molecules, digital actors, and other artifacts. International Journal of Robotics Research, 1119–1128 (1999)
20. Meegan, D.V., Tipper, S.P.: Reaching into cluttered visual environments: spatial and temporal influences of distracting objects. Quarterly Journal of Experimental Psychology 51(2), 225–249 (1998)
21. Meulenbroek, R.G.J., Rosenbaum, D.A., Jansen, C., Vaughan, J., Vogt, S.: Multijoint grasping movements - Simulated and observed effects of object location, object size, and initial aperture. Experimental Brain Research 138(2), 219–234 (2001)
22. Osheron, D.N., Kosslyn, S.M., Hollerbach, J.M. (eds.): Visual Cognition in Action: an Invitation to Cognitive Science. MIT Press, Cambridge (1990)
23. Park, J.: Optimal Motion Planning for Manipulator Arms Using Nonlinear Programming. In: Huat, L.K. (ed.) Industrial Robotics, Programming, Simulation and Applications, pp. 256–272. pIV pro literatur Verlag Robert Mayer-Scholz (2006)

24. Paulignan, Y., Frak, V.G., Toni, I., Jeannerod, M.: Influence of object position and size on human prehension movements. Experimental Brain Research 114, 226–234 (1997)
25. Rosenbaum, D.A.: Human Motor Control. Academic Press, San Diego (1990)
26. Rosenbaum, D.A., Meulenbroek, R.G.J., Vaughan, J.: Planning Reaching and Grasping Movements: Theroretical Premises and Practical Implications. Motor Control. 2, 99–115 (2001)
27. Rosenbaum, D.A., Meulenbroek, R.G.J., Vaughan, J.: Coordination of reaching and grasping by capitalizing on obstacle avoidance and other constraints. Experimental Brain Research 128(1-2), 92–100 (1999)
28. Rosenbaum, D.A., Meulenbroek, R.G.J., Vaughan, J., Jansen, C.: Posture-based motion planning: Applications to grasping. Psychological Review 108(4), 709–734 (2001)
29. Silva, R., Bicho, E., Erlhagen, W.: AROS: An anthropomorphic robot for human-robot interaction and coordination studies. In: Proc. of the Portuguese 8th Conference on Automatic Control - Controlo 2008, pp. 819–826 (2008)
30. Smeets, J.B.J., Brenner, E.: A New View on Grasping. Motor Control 3, 237–271 (1999)
31. Smeets, J.B.J., Brenner, E.: Independent movements of the digits in grasping. Experimental Brain Research 139, 92–100 (2001)
32. Soechting, J.F., Buneo, C.A., Herrmann, U., Flanders, M.: Moving Effortlessly in 3-Dimensions: Does Donders-Law Apply to Arm Movement. Journal of Neuroscience 15(9), 6271–6280 (1995)
33. Vaughan, J., Rosenbaum, D.A., Meulenbroek, R.G.J.: Planning Reaching and Grasping Movements: The Problem of Obstacle Avoidance. Motor Control 2, 116–135 (2001)
34. Vaughan, J., Rosenbaum, D.A., Meulenbroek, R.G.J.: Modeling Reching and Manipulating in 2- and 3-D Workspaces: The Posture-Based Model. ICDL, Bloomington (2006)

The Role of Anticipation on Cooperation and Coordination in Simulated Prisoner's Dilemma Game Playing

Maurice Grinberg and Emilian Lalev

Central- and Eastern European Center for Cognitive Science
21 Montevideo street, 1618 Sofia, Bulgaria
mgrinberg@nbu.bg, elalev@cogs.nbu.bg

Abstract. We present a connectionist model for the Iterated Prisoner's Dilemma game which we explored in different game-playing environments. The role of anticipation on cooperation and coordination was our main interest. The model was validated by comparisons with human subjects' experiments in which subjects played individually against a computer opponent. After reproducing several interesting characteristics of individual play, we used the model in multi-agent simulations of small societies in which agents interacted among each-other by playing the Iterated Prisoner's Dilemma game. In genetic simulations, we demonstrated how anticipation will evolve in the societies to achieve either higher cooperation rates or payoffs. Our results favor the assumption that anticipation is decisive for high level of cooperation and higher cooperative coordination in the simulated societies.

1 Introduction

The games defined in formal game theory (like e.g. the Prisoner's Dilemma game) are widely used to model social interactions (Colman, 2003). Recently, several influential research efforts (e.g. Axelrod, 1984 and Epstein and Axtell, 1996), based on Multi-Agent Simulations (MAS), have been carried out successfully in order to explain (and even try to influence) such important aspects of societies like cooperation and competition. The typical framework of such approaches consists of the use of simple agents interacting with an environment with simple rules or game playing. Although the phenomena arising in such environments are important enough to deserve detailed investigation, we have adopted a different approach here. We are interested in cognitively plausible agents whose performance can be compared against experimental data from human participants.

The above problem can be regarded as a development of the opposition of standard game theory and the bounded rationality framework (Colman, 2003). In standard game theory, players are described as perfectly rational and possessing perfect information about the game including knowledge about the possible moves and payoffs, and opponents. On the other hand, the bounded rationality

G. Pezzulo et al. (Eds.): ABiALS 2008, LNAI 5499, pp. 209–228, 2009.

view on cognition states that people are almost never perfectly rational (Colman, 2003) due to limitations in perception, time, thinking, and memory. Moreover, people tend to minimize the cognitive effort while making decisions. Finally, the results of experiments involving games demonstrate that people rarely play as prescribed by the normative game theory. One such famous example is the Prisoner's Dilemma (PD) game which will be dealt with in this chapter.

For instance the influence of cognitive constraints and mechanisms on decision making in the Iterated Prisoner's Dilemma Game (IPDG) and thus on the simulations describing social interactions has been studied in a series of investigations (see Hristova and Grinberg, 2004 and Lalev and Grinberg, 2007). The use of cognitively plausible agents can insure that the information gained by using them in simulations of complex social interactions will take into account specific cognitive mechanisms which are essential for the explanation of certain observed phenomena.

One such cognitive mechanism is anticipation and its role in explaining cooperation and coordination. Special attention will be devoted to the use of the anticipation model proposed by Lalev and Grinberg (2007) in MAS, where the role of anticipation on cooperation in IPDG has been investigated. The analysis of the model features and the comparison with previous experiments with human participants demonstrated the importance of prediction for adequate description of the behavioral data on cooperation. These results were obtained in the experiments and in the theoretical frameworks by using individual playing against a tit-for-tat opponent focusing on individual decision making. Here, we want to present results which demonstrate the role of anticipation in small societies of agents. The key characteristics of interest will be cooperation and coordination as related to the essence of social interaction.

1.1 The Prisoner's Dilemma Game

The Prisoner's dilemma is a two-person game and a famous example of a social dilemma game. The payoff table for this game is presented in Table 1. The players simultaneously choose their move - C (cooperate) or D (defect), without knowing their opponent's choice.

In Table 1, R is the payoff if both players cooperate (play C), P is the payoff if both players 'defect' (play D), T is the payoff if one defects and the other cooperates, S is the payoff if one cooperates and the other defects. The payoffs satisfy the inequalities $T > R > P > S$ and $2R > T + S$. This structure of the payoff matrix of that game offers a dilemma to the players: there is no obvious best move. The dominant D move ($T > R$ and $P > S$) would lead to lower payoffs if adopted by all the players (payoff P) although this is the choice prescribed by standard game theory. Cooperation seems to be the best strategy in the long run ($R > P$) but at the risk of one of the opponents to start to defect and the other to receive the lowest payoff S. This quite complicated situation is at the heart of the dilemma in this game and is the reason for the on-going interest in this game over the past 50 years and continuing today. The importance of the possibility to predict the opponents' moves is obvious

Table 1. Payoff table for the PD game. In each cell the comma separated payoffs are the Player I's and Player II's payoffs, respectively.

		Player II	
		C	*D*
Player I	*C*	R, R	S, T
	D	T, S	P, P

especially in the iterated version of the game. Reliable prediction would lead in some cases to trust in the opponent and higher cooperation while in other cases to 'punishment' of expected defection. In any case, anticipatory agents playing IPDG will be involved in specific interactions, which have to be investigated.

Rapoport and Chammah (1965) proposed the quantity CI = (R-P)/(T-S), called cooperation index, as a predictor of the probability of C choices, monotonously increasing with CI. In Table 2, two examples of PD games with different CI 0.1 and 0.9, respectively are presented.

Table 2. Examples of PD game matrices with different CI - 0.1 and 0.9, respectively. The first payoff in each cell is the payoff of the 'row' player and the second of the 'column' player.

CI=0.1		Player II	
		C	*D*
Player I	*C*	56, 56	0, 60
	D	60, 0	50, 50

CI=0.9		Player II	
		C	*D*
Player I	*C*	56, 56	0, 60
	D	60, 0	2, 2

CI does not explain all the cases in which people choose to cooperate in the Iterated Prisoner's Dilemma (IPDG); there are also non-CI influenced subjects (Hristova and Grinberg, 2004). This is an indication that there is more than one reason for the presence of cooperation and those reasons probably work together.

2 Approaches for Modeling Cooperation in IPDG

One possible reason for people to cooperate, apart from considering CI, could be that players learn their strategies according to positive or negative reinforcements associated to their past moves in the IPDG (Macy and Flache, 2002).

Players learn with experience that cooperation might be more rewarding than defection in the long run. Fictitious play (Brown, 1951) is a behavior in which a player evaluates reinforcements from situations that did not actually happen but were only imagined (Camerer and Ho, 1999). Additionally, in IPDG there is an opponent whose move players may try to guess. Therefore, it seems natural and realistic that knowledge of the opponent's moves be included in the models. Awareness of the presence of an opponent in IPDG implies that the player tries to make a model of the opponent's strategy which leads to a more complicated behavior related to attempts to utilize this model (Sutton and Barto, 1981). Provided players are able to predict the opponent's move, they may want to maximize their payoff by choosing the most profitable move given the predicted opponent's move. Or in case the players assume the opponents are trying to predict their strategy, they may try to mislead their opponents by pretending to play with different strategy (Camerer et al., 2002; Taiji and Ikegami, 1999). Such sophisticated relations, including theory of mind and social interactions, are related to anticipation in cognition (Rosen, 1985). They provide explanation of cooperation in IPDG based on forward-looking decision-making.

3 Connectionist Model with Anticipation

From a cognitive modeling point of view, the challenge is to understand the decision making mechanisms that would lead to the results observed in the experiments with human participants taking account all of the important characteristics (e.g. the dependence of cooperation on CI or of cooperation on the level of predictive capabilities). We are convinced that an adequate agent model should have a minimal but sufficient level of complexity and should perform in a environment similar to the environments of human experiment participants (e.g. they should perceive the payoff matrix of the game before making a move and take into account the opponent's moves and game outcomes). In the same time human players rely on past experience and on predictions of future events.

The model presented here, following Lalev and Grinberg (2007), is aimed at complying with these requirements. It has taken into consideration the results from extensive recent theoretical and experimental research on the cognitive processes involved in decision making in IPDG (see Hristova and Grinberg, 2004 and Lalev and Grinberg, 2007) using different approaches involving psychological experiments, eye-tracking studies, and modeling and simulations.

It is reasonable to argue that a realistic IPDG model should use different sources of information - such as history of game outcomes, gains, and CI - in parallel in order to be able to have the behavior observed in subjects. Based on all aforementioned assumptions derived from reinforcement learning theory (Macy and Flache, 2002; Camerer and Ho, 1999), anticipation in cognition (Rosen, 1985; Camerer et al., 2002), game theory (Colman, 2003), we have built a connectionist IPDG model player (Lalev and Grinberg, 2007) to account for several effects such as cooperation, received payoffs and CI dependence in the game.

3.1 Architecture

The basic unit in the architecture of the model is an Elman recurrent neural network (Elman, 1990) which has all available game information in time as inputs (see Figure 1). The outputs, in turn, are predictions about game information, such as the opponent's move, for the next game. The purpose of the network is to process the information flow within prolonged IPDG sessions.

All the inputs of the network were re-scaled within the range [0, 1]. As can be seen in Figure 1, the values of the payoffs from the current game matrix (excluding the payoff S which was always 0), as well as the past game payoff received, the player's and opponent's moves in the previous game were presented at the input nodes at each cycle.

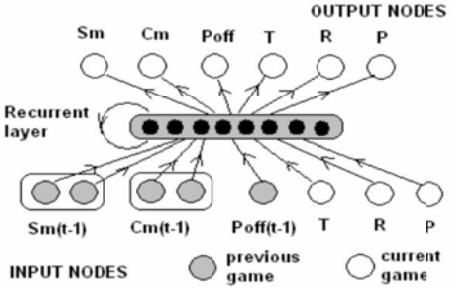


Fig. 1. Schematic view of the recurrent neural network and its inputs and outputs/targets. Notation: Sm and Cm are respectively the simulated subject and computer opponent (probability for) moves; Poff(t) is the player's received payoff at time t.

The past moves were recoded as [0,1] - for C and [1,0] - for D moves, so that activation would always come from any of the two couples of input nodes, no matter what the moves were - C or D.

The values of the T, R, and P payoffs from the current game had to be reproduced as an output by the model thus implementing an in-built autoassociator. There were two reasons to decide to include this component in the network architecture. The first was that this would force the network to establish representations of the games in its hidden layer which is crucial to account for the game payoff structure in the decision-making process. The second one was related to the anticipatory decision mechanism of the model where the output nodes concerning T, R, and P were used as predictions of the next games' payoffs.

At the output, the player's move ('Sm' node) and the computer-opponent's move ('Cm' node) nodes were interpreted as the probability for cooperation for the player and the prediction about the probability for cooperation of his/her opponent in the game at hand. The payoff ('Poff') node represented the expected gain from the current game.

3.2 Training

The network was trained using back-propagation on an input consisting of overlapping sequences of five games - the current game and the four previous games. Such sequences are further called micro-epochs. At each time step, a training micro-epoch was updated with the addition of the current PD game, and the last game was discarded. Then the network was trained as its initial weights were the existing weights from the previous time step.

The values at the six output nodes were used as predictions when the network was trained within the current micro-epoch. The 'T', 'R', and 'P' output nodes were expected to reproduce the corresponding input values in the input payoff matrices. The output of the 'Sm' node was trained with the model-player's probability for cooperation in the current game and the output at the node 'Cm' was the prediction for the cooperation probability of the opponent (after the end of the game when they were already known). The output at the 'Poff' node meant the expected game payoff. When both player and opponent had made their moves, and the payoff for the model-player was known, the network was trained with the inputs it was simulated with and the new targets.

3.3 Model Decision-Making

Decision-making of the model is done with the help of an anticipatory unit that uses predictions from the network and thus the model explores two possible own strategy paths - one, starting with a C move in the present game, and another, starting with a D move. The model uses this forward-looking fictitious play to evaluate corresponding to both choices payoffs and make move decisions that will lead to higher gains in the future. The payoffs from both sequences (PoffC for initial move C and PoffD for initial move D) were then considered.

The obtained payoffs from five fictitious games for each initial move choice were evaluated using a discount factor as follows:

$$Poff_{C,D} = \sum_{t=1}^{5} Poff_{C,D}(t)\beta^{t-1} \tag{1}$$

where Poff C,D(t) is the value of the payoff at moment t, for initial move C or D and β is a discount parameter that indicated to what extend the remote fictitious game payoffs were important for making decisions at present. If β was 0, only the first fictitious payoff would matter, and if β was 1, all the payoffs would be considered as equally important. In IPDG simulations of the model the parameter β was set to $\beta = 0.7$.

$$P(C) = \frac{e^{Poff_C^k}}{e^{Poff_C^k} + e^{Poff_D^k}} \tag{2}$$

where P(C) is the calculated cooperation probability and k is a parameter for the sensitivity of the function towards the difference between PoffC and PoffD. The smaller the value k had, the greater the sensitivity to the difference between the C and D alternative choices became.

4 Game Simulations with Individual Agents: Comparison with Experimental Results

In this Section, the most relevant results from Lalev and Grinberg (2007) will be presented as they are the basis of the MAS simulation presented in Section 4.

4.1 Comparison of the Model with Experimental Results

The agents play individually against a probabilistic Tit-for-two-Tats (Tf2T) computer strategy. Their moves depend on the player's two previous moves, thus being adaptive to their temporal cooperativeness without being easily predictable. The computer opponent probability for cooperation thus obtained is respectively: 0.5 for [C, D] and [D, C], 0.8 for [C, C], and 0.2 for [D, D]. This choice of a computer opponent is the same as the one in the experiments reported in (Hristova and Grinberg, 2004) and allows for a comparison with the experimental results.

The results presented in this section are based on 30 IPDG sessions of two-hundred games against the Tf2T computer strategy. For the comparisons with the experiment the first 50 games are taken to match the number of games played by human participants (see Hristova and Grinberg, 2004). From the experiment reported in Hristova and Grinberg (2004), only data from the first part and for the control condition was used in the comparison. In this experiment (see Hristova and Grinberg, 2004 for details) 30 participants played 50 PD games against the computer opponent described above. After each game the subjects got feedback about their and the computer's choice and could permanently monitor the total number of points they had won and its money equivalent. The subjects received information about the computer's payoff only for the current game and had no information about the computer's total score. This was made to prevent a possible shift of subjects' goal - from trying to maximize the number of points to trying to outperform the computer. In this way, the subjects were stimulated to pay more attention to the payoffs and their relative magnitude and thus indirectly to CI. Games of different CI, ranging from CI = 0.1 to CI = 0.9, were presented both to participants and in simulations with models A and B. Games were coming at random regarding their CI.

The best fit of the experimental results was obtained with the following parameters (see eqs. (1) and (2)): $\beta = 0.7$ and k = 0.05.

Mean Cooperation and Payoffs. The results for the mean cooperation and payoffs for the model and human participants' experimental data taken from Hristova and Grinberg (2004) are respectively presented in Figures 2 and 3. No statistically significant differences are found between the model simulation data and the experiment.

Dependence of Cooperation Rate on CI. The adequacy of the model can be further seen from the comparison of the influence of CI on cooperation displayed

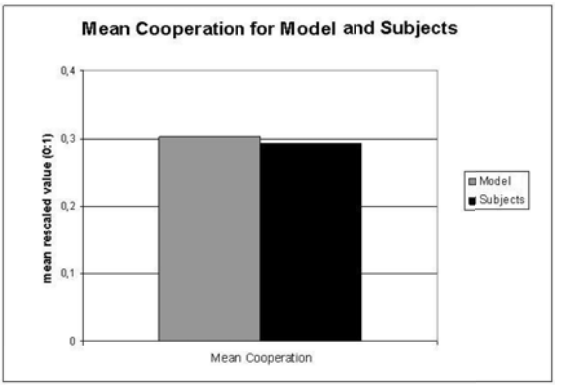


Fig. 2. Comparison of the mean cooperation between the model and the experimental data from (Hristova and Grinberg, 2004)

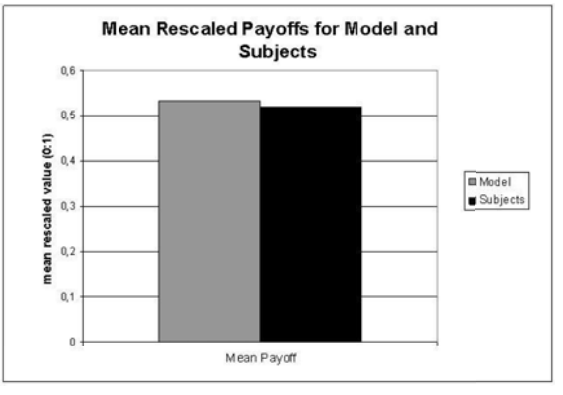


Fig. 3. Comparison of the mean payoffs between the model and the experimental data from (Hristova and Grinberg, 2004)

by the model and by human subjects (see Figure 4; main effect observed with $F=16.908$ and $p<0.01$).

In Figure 4, a detailed comparison, concerning the cooperation rate dependence on CI, between the predictions of the model and the experimental results is shown. It is seen from Figure 4, that the model gives a good description of the experimental results with no statistical differences between the mean cooperation of subjects and the model at all CI levels, and no main effect of the type of player (model or human) on cooperation ($F = 0.386$, $p = 0.856$).

As stated earlier, our main interest is related to the CI dependence of the cooperation rate. The ability to reproduce such details in the experimental data seems very important to us in order to assess the model's validity. The simulation by the model of possible games and moves and outcomes involves the prediction

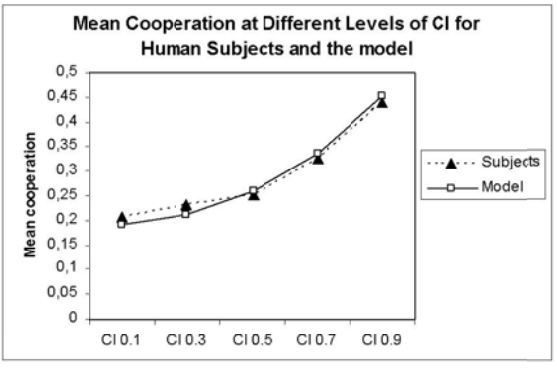


Fig. 4. Influence of CI on cooperation rates for the model and in the experiment from (Hristova and Grinberg, 2004)

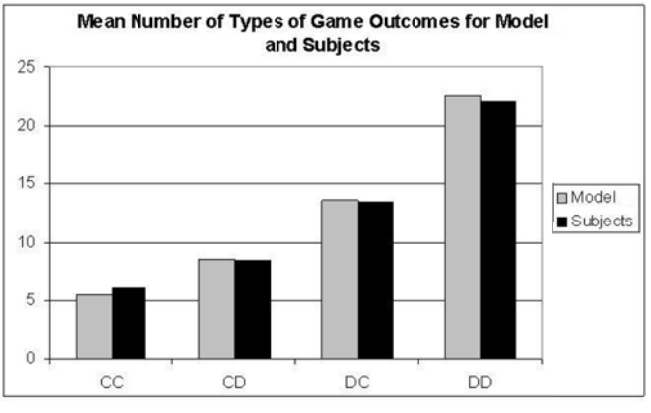


Fig. 5. Comparisons of types of game outcomes for the model with human subjects experiment taken from (Hristova and Grinberg, 2004) (with all values for the CI)

about the payoff structure of the game and thus indirectly of the CI. The main effect in the CI dependence found in the simulations comes from the specific anticipatory form of evaluation of the best move involving the payoffs of the game at hand and of anticipated payoffs reflecting the structure of the current game (see Lalev and Grinberg, 2007 for details).

Comparison of Game Outcomes. In Figure 5, the distribution of the possible types of game outcomes for the model and the subjects were compared and no significant difference was found. This statistics is very important as it shows not only the cooperation rate but gives information on the specifics of the interactions between players. Of special interest is the outcome CC in which both players cooperate.

5 Multi-agent Simulations

As seen from the comparison with experiments with human participants, the model presented in the previous sections gives a good account for human playing in IPDG against a Tf2T player. In this section, we present the results from simulations of the interactions in a society of artificial players implementing such a model. The aim of the simulations was to investigate what is the role of anticipation in a society of payoff-maximizing agents on cooperation and coordination among them.

5.1 Agent Societies

For this purpose, groups of ten agents, with different parameterizations of the model, played IPDG in simulated social environments. They played against eachother in randomly assigned couples. The length of the IPDG interaction sessions was 100 games for a pair of players. The PD game payoff matrices used in the simulations were identical to the ones used in the previous sections i.e. with CI from 0.1 to 0.9 (see the description in Section 4). In a society, only one pair of agents at a time played a whole game session. The pairs were chosen randomly with replacement so it was possible that one or both players from the previous IPDG session also play in the current one. There were 50 sessions in a simulation. After the end of a session, agents kept their trained network weights from their play with the opponent and these weights were kept as initial weights of the agent when it started a new IPDG with the next opponent. The sequences of last inputs and targets were also kept for each particular agent as experience from the previous session. These served as initial inputs and targets in the next IPDG sessions for the agents. When a new session began in the sequence of inputs the values of the new PD game's payoffs were used in the input vector along with the values for the last payoff, last own and opponent's moves. The overall performance of all players in the society determined its specific states and processes. When starting a new IPDG session, each player was influenced by its experiences in previous sessions with other opponents from the same society. In these simulations no mixing of agents from different societies has been done. This simulation scheme was chosen to have some common basis for comparisons with the simulations with Model A alone and with the experimental results reported earlier.

In order to investigate the role of anticipation, several parameters of the agents were varied like the number of the recurrent network's hidden units, the training method and the importance and number of fictitious games used for move evaluation (the parameter β in eq. (1). We considered five societies of agents by varying their capabilities to predict future opponent's moves and received payoffs:

1. Agents without anticipation of payoffs and opponent's move beyond the present PD game, i.e. β=0 in eq. (1) (further referred to as Low-Anticipation society);

2. Agents implementing exactly 'Model A' (30 hidden units) from (Lalev and Grinberg, 2007) used in the comparison with the experimental data in Section 4 (further referred to as Model-A-30 society);
3. Agents with a larger number of hidden units (50 hidden units) which should increase the predictive power for the model (Further referred to as Model-A-50 society);
4. Agents with 50 hidden units and the pseudo rehearsal training method used (see Ans et al., 2002 for details). The method circumvents the neural networks' catastrophic interference problem and improves the learning and therefore the predictive capability of the model by a rehearsal procedure using pseudo training vectors. The agents trained by using this method are very sensitive to the learned in the past in IPDG sessions with other opponents which makes their behavior difficult to predict (Further referred to as Pseudo-rehearsal society);
5. Agents with 50 hidden units and strengthened anticipation predispositions: the number of fictitious games was set to 10 (twice as more as in Model A) as well as the importance of remote games was increased by setting the discount parameter from β=0.7 to β=0.9 (Further referred to as High-Anticipation society).

5.2 Simulation Results and Discussions

In order to compare the five societies of agents formed on the basis of their anticipation capabilities, we have concentrated on the following characteristics: cooperation rate, payoffs, type of games outcomes, and coordination in cooperation (sequences of games in which both agents cooperated).

Cooperation Rates. In a simulated society, agents played ten IPDG sessions on average. With each next session the experience of players grew. In Figure 6, the cooperation rates for all agents in a society are averaged over their subsequent playing sessions from the first to the tenth. For example, the cooperation rates for all agents from their first IPDG session in the simulation are averaged, then for the second and so forth till the tenth.

There was no significant difference between the mean cooperation of the High-Anticipation and Pseudo-rehearsal simulations ($F=1.45$, $p=0.231$) (see Figure 7). These two societies had the highest cooperation rates among the societies as there was a significant difference between the mean cooperation of the Pseudo-rehearsal society and the Model-A-50 society ($F=18.72$, $p<0.01$). There was also no difference in the cooperation rates between the agents from the Model-A-30 and Model-A-50 societies ($F=1.93$, $p=0.168$). Their mean cooperation rates were higher than the mean cooperation in the Low-Anticipation agent society as in the comparison between Model-A-30 and the Low-Anticipation societies $F=69.95$ and $p<0.01$ (see Figure 7).

Overall, the results presented in Figure 6 show that the anticipatory capabilities of adaptive players in social settings may be basic for sustaining reasonable

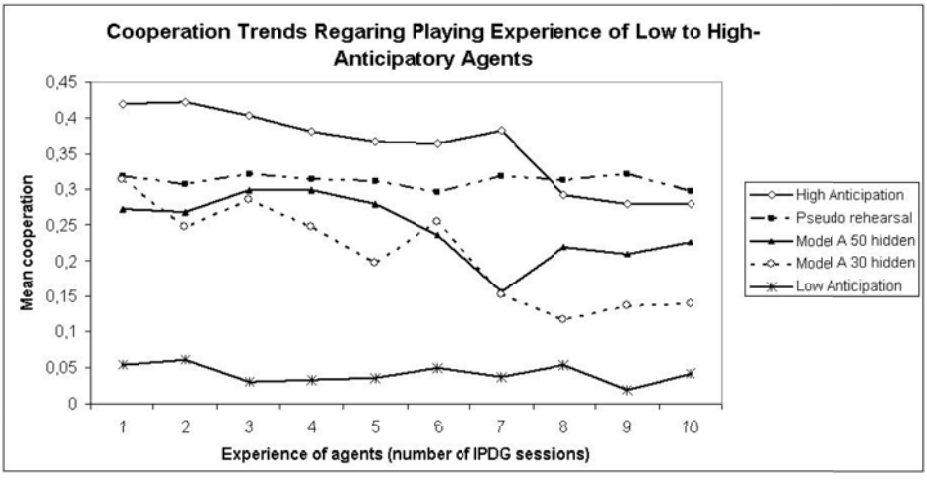


Fig. 6. Comparison between agent societies of the mean cooperation rates as a function of experience measured in terms of the number of IPDG sessions

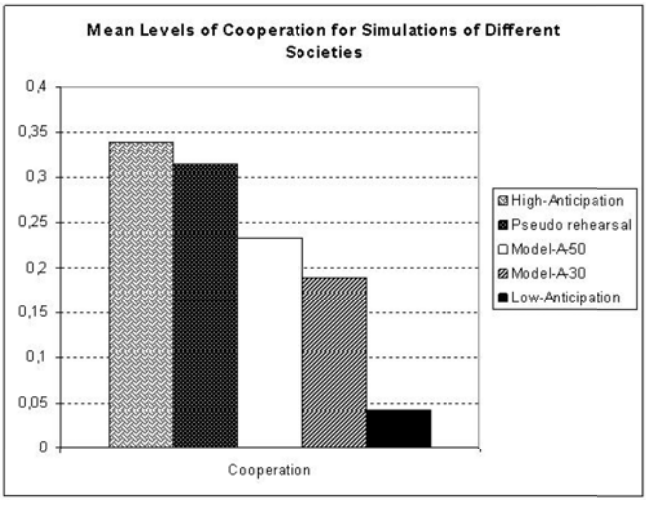


Fig. 7. Mean level of cooperation in simulations

levels of cooperation over time. Only in simulated societies where agents accounted to a higher extent for previous experience and used it to predict further behavior a stable level of cooperation among its players could emerge at least during the first ten IPDG sessions (as in the Pseudo-rehearsal society). In all other cases there was a tendency towards gradual decrease in the cooperation rate with time or low cooperation rate for all sessions.

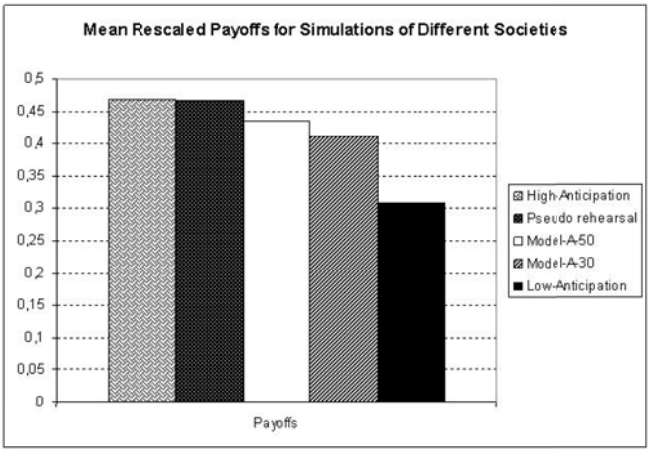


Fig. 8. Mean level of payoffs in simulations

In Figure 7 the mean cooperation in the agent societies is presented. It is seen that cooperation increases with anticipation capabilities and reaches about 0.3 for the High-Anticipation and Pseudo-rehearsal society while in the Low-Anticipation society it is below 0.05.

Payoffs. The mean payoff received by agents is another interesting characteristic because the agents use maximal payoff-based evaluation mechanism (see Figure 8). The High-Anticipation and the Pseudo-rehearsal societies did not differ in the mean payoffs that were received ($F=0.004$, $p=0.953$). They got payoffs higher than the Model-A-50 society: the difference between the Pseudo-rehearsal and Model-A-50 societies was significant ($F=7.82$, $p<0.01$). The payoffs of society Model-A-30 did not significantly differ from those of society Model-A-50 ($F=2.36$, $p=0.128$). The Low-Anticipation society got the lowest payoffs as its payoffs were lower than Model-A-30 society's ($F=62.21$, $p<0.01$).

As a whole, comparison of both the analyses of cooperation and payoffs (Figures 7 and 8) reveal a rule according to which in the simulations higher cooperation rates corresponded to higher payoffs.

Again, as with mean cooperation, the High-Anticipation and the Pseudo-rehearsal societies showed the largest number of CC games and the smallest number of DD games (see Figure 9). The number of CC games was not different for these two simulated societies ($F=0.74$, $p=0.39$). On the other hand, the DD game outcomes were more for the Pseudo-rehearsal society than in the High-Anticipation society ($F=4.99$, $p<0.05$).

The number of CC games was significantly lower for each next society (as follows, in Model-A-50, Model-A-30, and Low-Anticipation societies), and in the Low-Anticipation society they had the smallest number (see Figure 9). Concerning the mutual defection (DD) game outcomes the situation is inverse. In the High-Anticipation society the smallest number DD games was observed. The

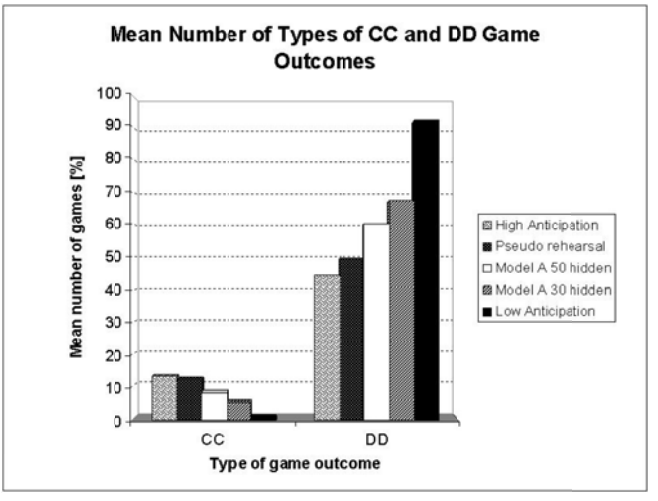


Fig. 9. Comparison of the mean number of CC and DD game outcomes calculated for the agent societies

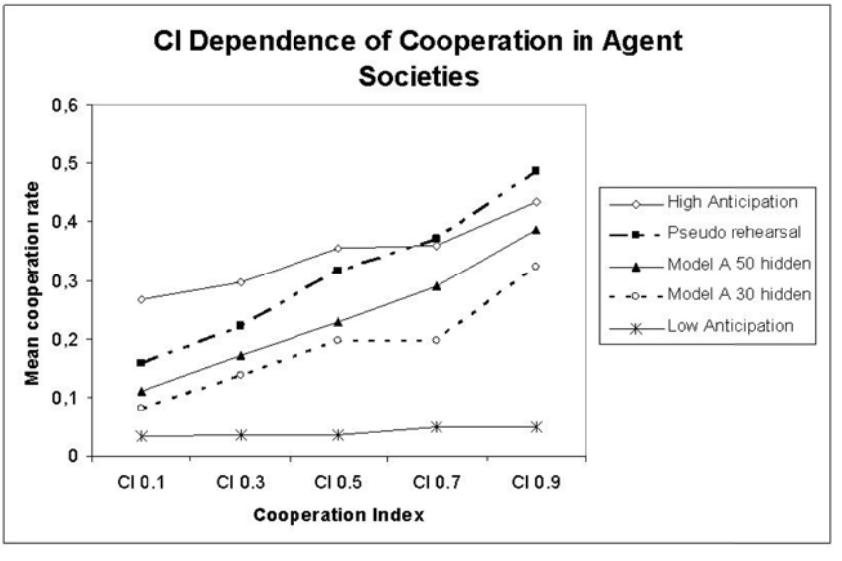


Fig. 10. CI dependence of the mean cooperation rate in the agent societies

largest mean number of DD games per IPDG session (more than 90 percent of the games) was reached in the Low-Anticipation simulation. For the DD game outcomes there was no difference only between Model-A-50 and Model-A-30 societies ($F=1.8$, $p=0.183$).

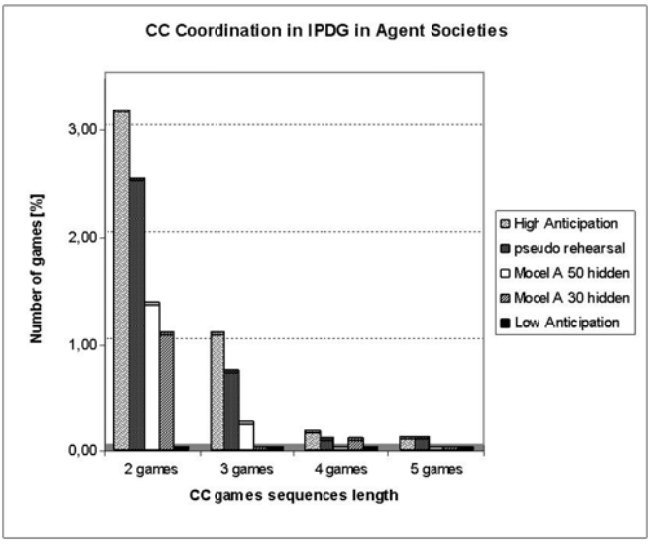


Fig. 11. Agents' coordination in terms of the mean length of the series of mutual cooperation (CC games) per IPDG session averaged over 50 IPDG sessions in each of the agent societies

A tendency of increase of the mean number of CC game outcomes per simulation is observed with increase of the anticipatory propensities of agents in the societies. The opposite is valid for the mean number of DD game outcomes per simulation regarding the anticipatory propensities of agents in the societies (Figure 9).

For each agent society, we calculated the mean cooperation rates of agents for games with a specific CI (see Figure 10). It was interesting to see if the dependence on CI will be preserved in games among the agents using only a recurrent network model and playing against a Tf2T opponent as it was the case in the experiment replication (see Section 3). In all societies the monotonously increasing dependence of cooperation on the CI is clearly observed except for the Low-Anticipation society. This confirms again the role of anticipation in getting this dependence as in the experimental results with human subjects (Rapoport and Chammah, 1965).

Coordination. We adopted as a first measure of the level of coordination between the agents the mean number of CC games played in a row per IPDG session. In Figure 11, the statistics for the agent societies are presented. The longest CC coordination lasted for five games and was present only in the High-Anticipation and Pseudo-rehearsal societies. Four-games-long sequences were observed also in the latter and in the Model-A-30 societies. In the Low-Anticipation society no sequences longer than two were found. Although the sequences are not very long (especially compared to DD sequences some of which were 100 games long) the influence of anticipation is considerable.

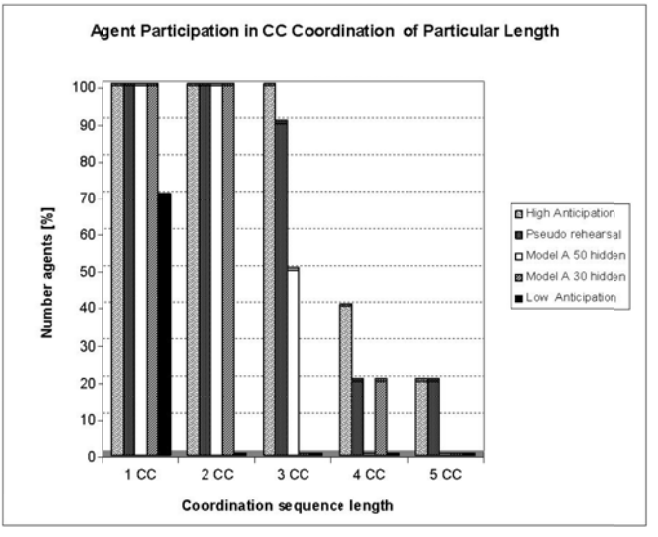


Fig. 12. Number of agents which played a series of CC games of a given length for each agent society

This conclusion is confirmed by a related analysis we performed: the number of agents in a society that participated in a CC game sequence of given length (see Figure 12). It is seen from Figure 12 that for example only 70 percent of the agents from the Low-Anticipation society ever played a CC game whereas for all other societies this percentage equals 100. Moreover, a considerable number of agents with sequences of CC games longer than two are observed only in societies with anticipation.

6 Evolution of Societies

For the parametrization of the agents in the simulated societies was predefined in the simulations in Section 5, the question arose whether anticipatory properties will also appear in societies of evolving agents. Therefore, we run and explored several simulations of the interactions within societies of game playing agents as their parameters for anticipation were subject to evolution. The aim of evolving such societies was either to achieve a highly cooperative society, or alternatively, to obtain a society with the highest possible group payoffs. An additional attempt was made to compare the behavior within a 10-agents' society, and in case there were were only 2 agents interacting with each-other, though the same overall number of games were played by each of the agents in both cases. Then we investigated the evolved parameters.

6.1 Settings of the Simulations

We maintained the general settings and rules of societies as described in Section 5. All agents in a society were identical in their architecture. This included the number of considered fictitious games, discount factor β, and number of hidden units. The interactions within one society from its start, until each agent had played approximately 500 games with other agents, was considered as an individual in the evolutionary algorithm. There were several such individuals that were run in a generation. In the history of each society, every agent had their anticipation parameters unchanged. Still, agents could learn during their play by changing their network weights. When a set of parameters turned out more evolutionary fit in one population, a new population of societies was generated to evolve these parameters further on.

Matlab Genetic Algorithm Toolbox was used to perform the evolutionary search for the best parameters. Here are some details on the used genetic algorithm in the Matlab environment:

1. Evolving populations consisted of 10 individuals (or 10 separate societies) for reasons regarding computational limitations
2. The fitness function evaluated how far the individual was from the ideal case, e.g. 100 percent cooperation
3. The best 5 individuals of a population were chosen to give procreation
4. There were 100 generations for each genetic simulation to last
5. As mutation function 'mutationadaptfeasible' was used
6. The crossover fraction was 0.5
7. The initial values of the parameters for each simulation were deliberately chosen low as follows: 1 fictitious game, β=0.1, and 5 was the number of initial hidden units
8. The allowed ranges of parameters was also considered: [1, 10] - for fictitious games, [0,1] - for β, and [1, 50] - for the number of network hidden units.

6.2 Evolving Cooperative Societies

With these simulations we tried to answer the question whether anticipation will evolve in the agents when cooperation is required. Also we checked if there would be any difference in cooperation by evolution of relatively big societies (10 agents) versus evolution of the minimal group of 2 agents.

Results. The general observation was that all three evolved parameters of the agents outgrew their initial low values during the simulations. The number of fictitious games usually reached 10 at the end of simulations, the discount factor β reached 1, and the number of hidden units reached higher than the initial values - usually about 35-40. This gives us the answer of the first question - in these settings, anticipatory properties evolve in the agents to achieve higher cooperation rates in the group. Cooperation rates grew accordingly robustly in the long run until they reached a plateau (see Figure 13).

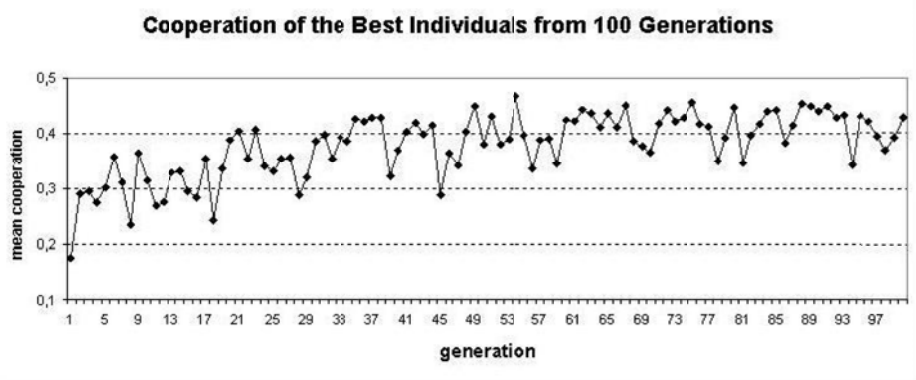


Fig. 13. Mean cooperation of the best individuals during 100 consecutive generations

It turned out that there was no influence of the count of agents in the society on the reached fitness value. In both cases when the society consisted of 10 or only 2 agents, they reached fitness values of about 0.6, corresponding to about 40 percent cooperation for the finally evolved societies.

6.3 Evolving Payoff Maximizing Societies

Using the same setting, we changed the fitness function to a payoff maximizing one. The expected result after evolution was that payoff maximizing, mediated by cooperation, would lead to increase in the anticipation of the agents.

Results. We observed how the parameters responsible for anticipation also grew in this case, thus augmenting the anticipatory properties of the agents and of the group as a whole. Compared to the finally evolved parameters in the evolved cooperation condition, there were, however, several things to notice. Apparently, anticipation developed less in the payoff maximizing condition as the number of fictitious games in payoff maximizing was 6 versus approximately 9 fictitious games in the cooperation maximizing condition. The importance of far forward fictitious payoffs β was about 0.6 in the payoff maximizing condition versus values close to 1 in the cooperation maximizing condition.

As for the size of the group, it turned out that the number of hidden units was higher in case there were only 2 agents in the society trying to maximize their payoffs, than in the case of 10 agents in the society. These numbers were 45 and 22, respectively. This was the only observable parameter difference between both types of evolved payoff maximizing groups. But it obviously had a big impact on the overall performance regarding the fitness in the societies. In the case of 2 agents, the fitness was 0.72 (equal to 0.38 average payoff for the evolved population), whereas in the case of 10 agents in the society, the fitness was much lower (better) - about 0.4 (equal to 0.6 average payoff for the population).

6.4 Discussion

As visible from these simulations, anticipation is a solution to the problem of sustainable high cooperation and payoffs. The evolving societies of agents reached anticipatory properties very close to those we initially predefined in the simulations from Section 5. Results were alike for cooperation no matter what the count of agents in the society was. But anticipation developed more when cooperation was directly linked to the fitness of the evolved populations, than did it develop when payoff was accounted for in the fitness of populations.

There was an interesting observation in evolving of payoff maximizing agents. Judging from the results, the mean payoff for a society could reach higher values when the society is composed of more agents, than when it has less members. We could speculate that when an agent has little opponents to play with in IPDG, it tries to increase its predictive capabilities in order to outguess the opponent's actions more often. This is valid for both members of the small group. They strive to get more of payoff T, but end up with less cooperation, and consequently, smaller payoffs. On the other hand, in a bigger group, there were more possible opponents who were coming randomly one after the other. This would disprove an agent from refining its predictions (increasing the number of network hidden units) and would result in higher cooperation and payoffs.

7 Conclusion

Several multi-agent models of social interaction based on anticipation were presented and discussed. Special attention was devoted to recurrent neural network model used to simulate IPDG playing in a society of agents. The model has been validated by comparison with human subjects experiments in a previous paper (see Lalev and Grinberg, 2007) in which participants played individually against a computer opponent. Several interesting effects could be reproduced which gave confidence that this model could be used in a multi-agent simulation modeling a small society of agents interacting among themselves by playing IPDG. We were interested in the role of anticipation of two essential for successful social functioning characteristics - cooperation and coordination. The agents were distributed in five types of societies based on their anticipatory abilities - from agents with low predictive ability to agents with high predictive one.

The results show that the higher the anticipatory ability is, the higher the cooperation rate and the coordination in cooperation between agents are. In the same time, anticipatory agents opposed to each other get involved into sophisticated behavior making mind-reading difficult. As human cooperation in IPDG is close in rates to the cooperation of our anticipatory agents, the prediction is that coordination series among human subjects may be in close ranges to those, observed in the simulations.

Using an evolutionary approach, we found out how anticipation develops in the agents, as long as the evolutionary fitness of societies is measured in overall cooperation or payoffs. This finding confirms once more the assumption that

anticipation and prediction are solutions to cooperation, coordination and gain maximization.

In general, there are no many investigations of anticipatory agent societies. Further research, e.g. based on the PD game and/or other games, is needed in order to explore the full importance of anticipation for social functioning.

Acknowledgements

This work was supported by the MindRACES project, Contract Number 511931, and euCognition network.

References

Ans, B., Rousset, S., French, R.M., Musca, S.: Preventing catastrophic interference in multiple- sequence learning using coupled reverberating elman networks. In: Proceedings of the 24th Annual Conference of the Cognitive Science Society (2002)

Axelrod, R.: The Evolution of Cooperation. Basic Books, New York (1984)

Brown, G.W.: Iterative solution of games by fictitious play. In: Activity Analysis of Production and Allocation. Wiley, New York (1951)

Camerer, C., Ho, T.-H., Chong, J.: Sophisticated ewa learning and strategic teaching in repeated games. Theory 104, 137–188 (2002)

Camerer, C.F., Ho, T.-H.: Experience-Weighted Attraction Learning in Games: Estimates from Weak-Link Games, in Games and Human Behavior: Essays in Honor of Amnon. Erlbaum, Hillsdale (1999)

Colman, A.M.: Cooperation, psychological game theory, and limitations of rationality in social interaction. Behavioral Brain Science 26, 139–153 (2003)

Elman, J.L.: Finding structure in time. Cognitive Science 14, 179–211 (1990)

Epstein, J.M., Axtell, R.: Growing Artificial Societies: Social Science from the Bottom Up. MIT Press, Cambridge (1996)

Hristova, E., Grinberg, M.: Context effects on judgment scales in the prisoner's dilemma game. In: sur Yvette, G. (ed.) Proceedings of the 1st European Conference on Cognitive Economics (2004)

Lalev, E., Grinberg, M.: Backward vs. forward-oriented decision making in the iterated prisoner's dilemma: A comparison between two connectionist models. In: Butz, M.V., Sigaud, O., Pezzulo, G., Baldassarre, G. (eds.) ABiALS 2006. LNCS, vol. 4520, pp. 345–364. Springer, Heidelberg (2007)

Macy, M., Flache, A.: Learning dynamics in social dilemmas. PNAS 99, 7229–7236 (2002)

Rapoport, A., Chammah, A.: Prisoner's Dilemma: A Study in Conflict and Cooperation. University of Michigan Press, Ann Arbor (1965)

Rosen, R.: Anticipatory Systems. Pergamon Press, Oxford (1985)

Sutton, R.S., Barto, A.G.: An adaptive network that constructs and uses an internal model of its world. Cognition and Brain Theory (4), 217–246 (1981)

Taiji, M., Ikegami, T.: Dynamics of internal models in game players. Physica D 134, 253–266 (1999)

A Two-Level Model of Anticipation-Based Motor Learning for Whole Body Motion

Camille Salaün, Vincent Padois, and Olivier Sigaud

Université Pierre et Marie Curie - Paris6
Institut des Systèmes Intelligents et de Robotique, CNRS UMR 7222,
4 place Jussieu, F-75005 Paris, France
{Camille.Salaun,Vincent.Padois,Olivier.Sigaud}@upmc.fr

Abstract. We present a model of motor learning based on a combination of Operational Space Control and Optimal Control. Anticipatory processes are used both in the learning of the dynamics model of the system and in the coordination between both types of control. In order to illustrate the proposed model and associated control method, we apply these principles to the control of a simplified virtual humanoid performing a stand-up task starting from a crouching posture.

1 Introduction

Early in the history of motor control studies, Bernstein raised the following paradox: thanks to the redundancy of their motor system, human beings can realize a task with an infinity of ways. However, for a given task, they always reproduce the same kind of stereotypic motion. The problem consists in explaining where these regularities come from, given the diversity of possible solutions [1].

An attractive solution to this problem consist in stating that motor control optimises some criterion, thus the performed motion among the many possible ones is the optimal one with respect to that particular criterion. Several potential criteria have been proposed such as the *Minimum jerk* [2], the *Minimum torque-change* [3] and their variants (*Minimum motor command change* [4] and *Minimum commanded torque change* [5]). These criteria are all based on the idea that human motion must optimise *smoothness*, but a principled explanation of why motion should optimise smoothness is missing.

By contrast, the *Minimum end-point variance* criterion comes with a convincing explanation: natural movements must reach their target accurately, in spite of a neural noise that is proportional to the motor signal. Thus motor control must minimise the error on the terminal position of the controlled effector. Harris and Wolpert [6] showed that an optimal controller based on this principle can reproduce all motor invariants on which previous criteria were based, but also additional invariants such as Fitts' law [7] which relates accuracy to the speed of execution.

The direct consequence of this criterion is the *Minimum intervention principle* [8] that stipulates that the motor system activates muscles as few as possible to

G. Pezzulo et al. (Eds.): ABiALS 2008, LNAI 5499, pp. 229–246, 2009.

achieve its task, so as to minimise the motor signal related noise. As a result, the generated control only acts in the dimensions that are relevant with respect to the task and rejects the noise to the other dimensions. This fact is validated experimentally by [9] who showed that fluctuations are larger on the *Uncontrolled manifold* consisting of the irrelevant dimensions than on the relevant ones.

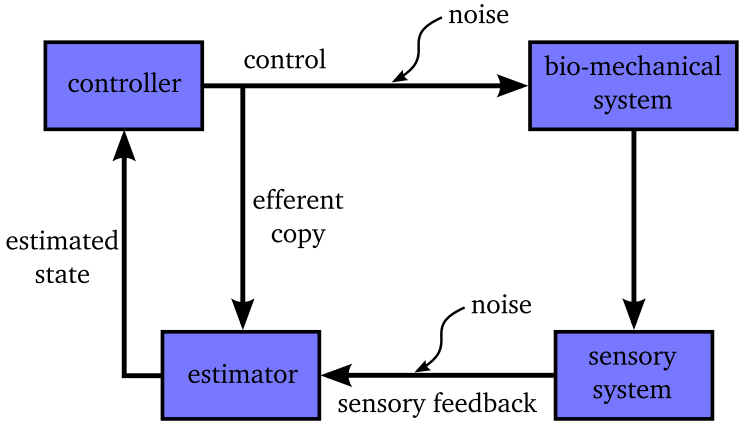


Fig. 1. A global view of SOFC: the controller must estimate the state of the system and maximise movement accuracy in the presence of sensory and motor noise

Based on these findings, the Stochastic Optimal Feedback Control (SOFC) method [10,11], illustrated in Fig. 1, and some variants (*Terminal Optimal Feedback Controller* [12,13,14] and *Task Optimisation in the Presence of Signal dependent noise* [15]) have received a wide agreement as good models of human motor control for elementary tasks for a single arm. However the computational cost of Optimal Control (OC) methods such as SOFC make them unsuitable to solve larger problems such as the synthesis of whole body motion. Thus an important issue in this domain consists in finding computationally less expensive approaches endowed with the same properties.

Another issue is concerned about learning and adapting to changing dynamic conditions. Several authors [16,17] propose a general framework in which human motor control is model-based, combines feedforward and feedback control processes [18] and calls upon optimisation processes as explained above. In this framework, motor adaptation results from learning the dynamics model of the system.

These principles were first implemented in *Modular Selection And Identification for Control* (MOSAIC) [19], a modular architecture where each module gets specialised to deal with particular dynamic circumstances. In MOSAIC, control learning is based on an handcrafted corrector that requires a priori knowledge

on the dynamics of the system. *Multiple Model-Based Reinforcement Learning* (MMRL) [20] solves this limitation by introducing a reinforcement learning process. Each module in MMRL calls upon an optimal control module, thus it still suffers from the computational cost of OC approaches.

One way to overcome this computational cost problem is suggested by Shadmehr's global view of the motor system [21]. From studies based on reaching and pointing movements, Shadmehr considers that motion generation can be decomposed into two parts: at the higher level, the central nervous system computes, in a visual frame of reference centred on the fixation point, a vector from the end-point effector to the target. This vector defines a trajectory in the operational visual space. Then this trajectory is converted into a muscular, low-level control of the joints angles by calling upon learned visuo-articular velocity and position mappings.

The most adequate robotics control tool to formalise Shadmehr's intuition consists in projection methods, such as Operational Space Control (OSC) [22]. These methods perform their computation in the so-called operational space, a space relative to the task, which is usually smaller than the joints space. They can thus be applied to large robotic systems [23]. They also give rise to a mathematically straightforward way of decoupling a set of tasks ranked by priority [24], but are sensitive to these priorities and cannot directly perform a global optimisation of a cost function associated to the motion of the system over time.

By contrast, optimisation methods, such as OC perform their computation in the joints space, but benefit from the easy definition of the constraints and performance criteria [25] of the tasks. However, the only way to express priorities among tasks within this framework consists in tuning the relative weights of the corresponding performance criteria.

In this contribution, we present an hierarchical model combining the assets of both control methods. Our model is consistent with Shadmehr's view in that a high level control loop computes a global trajectory in the operational space and then a low level control loop dynamically realizes it at the joints level. Besides, our model calls upon a velocity kinematics model and a dynamics model of the system, respectively. We show how these models can be learned from experience within an anticipatory process. Finally, the coordination of both control levels also results from an anticipatory process, the control set point being chosen as a point that is forward in time with respect to the current time. Our model is functional rather than neuromimetic: It focuses on the level of computational principles of human motor control and does not claim anything about the underlying neuroanatomy.

The contribution is organised as follows. Section 2 gives the technical background of each component of our model. The global structure of the controller itself is presented in section 3. In section 4, we describe an empirical study about the control of a simplified virtual humanoid performing a stand-up task starting from a crouching posture. We discuss the results in section 5.

2 Background

We present a model of human motor learning that combines the definition of tasks in the operational space, the use of OC at the joints level and experience-based, incremental model learning processes. An inverse velocity kinematics model is learned with the *Locally Weighted Projection Regression* (LWPR) non-linear function approximation algorithm [26] whereas a direct linearised model of the dynamics is learned with a classical Recursive Least Squares algorithm. Learning is performed on-line, during the control of the system. The different components of the resulting adaptive control architecture are described below.

2.1 Forward and Inverse Velocity Kinematics

The operational space or task space is the most natural space to describe the motion of the end effector of a system, whereas the joints space is the most natural space to control the system. The OSC approach consists in specifying the goal in the operational space and then using a linear transformation from the operational space to the joints space to control the system. Let $\boldsymbol{\xi}$ be an m-dimensional vector of operational coordinates used to describe the task (e.g. position and orientation of the end-point effector) and let $\boldsymbol{q}$ be an n-dimensional vector of generalised coordinates – *i.e.* the vector of parameters necessary to describe the configuration of the system without ambiguity. For systems with a fixed base, the generalised coordinates are often chosen as the joints angles. The relation $\boldsymbol{\xi} = f(\boldsymbol{q})$ is called *forward kinematics model* (FKM). It describes the operational coordinates as a function of the configuration of the system. Differentiating with respect to time, we obtain

$$\dot{\boldsymbol{\xi}} = J(\boldsymbol{q})\dot{\boldsymbol{q}}, \tag{1}$$

where $J(\boldsymbol{q}) = \dfrac{\partial f(\boldsymbol{q})}{\partial \boldsymbol{q}}$ is the Jacobian matrix of f.

To control the system, we need $\dot{\boldsymbol{q}}$ as a function of $\dot{\boldsymbol{\xi}}$. For a redundant system ($rank(J) < n$) for which the inversion of (1) has an infinity of solutions, one particular solution is:

$$\dot{\boldsymbol{q}} = J(\boldsymbol{q})^{\sharp}\dot{\boldsymbol{\xi}} \tag{2}$$

where $J(\boldsymbol{q})^{\sharp}$ is a weighted pseudoinverse of $J(\boldsymbol{q})$ [27].

Weighted pseudo-inverse of $J(\boldsymbol{q})$ are written:

$$J(q)^{\sharp} = W^{-1}J(\boldsymbol{q})^{T}\left[J(\boldsymbol{q})W^{-1}J(\boldsymbol{q})^{T}\right]^{-1}$$

where W is a symmetric and positive matrix of dimension n. This particular set of solutions to (1) minimises the Euclidean *W-weighted* norm $\sqrt{\dot{\boldsymbol{q}}^{T}W\dot{\boldsymbol{q}}}$ of the solution. In the case where $W = I_n$, this solution is called the Moore-Penrose or pseudoinverse of the Jacobian matrix and is noted J^{+}.

The general form of the solution can be written as:

$$\dot{\boldsymbol{q}} = J(\boldsymbol{q})^{\sharp}\dot{\boldsymbol{\xi}} + \left(I_n - J(\boldsymbol{q})^{\sharp}J(\boldsymbol{q})\right)\dot{\boldsymbol{q}}_0, \tag{3}$$

where $\left(I_n - J(\boldsymbol{q})^\sharp J(\boldsymbol{q})\right)$ is a projector onto the nullspace of $J(\boldsymbol{q})$ (*i.e.* the space of internal mobility with respect to the main task) and $\dot{\boldsymbol{q}}_0$ any vector of dimension n. The second term of the right-hand term of equation (3) gives access to the redundancy of the system in a hierarchical manner: any secondary task or constraint projected onto the nullspace of the Jacobian matrix is given less priority than the main task. It is achieved as long as it does not disturb this main task.

2.2 Operational Space Control

OSC consists in reaching a target $\boldsymbol{\xi_{ref}}$ from an initial position. This control method uses the inverse velocity kinematics model described by equation (2) as well as a proportional controller to compute the desired operational velocity:

$$\dot{\boldsymbol{\xi}}_{\boldsymbol{des}} = K_P\left(\boldsymbol{\xi}_{des} - \boldsymbol{\xi}\right), \tag{4}$$

where K_p is a positive definite matrix used as a proportional gain. This method is called *resolved rate motion control*.

Using the inverse velocity kinematics model described by equation (2), we have:

$$\dot{\boldsymbol{q}}_{des} = J(\boldsymbol{q})^\sharp \dot{\boldsymbol{\xi}}_{des}. \tag{5}$$

Thus, given a desired task velocity $\boldsymbol{\xi_{ref}}$ and knowing $J(\boldsymbol{q})^\sharp$, one can derive the desired joints velocities $\dot{\boldsymbol{\xi}}_{\boldsymbol{des}}$. If one wishes to specify the desired joints accelerations, one may use a progressive differentiation:

$$\ddot{\boldsymbol{q}}_{des} = \frac{\dot{\boldsymbol{q}}_{des} - \dot{\boldsymbol{q}}}{\Delta t} \tag{6}$$

Independently of the formalism used (Lagrange or Newton-Euler), the equations of motion of a fully-actuated, holonomic system in the contact-free case can be written:

$$\boldsymbol{\Gamma} = M(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{b}(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \boldsymbol{g}(\boldsymbol{q}) \tag{7}$$

where M is the symmetric positive definite inertia matrix of the system, $\boldsymbol{b}$ is the vector of nonlinear effects modelling centrifugal and Coriolis forces, $\boldsymbol{g}$ is the gravity vector and $\boldsymbol{\Gamma}$ represents the torques applied to the system. This inverse dynamics model is used to compute, for a given state, the torque that must be applied to the system to get the desired joint accelerations:

$$\boldsymbol{\Gamma} = M(\boldsymbol{q})\ddot{\boldsymbol{q}}_{des} + \boldsymbol{b}(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \boldsymbol{g}(\boldsymbol{q}) \tag{8}$$

where $\ddot{\boldsymbol{q}}_{des}$ is computed from $\dot{\boldsymbol{\xi}}_{\boldsymbol{des}}$ using equations (4), (5) and (6).

2.3 Linearisation of the Dynamics Model

It is possible to linearise the dynamics model around a state o:

$$F_{1_{\boldsymbol{o}}}\ddot{\boldsymbol{q}} + F_{2_{\boldsymbol{o}}}\dot{\boldsymbol{q}} + F_{3_{\boldsymbol{o}}}\boldsymbol{q} = \boldsymbol{\Gamma},$$

We can reformulate this as:

$$\begin{bmatrix}\dot{\boldsymbol{q}}\\ \ddot{\boldsymbol{q}}\end{bmatrix} = \begin{bmatrix}0 & Id\\ -F_{1_{\boldsymbol{o}}}^{-1}F_{3_{\boldsymbol{o}}} & -F_{1_{\boldsymbol{o}}}^{-1}F_{2_{\boldsymbol{o}}}\end{bmatrix}\begin{bmatrix}\boldsymbol{q}\\ \dot{\boldsymbol{q}}\end{bmatrix} + \begin{bmatrix}0\\ F_{1_{\boldsymbol{o}}}^{-1}\end{bmatrix}\boldsymbol{\Gamma}.$$

This formula is called state representation and can be rewritten $\dot{\boldsymbol{x}} = A\boldsymbol{x} + B\boldsymbol{u}$.

The state and action matrices A and B generally depend on the state. This will not be the case in the study hereafter, the dynamics model being linearised. In its discrete form, this equation is written:

$$\boldsymbol{x}_{k+1} = A'\boldsymbol{x}_k + B'\boldsymbol{u}_k \tag{9}$$

where $\boldsymbol{x}_k = \begin{bmatrix}\boldsymbol{q}_k\\ \dot{\boldsymbol{q}}_k\end{bmatrix}$ and $\boldsymbol{u}_k = \boldsymbol{\Gamma}_k$.

$\boldsymbol{x}_{k+1}$ is the future state whereas $\boldsymbol{x}_k$ and $\boldsymbol{u}_k$ are the current state and control input. A' and B' are the current state and action matrices.

In this study, we learn a model of the dynamics using the state space form of equation (9).

2.4 Optimal Control

Optimal control is a family of control methods that optimize a cost function over the achievement of a task. In the *Linear Quadratic Regulator* (LQR) framework in discrete time, the cost function (or optimality criterion) is written:

$$J = \sum_{k=0}^{\infty}(\boldsymbol{x_k}^T Q\boldsymbol{x_k} + \boldsymbol{u_k}^T R\boldsymbol{u_k}),$$

where Q and R are semi-definite positive and definite positive matrices associated to the cost on the state error and on the control input respectively.

The minimization of the cost function is obtained by solving the Riccati equation whose unknown is P:

$$A'^T PA' - \left(A'^T PB\right)\left(R + B'^T PB'\right)^{-1}\left(B'^T PA'\right) + Q = P,$$

where A' and B' are the state representation matrices defined above.

Solving this equation provides $L = \left(R + B'^T PB'\right)^{-1} B'^T PA'$, a constant state feedback matrix used to generate the following feedback controller:

$$\boldsymbol{u} = L\left(\boldsymbol{x}_{des} - \boldsymbol{x}\right),$$

where $\boldsymbol{x} = \begin{bmatrix} \boldsymbol{q} \\ \dot{\boldsymbol{q}} \end{bmatrix}$ and where $\boldsymbol{x}_{des} = \begin{bmatrix} \boldsymbol{q}_{des} \\ \dot{\boldsymbol{q}}_{des} \end{bmatrix}$. $\dot{\boldsymbol{q}}_{des}$ is computed using the inverse velocity kinematics model described by equation (2) whereas $\boldsymbol{q}_{des}$ is extrapolated from the current configuration:

$$\boldsymbol{q}_{des} = \boldsymbol{q} + \dot{\boldsymbol{q}}_{des} \Delta t.$$

A schematic view of LQR in shown in Fig. 2.

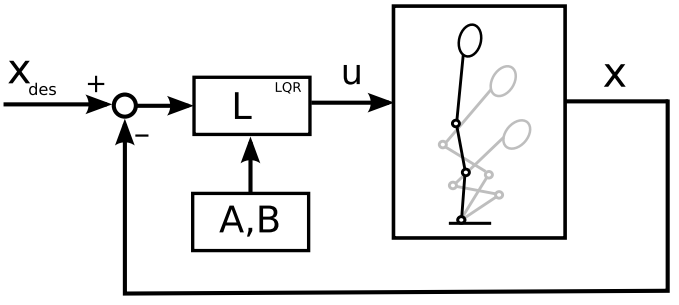


Fig. 2. LQR Feedback control: The feedback matrix L is computed from A' and B' using a discrete time Riccati equation

2.5 Model Learning

We want to learn models of our system incrementally and from experience. The nonlinear function we want to approximate can be written $\boldsymbol{y} = f(\boldsymbol{x})$. We use two techniques, one for obtaining a unique linear model β with $\boldsymbol{y} \approx \beta \boldsymbol{x}$ and the second for obtaining a nonlinear one $\hat{f}$ with $\boldsymbol{y} \approx \hat{f}(\boldsymbol{x})$ where $\hat{f}$ is a combination of linear models.

Least squares and recursive least squares. In the linear case, given some input-output data $X_k = [\boldsymbol{x_1}\ \boldsymbol{x_2}\ ...\ \boldsymbol{x_k}]$ and $Y_k = [\boldsymbol{y_1}\ \boldsymbol{y_2}\ ...\ \boldsymbol{y_k}]$ organised into a matrix, one can obtain the β matrix using the Least Squares approach: By multiplying $Y_k = \beta_k X_k$ by $X_k^T (X_k X_k^T)^{-1}$ on both sides, one gets $Y_k X_k^+ = \beta_k$ where $X_k^+ = X_k^T (X_k X_k^T)^{-1}$ and β_k represents the average for each term over β_1 ... β_k in the least squares sense.

This algorithm can be made incremental through a recursive formulation:

$$\beta_k = \beta_{k-1} + (\boldsymbol{y_k} - \beta_{k-1} \boldsymbol{x_k})\, \lambda$$

Thus, from this so-called Recursive Least Squares (RLS) method, we can get β_k for any k given the previous matrix and a forgetting factor λ that must be tuned.

LWPR. For most systems, linear models will not be accurate enough and must be replaced by nonlinear or piecewise-linear approximators such as neural networks, decision trees [28] or radial basis function networks [29].

The LWPR algorithm [30] is an incremental radial basis function approximator which provides accurate approximation in very large spaces in a $O(k)$ complexity, where k is the number of sample data. We use it here to learn the FKM of our robot. The algorithm uses a combination of linear models, which are valid on an elliptic zone of the input spaces and whose relative strengths are weighted by a gaussian. This space may evolve during training to match the training data. The domain of validity of each model is called a Receptive Field (RF). The prediction of an entire LWPR model on an input vector is the weighed sum of the results of all the active surrounding RFs. The RFs of a model are created when new input data are not part of any existing RF. Conversely, when a field overlaps another, some criterion can be used to delete it.

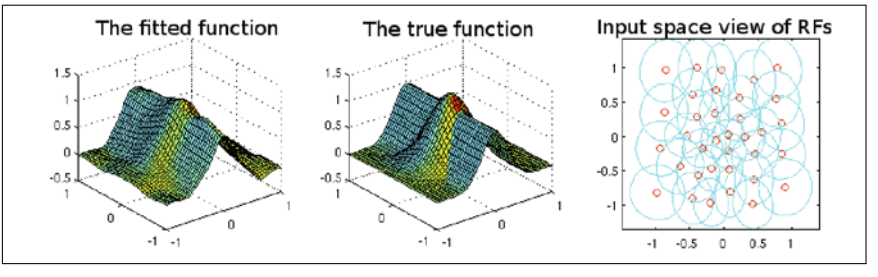


Fig. 3. Example of LWPR learning on a sample function

Each RF first projects the input vector on the most relevant dimensions for estimating the output vector by using Partial Least Squares [31]. During each update, the projector is updated and the algorithm checks whether increasing the complexity by adding another dimension to the input projection significantly reduces the estimation error. If it is so, it modifies the projector accordingly. The projected vector is then used in the n dimension linear model (n being the output dimension) to give the output of the RF.

During prediction with an input vector, the distance between the vector and the RF area is tested for activation and only the significant RFs are activated (see Fig. 3 for an example of RF repartition for function approximation).

The latest version of the algorithm also computes the first and second differential of the learned function with respect to each input dimension (Jacobian and Hessian matrices) when learning the model. LWPR can estimate them for any input value. This calculation is made easier by the fact that the model is a simple sum of multiple linear functions which are easily differentiated. We use this method to extract the Jacobian matrix from the learned model.

3 Our Motor Learning Model

Our global control scheme, illustrated in Fig. 4, consists of:

- a definition of the task in the operational space, resulting on the easy definition of tasks;
- learning the inverse velocity kinematics model with LWPR and a high level control law based on a proportional controller given by equation (4);
- learning the dynamics model with RLS and a low level optimal control law optimising a quadratic cost.

Our presentation below distinguishes three layers: high level control, low level control and the coupling between the two.

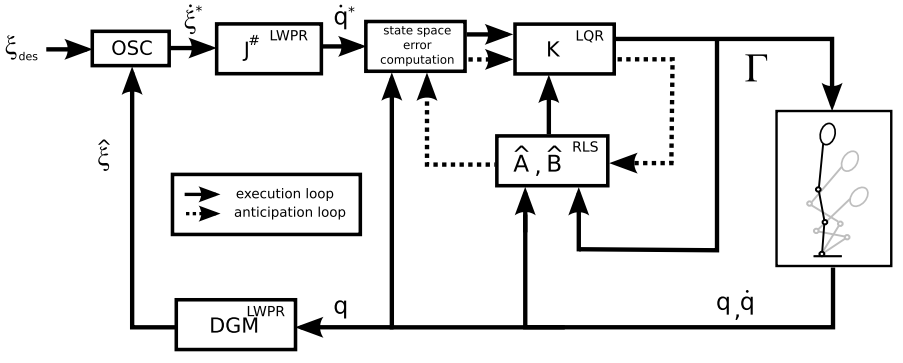


Fig. 4. Global control scheme

3.1 High Level Control

The operational space control part of our controller consists in the proportional controller given by equation (4) based on the difference between $\boldsymbol{\xi}_{des}$ and $\hat{\boldsymbol{\xi}}$. $\hat{\boldsymbol{\xi}}$ is the output of the FKM learned with LWPR, taking the joints positions as input and the operational space position as output.

After equation (4), we apply equation (5) to transform the error in the operational space into a desired velocity in the joints space. The inverse velocity kinematics model is learned with LWPR taking $\dot{\boldsymbol{\xi}}$ as input and $\dot{\boldsymbol{q}}$ as output, as shown in Fig. 5. This approach is similar to the one used in the work of D'Souza *et al.* in [32] who also choose to directly learn a specific inverse of the Jacobian matrix using the LWPR algorithm. Given the fact that there is an an infinity of inverses for the Jacobian matrix, the choice of a specific inverse leads to a loss of information regarding the redundant nature of the system. However, the learned

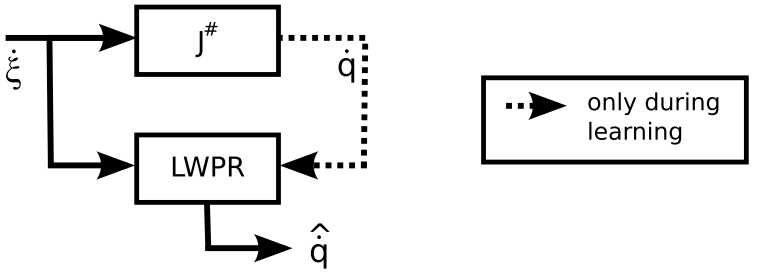


Fig. 5. Learning a generalised inverse with LWPR

inverse which we can expect to be a weighted pseudoinverse $J(\boldsymbol{q})^{\sharp}$, *i.e.* to have minimum norm properties, is the one that best fits the trajectories generated so far.

3.2 Low Level Control

At the low level, we use an LQR controller to track the desired joints velocities $\dot{\boldsymbol{q}}$ computed by the high level control. In the linearised dynamics model $\boldsymbol{x}_{k+1} = A\boldsymbol{x}_k + B\boldsymbol{u}_k$, the state $\boldsymbol{x}_k$ is $[\boldsymbol{q}\ \dot{\boldsymbol{q}}]^T$, the action $\boldsymbol{u}_k$ corresponds to the vector of joints torques, and the matrices A and B are learned incrementally with the RLS algorithm as shown in Fig. 6, taking $(\boldsymbol{x}_k,\boldsymbol{u}_k)$ as input and $\boldsymbol{x}_{k+1}$ as output. Note that, in order to bootstrap the learning process of RLS, rather than initialising A and B randomly, we rather store samples $(\boldsymbol{x}_k,\boldsymbol{u}_k,\boldsymbol{x}_{k+1})$ and apply a standard LS method until A and B are full rank.

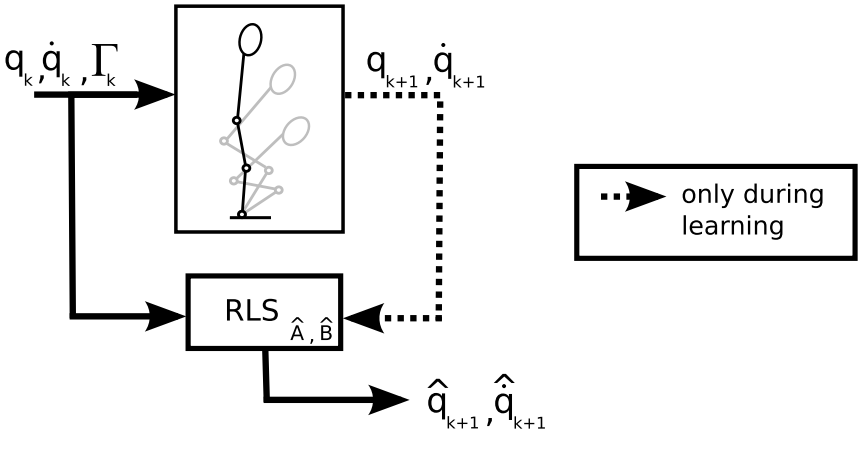


Fig. 6. Learning the dynamics model with RLS

3.3 Coupling Both Levels with Anticipation

If we take the current $\dot{\boldsymbol{q}}$ and $\boldsymbol{q}$ coming from the high level controller as reference, the tracking delay combined with the estimation errors results in an oscillatory behaviour. Thus, we couple the high level and low level control loops by anticipating the values of $\dot{\boldsymbol{q}}$ and $\boldsymbol{q}$ after some horizon H in the future and using these vectors in the low level control objective function to track them.

Algorithm 1. Anticipatory coupling algorithm.

1 $\boldsymbol{\xi}_0 = \boldsymbol{\xi}$;
2 **for** $i = 0$ *to* H **do**
3 $\quad \dot{\boldsymbol{\xi}}_{i+1} = K_p\left(\boldsymbol{\xi}_{des} - \boldsymbol{\xi}_i\right)$;
4 $\quad \dot{\boldsymbol{q}}_{i+1} = J^{\sharp}\dot{\boldsymbol{\xi}}_{i+1}$;
5 $\quad \boldsymbol{q}_{i+1} = \boldsymbol{q}_i + \dot{\boldsymbol{q}}_{i+1}\Delta t$;
6 $\quad \boldsymbol{\xi_{i+1}} = FKM\left(\boldsymbol{q}_{i+1}\right)$;
7 $\boldsymbol{u}_{0H} = LQR\left(A, B, [\boldsymbol{q}_H \dot{\boldsymbol{q}}_H]^T\right)$;

To perform such an anticipation, taking the current operational point $\boldsymbol{\xi}$ as initial value $\boldsymbol{\xi}_0$, we perform a loop on $\boldsymbol{\xi}_i$. First, we infer its derivative $\dot{\boldsymbol{\xi}}_i$ with a proportional controller (Algol. 1, line 3). This operational velocity can be translated into joints velocities through the weighted pseudoinverse of the Jacobian matrix (line 4). From these joints velocities, we estimate the next configuration (line 5) which, from the FKM, can be translated into an estimation $\boldsymbol{\xi_{i+1}}$ of the next operational position (line 6). We iterate this anticipation H times and this results in values for $\dot{\boldsymbol{q}}$ and $\boldsymbol{q}$ after H time steps. Finally, $\boldsymbol{u}_{0H}$ is the control input applied at moment 0 with horizon H.

Note that the way we estimate the next configuration does not take dynamics effects into account. We assume that our controller will reach the desired state at each step, which may not be true in practice if dynamics effects are not fully balanced by the low level controller.

By using the state representation $\boldsymbol{x}_i = \begin{bmatrix} \boldsymbol{q}_i \\ \dot{\boldsymbol{q}}_i \end{bmatrix}$, one can improve the estimation by using the $\boldsymbol{x}_{i+1} = A'\boldsymbol{x}_i + B'\boldsymbol{u}_i$ into the anticipation loop, but preliminary experiments have shown that this does not result in a significant difference in our context.

4 Empirical Study

4.1 A Simplified Virtual Humanoid System

To evaluate our control architecture, we work with a simplified 3 degrees of freedom (DOFs) mannequin model whose feet are fixed to the ground so as to alleviate the need to care about equilibrium. Thechoice of this simple model is intented to facilitate the presentation of our approach which we expect to be easily scalable to much higher dimension systems. The mannequin is simulated

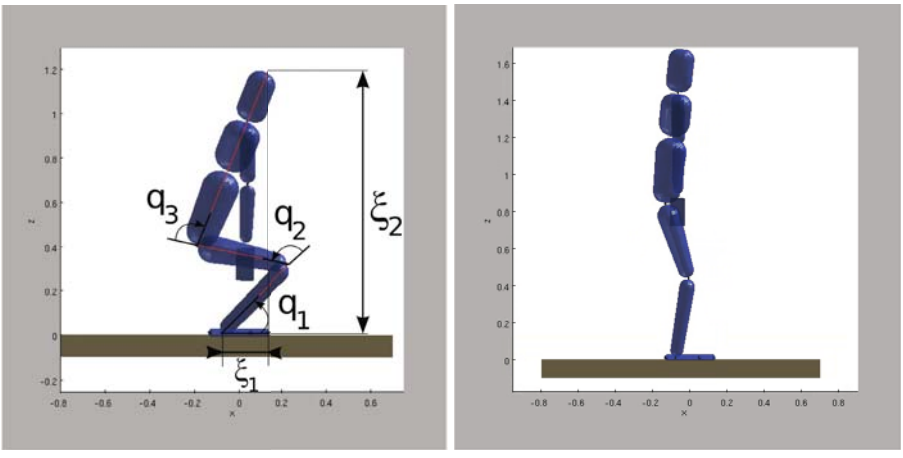


Fig. 7. Our 3 DOFs mannequin in its initial and final configurations

with Arboris [25], a Matlab simulator dedicated to the study of the dynamics of poly-articulated tree-like systems. Its parameters of our simplified mannequin model are extracted from anthropometric tables [33]. It is 1.74 meters tall and weighs 73 kg.

4.2 Task and Parameters

The task consists in making the mannequin stand up from a crouching posture, so that the vertex (upper extremity of the head) reaches 1.70 meters and stabilises over 1.65 meters, as shown in Fig. 7. Thus our operational task consists in reaching $\boldsymbol{\xi}_{des} = \begin{bmatrix}\boldsymbol{\xi}_x \; \boldsymbol{\xi}_y\end{bmatrix}^T = \begin{bmatrix}1.70 \; 0.05\end{bmatrix}^T$.

As for parameters, in (4), after testing diverse values, we take $K_P = 10$. Furthermore, a preliminary study of the low level control loop resulted in tuning the Q and R matrices of the LQR controller as follows:

$$Q = \begin{bmatrix} 0.4 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.6 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.05 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.3 \end{bmatrix} \quad R = 10^{-4} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

4.3 Empirical Results

Fig. 8 shows that, for a controller without an anticipation loop, the system is able to stand-up, but a vertical oscillation may be observed. With a short anticipation ($h = 2$ to 6), the system still reaches the operational target but is able to stay

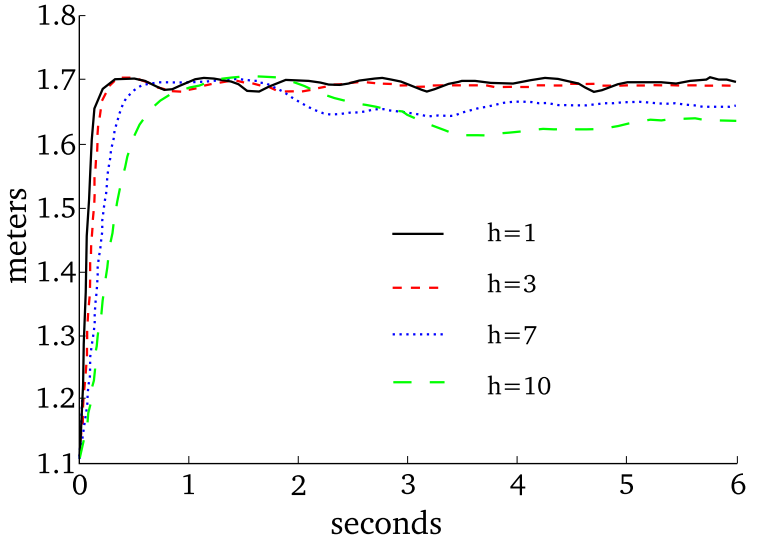
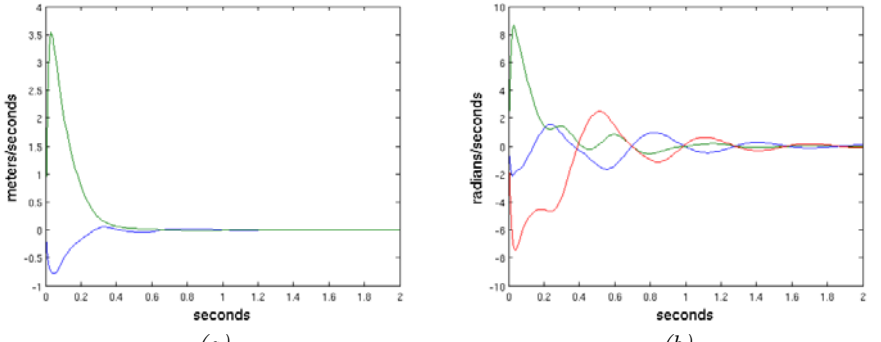


Fig. 8. Trajectory as a function of the anticipation horizon H

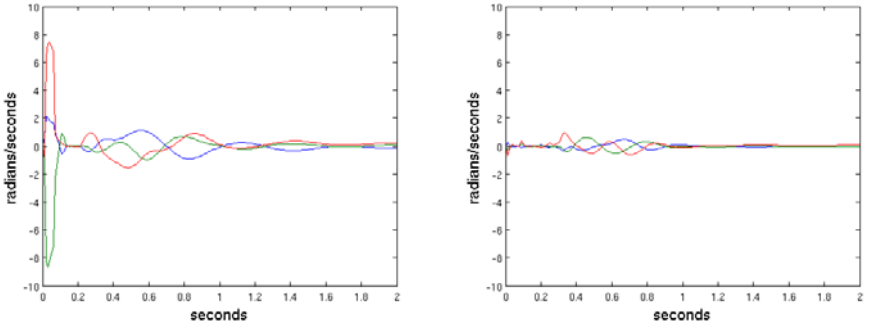


(a) *(b)*

Fig. 9. Learning the weighted pseudoinverse Jacobian matrix of the vertex of the mannequin. (a) Inputs are operational velocities ; (b) Output are joints velocities.

there for a longer time period and without oscillation. For a larger anticipation, the system is not able to maintain the desired position any more. Fig.9 shows an example of the input/output data used by LWPR to learn. After building the corresponding model, LWPR can predict the output $\hat{\dot{\boldsymbol{q}}}$ from the input $\dot{\boldsymbol{\xi}}$.

Fig.10 shows that, if we perform only one learning epoch along the trajectory, the predicted output cannot be used. We must perform 20 learning epochs where each set of input/output data is presented ten times so as to get a satisfactory model of the weighted pseudoinverse of the Jacobian matrix.



(a) (b)

Fig. 10. Error in prediction of joints velocities after one (a) and twenty (b) learning epochs where each input/output data was presented ten times

Note that, once the system is stabilised, we must stop learning with LWPR. Indeed, if the system is stable, its output is constant and the learned models degenerate. The persistent activation principle used in adaptive control [34] can be used here to stop the learning process.

5 Discussion and Perspectives

5.1 Learning the Dynamics Model

Learning a linear model of the dynamics with RLS is a limited approach that can tackle easy problems such as the control of our 3 DOFs system, but as soon as the dynamics gets more complicated, the approximation error will get too large. Our most immediate future work will consist in using LWPR to learn a piecewise linear model of the dynamics. As explained in section 2.5, one can learn a nonlinear evolution of the A and B matrices with the state $[\boldsymbol{q}_k \ \dot{\boldsymbol{q}}_k]^T$ and the control input $[\boldsymbol{u}_k]$ as input at any instant and the next state $\left[\boldsymbol{q}_{k+1}, \dot{\boldsymbol{q}}_{k+1}\right]^T$ as output.

However, it is not trivial to exploit the explicit dynamics model learned by LWPR under the form of matrices as it was done with RLS. The output given by LWPR is a weighted sum of the outputs of linear models multiplied by a gaussian. The global dynamics model matrices cannot be reconstructed easily from this collection of linear models. Nevertheless, we are working on directly learning an inverse dynamics model in order to control our system with a computed torque [35] approach. Furthermore, we need to control the internal mobility (see equation 3) of the system when it is redundant, which boils down to learning a specific weighted pseudoinverse. This is a matter for future work.

5.2 Related Models

An important related model is the work of Mitrovic [36], who combines learning a dynamics model of a 6 DOFs arm with LWPR and performing *Iterative Linear Quadratic Gaussian* (iLQG) optimal control [37] based on that learned model. To our knowledge, Mitrovic *et al.* are the only authors who succeeded in using a dynamics model learned with LWPR so far. By contrast, N'Guyen-Tuong [38] performed such a learning process on more general systems in the context of a comparison with other learning approaches, but without using the model to control a system. Note that iLQG does not use the operational versus joints space distinction, thus it does not benefit from the dimensionality reduction properties of OSC and is limited to addressing systems with no more than 10 DOFs.

Another important related model is MMRL. As in MMRL, the low level controller of our model is based on LQR. So far, our model does not benefit from the modular specialisation mechanism that MMRL shares with MOSAIC. However, such a modularity property will come into play as soon as we will use LWPR at the dynamics level, since LWPR calls upon a collection of local linear models. But here again, the main originality of our approach results from the way we use OSC to specify the task in the operational space. To our knowledge, the only other model that is endowed with the same property is presented in [39] and is based on neural networks.

5.3 State Estimation

In our model, we considered the state of the system exactly known at any moment. In robots, if the FKM of the system is known and if fast enough sensors in servo-motors can give an accurate estimation of the configuration, one can get a good estimation of the current operational state of the system. But when the FKM is learned and inaccurate, state estimation must be used. To perform such an estimation, one must combine informations coming from several exteroceptive and proprioceptive sensors whose accuracy is varying under the environmental conditions and whose delay is generally not compatible with the constraints of feedback control-based motion. Even motor control itself is noisy, as we highlighted in the introduction. Thus maintaining an accurate estimation of the joints state of the musculoskeletal system is a central issue that we must address in the future.

6 Conclusion

We have presented a model of human motor learning that combines two levels. At the higher level, we specify a task in the operational space and use the OSC approach to derive a target trajectory that will be tracked in the joints space by the lower level. The transformation from the operational space to the joints space is performed with an inverse Jacobian matrix that learned with the LWPR algorithm. At the lower level, we use LQR control to compute the optimal joints

torques that realizes the trajectory specified in the operational space, based on a simple dynamics model learned with RLS. In our approach, optimal control does not lead to a globally optimal motion but rather to a piecewise optimal solution with respect to the learned system dynamics. In that respect, we can expect our method to be more easily scalable to large dimension systems and it would be of interest to study the best compromise between the computational complexity and the chosen optimal control horizon.

We illustrated this model on a stand-up task. One advantage of the combination of OSC with learning is its flexibility in the definition of the task. Indeed, if we change the end-point effector with respect to which the goal is expressed, or if we change the actuators with which the low-level control is performed, we should just need to learn again the corresponding dynamics to obtain a different controller. Validating this property is in our future work agenda.

Furthermore, we have shown that, due to the approximation error of the linear model learned with RLS, we need to stabilise the control architecture by tuning the anticipation horizon of the low level control loop with respect to the high level one. Thus anticipation appears in our control architecture both in model learning processes and in the coupling between the high and the low level control loops.

References

1. Bernstein, N.: The Co-ordination and Regulation of Movements. Pergamo, Oxford (1967)
2. Flash, T., Hogan, N.: The Coordination of Arm Movements: An Experimentally Confirmed Mathematical Model. Journal of Neuroscience 5(7), 1688–1703 (1985)
3. Uno, Y., Kawato, M., Suzuki, R.: Formation and control of optimal trajectory in human multijoint arm movement. Biological Cybernetics 61(2), 89–101 (1989)
4. Kawato, M.: Optimization and learning in neural networks for formation and control of coordinated movement. In: Attention and performance XIV (silver jubilee volume): synergies in experimental psychology, artificial intelligence, and cognitive neuroscience, pp. 821–849. MIT Press, Cambridge (1993)
5. Nakano, E., Imamizu, H., Osu, R., Uno, Y., Gomi, H., Yoshioka, T., Kawato, M.: Quantitative examinations of internal representations for arm trajectory planning: Minimum commanded torque change model. Journal of Neurophysiology 81(5), 2140–2155 (1999)
6. Harris, C.M., Wolpert, D.M.: Signal-dependent noise determines motor planning. Nature 394, 780–784 (1998)
7. Fitts, P.M.: The information capacity of the human motor system in controlling the amplitude of movement. Journal of Experimental Psychology 47(6), 381–391 (1954)
8. Todorov, E., Jordan, M.: A minimal intervention principle for coordinated movement. In: NIPS, pp. 27–34 (2003)
9. Scholz, J.P., Schöner, G.: The uncontrolled manifold concept: identifying control variables for a functional task. Experimental Brain Research 126(3), 289–306 (1999)
10. Todorov, E., Jordan, M.I.: Optimal feedback control as a theory of motor coordination. Nature Neurosciences 5(11), 1226–1235 (2002)

11. Todorov, E.: Optimality principles in sensorimotor control. Nature Neurosciences 7(9), 907–915 (2004)
12. Guigon, E., Baraduc, P., Desmurget, M.: Computational motor control: Redudancy and invariance. Journal of Neurophysiology 97(1), 331–347 (2007)
13. Guigon, E., Baraduc, P., Desmurget, M.: Optimality, stochasticity and variability in motor behavior. Journal of Computational Neuroscience 24(1), 57–68 (2008)
14. Guigon, E., Baraduc, P., Desmurget, M.: Computational motor control: Feedback and accuracy. European Journal of Neuroscience 27(4), 1003–1016 (2008)
15. Miyamoto, H., Wolpert, D.M., Kawato, M.: Computing the optimal trajectory of arm movement: the TOPS (task optimization in the presence of signal-dependent noise) model. In: Biologically inspired robot behavior engineering, pp. 395–415. Physica-Verlag GmbH, Germany (2003)
16. Wolpert, D.M., Ghahramani, Z.: Computational principles of movement neuroscience. Nature Neuroscience 3, 1212–1217 (2000)
17. Wolpert, D.M., Kawato, M.: Multiple paired forward and inverse models for motor control. Neural Networks 11(7-8), 1317–1329 (1998)
18. Davidson, P.R., Wolpert, D.M.: Widespread access to predictive models in the motor system: a short review. Journal of Neural Engineering 2(3), S313–S319 (2005)
19. Haruno, M., Wolpert, D.M., Kawato, M.: MOSAIC model for sensorimotor learning and control. Neural Computation 13(10), 2201–2220 (2001)
20. Doya, K., Samejima, K., Katagiri, K., Kawato, M.: Multiple model-based reinforcement learning. Neural Computation 14(6), 1347–1369 (2002)
21. Shadmehr, R., Wise, S.: The Computational Neurobiology of Reaching and Pointing. MIT Press, Cambridge (2005)
22. Khatib, O.: A unified approach for motion and force control of robot manipulators: The operational space formulation. IEEE Journal of Robotics and Automation 3(1), 43–53 (1987)
23. Sentis, L., Khatib, O.: Control of free-floating humanoid robots through task prioritization. In: IEEE Conference on Robotics and Automation (ICRA), pp. 1718–1723 (April 2005)
24. Chiaverini, S.: Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators. IEEE Transactions on Robotics and Automation 13(3), 398–410 (1997)
25. Barthlemy, S., Bidaud, P.: Stability measure of postural dynamic equilibrium based on residual radius. In: RoManSy 2008: 17th CISM-IFToMM Symposium on Robot Design, Dynamics and Control (2008)
26. Vijayakumar, S., DSouza, A., Schaal, S.: LWPR: A scalable method for incremental online learning in high dimensions. Technical report. Press of University of Edinburgh, Edinburgh (2005)
27. Golub, G.H., Van Loan, C.F.: Matrix Computations. Johns Hopkins University Press, Baltimore (1996)
28. Potts, D., Sammut, C.: Incremental learning of linear model trees. Machine Learning 61(1-3), 5–48 (2005)
29. Sun, G., Scassellati, B.: A fast and efficient model of learning to reach. International Journal of Humanoid Robotics 2(4), 391–414 (2005)
30. Vijayakumar, S., Schaal, S.: Local dimensionality reduction for locally weighted learning. In: IEEE International Symposium on Computational Intelligence in Robotics and Automation, pp. 220–225 (1997)
31. Tenenhaus, M.: La régression PLS: théorie et pratique. Editions Technip (1998)

32. D'Souza, A., Vijayakumar, S., Schaal, S.: Learning inverse kinematics. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), vol. 1, pp. 298–303 (2001)
33. Wieber, P.B., Billet, F., Boissieux, L., Pissard-Gibollet, R.: The HuMAnS toolbox, a homogeneous framework for motion capture, analysis and simulation. In: Proceedings of the ninth ISB Symposium on 3D analysis of human movement, Valenciennes, France. Academic, San Diego (2006)
34. Sastry, S., Bodson, M., Bartram, J.F.: Adaptive control: Stability, convergence, and robustness. The Journal of the Acoustical Society of America 88, 588 (1990)
35. Siciliano, B., Khatib, O.: Springer Handbook of Robotics. Springer, New York (2007)
36. Mitrovic, D., Klanke, S., Vijayakumar, S.: Adaptive optimal control for redundantly actuated arms. In: Asada, M., Hallam, J.C.T., Meyer, J.-A., Tani, J. (eds.) SAB 2008. LNCS, vol. 5040, pp. 93–102. Springer, Heidelberg (2008)
37. Todorov, E., Li, W.: A generalized iterative LQG method for locally-optimal feedback control of constrained nonlinear stochastic systems. In: Proceedings of the American Control Conference, pp. 300–306 (2005)
38. Nguyen-Tuong, D., Peters, J., Seeger, M., Scholkopf, B.: Learning inverse dynamics: a comparison. Technical report, Max Planck Institute for Biological Cybernetics, Spemannstrae 38, 72076 Tubingen - Germany (2008)
39. Butz, M.V., Herbort, O., Hoffman, J.: Exploiting redundancy for flexible behavior: Unsupervised learning in a modular sensorimotor control architecture. Psychological Review 114(4), 1015–1046 (2007)

Space Perception through Visuokinesthetic Prediction

Wolfram Schenck

Computer Engineering Group, Faculty of Technology, Bielefeld University, Bielefeld, Germany
wschenck@ti.uni-bielefeld.de

Abstract. A model of visual space perception within the framework of the "perception through anticipation" approach is proposed. In this model, objects are localized by generating a simulated sequence of motor commands which would move the end effector of the agent from its current location to a location where it touches the object. Space perception arises whenever the agent knows how to move to the object. The main components of the model are a visuokinesthetic forward model for sensory prediction and a visual memory for novelty detection. Movement sequences are generated by the optimization method "differential evolution". The approach was implemented and successfully tested on a robot arm setup in the domain of block pushing on a table surface. The results indicate that visuokinesthetic prediction is superior to purely visual prediction for an iterative internal simulation of future sensory states. Furthermore, it is demonstrated that the generated movement sequences encode the location of the target object in a straightforward way.

1 Introduction

In psychology, the two main theoretical approaches to visual perception are the constructivist and the ecological one [19]. In the constructivist approach, perception is viewed as an inferential process. The sensory signals are regarded as inherently insufficient for unequivocal perception. Instead, it is assumed that the sensory information has to be processed on the basis of stored schemata and unconscious thought-like processes before perception can arise. On the contrary, the ecological approach is build around the conception of the so-called "direct information pickup" [3]. In this view, perception is an active process in which an active observer explores his environment by deliberately moving his eyes, his head, and his whole body. Perception extends over space and time, and objects are not perceived by the knowledge-driven interpretation of cues found in a single retinal image, but instead by directly detecting the *affordances* the objects offer to the observer. For example, surfaces can be "stand-on-able", "climb-on-able", or "sit-on-able". These affordances are closely related to the shape of the body of the observer and to his repertoire of motor actions. Basically, perception in the ecological approach is the direct perception of the behavioral meaning of the objects in the environment.

G. Pezzulo et al. (Eds.): ABiALS 2008, LNAI 5499, pp. 247–266, 2009.

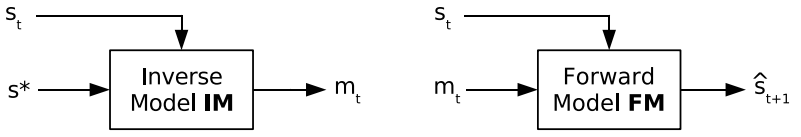


Fig. 1. Left: Inverse model (IM). It generates a motor command $\mathbf{m}_t$ to minimize the difference between the current sensory state $\mathbf{s}_t$ and the desired sensory state $\mathbf{s}^*$. Right: Forward model (FM). It predicts the future sensory state $\widehat{\mathbf{s}}_{(t+1)}$ as consequence of the current sensory state $\mathbf{s}_t$ and the motor command $\mathbf{m}_t$.

The general idea that perception is closely related to action has a long tradition in psychology and neurophysiology [2,6]. It is supported by a large body of experimental evidence showing that the brain integrates sensory and motor information at various processing levels (e.g. [21]). So-called internal models are widely used to describe these sensorimotor representations [24]. The most important classes of internal models are "inverse models" (IM) which act like motor controllers (see Fig. 1, left) and "forward models" (FM) which predict the sensory consequences of motor commands (see Fig. 1, right). Both aspects, the generation of motor commands and the prediction of their sensory consequences, are the basis for simulation theories of perception (e.g. [4]) and cognition (e.g. [6]). Their core assumption is that the brain uses inverse and forward models to iteratively predict into the future, in this way enabling perception and cognition.

Möller [15,16] suggested the "perception through anticipation" approach (PtA), which is related to the ecological view, but replaces the *direct* perception of affordances by a mental simulation process based on internal models. The main thesis of this approach is: "Perception of space and shape is based on the anticipation of the sensory consequences of actions that could be performed by the agent, starting from the current sensory situation. Perception and the generation of behavior are two aspects of one and the same (neural) process" ([16], p. 186). Starting from the current sensory situation, several motor actions are suggested (e.g., by random variation of the output of an IM). A corresponding FM predicts the sensory consequences of all suggested actions. On the basis of the predicted sensory situations, further motor actions are suggested, afterwards their consequences are predicted as well, and so on, until a maximum step size is reached or at least some simulated action sequences have led to sensory results with a clear positive or negative meaning to the agent (Fig. 2 illustrates the internal simulation process, omitting the IM and FM for clarity). In this way, a human agent can for example detect if an object is "sit-on-able", because at least one of the simulated movement sequences would result in a typical sitting posture with support for the body by the top surface of the object. Together with the affordances which have emerged from the other movement sequences simulated in parallel, this may result finally in the perception of a chair. Thus, in this approach, perception is an integrated sensorimotor process which relies on IMs, FMs, and the evaluation of sensory states.

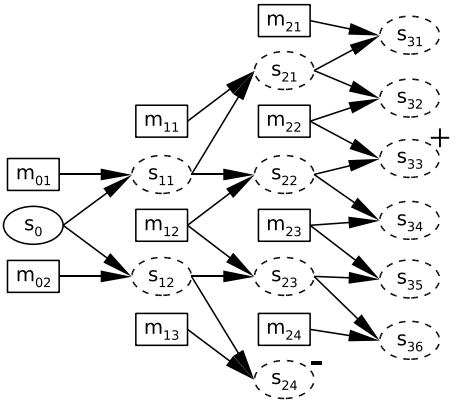


Fig. 2. Sketch of the internal simulation process in the "perception through anticipation" approach [16]. Starting from the current sensory state $\mathbf{s}_0$, different movement sequences with motor commands $\mathbf{m}_{ij}$ are simulated by internal prediction (the predicted sensory states are depicted in dashed ellipoids). The predicted sensory state $\mathbf{s}_{24}$ is evaluated as a negative outcome, thus it is not used for further simulation. When the simulation process encounters the predicted sensory state $\mathbf{s}_{33}$ with a positive rating, the simulation is halted (at a simulation depth of three steps) (adapted from [25]).

The behavior-based approach to perception is contradictory to classical artificial intelligence (AI). Classical AI assumes that visual perception relies on the construction of an explicit representation of the outer world [14]. This representation is created purely on the basis of sensory data and conceptual knowledge. In computer vision research, this approach is successful in well-controlled task domains with a restricted or well-known set of objects, but lacks the flexibility and adaptivity of human vision. Thus, it is highly questionable if classical information processing is a good starting point to model real perception. This view is meanwhile widely acknowledged [20].

Previous robot and simulation studies in the framework of the PtA approach aimed mainly on visual *shape* perception [9,7,18]. In the present study, a robot model of *space* perception in a restricted domain is proposed in which a robot arm pushes a small block on a table surface (see Fig. 4, left). The model has two main components: first, a visuokinesthetic FM which predicts the visual image of the gripper tool and the kinesthetic state of the robot arm after a small movement step, and second an abstract recurrent network which associates the visual image of the gripper tool and the visual image of the block during pushing while they touch each other. The ability to push the block around in small movement steps is the underlying motor capability of the agent which is not learned but prewired.

In the proposed model, space perception means to perceive the location of the block on the table surface by generating a movement sequence which would move the gripper of the robot arm from its current location to a location where it would touch the block (as during pushing). This movement sequence is not executed, but just internally simulated. Thus, space perception is not linked to

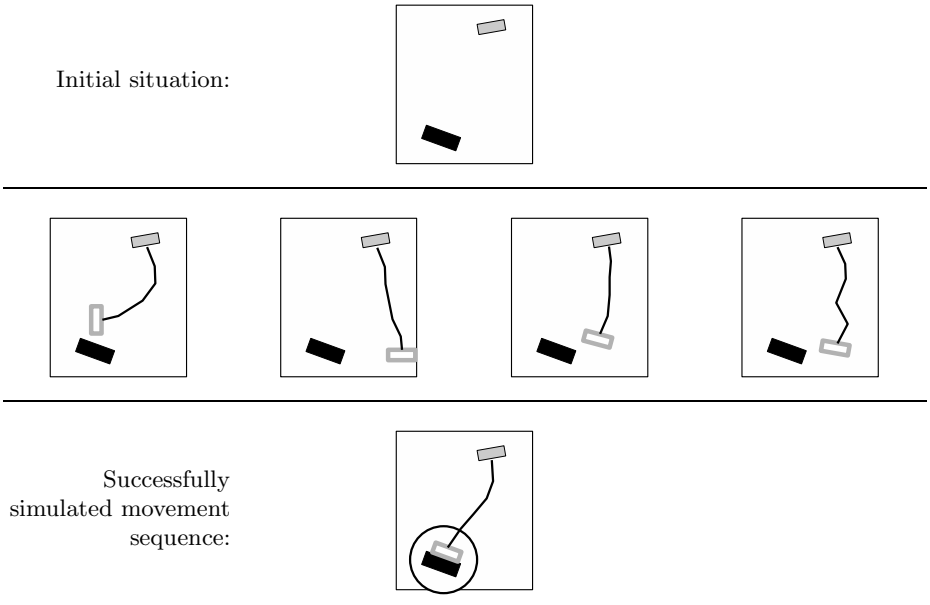


Fig. 3. The image in the first row shows the initial situation; the gripper tool (depicted in gray) and the block (depicted in black) are at different locations in the workspace. To perceive the location of the block, the agent internally simulates a multitude of movement sequences, resulting in various virtual gripper trajectories in the workspace (second row; the imagined final visual image of the gripper tool is shown as gray outline). Whenever the combination of the imagined gripper tool image and the real image of the block create a visual impression in which the gripper tool and the block are as close to each other as during pushing movements (third row), successful space perception arises.

a metric coordinate system, but arises whenever the system knows how to *move* to the goal object (here: the block).

The correct movement sequence is generated by an optimization process in which many sequences are tested in parallel. Each sequence has a final visual outcome (the visual image of the gripper tool after the last movement step as predicted by the FM). This outcome is overlayed with the current real image of the block, creating a composed visual state. Most of these states will show the gripper tool and the block at very different locations on the table surface; these visual states are irrelevant for space perception. Only if the (imagined) gripper tool and the block are as close to each other as during pushing, the corresponding movement sequence indicates the location of the block (for an illustration of this process, see Fig. 3). To distinguish between irrelevant and relevant movement sequences, the abstract recurrent network (the visual memory) is used as novelty detector (novel overlayed visual state never encountered during pushing → irrelevant, non-novel → relevant). As optimization method, "differential evolution"

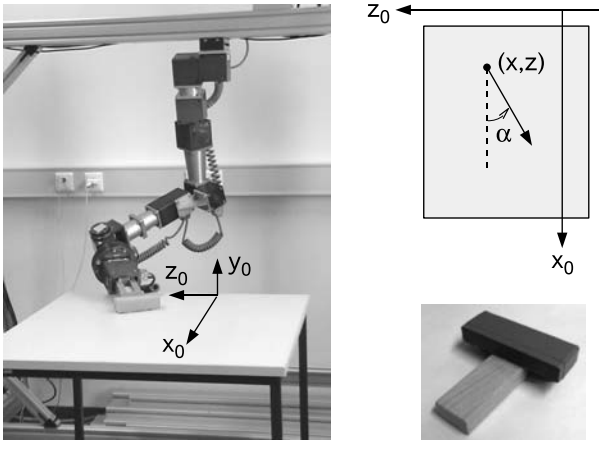


Fig. 4. Left: The robot arm in a pushing posture with the block in front of the gripper. Upper right: Base coordinate system on the table surface (see also left picture). The working area for pushing movements is shown in gray color. The kinesthetic state $\mathbf{s}_{\text{KIN}}$ is defined by the gripper position (x, z) and the pushing orientation α. Lower right: Tool held by the gripper during pushing (adapted from [26]).

(DE) [28] is applied. The optimization criterion enforces the minimization of the novelty of the overlayed visual states.

2 Setup and Method

The used robot arm setup is shown in Fig. 4 (left). The robot arm has six rotatory degrees of freedom and a gripper; it is built from PowerCube modules by Schunk. With the help of a special gripper tool (Fig. 4, lower right), the robot arm pushes a small block made of foam on the table surface. For this task, movements of the arm are restricted to a 2D plane at the white table surface. The posture of the robot arm is defined by the workspace coordinates x and z of the gripper tip and by an angle α indicating the pushing orientation. The remaining degrees of freedom are fixed, resulting in robot arm postures as shown in Fig. 4 (left). Collision-free operation is only possible for a restricted area of the table surface defined by $x \in [330\,\text{mm}; 730\,\text{mm}]$ and $z \in [-69.5\,\text{mm}; 250.5\,\text{mm}]$ ($\alpha \in [-40°; +40°]$) (Fig. 4, upper right). Visual data is collected with a camera that records the entire white table surface from above.

2.1 Sensory Processing

Three different sensory states are relevant for the overall model: the kinesthetic state of the robot arm ($\mathbf{s}_{\text{KIN}}$), the visual state related to the gripper tool ($\mathbf{s}_{\text{VG}}$),

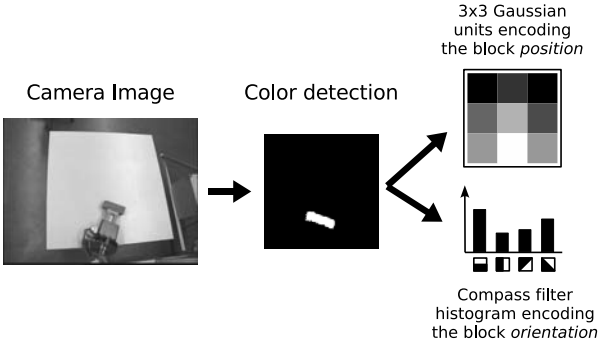


Fig. 5. Basic image processing steps, illustrated for the red block (for details see text)

and the visual state related to the block ($\mathbf{s}_{\mathrm{VB}}$). The kinesthetic state is just defined by $\mathbf{s}_{\mathrm{KIN}} = (x, z, \alpha)$. The visual states are based on the camera image; to facilitate image processing, the block and the gripper tool have opposite colors (red and green). To compute the visual block state $\mathbf{s}_{\mathrm{VB}}$, the camera image is first converted into a monochrome image in which all pixels of the block get maximum intensity and all other pixels zero intensity. From this image, a lowpass-filtered and subsampled version with only 3×3 pixels is created. The resulting 9 pixel intensity values encode the position of the block. The orientation of the block is encoded by the four values of a compass filter histogram. Four compass filters enhance the edges of the block segment in the full-size monochrome image in four different directions (0°, 45°, 90°, and 135°). After thresholding, the remaining pixels in each image are counted to give a value for the distribution of edges in a given direction [10] (see Fig. 5). Image processing for the visual gripper tool state $\mathbf{s}_{\mathrm{VG}}$ is carried out in an analogous way.

2.2 Visuokinesthetic FM

The visuokinesthetic FM receives the current kinesthetic state of the robot arm $\mathbf{s}_{\mathrm{KIN}}^{(t)}$, the current visual state related to the gripper tool $\mathbf{s}_{\mathrm{VG}}^{(t)}$, and a motor command $\mathbf{m}_t = (\Delta x_t, \Delta z_t, \Delta \alpha_t)$ for a small translational or rotational gripper movement as inputs. Although the motor command $\mathbf{m}_t$ and the kinesthetic state $\mathbf{s}_{\mathrm{KIN}}^{(t)}$ share the same coordinate system, and the computation of $\mathbf{s}_{\mathrm{KIN}}^{(t+1)}$ is straightforward with $\mathbf{s}_{\mathrm{KIN}}^{(t+1)} = \mathbf{s}_{\mathrm{KIN}}^{(t)} + \mathbf{m}_t$, it is important to note that this addition is never carried out in the model, in which the motor space and the kinesthetic space are conceptually different entities. The compatibility between $\mathbf{m}_t$ and $\mathbf{s}_{\mathrm{KIN}}^{(t)}$ was only exploited at the level of software implementation for the functions which evaluate learning success and which move the robot arm.

As output, the FM produces $\widehat{\mathbf{s}}_{\mathrm{KIN}}^{(t+1)}$ and $\widehat{\mathbf{s}}_{\mathrm{VG}}^{(t+1)}$ of the next time step (see Fig. 6, left). Learning this relationship is a function-approximation task; for this reason,

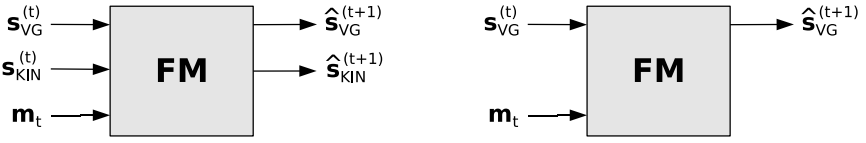


Fig. 6. Left: Visuokinesthetic FM. Right: Purely visual FM (for details see text).

the FM was implemented by a set of multi-layer perceptrons (MLP; [23]).[1] 37500 learning examples for the MLPs were generated by systematically moving the gripper of the robot arm along different trajectories through the working area, while it was pushing the red block. The movements were either translations in the current gripper direction α of a size of 10, 20, or 30 mm or rotations by a small angle $\Delta\alpha$ of 5 or 10 degrees. At the beginning and the end of each movement step, a camera image was recorded, so that a full learning example consisting of kinesthetic and visual data could be constructed. The systematic approach and the large number of learning examples ensured a uniform distribution of learning examples in the whole working area (including the gripper orientation α).[2] This process can be interpreted as a "motor babbling" stage in which the system learned the relevant sensorimotor relationships through its own experience.

The visual output $\widehat{\mathbf{s}}_{\mathrm{VG}}^{(t+1)}$ is divided into two parts: $\widehat{\mathbf{s}}_{\mathrm{VG/OR}}^{(t+1)}$ comprises the four compass filter values, and $\widehat{\mathbf{s}}_{\mathrm{VG/POS}}^{(t+1)}$ comprises the 9 pixel intensity values encoding the position of the gripper tool. For each of the outputs $\widehat{\mathbf{s}}_{\mathrm{KIN}}^{(t+1)}$, $\widehat{\mathbf{s}}_{\mathrm{VG/OR}}^{(t+1)}$, and $\widehat{\mathbf{s}}_{\mathrm{VG/POS}}^{(t+1)}$, a single MLP was trained. The first had no hidden layer, the latter two had 25 units in their hidden layers. Plain online gradient descent was used as learning algorithm (for 1000 complete iterations through the training set with exponentially decreasing learning rate from 0.004 to 0.0008). After training, the mean squared error (MSE) per pattern per output unit amounted to 0.068 for the $\widehat{\mathbf{s}}_{\mathrm{VG/OR}}^{(t+1)}$ network, to 0.027 for the $\widehat{\mathbf{s}}_{\mathrm{VG/POS}}^{(t+1)}$ network, and to 0.0 for the $\widehat{\mathbf{s}}_{\mathrm{KIN}}^{(t+1)}$ network (on normalized data).

Furthermore, a purely visual FM was trained to find out if the kinesthetic input is necessary for precise prediction. The structure of this FM is shown in Fig. 6 (right); it consisted only of two MLPs for the outputs $\widehat{\mathbf{s}}_{\mathrm{VG/OR}}^{(t+1)}$ and $\widehat{\mathbf{s}}_{\mathrm{VG/POS}}^{(t+1)}$ (otherwise, the specifications and the training were equal to the visuokinesthetic FM). As result, the MSE values amounted to 0.078 for the $\widehat{\mathbf{s}}_{\mathrm{VG/OR}}^{(t+1)}$ network and to 0.028 for the $\widehat{\mathbf{s}}_{\mathrm{VG/POS}}^{(t+1)}$ network. Thus, the performance of the visual FM seems to be only marginally worse than that of the visuokinesthetic FM.

[1] In a preceding study, this architecture proved to result in the most precise prediction performance [27].

[2] Since the data range in the different input and output dimensions varied considerably, all dimensions were normalized to a mean value of 0.0 and a variance of 1.0 before MLP training.

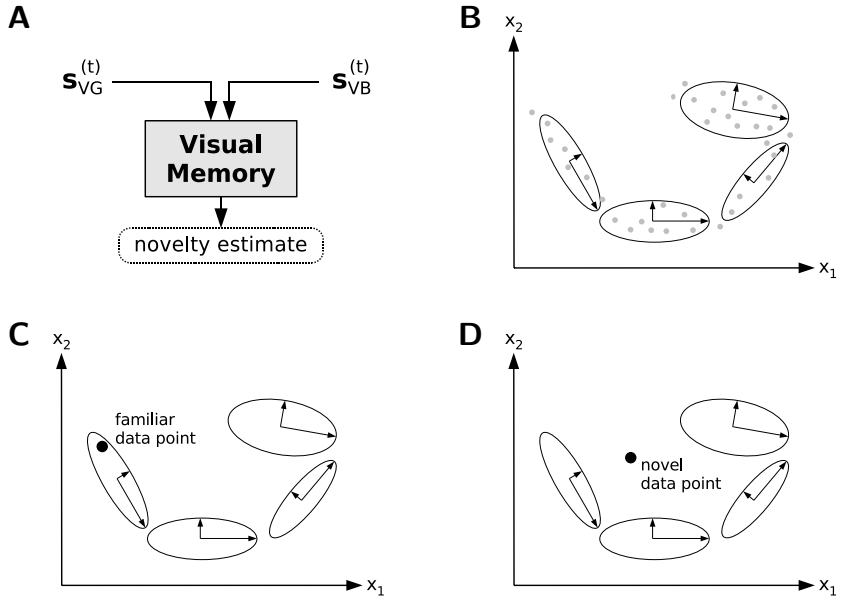


Fig. 7. A: Visual memory for novelty detection. B: NGPCA network composed from local PCA units (ellipsoids) which approximate a data manifold (gray dots). C: A new data point in close vicinity to the local PCA units is classified as familiar. D: A new data point far away from the local PCA units is classified as novel.

2.3 Visual Memory

The visual memory for novelty detection was implemented through an abstract recurrent network. These networks can be used for pattern association like dynamic recurrent networks, but they do not need to settle down to an attractor state first. In this study, a network architecture called NGPCA ("neural gas principal component analysis") was applied which is based on vector quantization and employs local PCA (principal component analysis) units instead of codebook vectors to represent data manifolds [17] (in Fig. 7b, a two-dimensional example is illustrated). We chose NGPCA because the resulting networks can easily be used for novelty detection after training (see below).

The NGPCA network for the visual memory consisted of 30 PCA units with 7 principal components each. It was adapted to 30000 training patterns which contained the vectors $\mathbf{s}_{\mathrm{VG}}^{(t)}$ (visual state related to the gripper tool) and $\mathbf{s}_{\mathrm{VB}}^{(t)}$ (visual state related to the block). This training data was also acquired during the abovementioned systematic pushing movements of the robot arm, thus the combined visual state $\mathbf{s}_{\mathrm{VIS}}^{(t)} = \left(\mathbf{s}_{\mathrm{VG}}^{(t)}, \mathbf{s}_{\mathrm{VB}}^{(t)}\right)$ represents scenes in which the gripper tool always touches the block. Since the $\mathbf{s}_{\mathrm{VIS}}$ space has overall 26 dimensions, a local dimensionality reduction was performed by using PCA units with only 7 principal components.

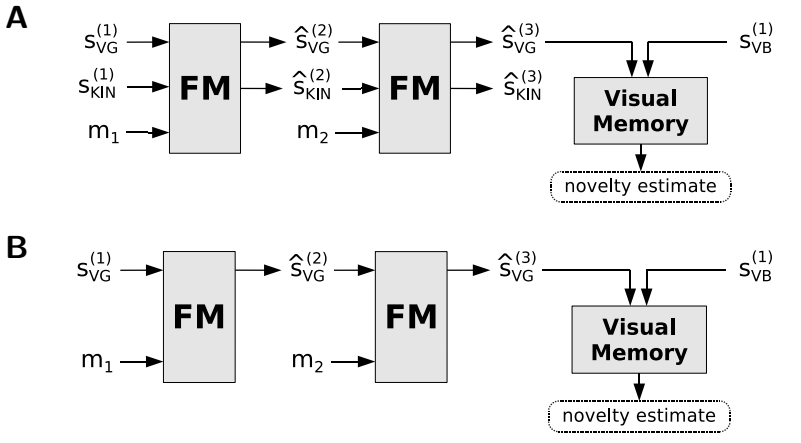


Fig. 8. Top: The iterative application of the visuokinesthetic FM, depicted exemplary as chain of two FMs ($N = 2$). The initial sensory state is used as input to the chain, the final output $\widehat{\mathbf{s}}^{(3)}_{\text{VG}}$ is combined with $\mathbf{s}^{(1)}_{\text{VB}}$ as input for the visual memory. The estimated novelty indicates how strongly $\left(\widehat{\mathbf{s}}^{(3)}_{\text{VG}}, \mathbf{s}^{(1)}_{\text{VB}}\right)$ differs from visual states showing the gripper tool as it touches the block. Bottom: Variation with a purely visual FM.

After the training of an NGPCA network, it can be used for novelty detection by computing the minimum of the distances[3] between a newly presented data point and all of the PCA units (which are interpreted as elliptical potential fields). Data points from the region(s) covered by the training set get a small minimum distance, data points outside this area get a large minimum distance (see Fig. 7c-d). In this way, novel data points can be distinguished from familiar data points. This mechanism was used for the visual memory to separate visual states $\mathbf{s}_{\text{VIS}}$ which show the red block close to the green gripper tool as during pushing (familiar data points) from visual states $\mathbf{s}_{\text{VIS}}$ which show the red block far away from the green gripper tool in the workspace (novel data points).

2.4 Iterative Prediction

The internal simulation process for motor planning and perception requires an iterative application of the visuokinesthetic FM. For $t = 1$ with known sensory input, an adequate motor command $\mathbf{m}_1$ has to be generated (without executing it). The FM predicts the sensory state $\widehat{\mathbf{s}}_2 = (\widehat{\mathbf{s}}^{(2)}_{\text{KIN}}, \widehat{\mathbf{s}}^{(2)}_{\text{VG}})$ of the next time step $t = 2$, a second motor command $\mathbf{m}_2$ is generated (without execution), the FM predicts the sensory state for $t = 3$ on the basis of the input $(\widehat{\mathbf{s}}_2, \mathbf{m}_2)$, and repeatedly so, until the number of prediction steps is equal to a predefined maximum N. Such an iterative application of an FM is illustrated in Fig. 8 for two prediction steps.

[3] As distance measure, the normalized Mahalanobis distance plus reconstruction error was computed [8].

After the last iteration, the final predicted visual state $\widehat{\mathbf{s}}_{\mathrm{VG}}^{(N+1)}$ (related to the gripper tool) is combined with the real visual block state $\mathbf{s}_{\mathrm{VB}}^{(1)}$ as input for the visual memory. This combination is interpreted here as an overlay of an imagined and an actually sensed visual impression, and the visual memory is used to determine if the agent has encountered this combined impression before. Technically, this was achieved by computing the minimum unit distance as novelty estimate as explained in the previous section (see also Fig. 8).

The PtA approach to visual perception hypothesizes that many movement sequences $\{\mathbf{m}_t\}$ are simulated in parallel. In previous studies, the motor commands were either generated by an IM with additional random variation of its motor output [18], they were determined on the basis of a movement heuristic or by recursive search [7], or they were computed by an optimization process [9]. The present study relies on the latter method because this leaves the generation of motor commands in a kind of "black box". One may even argue that the optimization process acts like an IM which produces an entire movement sequence from a given initial sensory state and a given sensory goal state.

The optimization problem is stated as follows. The initial sensory state is given by $\mathbf{s}_1 = (\mathbf{s}_{\mathrm{KIN}}^{(1)}, \mathbf{s}_{\mathrm{VG}}^{(1)})$, the number of iteration steps is set to a fixed number N. The free parameters in the optimization process are the motor parameters in the sequence $\{\mathbf{m}_t\}$ $(t = 1..N)$. The main optimization criterion is the minimization of the novelty estimate of the visual memory (thus, the minimization of the minimum unit distance). In each internal simulation step, the visuokinesthetic FM predicted first a translational movement in direction of the current estimated gripper orientation with length r_t, afterwards it predicted on the basis of the new estimated sensory state the rotation by an angle $\Delta\alpha_t$. Thus, in each iteration step t a double prediction was carried out to prevent the MLPs from operating in an untrained part of the input data space, since they were only trained with purely translational and purely rotational movements. For the same reason, Δx_t and Δz_t were not allowed to vary freely, but were instead computed as $\Delta x_t = r_t \cos(\widehat{\alpha}_t)$ and $\Delta z_t = -r_t \sin(\widehat{\alpha}_t)$ since the training data only contained purely translational movements in direction of the current gripper orientation. In summary, the free motor parameters were $(r_t, \Delta\alpha_t)$ for each movement step. Overall, this defined an optimization problem with $2N$ free parameters.

Differential evolution (DE) [28], an evolutionary optimization algorithm, was used as optimization method with a population size of $N_{\mathrm{DE}} = 30$ and a maximum number of $G_{\max} = 15$ generations.[4] Since the distance between the gripper and the block was not known beforehand, the optimization process had to be carried out with different numbers of iteration steps N. N was varied between 7 and 15. Thus, overall 4050 movement trajectories were internally simulated and evaluated for a single perception task. The movement sequence which resulted from the optimization trial with the smallest novelty estimate was picked as solution of the perceptual task.

[4] Further parameter settings according to the specification of DE in [25]: $\lambda = 0.7$, $\gamma = 0.7$, $p_{\mathrm{CR}} = 0.95$; these parameter values were carefully chosen to ensure optimum performance.

Table 1. Performance results of the different task conditions

	Type of forward model (FM)			
	visuokinesthetic		*visual*	
	Motor suppression		Motor suppression	
	strong	*weak*	*strong*	*weak*
e_{pos} [mm]	14.3 (10.2)	13.1 (10.0)	33.5 (20.4)	26.5 (16.6)
e_{α} [deg]	3.6 (4.0)	2.5 (2.2)	6.7 (5.1)	6.0 (4.8)
Steps	10.5 (2.7)	10.5 (2.6)	9.5 (2.5)	9.5 (2.4)

In the following experiments, two different factors were varied. The first factor determined the applied FM: visuokinesthetic vs. purely visual (see Fig. 8, top vs. bottom). The second factor affected the optimization criterion and was mainly introduced to explore the properties of the overall model at the technical level. In the task conditions with *strong* motor suppression, movement trajectories with movement parameters $(r_t, \Delta\alpha_t)$ outside the training range of the FMs were heavily punished during the optimization process. In contrast, in the task conditions with *weak* motor suppression, there was only a slight punishment. Overall, the two factors yielded four different task conditions (2×2).

3 Results

3.1 Performance

To test the performance of this computational approach, 100 perceptual tasks with random positions and orientations of the gripper and of the block were generated for each of the four task conditions (applying certain constraints to ensure that the required simulated movement was geometrically possible).[5] Each perceptual task was solved by the optimization process, finally yielding a sequence of motor commands. From this sequence, the (hypothetical) gripper position and

[5] The results reported in Sect. 3 were generated in a simulation study based on sensorimotor data from the real-world robot setup. For each perceptual task, the sensory states were retrieved from the pattern sets for the training of the visuokinesthetic FM and the visual memory. Two learning examples were drawn at random, one from each set. From the learning example for the FM, $\mathbf{s}_1 = (\mathbf{s}^{(1)}_{\mathrm{KIN}}, \mathbf{s}^{(1)}_{\mathrm{VG}})$ was extracted (sensory state before the small pushing movement encoded in the example), while $\mathbf{s}^{(1)}_{\mathrm{VB}}$ was determined from the learning example for the visual memory (visual block state). Certain constraints were applied to these randomly generated tasks to ensure that the required movement sequences were geometrically possible (e.g., the gripper position had to be closer to the base joint of the robot arm than the block position), that the overall orientation difference was not too large, and that gripper and block were not placed at the very border of the workspace. This procedure is largely equivalent to directly using the real-world setup for the creation of the perceptual tasks but saves experimental time and effort.

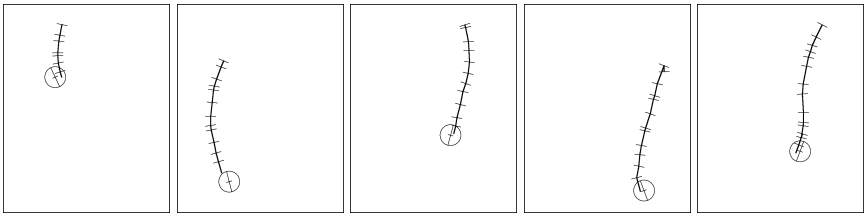


Fig. 9. Simulated trajectories for 5 different gripper and block positions/orientations (for details see text). The figures underneath each trajectory indicate the final position error (left; in mm) and the final orientation error (right; in degrees).

orientation after the last movement step were computed. Ideally, this should equal the position and orientation of the gripper after pushing the block to its actual location.

In Table 1, the results for the four different task conditions are reported. Performance measures are the Euclidean position difference e_{pos} between the final gripper position in the simulation and the ideal gripper position (x^*, z^*) in which the gripper tool would touch the block like during pushing, and the orientation difference e_{α} which is defined in an analogous way. Overall, the performance results are quite good for a workspace size on the table surface of 400 mm × 320 mm. Regarding the difference between the visuokinesthetic and the visual FM, it is obvious that the former exhibited a better performance. In summary, the final error values are only half as large for visuokinesthetic prediction compared to purely visual prediction. Thus, the slightly worse performance of the visual FM in a single prediction step (see Sect. 2.2) caused a considerable performance drop as soon as iterative prediction was required. Concerning the second experimental factor, strong motor suppression had a slightly negative impact. Thus, overall it seems to be advantageous if the FMs are sometimes allowed to operate in a region of motor space where they have to extrapolate.

Furthermore, Table 1 states the average number of simulation steps in the movement sequences. In this regard, visuokinesthetic prediction required more steps than visual prediction (10.5 vs. 9.5). An explanation might be that the less precise purely visual prediction yielded better results concerning novelty reduction the fewer prediction steps were involved. Based on these results, it was decided to focus this study on visuokinesthetic prediction and to base further analysis within this results section on the task condition with the visuokinesthetic FM and weak motor suppression.

Figure 9 illustrates five perceptual tasks, showing the best simulated trajectory from the location of the gripper to the location of the block. Each panel depicts the complete working area, the x-axis pointing in the vertical, the z-axis in the horizontal direction. The ideal final position (x^*, z^*) is indicated by a

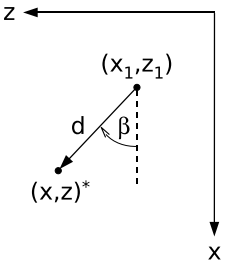


Fig. 10. Descriptors for the spatial location of the block: distance d and direction β. The gripper orientation is indicated by the dashed line. The relative orientation $\Delta\alpha^*$ of the block is not shown.

circle with a diameter of 20 mm in each panel. The longer bar of the cross at the center of the circle points into the ideal final orientation α^*. Single movement steps are separated by small ticks.

3.2 Space Perception

So far, it has been shown that the overall model consisting of the visuokinesthetic FM and the visual memory is capable of generating movement sequences which would move the gripper to a location where the gripper tool would touch the block as during pushing. This capability is interpreted as a *perceptual* one: The agent perceives the location of the block because he knows how to move to the block. This knowledge is encoded in a gripper-centric coordinate system because one and the same movement sequence encodes different locations in space depending on the initial gripper location. To perceive space in a body-centered reference frame, it would be necessary to consider the kinesthetic gripper state $\mathbf{s}_{\text{KIN}}^{(1)}$ at the beginning of the movement sequence as well. This property of the overall model is compatible with findings from neurophysiological studies which indicate that the brain encodes spatial information in different reference frames and computes the transformations between them [1]. In conclusion, body-centered space perception relies in the present model on both $\mathbf{s}_{\text{KIN}}^{(1)}$ and the motor sequence $\{\mathbf{m}_t\}$, while gripper-centered space perception only requires the latter. In the following, the ability to switch from gripper-centered space to body-centered space is taken for granted (actually, in the present model with its rather simple encoding of the kinesthetic state this transformation is trivial), and the main focus will be on gripper-centered space perception.

One may ask if it is computationally efficient to represent spatial locations through movement sequences. Of course, the answer depends heavily on the encoding of the motor commands and the motor task per se. Nevertheless, I will attempt such an analysis here. The goal of this endeavor is not to introduce a homunculus which "looks" internally at the motor representation and starts a perceptual process based on this data instead of the sensory data. Instead,

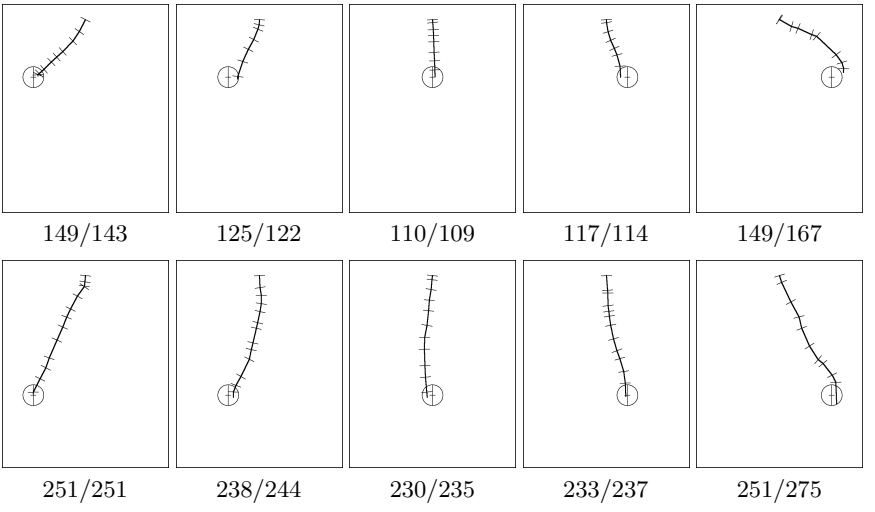


Fig. 11. Systematic tests illustrating the relationship between the distance d and the movement length R. The figures underneath each trajectory indicate d and R (in mm).

the goal is to demonstrate that information about spatial properties is easily available from the motor data without complex computations, and thus can be directly used for subsequent processing.

The following space descriptors define the position and orientation of the block in a gripper-centric coordinate system (see Fig. 10): the Euclidean distance d between the gripper position $(x^{(1)}, z^{(1)})$ and the ideal final gripper position (x^*, z^*), the direction $\beta = \tan\big((z^* - z^{(1)})/(x^* - x^{(1)})\big)$ in which the block is located relative to the gripper, and the relative orientation of the block $\Delta\alpha^* = \alpha^* - \alpha^{(1)}$. Thus, it has to be shown that d, β, and $\Delta\alpha^*$ can be derived from the parameters of the movement sequence $\{\mathbf{m}_t\}$ in a straightforward way. For this purpose, three different movement indicators were computed based on $\{\mathbf{m}_t\}$ with $\mathbf{m}_t = (\Delta x_t, \Delta z_t, \Delta\alpha_t)$:

1. the movement length $R = \sum_{t=1}^{N} \sqrt{(\Delta x_t)^2 + (\Delta z_t)^2}$,
2. the movement re-orientation $A = \sum_{t=1}^{N} \Delta\alpha_t$,
3. and the maximum slope of the trajectory $A_{max} = \sum_{t=1}^{\widehat{N}} \Delta\alpha_t$ with $\widehat{N} = \mathrm{argmax}_N \,|\sum_{t=1}^{N} \Delta\alpha_t|$.

For d and $\Delta\alpha^*$, the relation to the movement indicators is explicit. On a test set with 100 random perceptual tasks (generated as in Sect. 3.1), the following correlations were determined: $r(d, R) = 0.97$, $r(\Delta\alpha^*, A) = 0.99$ (see also Fig. 11 for an illustration of the relationship between r and D). Unfortunately, for the direction β such a simple relation does not exist. Fig. 12 shows that the maximum slope of the trajectory A_{max} varies depending on β. However, A_{max} is also

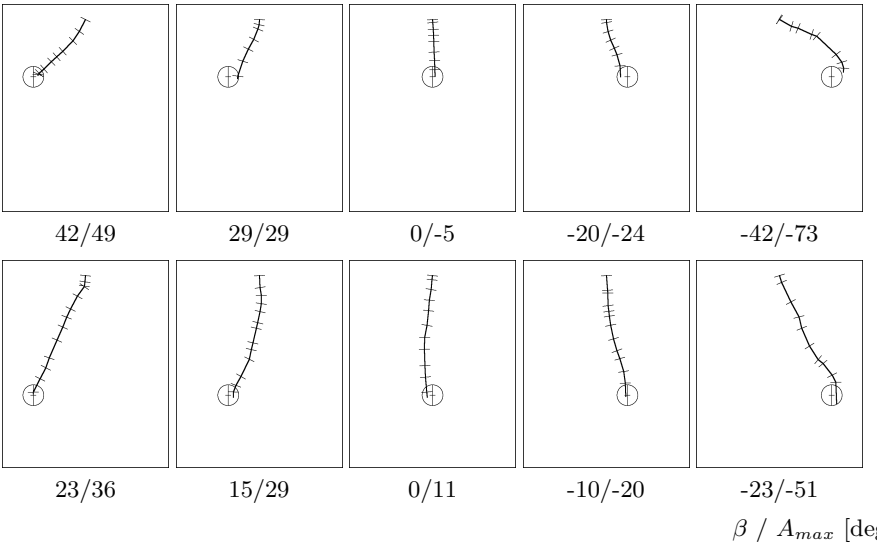


Fig. 12. Systematic tests illustrating the relationship between the direction β and the maximum slope A_{max}. The figures underneath each trajectory indicate β and A_{max} (in degrees).

directly related to $\Delta\alpha^*$ if β is kept constant as in Fig. 13. Since $\Delta\alpha^*$ is predicted very precisely by A, it is reasonable to assume that A and A_{max} form together a reliable set of predictors for β. This assumption was tested on a set of 100 perceptual tasks which were generated in a systematic way such that β and $\Delta\alpha^*$ were completely uncorrelated to avoid statistical artifacts. On this test set, the multiple correlation between the predictors A and A_{max} and the criterion β amounted to 0.73. Thus, a good approximation of β can also be derived from $\{\mathbf{m}_t\}$ by a rather simple computational model.

These results suggest that the encoding of spatial information by movement sequences is not hampered by a huge amount of computational overhead. Quite the contrary, the space descriptors are easily accessible. However, one has to admit that the present analysis is facilitated by the fact that the motor space and the spatial frame of reference are defined by the same dimensions. E.g., if the motor commands were specified as joint angle changes, the relation between movement indicators and space descriptors would be more complex.

Finally, I would like to point out that the model also allows for an interesting interpretation regarding the perceived distance. If we identify the movement length R with the percept of how far away the block is, the model would predict that the perceived distance depends on the block's orientation. Figure 14 illustrates this relationship for a fixed real distance d, a fixed value of $\beta = 0°$, and varying values of the relative block orientation $\Delta\alpha^*$. If d and β are kept constant as in Fig. 14, it becomes noticable that R varies depending on $\Delta\alpha^*$. In Fig. 14 with $\beta = 0°$, this is the case because larger absolute $\Delta\alpha^*$ values require

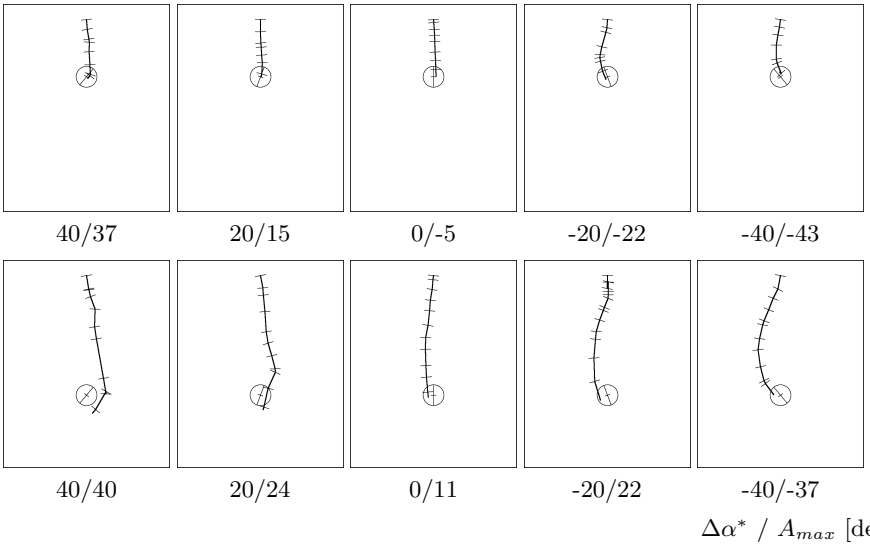


Fig. 13. Systematic tests illustrating the relationship between the relative block orientation $\Delta\alpha^*$ and the maximum slope A_{max} for a fixed $\beta = 0°$. The figures underneath each trajectory indicate $\Delta\alpha^*$ and A_{max} (in degrees).

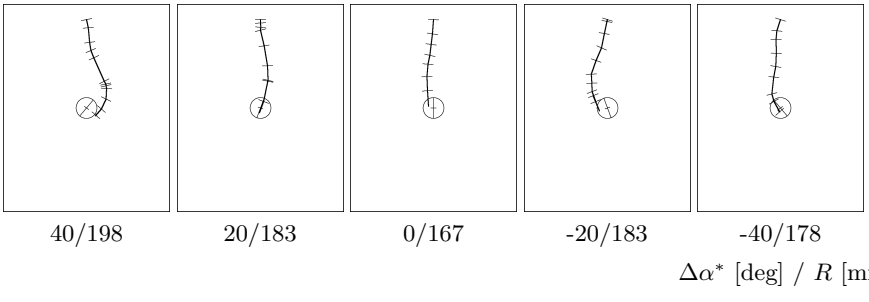


Fig. 14. Systematic tests illustrating the relationship between the relative block orientation $\Delta\alpha^*$ and the movement length R for a fixed $\beta = 0°$. The figures underneath each trajectory indicate $\Delta\alpha^*$ (in degrees) and R (in mm).

more curved trajectories resulting in larger R values. Thus, the model would predict for such a task configuration that the perceived distance depends on the object's orientation because of the varying required movement effort. Such a prediction is qualitatively in line with experimental results demonstrating that the distance perceived by human subjects depends partly on the anticipated motor effort [30,29].

4 Discussion

In the presented approach, visual space perception is linked to the localization of objects by identifying a sequence of motor commands which would move the end effector from its current location to a location where it touches the object. The experimental results show for the tested task domain that this approach is successful in generating movements sequences with sufficient precision. It has not been experimentally verified yet, but it is not expected that human subjects show a better performance in a similar perceptual task.

The proposed system architecture comprises two main components (a visuokinesthetic FM and a memory for visual states showing the end effector close to the object) and three main processes: iterative prediction by the FM, novelty detection by the visual memory, and the generation of movement sequences by an optimization process. This architecture is an instantiation of the PtA approach. It was argued that the generated movement sequences encode the location of the target object in an easily accessible way. Thus, the overall model demonstrates the feasibility of the PtA approach for visual space perception for a real-world agent in a specific task domain (see also [7,18]).

The comparison between iterative visuokinesthetic and iterative visual prediction showed that kinesthetic data can only be omitted if one is willing to sacrifice the precision of the final prediction. If one assumes that visual data is generally of higher dimensionality and complexity than kinesthetic data and thus more difficult to predict, this result might indicate that reliable long-term visual prediction is only viable in a multi-modal framework.

Extensions and Alternative Interpretations. The model can be extended to space perception of objects which are not directly reachable by incorporating movements of the whole body. Moreover, if the end effector is not visible in the beginning, the simulated movement sequence could rely on a visuokinesthetic prediction which is only driven by kinesthetic inputs until the prediction provides a valid visual state for the end effector. In this way, the proposed model of visual space perception might be extended to a more general approach.

The results on space perception based on visuokinesthetic prediction can also be interpreted in a different way. Since the agent knows the final predicted kinesthetic state, this state could be the basis for space perception in the sense of "I know how my arm would feel touching the object and thus I know where the object is". Following this line of thought, one can even propose a visuokinesthetic memory which associates images of the red block with the accompanying kinesthetic impression of the arm in the corresponding pushing posture. In this version, iterative prediction would not be necessary at all. One cannot rule out that shortcuts like this are used by the brain within the grasp space of the arm (or of the whole body if stretching movements of the other body parts are considered as well). Nevertheless, whenever an object is outside this directly reachable region, these approaches would fail. In this case, the agent has to rely on movement sequences for space perception which include lifting the body and walking towards the object. Unfortunately, the robot arm agent of the present

study lacks this capability, thus a test of such an extended model has to be postponed into the future (for now, see the study by Hoffmann et al. [9] for related work with a mobile robot). In conclusion, space perception on the basis of motor commands offers a more general account than space perception on the basis of kinesthetic information. Further support for a close link between motor commands and perception stems from studies on motor priming (the pre-activation of motor commands) by spatially corresponding stimuli [11].

Biological Plausibility. The main components of the model, a visuokinesthetic FM and a memory for visual states, are in principle biologically plausible. The same holds for the visual overlay hypothesis since studies on mental imagery show that visual mental images have clear neural correlates in the visual cortex [13]. For this reason, it is a plausible assumption that brain regions dedicated to visual processing can hold activation patterns which are partly induced by real sensory data and partly by internally generated ("imagined") data. These overlayed states could be used as retrieval patterns for the recall from visual memory. As a side effect, the visual overlay hypothesis mitigates the frame problem since only a precisely defined part of the world needs to be included in the prediction process (i.e. the gripper tool). Other aspects of the world, e.g. obstacles on the way from the initial gripper location to the hypothetical location close to the block are completely ignored.

One may critize from the cognitive and biological modeling perspective that the representation of the kinesthetic state and of the motor commands is too abstract. However, the conversion between $\mathbf{s}_{\text{KIN}}$ and the joint angles of the robot arm is purely kinematic, and the joint angles would be a plausible representation of the kinesthetic body state, even though their neural encoding in the brain may be rather complex. The visual "sensory" representations are also rather abstract. However, the compass filters work by extracting edges of a specific orientation and therefore act in analogy to the simple cells in the primary visual cortex [12].

From a general viewpoint, simulation theories of perception and cognition do not seem to be biologically plausible at first glance: For the internal simulation, the brain has to store the real sensory state and a series of hypothetical states simultaneously, and it has to keep track of which motor commands have been tested in which hypothetical sensory state. Moreover, the iterative prediction has to be very fast since a time interval of less than 30 ms of cortical activation seems to be sufficient for the recognition of visual stimuli [22]. Möller suggested a detailed neural model of the cerebral cortex which addresses these issues [15]. It is a modified version of Hebb's assembly theory [5]. The main mechanisms are first a distinction between real and hypothetical sensory states by the activation level of the assembly neurons, and second short-term synaptic plasticity to link hypothetical sensory states to motor commands. Furthermore, it is assumed that only a small number of motor sequences is tested (restriction to typical motor commands for a given situation). Nonetheless, further work on the neural underpinnings is needed to strengthen the simulation theories.

A Final Word on Perception. The term "perception" is ambiguous. On the one hand, it refers to the whole process which "transforms" a physical stimulus

into a conscious experience, on the other hand, it only refers to perception as a conscious event. It is clearly obvious that the presented robot model is not capable of conscious experience. Accordingly, this study shows in the first place that motor simulation might be an essential part of the *perceptual process*. Any further interpretation, e.g. identifying properties of the generated trajectories with the conscious experience of distance, is imposed on the system from an outside viewpoint. Nevertheless, I think interpretations like this are useful and legitimate as long as one is aware of their origin and their limitations.

Acknowledgements

I am grateful to Dennis Sinder who collected the training data for the visuokinesthetic forward model.

References

1. Battaglia-Mayer, A., Caminiti, R., Lacquaniti, F., Zago, M.: Multiple levels of representation of reaching in the parieto-frontal network. Cerebral Cortex 13(10), 1009–1022 (2003)
2. Berthoz, A.: The Brain's Sense of Movement. Harvard University Press, Cambridge (2000)
3. Gibson, J.J.: The Ecological Approach to Visual Perception. Houghton Mifflin Company, Boston (1979)
4. Grush, R.: The emulation theory of representation: Motor control, imagery, and perception. Behavioral and Brain Sciences 27(3), 377–442 (2004)
5. Hebb, D.O.: The Organization of Behaviour. Wiley, New York (1949)
6. Hesslow, G.: Conscious thought as simulation of behaviour and perception. Trends in Cognitive Sciences 6(6), 242–247 (2002)
7. Hoffmann, H.: Perception through visuomotor anticipation in a mobile robot. Neural Networks 20(1), 22–33 (2007)
8. Hoffmann, H., Möller, R.: Unsupervised learning of a kinematic arm model. In: Kaynak, O., Alpaydin, E., Oja, E., Xu, L. (eds.) ICANN 2003 and ICONIP 2003. LNCS, vol. 2714, pp. 463–470. Springer, Heidelberg (2003)
9. Hoffmann, H., Möller, R.: Action selection and mental transformation based on a chain of forward models. In: Schaal, S., Ijspeert, A., Billard, A., Vijayakumar, S., Hallam, J., Meyer, J.A. (eds.) From Animals to Animats 8, Proceedings of the Eighth International Conference on the Simulation of Adaptive Behavior, Los Angeles, CA, pp. 213–222. MIT Press, Cambridge (2004)
10. Hoffmann, H., Schenck, W., Möller, R.: Learning visuomotor transformations for gaze-control and grasping. Biological Cybernetics 93(2), 119–130 (2005)
11. Hommel, B., Müsseler, J., Aschersleben, G., Prinz, W.: The theory of event coding: A framework for perception and action planning. Behavioral and Brain Sciences 24(5), 849–937 (2001)
12. Hubel, D.H., Wiesel, T.N.: Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. Journal of Physiology 160(1), 106–154 (1962)
13. Kosslyn, S.M.: Image and Brain. MIT Press, Cambridge (1994)
14. Marr, D.: Vision: A Computational Approach. Freeman & Co, San Francisco (1982)

15. Möller, R.: Wahrnehmung durch Vorhersage — Eine Konzeption der handlungsorientierten Wahrnehmung. Ph.D thesis, Faculty of Computer Science and Automation, Ilmenau Technical University, Germany (1996)
16. Möller, R.: Perception through anticipation — a behavior-based approach to visual perception. In: Riegler, A., Peschl, M., von Stein, A. (eds.) Understanding Representation in the Cognitive Sciences, pp. 169–176. Plenum Academic/Kluwer Publishers, New York (1999)
17. Möller, R., Hoffmann, H.: An extension of neural gas to local PCA. Neurocomputing 62, 305–326 (2004)
18. Möller, R., Schenck, W.: Bootstrapping cognition from behavior — a computerized thought experiment. Cognitive Science 32(3), 504–542 (2008)
19. Norman, J.: Two visual systems and two theories of perception: An attempt to reconcile the constructivist and ecological approaches. Behavioral and Brain Sciences 25(1), 73–144 (2002)
20. Pfeifer, R., Scheier, C.: Understanding Intelligence. MIT Press, Cambridge (1999)
21. Rizzolatti, G., Fadiga, L.: Grasping objects and grasping action meanings: The dual role of monkey rostroventral premotor cortex (area F5). In: Novartis Foundation Symposium, vol. 218, pp. 81–103 (1998)
22. Rolls, E.T., Tovee, M.J.: Processing speed in the cerebral-cortex and the neurophysiology of visual masking. Proc. R. Soc. Lond. Ser. B-Biol. Sci. 257(1348), 9–15 (1994)
23. Rumelhart, D.E., Hinton, G., Williams, R.: Learning internal representations by error propagation. In: Rumelhart, D.E., McClelland, J.L. (eds.) Parallel distributed processing: Explorations in the microstructure of cognition. Foundations, vol. 1, pp. 318–362. MIT Press, Cambridge (1986)
24. Schaal, S., Schweighofer, N.: Computational motor control in humans and robots. Current Opinion in Neurobiology 15(6), 675–682 (2005)
25. Schenck, W.: Adaptive Internal Models for Motor Control and Visual Prediction. MPI Series in Biological Cybernetics. Logos Verlag, Berlin (2008)
26. Schenck, W., Sinder, D., Möller, R.: Combining neural networks and optimization techniques for visuokinesthetic prediction and motor planning. In: ESANN 2008 proceedings — European Symposium on Artificial Neural Networks, pp. 523–528. Bruges (Belgium), d-side publications (2008)
27. Sinder, D.: Roboterarm-Ansteuerung mit Hilfe von visuellen Vorwärtsmodellen, Diploma Thesis. Computer Engineering Group, Faculty of Technology, Bielefeld University (2006)
28. Storn, R., Price, K.: Differential evolution — a simple and efficient heuristic for global optimization over continuous spaces. Journal of Global Optimization 11(4), 341–359 (1997)
29. Witt, J.K., Proffitt, D.R.: Action-specific influences on distance perception: A role for motor simulation. Journal of Experimental Psychology: Human Perception and Performance 34(6), 1479–1492 (2008)
30. Witt, J.K., Proffitt, D.R., Epstein, W.: Perceiving distance: A role of effort and intent. Perception 33(5), 577–590 (2004)

Anticipatory Driving for a Robot-Car Based on Supervised Learning

Irene Markelić[1], Tomas Kulvičius[1], Minija Tamosiunaite[2], and Florentin Wörgötter[1]

[1] Bernstein Center for Computational Neuroscience, University of Göttingen, Bunsenstrasse. 10, 37073 Göttingen, Germany
{irene,tomas,worgott}@bccn-goettingen.de
http://www.bccn-goettingen.de
[2] Vytautas Magnus University, Kaunas, Lithuania
{m.tamosiunaite}@if.vdu.lt
http://www.vdu.lt

Abstract. Prediction and Planning are essential elements of successful human driving, making them equally important for autonomously driving systems. Many approaches achieve planning based on built-in world-knowledge. However, we show how a learning-based system can be extended to planning, needing little a priori knowledge. A car-like robot is trained by a human driver by constructing a database, where look ahead sensory information is stored together with action *sequences*. From that we achieve a novel form of velocity control, based only on information in image coordinates. For steering we employ a two-level approach in which database information is combined with an additional reactive controller. The result is a trajectory planning robot running at real-time, issuing steering and velocity control commands in a human manner.

Keywords: anticipatory behavior, example based learning, robot car driving, longitudinal control, lateral control, learning from experience.

1 Introduction

Automated system control is important in industry and has many applications for everyday life. For example, autonomously driving cars have the potential to increase safety and reduce costs. In driving, planning plays an important role. Look ahead information helps us decide which actions to take in response to upcoming events. We can either act immediately or prepare ahead of time for taking certain actions, thus reducing reaction time. For this reason we propose that an autonomously driving car should also be equipped with such capabilities as using look ahead and plan making, which is what we call anticipatory driving. The advantages are that it can a) react to upcoming events, b) cope with short lacks of sensory information, and c) use these plans for making predictions about its own state, which is useful for higher-level planning. For a more thorough list of the advantages of action sequence generation in general, see [1].

G. Pezzulo et al. (Eds.): ABiALS 2008, LNAI 5499, pp. 267–282, 2009.

In this paper we focus on the task of lane following, which is a basic skill in autonomous driving. Lane following is a visuomotor skill, i.e. one in which visual sensory input must be transferred into appropriate motor output. Many approaches rely on predefined control laws which require a map of the environment in Cartesian coordinates, the known state of the plant, and possibly the known states of other entities, e.g. other cars. Thus, the work consists of a) identifying the necessary control law(s), b) identifying the model for the plant as well as other desired object models, and c) transforming the (relevant) visual input into the required map. In this type of approach most knowledge, such as *what* is expected to be sensed (object models), *how* it is sensed (sensor models), and how to *act* upon it (control laws), is built int the system a priori. Examples of approaches matching this description to a great extent are [2,3,4,5]. We refer to them as "model-based". They have the property that forward simulations of the system are possible by using the state estimation process and the control law(s). This can be used for planning algorithms like path planning. Despite many advantages, the massive dependency on built-in world-knowledge presents a bottleneck. Everything that a system might need to act upon cannot be predefined.

As alternatives, machine learning based approaches to lane following can be implemented. A prominent example is ALVINN [6,7,8,9], where actions of a human driver were associated with concurrent visual input from a camera via a neural network. The inputs to the network were the pixel values of the (downscaled) camera image and the output was an appropriate steering angle. Velocity control was handled by a human. Two important points to note are: 1) the system learned to take the correct actions not by explicit control laws and state estimators, but instead based only on the provided examples, and 2) no transformation of the visual input into another representation was required, thus no conventional image processing (e.g. feature extraction, or reconstruction of 3d-information) was necessary since the visual input was directly mapped to a motor command. Further examples of machine learning based approaches are: [10] using reinforcement learning, and [11] using genetic algorithms. We refer to these approaches as "learning-based". A shortcoming of these systems is the lack of an explicit mechanism for planning, making them dependent on continuous sensor input.

Our goal is to utilize the most advantageous quality of the learning-based approaches, i.e. not having to rely on built-in knowledge, and to extend the method with an explicit planning component. Path planning in model-based approaches can be achieved by using the Cartesian map of the environment, a model of the system to be controlled, and a control law. How can one plan a path if all these items are not available? We solve this problem by equipping our system with very simple mechanisms, that are thought to play a role in human learning, too. Precisely, we give it the ability to make associations and to store and retrieve data (memory). First, a reactive controller is obtained from human control data, by associating visual information concerning the nearby street trajectory with a steering command. Second, a planner learns to associate visual information

about the entire observable street trajectory with action sequences. We show how this leads to robust lateral and longitudinal control of the robot, and how it also works in open-loop situations, i.e. when no sensory input is available. Our goal is not to compete with current state-of-the-art autonomous driving systems, which are quite advanced and also generally make use of many additional sensors besides visual ones, instead, we intend to present an alternative to many current approaches relying on task-specific knowledge.

As described our system is capable of generating speed control as well. This concept has been much less investigated than steering, at least for approaches that do not make use of environmental maps, for which a sensor model is necessary. Simpler controls also exists, such as Adaptive Cruise Control (ACC) systems, which use radar or laser to slow down the vehicle when detecting an obstacle in front, or Intelligent Speed Adapters and Limiters, ISAs and ISLs [12], which adjust, or limit, a vehicle's speed according to the given mandatory limits. Other approaches determine speed, with the help of a leading vehicle [13]. More related to this work are [14] and [15], which employ fuzzy neural networks trained on human control data to anticipate curves and regulate speed accordingly. In contrast to our approach, that work was done using simulations, and single actions per timestep were generated instead of action plans.

The structure of the paper is as follows: In the Experimental Setup section we describe the means for realizing this approach. In the Methods section we explain planner, reactive controller, and their combination, followed by their evaluation in the Results section. In the Discussion section we discuss our work and shortly compare it to predictive control based on Kalman filtering [16].

2 Experimental Setup

Experiments are carried out in an indoor environment on a four-wheeled robot (a modified VolksBot [17] of 50 cm x 60 cm size) with two motors, one for driving the wheels on each side (differential steering). The robot is equipped with a monochrome firewire camera operating at approx. 20 Hz, see. Fig. 1A. The laboratory setup simulates a street environment, where the driver can control the robot from a special station, see Fig. 1B. Here, one can see "through the robot's eyes" by means of a TV on which we display the camera output. The driver can manipulate the robot's actuators using a steering wheel and pedal set where the communication between human control output and robot sensory input is realized via a peer-to-peer architecture. A laptop placed on the robot, is connected to the camera and motors. In a cyclical fashion the robot acquires a camera image, sends it to the TV, and waits for a control input from the desktop computer connected to the steering wheel and pedal set. The control, or action input, is a steer and a velocity command, for which we use the following notation: a^{st} denotes the steer signal, and a^{v} velocity. Both signals take numerical values with $a^{st} \in [-128, 128]$ related to the steering angle and $a^{v} \in [-512, 512]$ related to the voltage sent to each motor. Throughout this text, we skip the superscripts

st or v, when referring to both action signals. They are generated by the human and sent to the robot-laptop via the desktop computer, which in turn passes them to the motors of the robot, (see Fig. 1C). Thus, every communication cycle defines a discrete timestep t where every incoming image frame I_t is related to the corresponding control a_t^v, and a_t^{st}. In Fig. 1D, we show a sketch of the track on which we trained the robot.

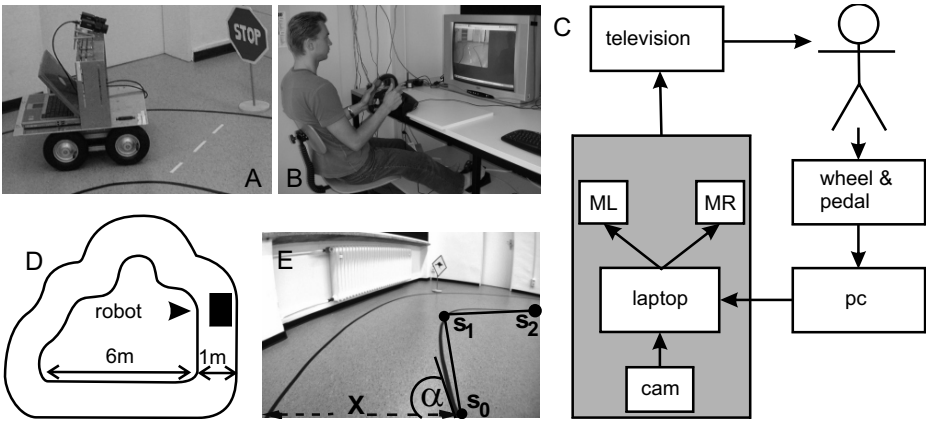


Fig. 1. A: A car-like robot. B: Control station. C: Information flow in the experimental setup: cam denotes camera, and ML and MR left and right motor, the shaded area indicates the robot. D: Sketch of the track used for training the robot. E: Short and long term visual information. x and α define short term information for the reactive controller and s_0, s_1, s_2, the corner points of the polygonized lane boundary, define the long term information for the planner.

During the supervised learning the robot associates visual information with human actions. This visual information is derived from the right street lane boundary that we detect in each image in real-time. We developed a simple and fast algorithm based on conventional edge detection (Canny [18]) which returns the detected boundary as an ordered 2d-curve.

3 Methods

Regulation of steering and speed are necessary for vehicle control. Steering control is considered to be a two-level process [19] using short-term and look ahead information, whereas we assume speed control to be based only on look ahead information. We use the word "short-term" to denote relevant visual information that is temporally and spatially close to the vehicle and "look ahead" to denote visual information that is relevant in the future, i.e. further away from the vehicle. As explained we use two modules, a reactive controller (RC) and

a planner, where the former maps short-term information to a single steering control value, and the latter is in charge of processing look ahead information and generating action plans, i.e. sequences for steering and speed control. The final steering command is a combination of planner and RC output. This setup is visualized in Fig. 2. In the following we describe both modules starting with the reactive controller.

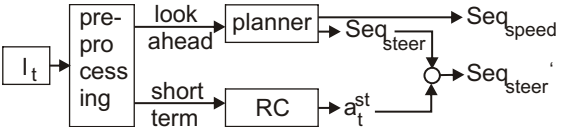


Fig. 2. The system setup. I_t denotes the image frame at time t and Seq a sequence of actions.

3.1 The Reactive Controller

The purpose of the reactive controller is two-fold. It is supposed to correct the planner if necessary, and it is used in the case that no sufficiently well suited plan is contained in the database. It is also learned from human actions and designed as follows: We define the immediate future (short-term information) of the robot-car by the tangent constructed from the beginning of the extracted street boundary and describe it by the angle α between tangent and horizontal border of the image and its starting position x on the x-axis at the bottom line of the image, see Fig. 1E. To acquire the supervisor's policy with respect to these parameters we assign human actions (from the training set) to the state space (see 3A). To fill the empty spaces, generalizing to unknown situations, we use k-nearest neighbor, shown in 3B. Of course, other approximation methods can be used instead. Note that this simple approach results in an extremely fast controller, only requiring the time necessary for looking up a steering signal in a matrix.

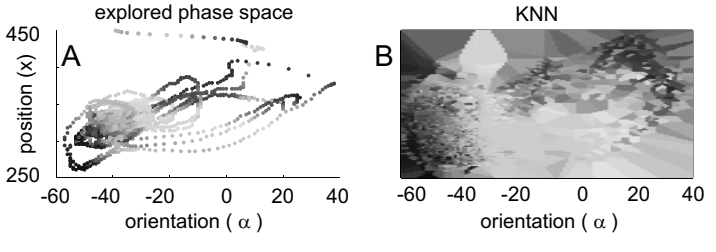


Fig. 3. A: The acquired policy from the supervisor. B: The interpolated policy using k-nearest neighbour. Different gray values denote different steering angles.

3.2 Planner

We follow the idea that a system should be able to associate experienced action *sequences* with visually perceived situations. When exposed to similar situations it should remember the previously conducted action maneuver. For example, if the system observes a right turn, it should remember that in the past it always conducted a similar sequence of actions after it had seen the right turn. Thus, right turns should be directly associated with right turns in the action space. Even if this associated action plan is not completely exact, for example the steering amplitude would not be exactly correct for taking the turn, it still provides guidance in the desired direction. We realized this idea by building a database, wherein the system stores triples containing a perceived situation description along with the corresponding sequence of steering and velocity actions. When driving in autonomous mode, the system queries the database with the currently perceived situation and receives (remembers) the assigned action plans. Based on these retrieved plans, it computes current and, if necessary, future actions. Thus, the following steps are necessary: a) database construction, b) database query at runtime, and c) control sequence calculation at retrieval time.

a) For the database construction a visual state or situation description, $\boldsymbol{s}$, is needed, comprising look ahead information. For that purpose we use a polygonized approximation of the right street lane, such that $\boldsymbol{s} = [s_0, s_1, ..s_l]$, where s_i, with $0 \leq i \leq l$, are the corner points of the polygon. The polygonization is done using the Douglas-Peucker method [20]. Note that the vertices of the vector $\boldsymbol{s}$ are ordered, i.e. s_0 is the first vertex at the bottom of the image and s_l describes the last vertex on the 2d-curve. The vector length l can vary. An example is shown in Fig. 1E and 4. It is a rough description of the observed street which contains look ahead information, but not explicitly extracted information like curvature or path length.

To each $\boldsymbol{s}_t$ corresponding control sequences are assigned. Control sequences are ordered series of actions, $Seq_{steer} = [a_t^{st}, a_{t+1}^{st}, ..a_{t+n}^{st}]$, and $Seq_{speed} = [a_t^v, a_{t+1}^v, .. \ a_{t+n}^v]$. The length n of a given sequence is supposed to resemble the number of actions that are executed while following the observed trajectory at a given timestep. That is, if only a short stretch of the street is visible we only assign a short action sequence to it and vice versa. Since we do not know exactly how many actions correspond to the observed street we use the experimentally determined value:

$$n = \lfloor \frac{1}{8} \sum_{i=1}^{l} |s_{i-1} - s_i| \rfloor. \tag{1}$$

A triple $(\boldsymbol{s}_t, Seq_{steer}, Seq_{speed})$ is stored in the database, unless a similar entry is already available, (i.e. $\epsilon \leq 10$, see below and equation 2). The database is complete if a predefined number of entries is reached, or no more triples are added by the routine. We denote the total amount of database entries as K. Thus, Seq_{steer}^k, with $1 \leq k \leq K$ is the steering sequence of the $k'th$ database entry. If we are referring to Seq_{steer} and Seq_{speed} interchangeably we skip the subscripts.

Fig. 4. Screenshot example of the planner operating mode. Left: The observed street (gray, originally red) is compared to the database entries and the best match is returned (black, originally blue). Right: The assigned steering sequence of the best match.

b) For the retrieval step, we need a metric to determine the difference ϵ between the extracted vectors, $\boldsymbol{s}$, which describe the street ahead. We use a weighted euclidean distance between vectors of same length l, normalized by l. The weighting enforces similar curves to be those that are especially similar in the beginning, which is the part of the street that is closest to the robot:

$$\epsilon = \frac{1}{l} \sum_{i=0}^{l} \boldsymbol{\omega}_i \sqrt{(\boldsymbol{s}_{q_i} - \boldsymbol{s}_{db_i})^2}, \tag{2}$$

where $\boldsymbol{s}_{q_i}$ denotes the i'th element of the queried vector and $\boldsymbol{s}_{db_i}$ the i'th element of a vector in the database, $\boldsymbol{\omega}$ is a vector containing weights where $\boldsymbol{\omega}_{i+1} < \boldsymbol{\omega}_i$ (we used 20, 10, 5, 5 for the first four $\boldsymbol{\omega}$ entries and 1 for all remaining ones). Equipped with such a database, the robot can use its current visual input for making queries. The return values are: 1) the difference ϵ to the best found match and 2) Seq_{steer} and Seq_{speed} that were assigned to it.

c) The action sequences from the database retrieval contain valuable information, not only for the current timestep t but also for $t+1$, $t+2$, ..., $t+n$. However, the database output as such only corresponds to the observed street to a certain degree. How can we drive on unknown streets? Even on the same track it is unlikely that identical images are retrieved multiple times. In other words, how can we generalize using the database output? Here, we postpone this generalization step until retrieval time which is typical for lazy-learning algorithms ([21,22,23,24]).

In principle there are two different ways in which the action sequences obtained from the database query can be used:
1) an action plan can be computed based on *single* retrieved sequences, or
2) based on *all* (or the latest N) retrieved sequences.

For the former we propose a method that we refer to as DIFF, because it is based on a difference equation, and for the latter a method that we refer to as AVG, because it is based on simple averaging. We will find that both methods yield comparable results, and because AVG is much simpler to implement we will only use this method later. Even so, we believe that chaining single action sequences together is an important concept that should be considered as well. For this reason we include a description of the DIFF method.

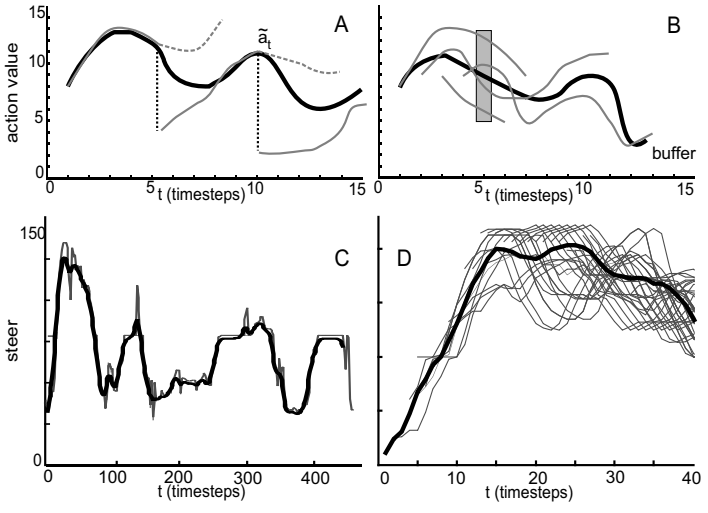


Fig. 5. A) and B) visualize schematically the data DIFF (A) and AVG (B) operate on. A) The solid gray lines denote parts of single retrieved action sequences that are used until a better match is found, which is indicated by the vertical slashed lines. As can be seen, there is no smooth transition from one retrieved action sequence to the next. The proposed difference equation smoothly joins two sequences together by taking into account future values from the previous sequence, which are depicted by the gray slashed line segments. In the equation we refer to these as $\tilde{a}$. As an example, we took $t = 10$ as the current timestep. Thus, $\tilde{a}_{t+i}$ denotes the action value from the previous sequence i timesteps ahead. The resulting action plan is drawn as the thicker black line denoting the function $a(t)$. B) The AVG method uses values from the last N retrieved action sequences, which must be held in a buffer and which are drawn as thin gray lines. The new action sequence is computed by simply taking the average from the action values in the buffer at each timestep, indicated the drawn rectangle, which denotes the vector v (see text). C) and D) show real data examples for DIFF (C) and AVG (D). Again, the thin gray lines denote retrieved sequences and the thick black line denotes the new sequence returned by each method.

As explained the DIFF method computes an action plan based on single action sequences obtained from database queries. At every timestep a database query is conducted and the returned sequence is compared to the one obtained in a previous timestep. If it is better, i.e. the affiliated error ϵ, which is returned together with the sequence by the database, is smaller than the error affiliated with the previous sequence, then the new result is kept and the old sequence is discarded and vice versa. (To acknowledge that a good match found a few timesteps ago, is less well suited at the current moment, we discount previous sequences by adding a discount factor, $\lambda = 5$, to their affiliated error.) The concept is visualized in 5A, where the gray line segments denote current action sequences. It can be seen that whenever a better match is found, there is a gap between two consecutive signals, indicated by the vertical, dashed lines in the figure. The purpose of the DIFF method is to create a smooth transition between

two such signals by taking the future values (the dashed gray line segments in the figure) of the previous signal into account. The action plan is given by a_t as shown in 3 and 4.

$$a_{t+1} = a_t + \Delta a_t \tag{3}$$

$$\Delta a_t = \sum_{i=0}^{n-1} \alpha_i \frac{\tilde{a}_{t+i\tau} - a_t}{(1 + \frac{a_t}{a_{max}})G}, \tag{4}$$

where a_t is representing a steering or velocity command, i.e. either a_t^{st} or a_t^v, and n is the length of the sequence currently being used. We denote action values from sequences from the database retrieval at the current timestep with $\tilde{a}_t$, see Fig. 5A. Thus, future values, i.e. those values in the sequence at $t+1$, $t+2$, $t+..$, $t+n$ are given by $\tilde{a}_{t+i}$. The variable τ is a constant determining the sampling frequency on the current sequence. It influences how fast to move from one signal to another. If a low value is chosen, the resulting control sequence lingers longer in the vicinity of the previous segment before reaching the values of the new segment and vice versa. From the training data we know that the human used steering and speed commands that did not exceed a certain amplitude. These upper and lower limits, which we denote with a_{max} should not be exceeded by the system either. Hence, the denominator decelerates the growth of the action plan function if the previous action was already close to these known limits. It is determined by the constant G, which we set to 10, and $\alpha_i = e^{\frac{-i^2}{\sigma^2}}$ a decay term, which discounts the influence of future values given by $\tilde{a}$. The σ is a constant, which we set to 4. In Fig. 5A the action plan a_t is drawn in black. In 5C the result of the difference equation is shown for real data.

We now turn to the second method, AVG, which also makes a query every timestep, but in contrast to DIFF keeps the returned sequence of each retrieval in a buffer. To determine an action value at a given timestep, we simply compute the average on the action values from the latest N retrievals, which are contained in a vector $\boldsymbol{v}$ as shown in the figure. The value N is usually also the number of values that the average is computed from. Thus:

$$a_t = \frac{1}{|v|} \sum_{i=0}^{|v|-1} v_i. \tag{5}$$

An example for this method is shown in Fig. 5B, and a result computed on real data in Fig. 5D.

3.3 Combination of Planner and Reactive Controller

The next step is the combination of RC and planner. The RC should correct the planner in critical, i.e. unfamiliar situations. Therefore, a measure is needed that informs about this state. We find that an appropriate measure is the error ϵ returned from the database query. If no sufficiently good match to the currently observed image is contained in the database the system performance decreases and this correlates with the value of ϵ.

We can now combine RC and planner as a function of ϵ, $f(\epsilon) \rightarrow \omega_c$ with $0 \leq \omega_c \leq 1$. The smaller ϵ the more we want to rely on the planner, the larger ϵ the more we consider RC output:

$$steer = \omega_c * RC + (1 - \omega_c) * planner, \tag{6}$$

$$\text{with } f(\epsilon) = e^a, \text{and } a = \frac{(\epsilon - \epsilon_{tolerable})}{150}, \tag{7}$$

where we set $\epsilon_{tolerable}$ to 700.

4 Results

To test the algorithms DIFF, AVG and RC, training and test data is produced from eight laps of human driving on the described track in the lab (always in the same direction). The database is constructed from five of these laps and the remaining data is used as test set. First we consider the performance of a single action generation for the current timestep, i.e. a_t. For velocity prediction with AVG we found best results when averaging over the last $N = 20$ buffer entries and for steering the last $N = 10$. The result is shown in Fig. 6A-E. It can be seen that all three methods capture the human behavior, where AVG and DIFF give smooth output and RC is comparably jerky.

We further compare the methods by plotting the root of the summed squared error between algorithmic output and human signal, $(error = \sqrt{(algorithm_{out} - human_{out})^2})$. This error and confidence interval (95%) are plotted in Fig. 7A for steer and 7B for speed. It can be seen that there is little difference between AVG and DIFF. The higher error for RC compared to AVG and DIFF can be explained by its jerkiness. It is also observable that the error for speed prediction on average is higher than for steer. This is understandable because there is more variance in the human velocity data than in the steering data. Consider for example the velocity plot in Fig. 6B or C between timesteps 300 and 500 on the x-axis. The depicted speed signal in this intervall can be considered to be constant, however, the small deviations between human and synthesized signal accrue to a relatively large error.

As this is a quantitative comparison, it is necessarily offline, and does not prove that the system behavior would also be acceptable if the controllers were used inside the closed-loop setup, i.e. when the generated action of the controller affects its future sensory input. Therefore, we let the robot run on the track in autonomous mode. We find that with all three controllers it can follow the road well, i.e. it stays on the track. The jerkiness of the RC output also results in a jerky lateral behavior. However, due to the inertia of the robot it is less strongly visible than what could be expected from the plotted signal.

Next we test wether or not the combination of planner and RC indeed improves the system performance as supposed. In case of an unfamiliar street environment that is not represented in the database, the robot should still be able to issue appropriate steering signals, albeit, without the ability to plan ahead. We trained the robot in one direction, and since our setup track is circular the robot

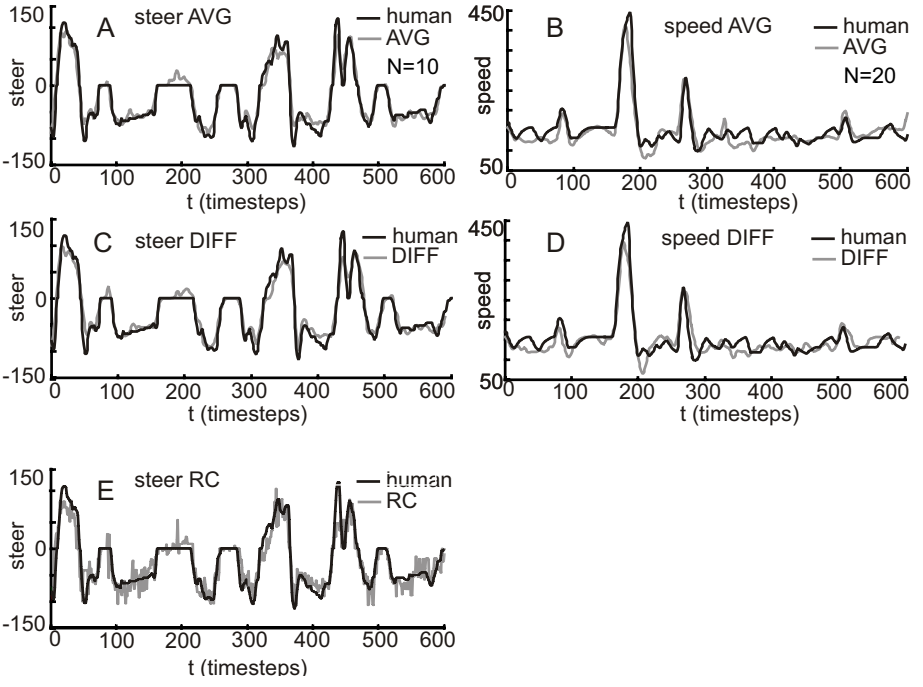


Fig. 6. A and B: Performance of AVG on generating a_t^{st} and a_t^v. "N" is the amount of entries in the buffer over which was averaged. C and D: Performance of DIFF on generating a_t^{st} and a_t^v. E: Performance of RC on generating a_t^{st}.

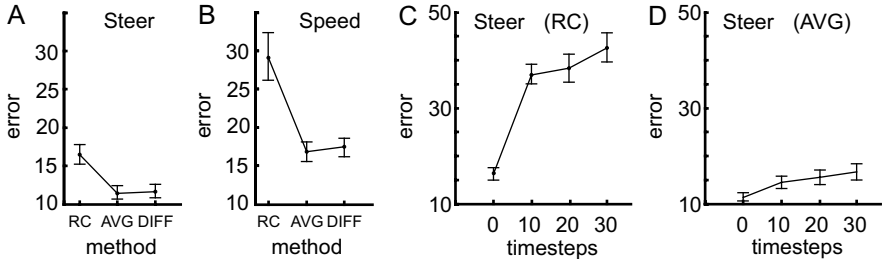


Fig. 7. A: Comparing the performance of AVG, DIFF and RC for steering generation for a_t. The plotted error is the root of the summed squared difference between the human action signal and the signal generated by each method. B: Comparing the performance of the methods for speed generation. C: The quality of steer predictions of RC for t, t+10..t+30 timesteps ahead. D: The quality of steer predictions of AVG for t, t+10..t+30 timesteps ahead. As expected, the error for AVG is much less than for RC, which indicates the capacity of AVG for action prediction.

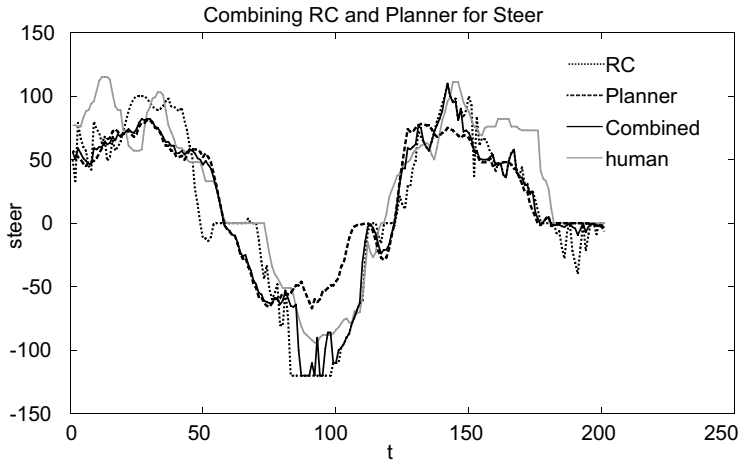


Fig. 8. Comparison of the combined signal to RC, Planner, and human output, where the robot was driven by the human. At around $t = 100$ it can be seen how the combined signal is better than the Planner output by being drawn closer to the RC signal.

is almost exclusively exposed to turns in the same direction, in this case to the right, thus, when turned around it is facing turns to the left, which are not part of its database. For a first evaluation we let the human drive the unknown track and at the same time record the suggested steering actions of RC, planner, and the combined signal. One would expect the latter to capture the human signal better than RC or planner output alone. We show an excerpt of the steering signal of this drive in Fig. 8, where at around timestep 100 on the x-axis this behavior can be well observed. The negative human steering value indicates a steep (left) curve, which is not well known by the robot. The amplitude of the signal is important as it describes how much is turned. Over- or understeering without correction leads the robot off the track. It can be seen that the suggested signals from the planner indicate less left steering, since it does not know what to do in this situation. The RC signal captures the amplitude of the human steering signal better. In this unfamiliar situation the combined output is more determined by the RC signal, therefore it also captures the human behavior better - however, it is also jerkier. In less critical situations the combined signal is smooth, since it is more determined by the planner.

As a second evaluation we let the robot drive on the unknown track using a) only the planner, b) only RC, and c) the combined signals. With the planner it drives smoothly but looses the track in difficult (high curvature) turns due to the explained reason that this situation is not represented in its database. Using RC it is able to stay on track as expected, however, the behavior is less smooth. Finally, when using the combination it drives smoothly on the known parts, which constitutes the majority of the encountered situations, and in addition it manages to stay on the track even during the described difficult turns. To evaluate the performance of the system concerning sequence prediction, which

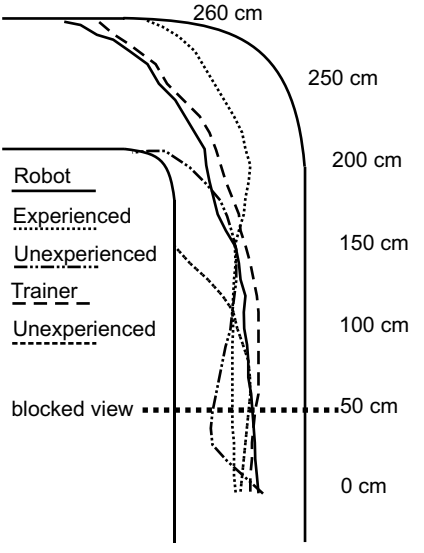


Fig. 9. Top view on part of the track. Shown is the driven trajectory of the drivers and the robot, sampled every 10 centimeters. The horizontal line denotes where the view was blocked.

is our main interest, we do not consider the DIFF method but only AVG, since DIFF is more complicated with more parameters to tune than AVG, yet fails to lead to significantly better results in generating single actions as shown above. Again we test quantitatively and qualitatively.

For the quantitative evaluation we apply AVG on the test set to generate an action a few timesteps ($t = 0, 10, 20, 30$) ahead, which we then compare to the signal elicited by the human at that timestep. We sum the difference over the entire test set and plot it in Fig. 7D. We also included RC[1] in this plot, mainly for comparison. This result is shown in Fig. 7C. It can be seen that RC's predictive capacity is very poor - as expected, and that AVG's predictive capacity is high in comparison, but the error increases with the number of timesteps to be predicted ahead. This indicates that the actions in the sequence generated by AVG are more precise in the beginning and less reliable with longer predictions, just as expected.

For qualitative testing we abruptly blocked the human controller's view during driving. This can be interpreted as a short sensor "black-out", which might occur due to technical problems. We then measure the number of timesteps the human was able to stay on the street without visual feedback. For that we only let the human control steering. The speed signal is set to a constant value uninfluenced by the driver, (during human performance, not for the robot). This is done

[1] Since the RC cannot predict sequences we had to "trick" here. To predict the action for $t = 10$, we constructed the RC by mapping $(\alpha_t, x_t) \mapsto a_{t+10}$. We proceeded analogously for $t = 20$ and $t = 30$.

because the drivers stop the robot during the experiment as soon as they cannot see the street anymore. Furthermore, we decide to block the view shortly before a curve, requiring a real change in actions. We repeat this with three more drivers: two are not trained in driving the robot, one intermediate driver, and the expert, who also generated the training data set for the robot. The result is shown in Fig. 9. It can be seen that the robot does perform the turn, which means that it successfully uses its generated plan and executes it it similarly to the trainer. It also shows that the less well trained humans lose the track quicker than the robot.

5 Discussion

We presented a robot-car that learns anticipatory driving from a human supervisor and visual sensory data. Anticipatory means that it learns to generate action sequences and to react to upcoming events, which is necessary for velocity control (e.g. speed must be decreased when approaching sharp turns). It runs at real-time and issues steering and velocity controls in a human-like way. Its planning capability allows it to cope with missing visual input.

In contrast to many current approaches to vehicle control, which are mostly model-based, very little a priori knowledge was required. Instead the system achieved its behavior by being equipped only with mechanisms for associative learning and memory.

First, a reactive controller associated short-term visual information with single actions from a human teacher. Following the idea that a system that repeatedly executes similar action sequences after observing similar images should be able to also associate these things, a planner learned to correlate observed street trajectories with subsequently performed action sequences. During performance the combined signal between reactive controller and planner was shown to lead to robust lane following behaviour. As described in the Results section, the combined signal is jerkier when relying on RC in unknown situations and smoother when using the planner in well known situations. This appears natural when considering that humans also produce smoother action sequences (in dancing for example) after training. In particular, it was not necessary to build a map of the environment from the visual sensor input to acquire action plans. Visual information could be processed directly in image coordinates. No sensor model was needed, thus it was not necessary to know the camera geometry or to undistort image frames. This makes this approach easy to implement and to use.

Concerning velocity control this work makes a novel contribution with respect to the work in autonomous driving that is not based on constructing environmental maps, namely the ability of the system to generate speed control based on the visually perceived upcoming curves.

Since this work is related to predictive control, we shortly compare it to methods usually used in this context. All of the cited model-based work in the Introduction uses state estimators for generating action control. As state estimators, a variant of the Kalman filter [16] is often used. Such a filter requires knowledge about the state-transition probability of the system, i.e. it must be known

how the system's state changes under the influence of actions or time. Based on this, and possibly also on knowledge about the sensing process, a probable future state can be predicted. Then actions can be chosen with regard to the predicted future state of the system. If this is done repeatedly (like a mental simulation), action sequences can be obtained. The main difference between this way of achieving predictions and our method is that we skip the state prediction. We generate action predictions not by inferring them from a predicted state, but by memorizing entire sequences. We see two advantages in that: 1) it is faster, simply because the step of state generation is not necessary; 2) it is less prone to error, because fixed sequences are stored and do not have to be generated step by step based on predicted states that get more and more erroneous. Of course, not being able to predict future states is a disadvantage. For example, we cannot link multiple sequences together, which would be possible if we knew the state of the system after the execution of an action sequence. However, this approach could be extended to also predict future states.

Acknowledgments. This work was supported by the European Comission grant DRIVSCO.

References

1. Sun, R., Sessions, C.: Learning plans without a priori knowledge. Adaptive Behavior 8(3-4), 225–253 (2000)
2. Dickmanns, E.D., Graefe, V.: Dynamic monocular machine vision. Machine Vision and Applications 1, 223–240 (1988)
3. Turk, M.A., Morgenthaler, D.G., Gremban, K.D., Marra, M.: Vits-a vision system for autonomous land vehicle navigation. IEEE Trans. Pattern Anal. Mach. Intell. 10(3), 342–361 (1988)
4. Thrun, S., Montemerlo, M., Dahlkamp, H., Stavens, D., Aron, A., Diebel, J., Fong, P., Gale, J., Halpenny, M., Hoffmann, G., Lau, K., Oakley, C., Palatucci, M., Pratt, V., Stang, P., Strohband, S., Dupont, C., Jendrossek, L.E., Koelen, C., Markey, C., Rummel, C., van Niekerk, J., Jensen, E., Alessandrini, P., Bradski, G., Davies, B., Ettinger, S., Kaehler, A., Nefian, A., Mahoney, P.: Stanley: The robot that won the darpa grand challenge. J. Robot. Syst. 23(9), 661–692 (2006)
5. Urmson, C., Anhalt, J., Bagnell, D., Baker, C., Bittner, R., Dolan, J., Duggins, D., Ferguson, D., Galatali, T., Geyer, C., Gittleman, M., Harbaugh, S., Hebert, M., Howard, T., Kelly, A., Kohanbash, D., Likhachev, M., Miller, N., Peterson, K., Rajkumar, R., Rybski, P., Salesky, B., Scherer, S., Woo-Seo, Y., Simmons, R., Singh, S., Snider, J., Stentz, A., Whittaker, W.R., Ziglar, J.: Tartan racing: A multi-modal approach to the darpa urban challenge. Darpa Technical Report (2007)
6. Pomerleau, D.: Alvinn: An autonomous land vehicle in a neural network. In: Advances in Neural Information Processing Systems, vol. 1, Morgan Kaufmann, San Francisco (1989)
7. Pomerleau, D.: Efficient training of artificial neural networks for autonomous navigation. Neural Computation 3(1), 88–97 (1991)
8. Pomerleau, D.: Neural network based autonomous navigation. In: NAVLAB 1990, pp. 558–614 (1990)

9. Pomerleau, D.A.: Neural network vision for robot driving. In: The Handbook of Brain Theory and Neural Networks. M. Arbib (1999)
10. Riedmiller, M., Montemerlo, M., Dahlkamp, H.: Learning to drive a real car in 20 minutes. In: Proc. Frontiers in the Convergence of Bioscience and Information Technologies, FBIT 2007, pp. 645–650 (2007)
11. Togelius, J., Lucas, S.: Evolving robust and specialized car racing skills. In: Evolutionary Computation IEEE Congress on Proc. CEC 2006, pp. 1187–1194 (2006)
12. Brookhuis, K., de Waard, D.: Limiting speed, towards an intelligent speed adapter (isa). Transportation Research Part F: Traffic Psychology and Behaviour 2, 81–90 (1999)
13. Tahirovic, A., Konjicija, S., Avdagic, Z., Meier, G., Wurmthaler, C.: Longitudinal vehicle guidance using neural networks. In: Computational Intelligence in Robotics and Automation, CIRA 2005 (2005)
14. Partouche, D., Pasquier, M., Spalanzani, A.: Intelligent speed adaptation using a self-organizing neuro-fuzzy controller. In: Proc. IEEE Intelligent Vehicles Symposium, pp. 846–851 (2007)
15. Kwasnicka, H., Dudala, M.: Neuro-fuzzy driver learning from real driving observations. In: Proceedings of the Artificial Intelligence in Control and Managamnent (2002)
16. Kalman, R.E.: A new approach to linear filtering and prediction problems. Transaction of the ASME Journal of Basic Engineering, 33–45 (1960)
17. Volksbot (2000), http://www.volksbot.de
18. Canny, J.F.: A computational approach to edge detection. IEEE Trans. Pattern Anal. Machine Intell. 8, 679–698 (1986)
19. Donges, E.: A two-level model of driver steering behaviour. Hum Factors 20, 691–707 (1978)
20. Douglas, D.H., Peucker, T.K.: Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. Cartographica: The International Journal for Geographic Information and Geovisualization 10, 112–122 (1973)
21. Aha, D.W. (ed.): Lazy Learning. Artificial Intelligence Review, vol. 11, pp. 7–10. Kluwer Academic Publishers, Dordrecht (1997)
22. Bottou, L., Vapnik, V.: Local learning algorithms. Neural Computation 4, 888–900 (1992)
23. Santamaria, J.C., Sutton, R.S., Ram, A.: Experiments with reinforcement learning in problems with continuous state and action spaces. Adaptive Behavior 6, 163–217 (1997)
24. Smart, W.D., Kaelbling, L.P.: Practical reinforcement learning in continuous spaces. In: Proceedings of the Seventeenth International Conference on Machine Learning (2000)

Prediction Time in Anticipatory Systems*

Birger Johansson and Christian Balkenius

Lund University Cognitive Science
Kungshuset, Lundagård
222 22 LUND, Sweden

Abstract. We investigated the role of the length of the future time interval in which an agent predicts what will happen. A number of simulated robot experiments were performed where four thieves try to collect pieces of gold from a house that is guarded by a single robot. The thieves try to anticipate the movement of the guard to select behaviors that will allow them to steel the gold without being seen. This scenario was investigated in four experiments with different visual fields of the guard and different strategies of the thieves. The results show that it is not always better to predict longer into the future and that best behavior would results when the agents match their predictions to the time it will take to perform their tasks.

1 Introduction

How important is it to anticipate what will happen in the future? Is it better to anticipate far into the future or to focus on the next few seconds? We have investigated this question using a number of detailed computer simulations of robots that collect pieces of gold while trying to avoid being seen by a guard that patrols the environment. A central ability of the simulated robots is to anticipate where the guard will be and select their actions accordingly. Anticipation is also important for predicting the agent's own behavior and thus bridge sensory delay. To be able to predict or anticipate we need a way to represent a future state. Rosen [1] might have been one of the first to put this idea into a useful definition: "An anticipatory system is: [...] a system containing a predictive model of itself and/or its environment, which allows it to change state at an instant in accord with the models predictions pertaining to a latter instant."

Davidsson [2] used simulations to investigate the benefits of anticipation. Two different types of experiments were conducted. The first investigated competition between agents and in the second, the agents were cooperative. In the experiments, the task of the agents was to pick up targets in a two dimensional grid-world in a particular order. By using a linearly quasi-anticipatory agent architecture, one agent could realize that it would not reach the target before the other agent and would instead start to move toward the following target.

* This work was supported in part by the EU funded project MindRACES, FP6-511931.

G. Pezzulo et al. (Eds.): ABiALS 2008, LNAI 5499, pp. 283–300, 2009.

Only one of the robots used an anticipatory behavior. In the second experiment, the agents cooperated, which lead to a decreased total time for fetching all target objects. More recently, Sharifi et al. [3] describe a system for the simulation league of RoboCup where the anticipated future state is used to decide which robot will posses the ball next, while Veloso et al. [4] anticipate the state of the whole team.

When studying anticipation, one of the most challenging tasks is to analyze the models used for prediction. The challenge lies particularly in the amount of states that the model must handle to be able to predict. Seemingly simple tasks have an enormous amount of current and future states. To understand the difficulties of predictive models, let us make a short and far from complete analyze of a simple everyday situation. Your task is to cross a busy street. During years of training, you have developed a feeling for how long it will take for you to cross the street depending on its width. You also have a feeling for how the traffic changes. Here we focus on the traffic prediction model. This model must predict how long it will take for the cars to travel to the point where you are planning to cross the street.

A simple model could use linear extrapolation of the cars current positions to where they will be when you are planning to cross the street. This prediction can then be used together with the prediction of how long it will take for you to cross the street to determine if you should start crossing the street immediately or if you should wait for a better opportunity. In this scenario, the input for the model is just the position of the car. In reality, anything that influences the prediction should be considered by the model. This can require a very complex model. It must handle vehicles with different velocities, colors, sizes, weather conditions and other things that may influence the prediction. Although the predicted position will be independent of the color of the car in most cases, a red car may be a red fire truck with its sirens on. In this case, the sound will influence the prediction and we may assume the fire truck has a higher velocity than an ordinary red car. Other assumptions that can be made are that other drivers will react differently to the fire truck and that the path for the fire truck may be more important than your crossing the road.

Another problem that is always present when dealing with the real world is sensory latencies and processing delays. When running the robot system used in this paper, the camera and tracking parts are slow and computationally costly. To grab an image and visually track all obstacles and robots takes approximately 200 ms. These processes are allowed to use up to 250 ms. The tracking is run on a dedicated tracker node and the communication to the rest of the nodes takes an additional 250 ms. This leaves us with a total delay of half a second before the sensory information reaches the system.

These types of system latencies are often minimized by faster hardware with more processing power. Another way is to compensate for the latency using predictive models and this is the method used here. A similar approach was used by Benke et al. who used predictive modelling to minimize control latency in their RoboCup team [5].

In previous work, we investigated the importance of anticipation in navigation tasks [6]. The results of this work show that a multi-robot system will benefit from anticipation compared to a system without anticipation. However, the models used to anticipate must have high precision, otherwise a reactive or a pure planning strategy will perform equally well as one with anticipation. The benefits of anticipation also depend on the task. A complex task will increase the usefulness of anticipatory mechanisms [6]. In a simple task, a reactive or planning strategy that did not take the future into account would perform better than an anticipatory strategy.

In this paper we study one additional variable that we call the prediction time. The prediction time sets the boundary for the future time interval for which a prediction will be made. How does the prediction time influence the success of a task? We have investigated this question within a multi-agent scenario where we compared the performance of agents with different predictive abilities.

2 A Task with Guards and Thieves

In the guards and thieves scenario that we use to study predictive models, we use 5 agents in a dedicated robot arena. The task for the thief agents is to collect gold from different locations. In our setup we use one guard and four thieves. The guard protects a building where the gold is stored by patrolling the area around the building along a fixed route. The thieves hide in the home zone and, when there is an opportunity, they sneak out and try to collect gold. If a thief is seen by the guard, the thief seeks shelter in the home zone. In this scenario, we are not interested in the guards behavior. We only measure the anticipatory behaviors of the thieves.

The guard and thief scenario shares some features with the street crossing scenario described above. The thief must predict where the guard will be and use this to decide if it should try to fetch the gold or not. With the guard and thief scenario, we manage to eliminate a large amount of possible states. The model used in our experiments is very simplified, but handles simple contexts, occlusion, velocity and the visual field of agents. The model is minimalistic which gives us a chance to analyze it, but is still complex enough to study how the prediction time inuences the task.

The environment (Fig. 1) is similar to that used in an earlier experiment where the robots had to switch places with each other in environments of different complexity [6]. In this setup, the guard has intentionally been made less gifted than the thieves. The behavior of the guard is to follow an already defined route around the buildings. It would be possible to give the guard more complex and more realistic behavior like letting it anticipate the thieves and having it patrol in an autonomous way. Behaviors like that would give the guard and thieves scenario more dynamics and more interesting behaviors, but at the expense of results that would be much harder to interpret.

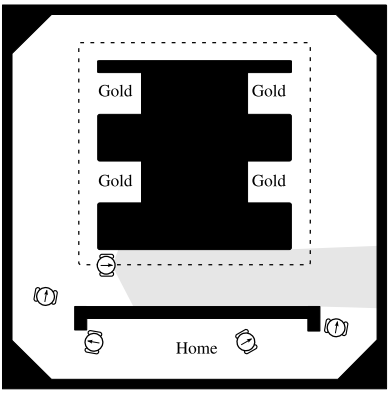


Fig. 1. The robot area with robots and obstacles. Black indicates obstacles. Each robot is trying to collect gold from one of the four openings and to bring it back home without being seen by the patrolling guard which is draw on top of its dashed path. If the guard sees the thief during an attempt to collect gold, the thief looses its gold.

3 AARC Architecture

All the simulations made in this paper used the AARC architecture [7]. The AARC architecture consists of a large number of modules connected into a complex system using the Ikaros framework [8]. The AARC architecture needed to control five robots consists of more then 300 modules and 1000 connections, where each module performs a task such as visual tracking, Kalman filtering, planning or motor control. The relatively large complexity of the system has to do with the internal compensation for many different types of processing delays. The architecture can be used both as a pure simulation and to control real robots. Here we use a simulated e-puck. The software system consists of four interacting layers (Fig. 2). The bottom layer is the host operating system (OS) which executes one or several Ikaros processes. The simulations reported here run on an eight node Linux cluster. Ikaros provides software support for real-time execution and message passing, as well as tools for the design of complex networks of interacting computational modules. The third layer is the AARC architecture which is implemented as a set of interacting Ikaros modules. Finally, the top layer consists of task specific control which sets the overall goals and tasks for the system.

3.1 Anticipatory Learning

To be able to predict, we need a way to represent a future state. The AARC architecture uses linear associators in combination with traditional planning algorithms to build models. The thieves model the behavior of the guard. The route of the guard is learned using a linear associator which learns the association between the current position and a future position.

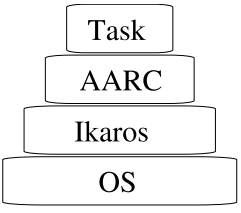


Fig. 2. The four main levels of the implemented system

The learning can be done both online or offline. With online learning, the model will be rather imprecise during learning which will inuence the result of the experiment. Instead, in these experiments, the learning was made off line. No learning is done during experiments to avoid the duration of the experiment influencing the predictive model. The online learning used data from 10 laps to build the model of the guards route.

The linear predictors learns a function from a number of observed positions $p(t-n), \ldots p(t-1)$ to the estimated position $p^*(t)$ at time t. Any of a number of learning algorithms could learn such a function by minimizing the prediction error $e(t) = p(t) - p^*(t)$. The learned function constitutes an anticipatory model of the target motion.

We now add the constraint that the perception of the target, including its localization, takes τ time units. In this case the problem translates to estimating $p^*(t)$ from $p(t-n), \ldots p(t-\tau)$, since the rest of the sequence is not yet available. In addition, this means that the system only has access to the prediction error e(t) after τ additional time steps, that is, learning has to be set off until the error can be calculated and the estimate of $p^*(t)$ has to be remembered until time $t+\tau$ when the actual target location $p(t)$ becomes available. The important point here is that a system of this kind will never have access to the current position of the target until after a delay. The central problem for the predictor is thus to learn the mapping

$$p^*(t) = f(p(t-n), \ldots, p(t-\tau)|c),$$

where c is a set of parameters. With an appropriate model f, a system will be able to anticipate the target location p^* and direct its attention or actions toward it. Any of a number of learning mechanisms can be used to learn f. We have found that in many cases, such as in tracking a regularly moving object, a linear association trained with a gradient descent method is sufficient although other methods may give faster convergence and better noise sensitivity. Here, a number of parallel predictors are used to estimate a number of future positions of the thief $p_i^*(t)$ using different τ_i.

An example of the guards predicted route is presented in Fig. 3. In simulation, it is generally possible to chose to make a perfect model, but in real robot experiments even the simplest environment results in noisy predictions. The noise in the predicted path in Fig. 3 depends on the inaccuracies of the control

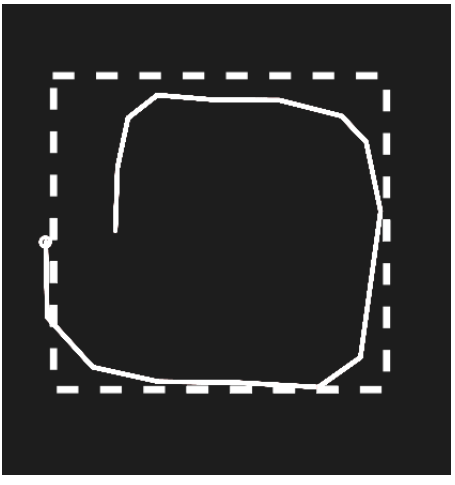


Fig. 3. An example of the prediction of the future positions of the guard. The circle is the current position and the white line in the position for the next 85 seconds. The white dashed line is the path that the guard attempts to follow.

system of the gaurds as well as on sensory noise in the thieves and limitations of the prediction method used.

Although we have chosen a linear associator in this setup there exist many other possible algorithms for predictive models. Some examples are neural networks [9], Kalman filters [10], anticipatory classifier systems [11] or Bayesian approaches [12]. We chose linear associators because they are simple and fit well for this type of task.

3.2 Thief Control

Using the learned route of the guard, the thieves can predict the position of the guard t time steps ahead based on its current observation of the guard. The output from the prediction system is used to mark regions that are or will be visible to the guard and the thieves try to avoid these areas (Fig. 4). When the guard is at location p, the polygon representing its visual field is denoted by $V(p)$ and is calculated using a ray-casting algorithm in the model of the environment. The visible part of the environment is formed as the union of all the visibility polygons generated by each location where the guard is predicted to pass in the next t time steps:

$$\bigcup_{0 \leq i \leq t} V(p_i^*).$$

3.3 Delay and Timing

In the system, the internal model of the world is running in phase with the real world, even through it does not yet have access to the sensory information for

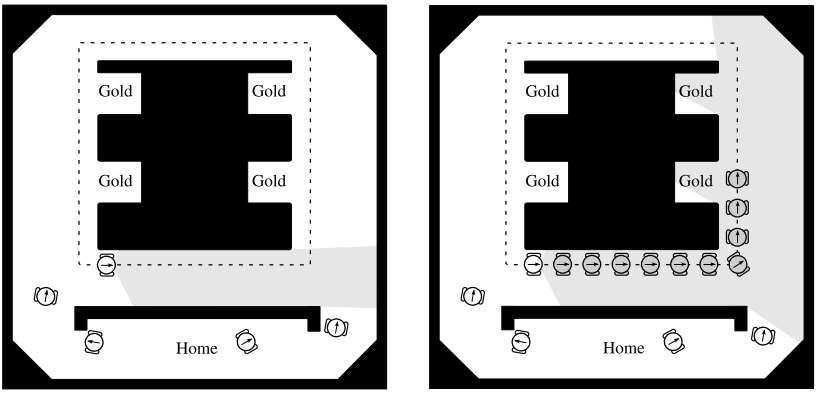


Fig. 4. Left: The robot simulator provides an input to the learning system, which is used to predict the future positions of the guard. This prediction is subsequently used to determine a local goal for the agent. Right: The agent marks areas that are visible to the guard and tries to avoid these.

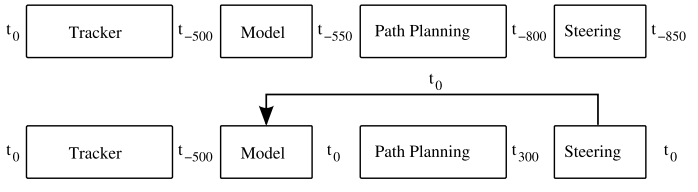


Fig. 5. Top: The total processing time for the whole system is 850 ms. The Tracker takes 500 ms, the Anticipatory Model 50 ms, Path Planner 250 ms and Steering and Robot control 50 ms. Bottom: The different parts operate in different time frames. The real environment, model and the steering work in synchrony in the same time frame despite different delays.

this state. To manage this, the system predicts the state of the environment 500 ms into the future. A latency of 500 ms is fairly large. With a speed of 13 cm/s, the robot can move 6.5 cm before it will influence the prediction model. The system thus depends on the predicted state of the world for control.

Not only is there a delay of the sensory input, there is also a delay for the outgoing motor commands. To overcome this, the path planning uses the positions predicted by the anticipatory model to calculate a possible path to the goal. This calculation takes approximately 250 ms and the path is subsequently forwarded to the steering module and finally to the robot module, which takes an additional 50 ms. To compensate for these delays, the path planning system uses the current position, but compensates for its own execution time when calculating the future path. Overall, the system compensates for a total delay of over 850 ms from sensors to effectors using calculations in different time frames for different

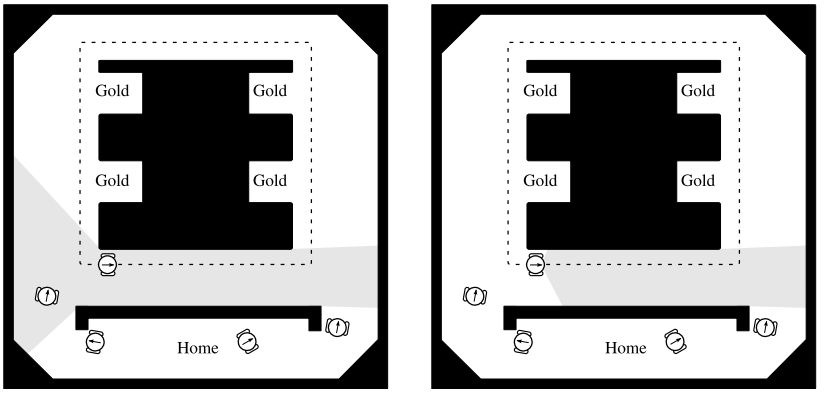


Fig. 6. Left: Guard agent with 360 degree field of view. Right: Guard with a visual field of 135 degrees.

parts of the system (Fig. 5). The anticipatory model and the steering work in the same time frame although they base their processing on differently delayed information. When the tracker has processed an image, the output is already old. As a consequence, the path planning compensates for its own computation delay as well as the reaction time of the steering. The anticipatory model is set initially to a future state and by providing it with motor commands from the same timeframe, it will stay in phase with the actual robot in the environment.

4 Simulations

In the simulations, the thieves try to steal gold from the guarded house. One lap for the guard's route takes approximately 85 seconds (t_g). During its route, the guard cannot simultaneously defend all the pieces of gold and this is used by the thieves to collect gold without being seen. In the first experiment, the visual field of the guard can, at most cover, two of the gold areas and will cover the same location within a minimum of a half lap (Fig. 6). If the guard has a 360 degree field of view, the time where the thief agents can steal gold is:

$$t_{ns} = \frac{t_g}{2}$$

Where t_{ns} is the time when the guard cannot see the thief.

Each of the robots will collect gold at different places (Fig. 7) and, depending on the distance to the gold and the direction of the guard's route, each agent faces a different degree of difficulty. The time taken for the agent to fetch a piece of gold, without interference from the guard, is called its action time (t_a). Thief A, B, C, D have action times of 30, 45, 45, 30 seconds.

Four experiments have been conducted in this paper. In the first experiment, the guard's visual field is 360 degrees and in the second, 135 degrees. In the 360

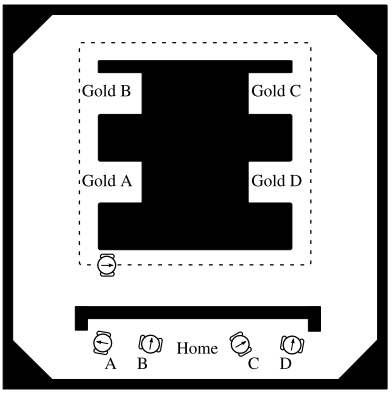


Fig. 7. Each of the thieves fetches gold at different places. Each with its own level of difficulty.

visual field experiment, the agent can cover up to half the environment, with the 135 degrees visual field, only a maximum of one fourth of the environment can be covered. With the 135 degrees visual field, the thieves can run behind the guard's back. This gives the thieves more time to collect the gold:

$$t_{ns} = \frac{t_g}{4}$$

How far ahead each thief predicts is called its prediction time t_p. Each of the experiments, except for the last one, is running repeatedly with different prediction times for the thieves. The prediction time used in the experiment ranges from no prediction at all to prediction of a whole lap for the guard (0 - 85 seconds). With a small prediction time, the thief will only use the next few seconds to calculate a path to the gold. If the thief is seen by the guard, it will return home without any gold. If the thief predicts that it will be seen it tries to find a location that is hidden from the guard's visual field. When predicting that the guard will see it, the hiding place will be in the direction of its home zone and not towards the gold. During the experiment, each thief recalculates the prediction of the guard continuously. With a small prediction time, the robot can start to run towards the gold and halfway realize that it will soon be visible for the guard and be forced to turn back.

Each of the agents is given 100 trials to collect gold pieces for each prediction time and for all experiments the amount of gold and the elapsed time is stored. Initially, the thieves are located in the safe zone and are not seen by the guard who is located in the upper left corner of the environment.

Each trial starts with a waiting period for the thief. The thief waits a random period of time from 0 to the time for the guard to finish one lap (85 seconds). The waiting period gives each trial a random position of the guard which neutralizes the risk of a thief getting stuck in a phase where the initial state is the same

for each trial. After the waiting period, the thief tries to collect gold. When the guard is at location p, the extent of its visual field is denoted by $V(p)$ and is calculated using a ray-casting algorithm in the model of the environment. If the agent does not use any predicting the following criteria must be fulfilled for the thief to head for a piece of gold:

$$t_p = 0 \ \wedge \ p_T \notin V(p_G)$$

If the thief does not use any prediction, it will try to fetch gold if it is not currently visible to the guard. When the agent starts to use prediction, the condition are extended to the following:

$$t_p \neq 0 \ \wedge \ p_T \notin V(p_G) \ \wedge \ p_T(t) \notin \bigcup_i V(p_i^*)$$

With prediction the agent is allowed to try to fetch a piece of gold if it is not currently visible or predicted to be visible.

If the conditions described above are not satisfied, the robot will seek shelter instead of going for the gold. This will lead to a waiting behavior if a trial has started but the guard is currently at a location where it will see the thief. The thief will wait in a safe position near its home zone until the path is clear from the guard and then continue its mission to fetch a piece of gold.

If the thief agent has managed to leave the home zone before it realizes that it soon will be visible to the guard, it will seek shelter. The agent always tries to find a hiding place towards its home and this can lead to the guard chasing the thief all the way back to the home zone. If the thief is seen during this retreat, the trial is not valid. Otherwise, it makes another try to fetch the gold.

For the thief agent to collect a valid piece of gold, one of the following conditions must be fulfilled: With no prediction at all, the agent can mange to collect a piece of gold if it is not seen by the guard during an attempt to fetch the gold

$$\forall t: \ 0 \leq t \leq \ t_a \ \Rightarrow t_p = 0 \ \wedge \ p_T(t) \notin V(p_G(t)).$$

If the agent uses prediction, a piece of gold is collected if its not seen by the guard or predicts to be seen by a guard in the second half of its mission:

$$\forall t: \ 0 \leq t \leq t_a \ \Rightarrow t_p \neq 0 \ \wedge \ p_T(t) \notin V(p_G(t)) \ \wedge \ \left[p_T(t) \notin \bigcup_i V(p_i^*) \vee G_T \right],$$

were G is true when the thief is carrying gold.

To summarize, four experiments have been conducted in this paper. In Experiment 1, the guards visual field was 365 degrees, in Experiment 2, the guards visual field was 135 degrees, in Experiment 3, the thief agents know how long time it takes to fetch a piece of gold and finally in Experiment 4, the thief agents use the provided action time to calculate the prediction time needed to fetch a piece of gold.

5 Results

In all experiments except the last, the agent tries to collect gold 100 times. An overall result for all the simulations is that longer prediction times cause the total time for the experiment to increase. The maximum experimental time that the thieves needed to fulfil their task was 3 hours and 30 minutes. The fastest experiments were without any prediction. In these cases the total time for the experiment was less then 2 hours. The experiments simulated a t_p from 0 to 85 seconds with a resolution of 1 second, which gives a total of 256 simulations for all the experiments.

5.1 Experiment 1

In the first experiment, the guard had a 360 degrees visual field. With a large visual field, the guard will cover more of its surroundings and this will give the thieves less places to hide.

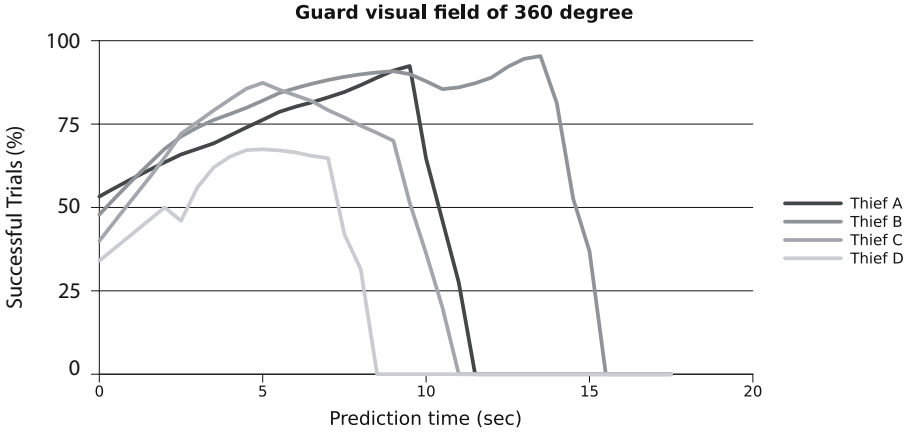


Fig. 8. The number of number successful trails for the thieves for each prediction time when the guard has a 360 degrees field of view

The results show that thief agents A and B, who collect gold from the left side of the environment, reach a high rate of success (Fig. 8). Thief A collected a maximum of 98 pieces of gold in 100 trials and both thief A and thief B were collecting pieces of gold in 80% of the trials with only a 5 second prediction time.

Thief A is able to fetch gold until the prediction time reaches 12 seconds and thief B until 15 seconds. With a longer prediction time, the agents are not able to collect any more gold.

Thief C and D collected less gold compared to thieves A and B. At maximum, thief C manages to collect 90 pieces of gold and thief D collect only 68 pieces of gold. Neither thief C nor D collected as much gold as thief A or B. With a 5 second prediction time, thief C collects gold on 80% of its trials and thief D on

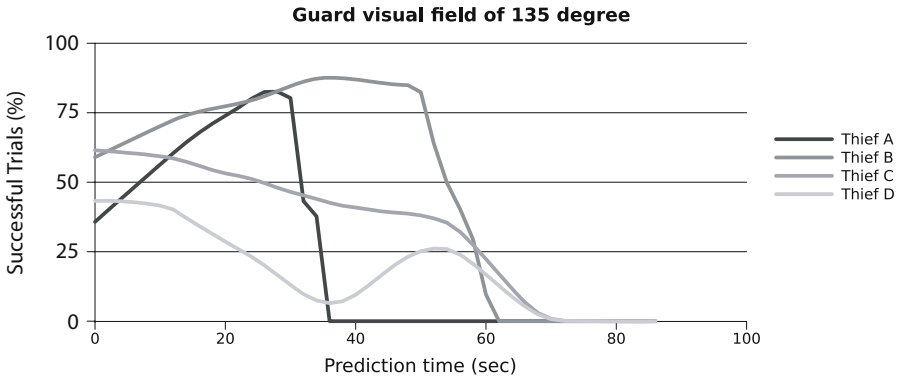


Fig. 9. The graph shows the mean number successful trails for the thieves when the guard has a 360 degrees vision field

65%. With prediction time longer than 5 seconds, the success rate stabilizes and drops to 0 after 8 seconds for thief C and after 12 seconds for thief D.

5.2 Experiment 2

In this experiment, the guard's visual field was 135 degrees (Fig. 9). With a narrower visual field, the guard cannot cover an as large area, which gives the thieves more time to fetch gold. In this experiment thief A and B reached a success rate of 90%.

With a prediction time over 25 seconds, the gold rate stabilizes at a high success rate. Thief A fetches gold until its prediction time reaches 35 seconds and thief B until the prediction time is over 58 seconds. Thief C starts at the same level as thief A and thief D as thief B. Both thief C and thief D decrease their performance with longer prediction time. With a prediction time of over 58 seconds, none of the thieves manage to collect any gold. The success rate for thief D decreases temporarily down to almost zero at a prediction time of 35 seconds and then raises to a success rate of 40%.

5.3 Experiment 3

In this experiment, the thief agents had a 135 degrees visual field and know their action times. With this information, an agent no longer tries to fetch a piece of gold if it cannot predict that it will succeed. The result shows that agents only try to fetch gold with a prediction time longer than the agents' action time (Fig. 10). At this point, the thief will have an optimal success rate until the prediction time reaches approximately 60 seconds. The result is identical to the result form the previous experiment when the prediction time is longer than the action time.

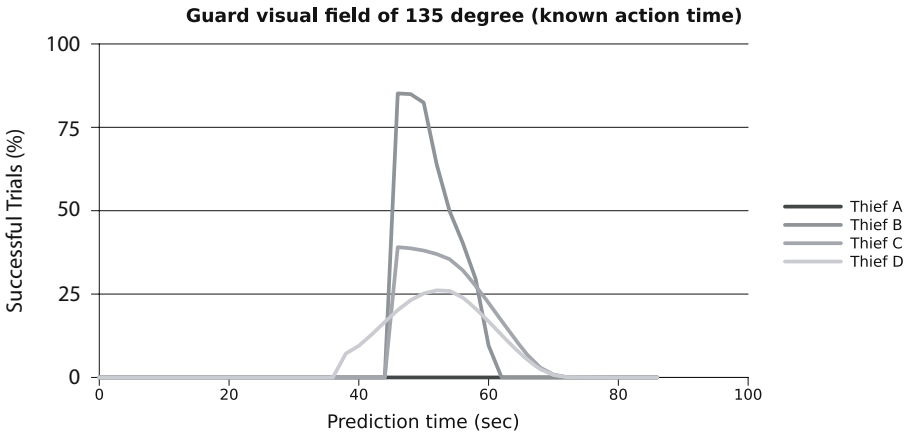


Fig. 10. The graph shows number successful trails for the thieves when the guard has a 135 degrees visual field

5.4 Experiment 4

In the last experiment, the thief agents again had a 135 degrees visual field and use the provided action time to calculate its prediction time. Instead of always predicting a static length ahead, the agent uses its action time to predict only the time necessary to fulfill the task. The agents predicts, at least as long as the action time in the beginning of a trial. During a trial the prediction time is decreased according to how much the agent manages to fulfill its mission. Instead of always predicting from the starting location to the gold point and back again, it only predicts the states from where it is currently, to the end of the task.

The result for this experiment is not the same as in previous experiment series, where success rate depend on the prediction time. In this experiment, the prediction time is constantly changing within the system. This gives us an optimal gold rate for the thief agents. The optimal rate is 0.78, 0.86, 0.51, 0.58 for thief A to D.

6 Discussion

The four experiments conducted in this paper used four different anticipatory behaviors. The first experiment uses a simple strategy where the guard can see in all directions. In the second experiment, the guard's visual field is limited to 135 degrees and in the third, the action time is used when choosing an action for the agent. In the last experiment, the action time is used to optimize the prediction time throughout different situations in the experiment.

When the agent does not use any prediction ($t_p = 0$), it only tries to fetch a piece of gold if it is not currently visible to the guard ($p_T \notin V(p_G)$). With no prediction, the success rate is higher in the second experiment than the first. This is due to the limited field of vision of the guard in the second experiment.

With a broader visual field for the guard, the areas where the thief agents can hide is reduced. Thief A and B cannot leave their home while the guard is on the right side of the gold area or between the home zone and the gold area. If the prediction time is the same or larger than the action time ($t_p \leq t_a$), an agent will be able to predict the whole route back and fourth to the gold.

Safety margins are an important feature when predicting a task. When the prediction time is longer than the action time ($t_p > t_a$) the agent will be able to predict the whole task with safety margins. For example, when crossing a street with traffic, you may want a large safety margin between the cars and yourself. An erroneous prediction of the cars' movements, could lead to devastating consequences.

When agents in the first experiment use prediction, the gold rate will increase with the prediction time, until the t_p reaches the t_{ns}. The t_{ns} is the time slot that the thieves have in which they cannot be seen by the guard.

The safety margin is closely connected to the performance of the model. With a precise model you are less dependent on the safety margin, but usually the models need to be fast, at the expense of accuracy.

A safety margin that is too long, may reduce the performance of the task. With a safety margin that is too long, you will never be able to cross the road. In the agents' case, a prediction time that is too long will result in agents not leaving their home zone.

In the second experiment, the guard's visual field is limited, which gives more interesting behaviors to the system. The guard is no longer aware of everything around it. Instead it has limited resources and focuses its attention on the environment in front of it. The guard's limited resources gives the thieves more time to fetch gold.

The limitation of the visual field for the guard gives the system two interesting features. First the direction of the guard is now more important for the outcome of the experiment. In the previous experiment, the thieves that moved along with the guard when going after the gold were having a lower success rate than the other thieves. Thieves C and D reach the state where they do not leave the home zone at all earlier than thief A and B.

When comparing the result from Experiment 1 with Experiment 2, one can see a clear difference between the two pairs of thieves. For thieves A and B, the success rate is increasing with longer prediction time in Experiment 1 and decreasing in Experiment 2. This shows how small changes like the broadness of the guard's visual field can have a large impact on the success of the task.

Another difference between the experiments, are the interesting behaviors that start to appear in the simulations. The thieves start to exploit the guard's directed attention to avoid being seen when fetching pieces of gold. Instead of waiting until the guard is out of their direct line of sight, they sneak out as soon as the guard has passed. In Experiment 2, the thieves also get trapped when trying to fetch gold and put themselves in more dangerous situations.

When the thieves start sneaking behind the guarding agent's back, it gives the thief a much more dynamic behavior. The behavior is looking more animal

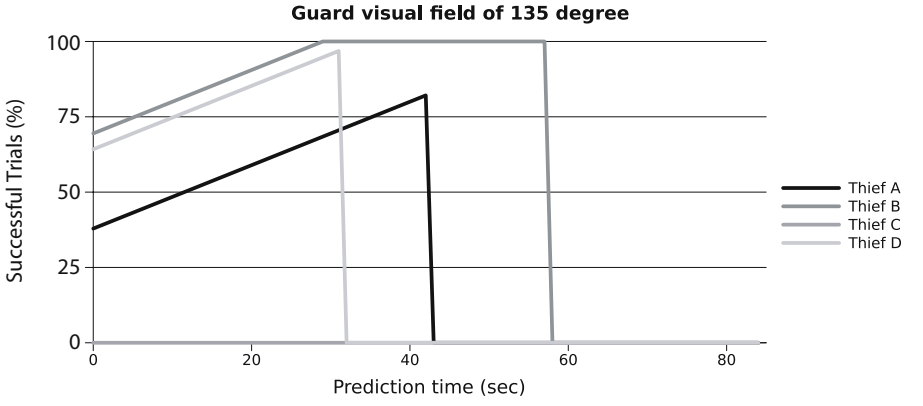


Fig. 11. The graph shows number successful trails for the thieves

like. Not only will the thieves leave their hiding place earlier, they also start to sneak behind the guard's back to avoid being seen while fetching a piece of gold. An example of this is when the thief is just behind a guard that is about to turn. When turning the guard may see the thief and the thief is forced to seek shelter. By using the limitation of the guard, the thief can choose an alternative behavior. Instead of going back home, it finds a closer hiding place behind the moving guard's back. While the guard is turning, the thief follows behind the guards back.

The change of the guard's visual field can also lead to trapping situations where the thief predicts a clear path but, during its mission, the guard manages to cut off the path between the thieves and their home zone. The result of a trap could be seen clearly for thief D at a prediction time of 35 seconds, where the success rate is almost zero (Fig. 9). When the prediction time is more then 35 seconds the thief start to predict the trap and will avoid it.

The last and most important change between Experiment 1 and Experiment 2, is that the thieves are active closer to the guard in the second experiment. In the first experiment the thieves had large margins between the guard and themselves because of the broadness of the guard's visual field. Now they are more aggressive and start going after the gold right after the guard has passed.

This is especially true for thief C and D. They are sneaking after the guard for as long as half their action time which puts them at a higher risk of being seen compared to a thief that has a larger distance to the guard or is located where it is not visible to the guard.

The theoretical result and the simulation result are compared in Fig. 11. The theoretical result indicates that a longer prediction will increase the performance of the task and this is compatible with the results in Experiment 1 and thief A and thief B in Experiment 2. The decreasing performance, for thief C and D in Experiment 2, is harder to calculate theoretically as it depends on the precisions of the whole system. With a perfect model, thieves C and D would probably also increase their performance in the second experiment. The result

from Experiment 2 indicates that with lower safety margins the model must be able to handle this, otherwise the result could lead to unwanted behaviour.

In Experiment 3, the thieves use their given action time to start a trial only if it can predict the success of the trial. In our guard and thief scenario, the thief looses its gold if it is seen but can continue with a new trial afterwards. A thief behavior like this would be crucial if the task were to be more important. For example, if the whole experiment ended if a thief was seen, a behavior like this could give a better result than the behaviors used in Experiment 1 and 2. This type of importance when making decisions is always present in human decision making. If we were in a life and death situation, we would be prone to use a strategy like this, because a failure is worse than not fulfilling the task in the most efficient way.

In Experiment 4, we let the thief use its action time to adjust its prediction time. The agent only needs to predict to the estimated end of fetching the gold. Instead of having a constant prediction time, the agent adjusts its prediction to fit the particular task better. This way of handling prediction is similar to how humans handle situations which need prediction.

We use this type of anticipation every day. An example of just this type of anticipation is when driving a car. When approaching a roundabout, we predict the other cars in the roundabout and use this together with the prediction for ourselves to enter the roundabout. This is similar to the anticipation in the third experiment.

The result from Experiment 4 indicates that a longer prediction time will actually reduce the performance of the task. The reason for this is that, with a longer prediction time, the guard and thief will stay closer to each other so the risk of being seen increases. This is a result of this particular environment where the thief tries to avoid being seen by the guard on the next lap by moving closer to on this lap.

In the third experiment, the thieves have a safety margin when the guard is going towards them but a smaller safety margin when they are going in the same direction as the guard.

To really benefit when using the prediction, the prediction time must be set depending on the task. In the guard and thief scenario the most important factor when deciding the time needed for prediction is the action time. If we know how long it will take to fetch a piece of gold we can predict according to current situation.

The action time in the guard and thief scenario is given but could easily be learnt be the thieves. One scenario could be to let the thieves have a prediction strategy, as in the first experiment when the action time is unknown, and during the experiment they learn it and change their strategy to the one used in experiment four.

The prediction strategy, used in Experiments 1 and 2, may be more realistic when there is a number of possible paths to the gold. If the thief has an option between two routes to the gold and only one is patrolled by the guard, the strategy in Experiments 1 and 2 would work very well. With a low t_p, the path

where the guard could appear would be used, as the t_p increases, the other route will be chosen more frequently.

The next step would be to apply these simulations to more interesting behaviors. The complexity could be increased, by using guards with an attention system that can be used to turn the visual field in a desire direction or to make the task more complex by changing the environment. Here, the thieves always use the shortest path to the guarded houses, but a longer path could potentially be safer. With the very limited intelligences of the guard, it cannot turn its attention in a way other than in the direction, in which it is heading. If they were allowed to direct their attention towards interesting areas, it would influence the task even more than the narrowing of the visual field. In previous work we have used attention together with event learning [13] and this will be combined with the current system for future experiments with even more interesting behaviors. The most exciting extension would be to let the thieves start to manipulate the prediction with their own actions. In the simplest case by just regulating their speed in an intelligent way. Just like when approaching a roundabout we make a prediction and decide to change the predicted state by accelerating or by slowing down to get a more satisfactory future state.

Simulations with this type of anticipation will be conducted using the AARC system in the future and some of the experiment will also move from simulation to real robots.

The results in this paper indicate that a system can gain much using predictive and anticipatory behaviors. However, a prediction must be used in the right way, otherwise the result get worse with prediction compare to without.

References

[1] Rosen, R.: Anticipatory systems. Pergamon Press, Oxford (1985)
[2] Davidsson, P.: Learning by linear anticipation in multi-agent systems. Distributed Artificial Intelligence Meets Machine Learning 1221, 62–72 (1996)
[3] Sharifi, M., Mousavian, H., Aavani, A.: Predicting the future state of the robocup simulation environment: heuristic and neural networks approaches. Systems, Man and Cybernetics 1, 27–32 (2003)
[4] Veloso, M., Stone, P., Bowling, M.: Anticipation as a key for collaboration in a team of agents: A case study in robotic soccer. In: Schenker, P.S., McKee, G.T., eds.: Proceedings of SPIE Sensor Fusion and Decentralized Control in Robotic Systems II, Bellingham, vol. 3839, pp. 134–143 (September 1999)
[5] Behnke, S., Egorova, A., Gloye, A., Rojas, R., Simon, M.: Predicting away robot control latency. In: Polani, D., Browning, B., Bonarini, A., Yoshida, K. (eds.) RoboCup 2003. LNCS, vol. 3020, pp. 712–719. Springer, Heidelberg (2004)
[6] Johansson, B., Balkenius, C.: Robots with anticipation and attention. In: Funk, P., Rognvaldsson, T., Xiong, N. (eds.) Advances in Artificial Intelligence in Sweden, pp. 202–204. Mälardalen University, Västerås (2005)
[7] Johansson, B., Kolodziej, A., Balkenius, C.: Anticipation and attention in robot control. LUCS Minor 14, Lund University Cognitive Science (2008)

[8] Balkenius, C., Morén, J., Johansson, B., Johnsson, M.: Ikaros: Building cognitive models for robots. In: Hülse, M., Hild, M. (eds.) Workshop on current software frameworks in cognitive robotics integrating different computational paradigms, Nice, France, in conjunction with IROS 2008 (2008)
[9] Jordan, M., Rumelhart, D.: Forward models: Supervised learning with a distal teacher. Cognitive Science 16, 207–354 (1992)
[10] Kalman, R.E.: A new approach to linear filtering and prediction problems. Transactions of the ASME Journal of Basic Engineering 82, 35–45 (1960)
[11] Butz, M.V.: Anticipatory learning classifier systems. Kluwer, Boston (2002)
[12] Wolpert, D.M., Flanagan, J.: Motor prediction. Current Biology (11), R729–R732 (2001)
[13] Balkenius, C., Johansson, B.: Event prediction and object motion estimation in the development of visual attention. In: Berthouze, L., Kaplan, F., Kozima, H., Yano, H., Konczak, J., Metta, G., Nadel, J., Sandini, G., Stojanov, G., Balkenius, C. (eds.) Proceedings of the Fifth International Conference on Epigenetic Robotics. Lund University Cognitive Studies, vol. 123, pp. 17–22 (2005)

Multiscale Anticipatory Behavior by Hierarchical Reinforcement Learning

Matthias Rungger, Hao Ding, and Olaf Stursberg

Institute of Automatic Control Engineering
Technische Universität München
D-80290 Munich, Germany
{matthias.rungger,hao.ding,stursberg}@tum.de

Abstract. In order to establish autonomous behavior for technical systems, the well known trade-off between reactive control and deliberative planning has to be considered. Within this paper, we combine both principles by proposing a two-level hierarchical reinforcement learning scheme to enable the system to autonomously determine suitable solutions to new tasks. The approach is based on a behavior representation specified by hybrid automata, which combines continuous and discrete behavior, to predict (*anticipate*) the outcome of a sequence of actions. On the higher layer of the hierarchical scheme, the behavior is abstracted in the form of finite state automata, on which value function iteration is performed to obtain a goal leading sequence of subtasks. This sequence is realized on the lower layer by applying policy gradient-based reinforcement learning to the hybrid automaton model. The iteration between both layers leads to a consistent and goal-attaining behavior, as shown for a simple robot grasping task.

Keywords: Reinforcement learning, hierarchical model, hybrid automaton, behavioral programming, artificial intelligence, planning.

1 Introduction

A characteristic property of intelligent autonomous systems is the capability to determine goal-attaining behavior for tasks that are posed to the system for the first time. A crucial point in determining such behavior is to anticipate what the outcome of an own action is and how the environment reacts to the action, in order to be able to select the best choice. Several approaches for anticipatory behavior of learning systems have been developed in recent years and are described, e.g. in [5,6]. Reinforcement learning (RL) is one of the main approaches to establish anticipatory behavior [20,3]. RL uses an estimate of the outcome of future actions and selects the actions for which a reward is maximized. The estimate of the outcome is either based on the observation of past behavior (i.e. the system runs iteratively through similar evolutions and assesses which actions lead to an preferable outcome) or on model-based computation. The latter approach, which is chosen in this paper, allows the system to evaluate the effects of a large

G. Pezzulo et al. (Eds.): ABiALS 2008, LNAI 5499, pp. 301–320, 2009.

variety of actions, even those which are potentially harmful for the system or its environment (i.e the system should not encounter corresponding situations in reality). Depending on the type of model used within RL and the period of time over which future behavior is anticipated, one can distinguish between reactive and deliberative planning. Reactive planning (often referred to as bottom-up approach) considers the momentary situation and produces in response a single action or a sequence of actions over a short period of future time. Often this response is suitably determined based on a sophisticated model of the situation or on a learned knowledge-base. Deliberative planning, in contrast, usually does not only consider the momentary situation but, in addition, the future evolution up to the point at which a task is accomplished – this means often that a (longer) sequence of future actions has to be determined. The model must be appropriate for anticipating the outcome of this sequence, i.e. the anticipation typically has to cover a longer time horizon and, as an implication, the underlying model encodes behavior in a more abstract setting to enable real-time computation. For planning of animals or humans, it is natural to combine both types of planning and learning in a hierarchical setting, i.e. deliberative planning leads to a rough plan for accomplishing a task, and this plan is further refined into concrete behavior using repetitively reaction to a changing environment along the envisaged plan.

To employ this principle in technical systems, a number of approaches have been proposed in the last decades, as e.g. multi-modal control, also referred to as behavior-based robotics in [1]): a behavior based architectures consists of a reactive controller and deliberative planner. The reactive controller is designed as a basis behavior with direct access to sensor and actuator signals. The planner acts on the behavior modules and is responsible for the interconnection of the different behaviors, which may be executed in parallel. A crucial point within these approach is the behavior coordination [16], such that emergent behaviors, not designed by the programmer, may arise. In [12] the behavior-based approach is used to speed up RL.

Other relevant published approaches combining learning with hierarchical planning include the following: Parameterized nonlinear differential equations are used in [19] to define motion primitives, which can be concatenated to build complex behavior. Hierarchical reinforcement learning schemes, like MAXQ [7], Options [17], and HAMS [15] use Semi-Markov processes to define subtasks – this, however, in a rather rigid scheme since the exit states for the subtasks are defined in advance. A hierarchical reinforcement learning approach on continuous dynamics is described in [14], where a two-level hierarchy is introduced to speed up learning. The higher level algorithm identifies subgoals in predefined distances within the state space of the dynamic system, which are then used to guide the system faster into the desired goal state.

The method presented in this paper is distinct from previously published ones in the following respects: To represent behavior on two layers of a learning and planning hierarchy, we use two types of formal dynamic models: On the lower layer, behavior is formulated in terms of continuous dynamics (specified by ordi-

nary differential equations) which changes discretely if certain logical conditions become true or false. Hybrid automata, as introduced in [11], are suitable to express such combined continuous-discrete behavior which is either controlled by the discrete mode or in which controllers of the continuous dynamics imply a certain sequence of discrete modes, see. e.g. [4,10,8,13]. For the example of a robotic arm for transporting objects, the mode (or logical condition) may model that different trajectories need to be realized depending on whether an object is currently grasped or not. Starting from the hybrid automaton on the lower layer, a more abstract representation of the behavior is derived for the higher layer to allow deliberative planning. Finite state automata are chosen as the type of the model, where the states represent different subgoals, and the transitions encode the continuous evolution of the system in between two subgoals. The rewards assigned to the transitions on the higher layer are calculated on-line on the lower level. Based on the finite state automaton, value iteration [3] is used to find the sequence of transitions with highest reward. This sequence is refined on the lower layer by applying reinforcement learning to the continuous dynamics of the hybrid automaton for each mode which corresponds to a transition of the higher layer sequence. The result is a goal-attaining sequence of actions which are obtained as a sequence of continuous control trajectories. Compared to the approach proposed before by the authors in [18], we here combine reinforcement learning on two layers, and the subgoals represented on the higher layer are calculated on-line. Using this hierarchical scheme, two scale anticipatory behavior is enforced in the sense that model-based anticipation of the reward of future actions is the basis for a suitable (or even best) choice of actions.

The paper is organized as follows: Section 2 defines the hybrid automaton and the problem of computing the respective action sequence for a given task. The abstraction of the hybrid automaton to the subgoal representation by a finite state automaton, is described in Sec. 3. The learning algorithms on the two layers are introduced in Sec. 4. As the main result, the overall algorithm for combining the two layers is specified in Sec. 5. An illustrating example is introduced in Sec. 7, and Sec. 8 provides conclusions and an outlook on future work.

2 Model and Problem Formulation

2.1 Lower Layer Model: Hybrid Automaton

Hybrid automata, as the type of the model chosen for the lower layer of the learning hierarchy, enable the modeler to formulate distinct continuous behavior for different modes of operation. In a first modeling step, the set of possible modes is identified and a discrete state, referred to as *location*, is assigned to each mode. Next the possible transitions between pairs of locations are identified and are formally defined as the transition structure of the hybrid automaton. For each location, a set of differential equations is identified to suitably describe the change of the relevant continuous state variables over time. This change usually depends on continuous input variables and is expressed by first order differential

equations. Well-known principles of balancing energy, mass, or impulse lead for many systems straightforwardly such rigorous dynamic models; the identification based on measured is a possible alternative if the physical principles of the system to be modeled are not well understood. Finally, the transitions are made dependent on discrete inputs and on the continuous dynamics by specifying conditions for the continuous state variables under which a transition is enabled.

Formally a hybrid automaton HA can be defined as a tuple

$$HA = (Z, V, X, U, inv, \Theta, g, \boldsymbol{r}, \boldsymbol{f})$$

consisting of:

- $Z = \{z_1, \ldots, z_{n_z}\}$ as the finite set of discrete *locations* to represent the discrete modes of operation;
- $V = \{v_1, \ldots, v_{n_v}\}$ as the finite set of *discrete inputs*, triggering the transitions by specifying the follow-up location[1];
- the *continuous state* $\boldsymbol{x}$ defined on the continuous state space $X \subseteq \mathbb{R}^{n_x}$;
- the *continuous input* $\boldsymbol{u}$ defined on the continuous input space $U \subseteq \mathbb{R}^{n_u}$;
- $inv : Z \to 2^X$ represents the assignment of *invariants* to locations; these invariants, which are compact subsets of $\mathbb{R}^{n_x}$, represent the permitted values of $\boldsymbol{x}$ as long as HA is in the respective location z;
- the finite set of *discrete transitions* $\Theta \subseteq Z \times Z$;
- a mapping $g : \Theta \to 2^X$ which assigns the so-called *guard sets* to the transitions as the subset of continuous states $g((z_i, z_j)) \subseteq X$ for which a transition $(z_i, z_j) \in \Theta$ is enabled;
- the *reset function* $\boldsymbol{r} : \Theta \times X \to X$ which is evaluated when a transition occurs and which updates the continuous state upon execution of a transition;
- the *continuous state dynamics* $\boldsymbol{f} : Z \times X \times U \to \mathbb{R}^{n_x}$ defining for every location z the evolution of the continuous states over time by a set of ordinary differential equations $\dot{\boldsymbol{x}} = \boldsymbol{f}_z(\boldsymbol{x}, \boldsymbol{u}) := \boldsymbol{f}(z, \boldsymbol{x}, \boldsymbol{u})$.

For these syntactical elements, the evolution of the hybrid automaton can be written formally as follows: Let the ordered set of event times $T = \{t_0, t_1, t_2, \ldots\}$ contain the initial time t_0 and all points of time at which a discrete transition is taken. Let $\overline{z}(t)$ denote the piecewise constant trajectory of the discrete locations with $z_k := \overline{z}(t)$ for $t \in]t_k, t_{k+1}]$. Likewise, $\overline{v}(t)$ is the piecewise constant trajectory of discrete inputs, $\overline{\boldsymbol{u}}(t)$ the continuous input trajectory, and $\overline{\boldsymbol{x}}(t)$ the continuous state trajectory. Define $\boldsymbol{x}_k := \boldsymbol{x}(t_k)$ and $\boldsymbol{x}_k^+ := \boldsymbol{x}(t_k^+)$ with $\boldsymbol{x}(t^+)$ denoting the right hand limit of $\boldsymbol{x}$ at t.

An *admissible hybrid state trajectory* $(\overline{z}(t), \overline{\boldsymbol{x}}(t))$ resulting from a given *control trajectory* $(\overline{\boldsymbol{u}}(t), \overline{v}(t))$ is then obtained as follows: After initialization to $z_0 = z(t_0)$ and $\boldsymbol{x}_0^+ = \boldsymbol{x}(t_0)$, and assuming that no immediate transition occurs at t_0, the progress of HA between two event times t_k and t_{k+1} is given by:

[1] As transitions may occur non-deterministically when the guard sets overlap, the discrete input selects the desired transition.

– the continuous evolution $\overline{\boldsymbol{x}}(t), t \in]t_k, t_{k+1}]$ as existing unique solution of

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(z_k, \boldsymbol{x}(t), \boldsymbol{u}(t)),$$
$$\boldsymbol{x}(t_k) = \boldsymbol{x}_k^+$$

with $\overline{\boldsymbol{x}}(t) \in inv(z_k)\ \forall t \in]t_k, t_{k+1}]$;
– followed by a transition $(z_k, z_{k+1}) \in \Theta$ which is subject to the guard set according to $\boldsymbol{x}_{k+1} \in g((z_k, z_{k+1}))$ and triggered by $z_{k+1} = v(t_{k+1})$. The updated continuous state is then obtained from:

$$\boldsymbol{x}_{k+1}^+ := \boldsymbol{r}((z_k, z_{k+1}), \boldsymbol{x}_{k+1}) \in inv(z_{k+1}).$$

Along a hybrid state trajectory $(\overline{z}(t), \overline{\boldsymbol{x}}(t))$, the guard sets can be interpreted as a sequence of subgoals into which the continuous trajectory is driven within each location. When a guard set is reached, the system can change from one location to another. The combination of the continuous dynamics within an active location and the subgoal is understood as a *subtask* to be accomplished to realize the hybrid state trajectory. A subtask is solved by determining the part of the input trajectory $(\overline{\boldsymbol{u}}(t), \overline{v}(t))$ which refers to the particular location.

2.2 Control Synthesis

A task to be solved can be defined such that a system has to be driven from a current (or initial) state $\boldsymbol{x}_k$ into a given future (or final) state $\boldsymbol{x}_{k+p}$ (p events later) – obviously, accomplishing the task means to solve a sequence of subtasks. If solving the task is based on a behavior representation given by HA, a straightforward interpretation is that the model is used to *anticipate* the behavior of the system under the effect of a chosen input trajectory. We formalize the evolution from an initial state into a goal state by introducing the following sets:

– an initial hybrid state set (z_0, X_0) consisting of hybrid states build from one initial location $z_0 \in Z$ and possible continuous initial states $\boldsymbol{x}_0 \in X_0$ and $X_0 \subset inv(z_0)$,
– a final hybrid state set (z_F, X_F) in which any element is composed of one final discrete location $z_F \in Z$ and a possible continuous final state with $\boldsymbol{x}_F \in X_F$ and $X_F \subset inv(z_F)$.

The *control synthesis task* is to find an input trajectory $(\overline{\boldsymbol{u}}(t), \overline{v}(t))$ for HA such that:

– an admissible trajectory $(\overline{z}(t), \overline{\boldsymbol{x}}(t))$ results for any $\boldsymbol{x}_0 \in X_0$;
– the end state lies within the final set $z(t_e) = z_F$ and $\boldsymbol{x}(t_e) \in X_F$.

In general it is desired to find not only a feasible solution, but one which is optimal with respect to some performance criteria, e.g. the time to accomplish the transfer from the initial state to the goal state. To circumvent the difficulty to solve an optimal control problem for a hybrid automaton as the underlying dynamical

system, local performance criteria are introduced for each location. The continuous control $\overline{\boldsymbol{u}}(t)$ is calculated to maximize the given local performance criteria.

An example is illustrated in Fig. 1: It consist of two locations $\{z_1, z_2\}$ representing two different continuous dynamics. The invariants for each location, defining the state space of each continuous subsystem are illustrated by the gray shaded regions. The two continuous dynamics $\{\boldsymbol{f}_1, \boldsymbol{f}_2\}$, defined on $\mathbb{R}^2$, are restricted to the subsets $inv(z_1)$ and $inv(z_2)$ respectively. A trajectory starting within location $z(t_0) = z_1$ and $\boldsymbol{x}(0) \in X_0$, is depicted by the bold black line. When the trajectory reaches the guard set g_{12}, the discrete input is set to $z(t_1) = v(t_1) = z_2$. Thus, the discrete transition is taken and the reset function $\boldsymbol{r}$ is evaluated for the state $\boldsymbol{x}(t_1)$. The evolution of the trajectory starts again in the co-domain of the reset function, governed then by the new dynamics $\boldsymbol{f}_2$, and progresses until the final set X_F is reached. A corresponding technical example

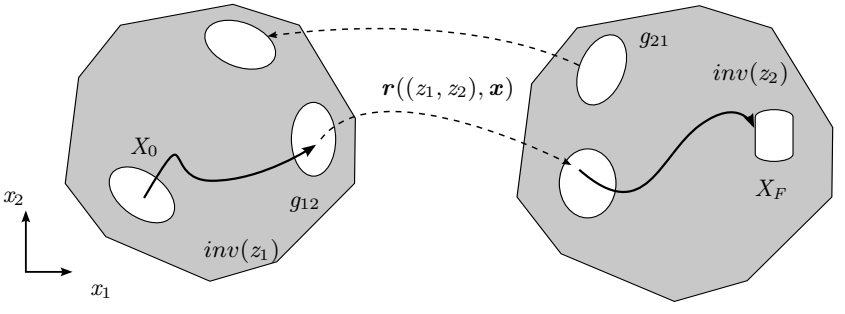


Fig. 1. An illustrating example of an hybrid automaton with two discrete locations z_1 and z_2

from the area of robotics is the transition from a locomotion task to a grasping task. The robot chassis dynamics is active while approaching the object to grasp. When the object is within reach, the grasping dynamics of the robot arm become active.

2.3 Higher Layer Model: Subgoal Automaton

Before introducing the algorithm for solving the control synthesis task, the subgoal automaton for modeling the system behavior in an abstract form on the higher layer is introduced. As mentioned in Sec. 1, the reason for using a second, less detailed behavior representation is that goal attainment by deliberative planning usually requires to cover longer time horizons – searching for control trajectories to solve the aforementioned control problem for HA and for long time horizon often turns out to be too complicated to be accomplished in real time. Thus, we here choose the approach to vertically decompose the problem by temporarily searching for a solution to a task on a simplified behavior representation. This model must still have enough structure to represent the behavior

on a qualitative scale as well as a sufficiently small state space to allow finding an action sequence quickly.

As mentioned above, the trajectory $(\overline{z}(t), \overline{\boldsymbol{x}}(t))$ of a hybrid automaton consists of the alternating sequence between the continuous evolution within one location and the discrete transition to change locations, which represent particular modes of operation or capabilities of the technical system. In this regard, the continuous evolution of the system can be considered as a subtask while the subgoal, associated with the subtask, is given in terms of the guard set. The continuous evolution within one location of the HA is represented by a transition of the *subgoal automaton* SGA, and a state of SGA corresponds to a guard set of HA for representing a particular subgoal. For example, the hybrid trajectory from the previously introduced example (see Fig. 1), is modeled by the sequence (s_0, s_{12}, s_F). For each guard set as well as for the initial set and the final set, a discrete state is introduced in SGA – the corresponding model is shown in Fig. 2.

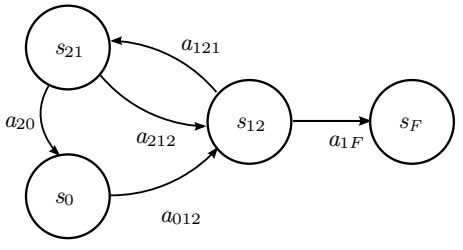


Fig. 2. SGA model for the example in Fig. 1

Formally, the automaton is defined by:

$$SGA = (S, A, h)$$

with:

- the finite set of discrete states $S = \{\ldots, s_{ij}, \ldots\} \cup \{s_0, s_F\}$ with one state s_{ij} for each guard set g_{ij} defined for HA, complemented by the initial state s_0 and the desired final state s_F;
- the set $A = \{\ldots, a_{ss'}, \ldots\}$ of actions $a_{ss'}$ which represent the hybrid evolution of all trajectories originating from the guard set g_s and leading to $g_{s'}$;
- the transition function is defined by $h(s, a_{ss'}) = s'$ for a transition from the state s to the state s' under effect of the action $a_{ss'}$ (introduced for any possible transition of HA). Additionally, transitions for the initial state s_0 and the final state s_F are defined, i.e. s_0 is connected to all states $s \in S$ representing a guard set contained in the invariant $g_s \in inv(z_0)$, and a similar construction is used for the final state s_F.

2.4 Example

The modeling procedure is illustrated for a simple example consisting of a 1-DOF robot arm which can move forward and backward. The task is to grab the ball at position p_0 and move it to p_1.

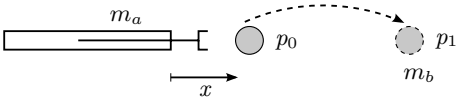


Fig. 3. Robot arm with the aim to move the ball to position p_1

The HA for the robot is modeled by two locations (z_1, z_2), where one represents free arm movement and one represents the movement of the arm with the ball. The dynamics within the locations are given by

$$z_1 : \ddot{x} = \frac{u}{m_a}, \qquad z_2 : \ddot{x} = \frac{u}{m_a + m_b},$$

and the guard sets by:

$$g_{12} = \{x \,|\, x = p_0\}, \quad g_{21} = \{x \,|\, x = p_1\}.$$

The grabbing/releasing of the ball is considered to be triggered by the discrete input signal $v \in \{z_1, z_2\}$.

The states of the corresponding abstract model SGA are given by s_0 (the initial state as shown in Fig. 3), s_{12} (representation of the guard set g_{12}), s_{21} (corresponding to the guard set g_{21}), and s_F : (the final state). A task for this system would be to design a controller which drives the robot arm from $\boldsymbol{x}_0$ to p_0, set $v(t) = z_2$ such that the ball is grabbed and the transition is taken. Then, the ball has to be moved to p_1 and the discrete input is reset to $v(t) = z_1$.

In general, the control synthesis task for the hybrid automaton is solved by: **a)** determining an appropriate sequence of locations, **b)** identifying the appropriate switching times/states for triggering the discrete transitions, and **c)** finding the continuous control laws within each location for driving the system to the identified states within the chosen guard sets.

3 Solution Algorithm

In the following, value iteration is used to find the sequence of locations on the higher layer and reinforcement learning is used to calculate the continuous control law on the lower layer. The reward signal on the higher layer, as the underlying driving force of the value iteration algorithm, is first initialized to a guess and then iteratively updated based on computation on the lower layer. The reward signal on both layers is designed such that a positive reward is assigned if a desired state is reached, and a negative reward if not.

3.1 Value Iteration

Given the SGA model on the higher layer, value iteration is applied to find an action sequence which leads the system from the initial state to the desired goal state.

For a policy $\pi : S \rightarrow A$, which provides for each state an appropriate action, the accumulated reward from the initial state s^0 to the goal state s^N is given by:

$$W^{\pi}(s^0) = \sum_{k=0}^{N-1} c(s^k, h(s^k, \pi(s^k))),$$

where $(s^0, s^1, \ldots, s^N)$ is the indexed state sequence of the resulting trajectory. The calculation of the rewards $c_{ss'} := c(s, s')$ associated with every action $a_{ss'}$ will be described in detail in Sec. 4. For realizing the goal oriented behavior, the policy (a state to action mapping) is derived such that the reward-to-come is maximized:

$$W(s^0) = \max_{\pi} \sum_{k=0}^{N-1} c(s^k, h(s^k, \pi(s^k))).$$

Applying the Bellman Principle, the maximal reward-to-come, referred to as the *value function*, is formulated recursively

$$W(s) = \max_{a}\{c(s, h(s, a)) + W(h(s, a))\}, \quad \forall s \in S. \tag{1}$$

A greedy policy is directly derived by maximizing the one step reward plus the expected/anticipated reward-to-come from the resulting state.

The subgoal automaton is a deterministic finite state automaton. Thus, value iteration (see [3]) with a look-up table representation is a viable solution method for the calculation of the value function.

After initialization of all values to zero, $W_0(s) = 0$ for all s, the value function is iteratively updated for all states by:

$$W_i(s) = \max_{a}\{c(s, h(s, a)) + W_{i-1}(h(s, a)\},$$

where i is the iteration index. Since no uncertainty of the outcome of actions need to be considered, the incremental update rule (normally included in value iteration schemes) is omitted. The pseudo code of the iterative procedure is listed in Alg. 1. The state sequence from the initial state to the final state is obtained by using the greedy policy.

3.2 Continuous Valued and Time Reinforcement Learning

In this section, the algorithm for the realization of the control commands on the lower layer is introduced. The objective is to implement a solution which is consistent to the sequence of discrete actions on the higher layer. The same

Algorithm 1. Value iteration

INITIALIZATION: $\forall s : W_0(s) = 0$, $i = 0$, $\delta = \infty$
VALUE ITERATION:
while $\delta > \Delta$ (a small threshold) **do**
  for all $s \in S$ **do**
    $W_{i+1}(s) := \max_{a \in A}\{c(s, h(s, a)) + W_i(h(s, a))\}$
    $\delta = \min(\delta, |W_{i+1}(s) - W_i(s)|)$
    $i := i + 1$
  end for
end while
STATE SEQUENCE:
return $(s^0, \ldots, s^n)$

reward principle as on the higher layer is used. The algorithm for the implementation of a continuous time, continuous-valued version of reinforcement learning was introduced in [9]. It is stated here in condensed form: Since each location is considered separately, the system dynamics is specified by:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t)),$$

where the index for the location z is here omitted for simplicity of notation. The reward signal is denoted by $l(\boldsymbol{x}(t), \boldsymbol{u}(t))$ and control inputs are calculated to maximize the cumulative reward – this may model, e.g., the inverse time to reach the next subgoal or the negative quadratic distance to the subgoal.

Similarly as previously introduced for SGA, the reward-to-come for a policy $\boldsymbol{u}(t) = \mu(\boldsymbol{x}(t))$ is given by:

$$V^{\mu}(\boldsymbol{x}(t)) = \int_t^{t_F} e^{-\frac{s-t}{\tau}} l(\boldsymbol{x}(s), \boldsymbol{u}(s))\mathrm{d}s,$$

where t_F is the time at which the trajectory $\boldsymbol{x}(t)$ reaches the guard set, or the subgoal respectively. τ is the time constant for discounting future rewards. The optimal value function maximizing the cumulative future reward is given as:

$$V^*(\boldsymbol{x}(t)) = \max_{\overline{\boldsymbol{u}}(s)} \int_t^{t_F} e^{-\frac{s-t}{\tau}} l(\boldsymbol{x}(s), \boldsymbol{u}(s))\mathrm{d}s,$$

with $\boldsymbol{u}(s)$, $s \in [t, t_F]$. Applying the Bellman principle of optimality leads to a discounted version of the Hamilton-Jacobi-Bellman equation (see [9] for details):

$$\frac{1}{\tau} V^*(\boldsymbol{x}(t)) = \max_{\boldsymbol{u}}\{l(\boldsymbol{x}(t), \boldsymbol{u}(t)) + \frac{V^*(\boldsymbol{x}(t))}{\boldsymbol{x}} \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t))\}, \tag{2}$$

and the policy at a certain time is derived from the right-hand side to:

$$\boldsymbol{u} = \mu(\boldsymbol{x}) = \arg\max_{\boldsymbol{u}}\{l(\boldsymbol{x}, \boldsymbol{u}) + \frac{V^*(\boldsymbol{x})}{\boldsymbol{x}} \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u})\} \tag{3}$$

Since a continuous time, continuous-valued dynamical system is considered (within one location), a look-up table representation of the value function is not appropriate. Instead a function approximation architecture is used to represent the value function:

$$V^{\mu}(\boldsymbol{x}(t)) \approx V(\boldsymbol{x}(t), \boldsymbol{w}).$$

Learning of the value function is performed in terms of updating the parameter $\boldsymbol{w}$ of the function approximation. The self-consistency condition, which follows from Eq. (2) to $\dot{V}^{\mu}(\boldsymbol{x}(t)) = \frac{1}{\tau}(V^{\mu}(\boldsymbol{x}(t)) - l(\boldsymbol{x}(t), \mu(\boldsymbol{x}(t)))$, is used to evaluate the current estimate $V(\boldsymbol{x}(t))$ of the value function. The weights of the approximation are adapted such that the error:

$$E(t) = \frac{1}{2}|\dot{V}(\boldsymbol{x}(t)) - V(\boldsymbol{x}(t)) + l(\boldsymbol{x}(t), \mu(t))|^2 \tag{4}$$

is minimized. As described in [9], a potential problem using $\dot{V}$ to update the weights is the symmetry in time. An approach to update the past estimates of $V(\boldsymbol{x}(t))$ without affecting the future estimates is to employ an Euler approximation $\dot{V}(\boldsymbol{x}(t)) = (V(\boldsymbol{x}(t)) - V(\boldsymbol{x}(t - \Delta t)))/\Delta t$. The gradient of the squared error (4) with respect to the parameter w_i results then in:

$$\frac{\partial E(t)}{\partial w_i} = \delta(t)\frac{1}{\Delta t}\left[\left(1 - \frac{\Delta t}{\tau}\right)\frac{\partial V(\boldsymbol{x}(t), \boldsymbol{w})}{\partial w_i} - \frac{\partial V(\boldsymbol{x}(t - \Delta t), \boldsymbol{w})}{\partial w_i}\right]$$

with:

$$\delta(t) = \frac{1}{\Delta t}\left[\left(1 - \frac{\Delta t}{\tau}\right)V(\boldsymbol{x}(t)) - V(\boldsymbol{x}(t - \Delta t))\right] + l(\boldsymbol{x}(t), \mu(\boldsymbol{x})) \tag{5}$$

as the *temporal difference* error. It coincides with the conventional TD error (see [20]). A gradient descent algorithm to search for the w_i, which minimizes the error, uses the rule:

$$\dot{w}_i = -\eta\delta(t)\left[\left(1 - \frac{\Delta t}{\tau}\right)\frac{\partial V(\boldsymbol{x}(t), \boldsymbol{w})}{\partial \boldsymbol{w}_i} - \frac{\partial V(\boldsymbol{x}(t - \Delta t), \boldsymbol{w})}{\partial w_i}\right] \tag{6}$$

with η as the learning rate. This update scheme corresponds to the residual-gradient algorithm (see [2]).

The control law: The proposed procedure is an on-line learning approach, thus the control law stated in Eq. (3) has to be solved in every simulation step. Depending on the complexity of the reward $l(\boldsymbol{x}, \boldsymbol{u})$ and the system dynamics $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u})$, the solution of this static optimization problem is in general difficult to obtain. A possible approach is to establish an *actor-critic architecture*: the feedback mapping $\mu : X \rightarrow U$ is approximated by a function approximator and is learned online. Under the assumptions that the reward function l is convex in $\boldsymbol{x}$ and the dynamics is linear with respect to $\boldsymbol{u}$, an analytic solution can be derived by differentiating Eq. (3), leading to:

$$0 = \frac{\partial l(\boldsymbol{x}, \boldsymbol{u})}{\partial \boldsymbol{u}} + \frac{\partial \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u})^T}{\partial \boldsymbol{u}}\frac{\partial V(\boldsymbol{x})^T}{\partial \boldsymbol{x}}. \tag{7}$$

The solution with respect to $\boldsymbol{u}$ results in the control law. In [9], this is referred to the *value-gradient based policy*.

The algorithm which realizes the described steps is specified below as(Alg. 2. For each of a chosen number of N trials, the dynamics and parameter equation is numerically simulated for the time interval $[0\, P]$, where P is an estimated upper bound of time required to reach the corresponding subgoal.

The algorithm (Alg. 2) is embedded into the overall scheme according to Alg. 1 as the step to compute the rewards c, see the next section. A value function V_g for each guard set g is determined when the corresponding continuous evolution into g is requested as part of the action sequence computed on the higher layer.

Algorithm 2. Value-Gradient based value iteration within one location

PARAMETERS: N, P
INITIALIZATION: $\boldsymbol{w}(0) := 0$
PROGRESS:
while $j < N$ **do**
  SET: $\boldsymbol{x}(0) := \boldsymbol{x}_0$, $t := 0$
  while $t \leq P$ **do**
    Simulate $\boldsymbol{x}(t)$ and $\boldsymbol{w}(t)$ with
      $\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t))$
      $\dot{w}_i(t) = -\eta\delta(t)\left[\left(1 - \frac{\Delta t}{\tau}\right)\frac{\partial V(\boldsymbol{x}(t),\boldsymbol{w})}{\partial w_i} - \frac{\partial V(\boldsymbol{x}(t-\Delta t),\boldsymbol{w})}{\partial w_i}\right]$
  end while
  $j := j + 1$
end while

4 The Hierarchical Learning Approach

Based on the algorithms 1 and 2, this section describes the overall procedure of hierarchical learning. A crucial point of this procedure is the calculation of the transition rewards for the algorithm 1. The rewards c_{ijk} represent the cumulative reward obtained along the trajectory of the system within one location. Additionally, for each state s of the SGA, a goal state $\boldsymbol{x}_s \in g_s$ for the corresponding subtask is assigned. It is used on the one hand to set the discrete control input v when the continuous state trajectory enters a subgoal, and on the other hand to guide the system within one location, i.e. to determine the low-level reward signal l.

4.1 Subtask Reward Calculation

A transition $a_{ss'}$ of the subgoal automaton SGA represents the transfer of the system from the entry into a location (e.g. by the preceding reset) into the next guard set of the hybrid automaton HA. The transition can be interpreted as the set of all possible trajectories connecting g_s with $g_{s'}$, i.e. the rewards $c_{ss'}$ in the SGA represent the cumulative reward arising from such a transfer.

For the calculation of the transition reward, the value function $V_{s'}$ of the goal location is used in combination with the continuous goal state $\boldsymbol{x}_s$ of the preceding location:

$$c_{ss'} = V_{s'}(\boldsymbol{r}_s(\boldsymbol{x}_s)). \tag{8}$$

As will be described below, the continuous subgoal state $\boldsymbol{x}_s$ is calculated online and may change over the iterations. In the case that the subtask cannot be achieved, i.e. the subgoal state is not reachable, a low reward value is assigned to the corresponding transition of SGA.

4.2 Subgoal State Calculation

Until now, the focus was on the calculation of the sequence of goal leading locations and the corresponding continuous control trajectories $\overline{\boldsymbol{u}}(t)$. To obtain the control for setting the discrete input v, and thus to trigger a particular transition, the *subgoal state* $\boldsymbol{x}_s \in g_s$ is examined further. The subgoal state of a subtask is restricted to lie within the particular guard set g_s which terminates the subtask.

Starting the continuous evolution $\boldsymbol{x}(0) \in inv(z_i)$ of the system within location z_i, the discrete input is set when the trajectory reaches the subgoal state

$$v = z_j \quad \text{if } \boldsymbol{x}(t) = \boldsymbol{x}_{s_{ij}} \in g_{s_{ij}}.$$

By determining the subgoal state within a guard set, the decision where and when to switch is determined autonomously.

The subgoal plays a crucial role for the decision where to change location. The subgoal state is the goal state for the current subtask. It is chosen to complete the subtask (inside the guard set) and to give an initialization for the next subtask. Thus, the subgoal state is calculated within the corresponding guard set and to maximize the reward-to-come for the subsequent subtask s':

$$\boldsymbol{x}_s = \arg\max_{\boldsymbol{x} \in g_s} V_{s'}(\boldsymbol{r}_s(\boldsymbol{x})). \tag{9}$$

Thus, the choice of $\boldsymbol{x}_s$ is based on an anticipation of the reward-to-come of the next subtask. If the optimization result is not unique, the subgoal state is picked randomly from the set of possible states.

4.3 Algorithm

The formulae Eq. (8) and Eq. (9) complete the set of components required to formulate the overall learning algorithm. After the initialization of the value function V_s for each state s of the SGA, the iteration is started. The first value iteration results in a shortest path sequence from the initial state s_0 to the final state s_F, since no value function on the lower layer is trained and no subgoal $\boldsymbol{x}_s$ is reached yet (equal rewards for all transitions). After the sequence of guard sets $(s^0, \ldots, s^n)$ is determined by Alg. 1, the following is carried out for each

transition $a_{ss'}$ referring to this sequence: first, the continuous subgoal state $\boldsymbol{x}'_s$ is calculated, and then Alg. 2 is executed to learn $V_{s'}$. If the system trajectory reaches the subgoal state $\boldsymbol{x}(t) = \boldsymbol{x}_{s'}$, the transition $a_{ss'}$ proposed by the higher layer is realized and the learning of V_s continues for the next location. If the subgoal state was found to be not reachable, the learning for the particular sequence $(s^0, \ldots, s^n)$ stops, the value function results in a low reward, and thus the corresponding transition of the SGA is avoided subsequently. Then, the value iteration on the higher layer resumes, this time with updated value functions V_s to adapt the transition rewards. In this manner, the higher layer value iteration is steered towards a state sequence for which the the guard sets on the lower layer are reachable, and accordingly the *control synthesis task* in Sec. 3.1 is solved. The algorithm, which is listed in Alg. 3, stops when the final state is reached. Of course, the iteration may be repeated to enhance the performance.

Algorithm 3. Algorithm for hierarchical reinforcement learning

```
INITIAL/FINAL STATE: x_{s_0} := x_0, x_{s_F} := x_F
INITIALIZATION: render SGA, w_s(0) := 0,
PROGRESS:
while ||x(t) − x_F|| < ε do
  for all a_{ss'} ∈ A do
    c_{ss'} := V_{s'}(r_s(x_s))
  end for
  determine (s^0, ..., s^n) by Alg. 1
  for i = 1 : n do
    x_{s^{i+1}} := arg max_{x ∈ g_{s^{i+1}}} V_{s^{i+2}}(r(x))
    use Alg. 2 with x_0 := x_{s^i}, x_G := x_{s^{i+1}}
    if x(t) ≠ x_{s^{i+1}} then
      break and resume with outer while loop
    end if
  end for
end while
```

5 Simulation Results

In this section, the procedure is illustrated by means of a 2-DOF robot arm. Similar to the previously introduced example, the task is specified as a transportation problem, in which the robot arm has to move to position p_0, grab the ball, move it to position p_1, and release it there (see Fig. 4). To illustrate the proposed approach, the behavior of the robot arm is fixed to separated linear and rotational motion. The resulting hybrid automaton consists of 4 locations, representing *linear motion without ball*, *linear motion with ball*, *rotational motion*

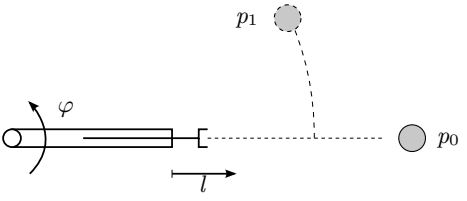


Fig. 4. The robot arm aims to move the ball from position p_0 to position p_1 by using its revolute and prismatic joints

without ball, and *rotational motion with ball*. The dynamics for the locations is given as:

$$z_1 : \dot{\boldsymbol{x}} = \begin{pmatrix} u_1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \qquad z_2 : \dot{\boldsymbol{x}} = \begin{pmatrix} u_1 \\ 0 \\ u_1 \\ 0 \end{pmatrix}$$

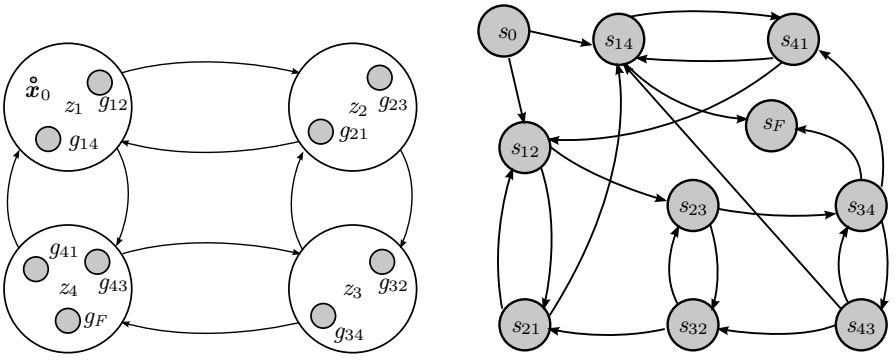
$$z_3 : \dot{\boldsymbol{x}} = \begin{pmatrix} 0 \\ u_2 \\ 0 \\ 0 \end{pmatrix} \qquad z_4 : \dot{\boldsymbol{x}} = \begin{pmatrix} 0 \\ u_2 \\ 0 \\ u_2 \end{pmatrix}$$

with $\boldsymbol{x} = (l, \varphi, l_B, \varphi_B)^T$, denoting the translational positions (l, l_B) and the angles (φ, φ_B) of the robot's end-effector, and the ball respectively (index $_B$). Neglecting the inertial forces, it is assumed that the translational control u_1 and the rotational control u_2 are commanded directly. The invariants of all locations are given by $inv(z) = [0.5,\, 1.2] \times [-\pi,\, \pi] \times [0.5,\, 1.2] \times [-\pi,\, \pi]$.

Fig. 5(a) displays the hybrid automation with the four locations and the corresponding guard sets and transitions. The transition from location z_1 to z_2, i.e. from linear motion without ball to linear motion with ball is bound to the guard set $g_{12} = \{\boldsymbol{x} \,|\, |x_1 - x_3| < 0.01\}$ indicating the state space, where the end-effector approaches the ball (grabbing the ball is neglected). Since it is everywhere allowed to release the ball, the guard set g_{21} for the reverse transition coincides with the invariant of the location. The guard sets corresponding to the transitions of the system from linear motion to rotational motion (z_1 to z_4 and z_2 to z_3) also coincide with the invariant, and such that it is always possible to take the transition. The reverse transition (from rotational to linear motion) is restricted to the part of the state space in which the angular displacement is zero: $g_{32} = g_{41} = \{\boldsymbol{x} \,|\, x_2 = 0\}$. The task initial state $\boldsymbol{x}_0 = (0.5\,0\,1.1\,0)^T \in inv(z_1)$ and final position $\boldsymbol{x}_F = (0.8\,0\,0.8\,\pi/3)^T \in inv(z_4)$ are marked in the figure.

The generated SGA is shown in Fig. 5(b). Its state set consists of one state each for representing the guard sets s_{ij}, the initial state s_0 and the final state s_F.

According to the given task, the algorithm 3 is able to determine the following solution, which intuitively is the correct one: The end-effector is first moved from $\boldsymbol{x}_0 \in inv(z_1)$ to the ball position $x_1 = x_B$, thus entering the guard set g_{12} and triggering the transition to z_2. Then the ball is moved to $x_1 = 0.8$ and



(a) The hybrid automation with its guard sets for the considered robot arm.

(b) The generated transition automaton.

Fig. 5. Dynamic models for the robot example

the transition to rotational motion occurs by which ϕ is changed to $x_2 = \pi/3$. The point ($\boldsymbol{x}_{s_{34}} = (0.8\,0\,0.8\,0)^T$) is determined iteratively by Eq. (9) within the algorithm. The ball is then released and the robot arm moved to $x_2 = 0$. The corresponding state sequence for the SGA is $(s_0, s_{12}, s_{23}, s_{34}, s_F)$.

To obtain this result, the continuous control on the lower layer is achieved by formulating the reward for guiding the end-effector to the desired subgoals $\boldsymbol{x}_s$ within the different locations as:

$$r(\boldsymbol{x}, \boldsymbol{u}) = -|\boldsymbol{x}(t) - \boldsymbol{x}_s|^2 - \int_0^{u_i} \nu \tan\left(\frac{\pi}{2}\frac{u}{u_i^{\max}}\right) \mathrm{d}u,$$

such that the control law using Eq. (7) results in:

$$\mu_z(\boldsymbol{x}) = \frac{2}{\pi} u^{\max} \arctan\left(\frac{1}{\nu}\frac{\partial \boldsymbol{f}_z(\boldsymbol{x}, \boldsymbol{u})^T}{\partial \boldsymbol{u}}\frac{\partial V(\boldsymbol{x}, \boldsymbol{w})^T}{\partial \boldsymbol{x}}\right).$$

The constant ν is chosen to 0.01, and the underlying approximating function consists of linearly weighted Gaussian bell-shaped functions, also known as radial basis functions (RBF).

For the iterative computation on the higher layer, the transition rewards for SGA are initialized with -0.1, and the weights of the RBF-network are initialized to 0. The first value iteration results in the shortest path sequence since all transitions are initialized with the same reward, i.e. the sequence is:

$$(s_0, s_{14}, s_F).$$

As a result, the guard set g_{14} on the lower layer is the subgoal in the first step. After the calculation of the particular state $\boldsymbol{x}_{s_{14}}$ in g_{14} by evaluating $\arg\max_{\boldsymbol{x}} V_{s_F}$, (see Eq. (9)) algorithm 2 is evoked with $N = 10$ trials for $P = 10\,\mathrm{sec}$. The trajectory within the last trial reaches the guard set g_{14}, thus the transition is

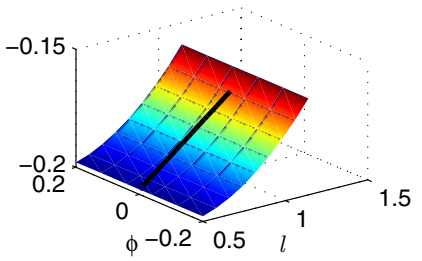
taken and the algorithm continues within location z_4. Alg. 2 is started again, now with $\boldsymbol{x}_{s_F} = (0.8\,0\,0.8\,\pi/3)$ targeting the final state. Since the last trajectory within the iteration does not reach the final state (the ball is still at the initial position), the iteration is interrupted, and the value iteration (Alg. 1) for SGA is started again. This time, the rewards for the transition (s_0, s_{14}) and (s_{14}, s_F) are updated by the values from $V_{g_{14}}$ and V_{g_F} trained in the previous iteration. While learning V_{g_F}, the goal is never reached, thus the reward diminished to -0.30 (see Tab. 1). Thereafter, the value iteration results in the desired sequence $(s_0, s_{12}, s_{23}, s_{34}, s_F)$, but it can not be realized on the lower level since the corresponding value functions for the low level are not yet trained well enough. Different sequences on the higher level are evaluated until again the desired sequence $(s_0, s_{12}, s_{23}, s_{34}, s_F)$ is selected in the 5^{th} iteration, which eventually can be realized on the lower layer.

Table 1. Value function for SGA. Next to each value the index within the state sequence computed for SGA is listed.

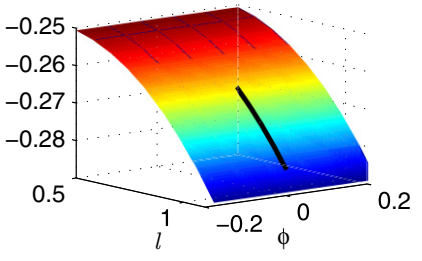
	W^0		W^1		W^2		W^4		W^5	
s_0	-0.20	1	-0.40	1	-2.48	1	-2.50	1	-3.16	1
s_{12}	-0.30	–	-0.30	2	-0.50	2	-1.64	–	-1.64	2
s_{23}	-0.20	–	-0.20	3	-0.20	–	-0.20	–	-0.20	3
s_{34}	-0.10	–	-0.10	4	-0.10	6	-0.10	4	-0.10	4
s_{41}	-0.20	–	-0.40	–	-0.40	–	-0.40	–	-1.74	–
s_{14}	-0.10	2	-0.30	–	-0.30	4	-0.30	2	-1.84	–
s_{21}	-0.20	–	-0.40	–	-0.40	3	-0.40	–	-1.74	–
s_{32}	-0.30	–	-0.30	–	-0.30	–	-0.30	–	-0.30	–
s_{43}	-0.20	–	-0.20	–	-0.20	5	-0.20	3	-0.20	–
s_F	0	3	0	5	0	7	0	5	0	5

The value functions $V_{g_{12}}, V_{g_{23}}, V_{g_{34}}, V_{g_F}$ used for the calculation of the continuous control law for the final sequence are plotted in Fig. 6(a-d). The black solid lines show the trajectory of the end-effector within the last iteration. The value functions have their maxima where the corresponding highest rewards are observed. For example, $V_{s_{12}}$ is the value function driving the system from the linear motion without ball to linear motion with ball. The transition occurs when the end effector reaches the ball position at $x_1 = 1.1$. It can be seen that $V_{g_{12}}$ has its maximum at this value, and thus the end effector is driven to the ball position.

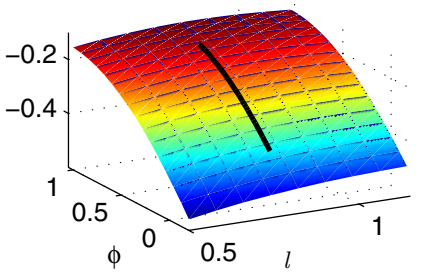
The subgoal state $\boldsymbol{x}_{s_{23}} \in [0.5\,1.2] \times 0 \times [0.5\,1.2] \times 0$ triggering the transition from linear motion to rotational motion with ball is determined by evaluating $\arg\max_{\boldsymbol{x}} V_{s_{34}}$. The guard set g_{34} represents the transition from the rotational motion with ball to the rotational motion without ball. It is triggered at $\boldsymbol{x}_{s_{34}} = (0.8\,\pi/3\,0.8\,\pi/3)^T$. The value function $V_{s_{34}}$ is plotted over the guard set g_{23} in Fig. 6(e). It can be observed that the maximum is at $x_1 = 0.8$, hence $\boldsymbol{x}_{s_{23}} = (0.8\,0\,0.8\,0)^T$. The trajectory of the end-effector for completing the task is plotted over time in Fig. 6(f).



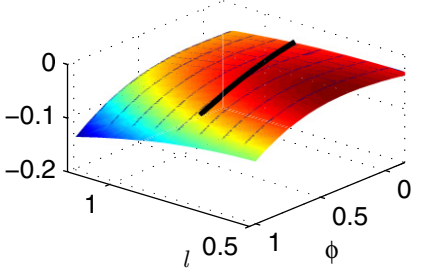
(a) Value function $V_{g_{12}}(\boldsymbol{x})$



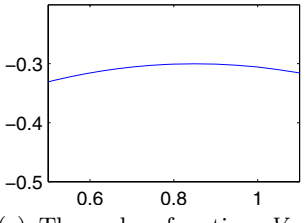
(b) Value function $V_{g_{23}}(\boldsymbol{x})$



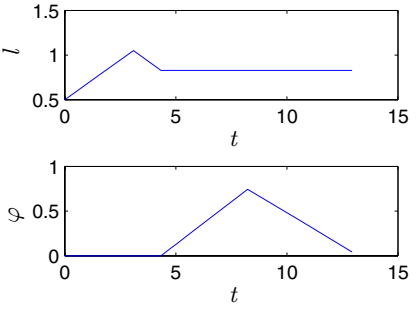
(c) Value function $V_{g_{34}}(\boldsymbol{x})$



(d) Value function $V_{g_F}(\boldsymbol{x})$



(e) The value function $V_{s_{34}}$ evaluated over g_{23}.



(f) Goal leading trajectory of the end-effector over time.

Fig. 6. Numerical results for the hierarchical reinforcement learning algorithm

6 Summary and Conclusion

A hierarchical algorithm is proposed by which a technical system can establish autonomous goal-attaining behavior for new tasks. To select between (sequences of) possible actions to accomplish the task, model-based anticipation of the outcome of actions is used. The starting point, a hybrid automaton model, represents the different capabilities of the system, but is often to complex for finding control trajectories that solve the given task. Thus, the suggestion is to generate the subgoal automaton (SGA), for which value iteration leads to a

coarse and potentially goal-attaining sequence of subtasks. The rewards of the transitions of SGA are updated iteratively from the lower level execution. The vertical decomposition of the task solution together with the model-based anticipation contribute to finding plans an control actions for rather complicated task without the necessity of exploring the complete hybrid state space. On both layers of the hierarchy an anticipated estimate of the future reward outcome of possible action are computed – since the complete computation is model-based, the system does not need to experiences behavior which is not successful (or possibly harmful) in reality.

The introduced example demonstrates the viability of the proposed approach for task solving, when different dynamics need to be activated in sequential manner. The benefits of the approach is that the complicated tasks is split into a (deliberative) planning of abstract action sequences on the higher layer and the realization (reactive planning) on the low layer. Even if the solution is completely unclear to the system when the task is posed, the integrated solution scheme achieves to find a feasible solution after a relatively low number of iterations without exploring large parts of the state search space of the original problem (defined for HA). Thus, the hierarchical approach seems promising to render reinforcement learning applicable to relative complex problems, by enforcing motion constraints and defining simple basic motion primitives, as in the example where the robot arm is restricted to activate linear motion or rotational motion sequentially. It is a matter of current work to investigate in detail what complexity of tasks can be accounted for by the proposed approach.

Future work will focus on a formal convergence proof of the approach as well as on a reduction of the number of hand tuned parameters, like the duration and number of trials for the continuous time reinforcement learning.

References

1. Arkin, R.C.: An Behavior-based Robotics. MIT Press, Cambridge (1998)
2. Baird, L.: Residual algorithms: Reinforcement learning with function approximation. In: Proceedings of the Twelfth International Conference on Machine Learning, pp. 30–37 (1995)
3. Bertsekas, D.P., Tsitsiklis, J.: Neuro-Dynamic Programming. Athena Scientific, Belmont (1996)
4. Branicky, M.S.: Behavioral Programming. In: Working Notes AAAI Spring Symp. on Hybrid Systems and AI (1999)
5. Butz, M.V., Sigaud, O., Gérard, P.: Anticipatory Behavior: Exploiting Knowledge About the Future to Improve Current Behavior. In: Butz, M.V., Sigaud, O., Gérard, P. (eds.) Anticipatory Behavior in Adaptive Learning Systems. LNCS, vol. 2684, pp. 1–10. Springer, Heidelberg (2003)
6. Butz, M.V., Sigaud, O., Gérard, P.: Internal Models and Anticipations in Adaptive Learning Systems. In: Butz, M.V., Sigaud, O., Gérard, P. (eds.) Anticipatory Behavior in Adaptive Learning Systems. LNCS, vol. 2684, pp. 86–109. Springer, Heidelberg (2003)
7. Dietterich, T.G.: Hierarchical reinforcement learning with the MAXQ value function decomposition. Journal of Artificial Intelligence Research 13, 227–303 (2000)

8. Ding, H., Rungger, M., Stursberg, O.: Intelligent Planning of Manufacturing Systems with Hybrid Dynamics. In: IFAC Conf. on Manufacturing Modeling, Management, and Control, pp. 181–186 (2007)
9. Doya, K.: Reinforcement learning in continuous time and space. Neural Comput. 12(1), 219–245 (2000)
10. Egerstedt, M.: Behavior Based Robotics Using Hybrid Automata. In: Lynch, N.A., Krogh, B.H. (eds.) HSCC 2000. LNCS, vol. 1790, pp. 103–116. Springer, Heidelberg (2000)
11. Henzinger, T.: The Theory of Hybrid Automata. In: Proceedings of the 11th Annual IEEE Symposium on Logic in Computer Science (LICS 1996), pp. 278–292 (1996)
12. Mataric, M.J.: Reward functions for accelerated learning. In: Proc. of the 11th Int. Conf. on Machine Learning, pp. 181–189. Morgan Kaufmann, San Francisco (1994)
13. Tejas, R.: Mehta and Magnus Egerstedt. Multi-modal control using adaptive motion description languages. Automatica 44, 1912–1917 (2008)
14. Morimoto, J., Doya, K.: Acquisition of stand-up behavior by a real robot using hierarchical RL. Robotics and Autonomous Systems 36(1), 37–51 (2001)
15. Parr, R., Russell, S.: Russell Reinforcement learning with hierarchies of machines. In: Advances in Neural Information Processing Systems, vol. 10, pp. 1043–1049. The MIT Press, Cambridge (1997)
16. Pirjanian, P.: Multiple objective behavior-based control 31, 53–60 (2000)
17. Precup, D., Sutton, R.S., Singh, S.P.: Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. Artificial Intelligence 112(1-2), 181–211 (1999)
18. Rungger, M., Stursberg, O., Spanfelner, B., Leuxner, C., Sitou, W.: Efficient Planning of Autonomous Robots using Hierarchical Composition. In: 5th Int. Conf. on Informatics, Control, Automation, Robotics, pp. 262–267 (2008)
19. Mohajerian, P., Schaal, S., Ijspeert, A.: Dynamics Systems vs. Optimal Control – A Unifying View, ch. 27, pp. 425–445. Elsevier, Amsterdam (2007)
20. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press, Cambridge (1998)

Anticipatory Learning Classifier Systems and Factored Reinforcement Learning

Olivier Sigaud[1], Martin V. Butz[4], Olga Kozlova[1,2], and Christophe Meyer[3]

[1] Université Pierre et Marie Curie - Paris6
Institut des Systèmes Intelligents et de Robotique (ISIR), CNRS UMR 7222,
4 place Jussieu, F-75005 Paris, France
Olivier.Sigaud@isir.fr
[2] Thales Security Solutions & Services, Simulation
1 rue du Général de Gaulle, Osny BP 226
F95523 Cergy Pontoise Cedex, France
Olga.Kozlova@thalesgroup.com
[3] Thales Security Solutions & Services, ThereSIS Research and Innovation Office
Route départementale 128
F91767 Palaiseau Cedex, France
Christophe.Meyer@thalesgroup.com
[4] University of Würzburg
Röntgenring 11
97070 Würzburg, Germany
mbutz@psychologie.uni-wuerzburg.de

Abstract. Factored Reinforcement Learning (FRL) is a new technique to solve Factored Markov Decision Problems (FMDPs) when the structure of the problem is not known in advance. Like Anticipatory Learning Classifier Systems (ALCSs), it is a model-based Reinforcement Learning approach that includes generalization mechanisms in the presence of a structured domain. In general, FRL and ALCSs are explicit, state-anticipatory approaches that learn generalized state transition models to improve system behavior based on model-based reinforcement learning techniques. In this contribution, we highlight the conceptual similarities and differences between FRL and ALCSs, focusing on the one hand on SPITI, an instance of FRL method, and on ALCSs, MACS and XACS, on the other hand. Though FRL systems seem to benefit from a clearer theoretical grounding, an empirical comparison between SPITI and XACS on two benchmark problems reveals that the latter scales much better than the former when some combination of state variables do not occur. Based on this finding, we discuss the mechanisms in XACS that result in the better scalability and propose importing these mechanisms into FRL systems.

1 Introduction

This paper is about two classes of explicit state-anticipatory systems [1] that learn generalized state transition models to improve their behavior based on model-based reinforcement learning techniques.

G. Pezzulo et al. (Eds.): ABiALS 2008, LNAI 5499, pp. 321–333, 2009.

On the one hand, Learning Classifier Systems (LCSs) are rule-based systems where the rules (called classifiers) are learned from experience. Due to genetic algorithm-based generalization mechanisms, LCSs were shown to build compact representations of Markov Decision Problems (MDPs) and learn to behave optimally. Anticipatory Learning Classifier Systems (ALCSs) [2] deviate from this classical framework on one fundamental point. Instead of [Condition] → [Action] classifiers, they manipulate [Condition] [Action] → [Effect] classifiers, where the [Effect] part represents the expected effect of the action in all situations that match the [Condition] part of the classifier. A set of classifiers constitutes a *model of transitions*, as it is called in the Reinforcement Learning (RL) literature Thus, ALCSs are an instance of model-based RL architectures—a category of systems whose prototype is the DYNA architecture [3]. As a result, ALCSs can be seen as combining two crucial properties of RL systems: Similar to the DYNA architectures, they learn a model of transitions, which endows them with anticipation and planning capabilities and can speed up the learning process. Similar to classical LCSs, they benefit from generalization mechanisms, which enable them to build much more compact models than tabular DYNA architectures [2,4].

On the other hand, in the RL literature, the Factored Markov Decision Processes (FMDPs) framework was introduced to represent large and structured MDPs compactly [5]. In this approach, a state is implicitly described by an assignment of values to some set of state variables—a representation that shares strong similarities with the one used in LCSs where the variables are termed "attributes". But the structure of the model of transitions is assumed to be known in FMDPs, which stands in contrast with the ALCS framework where this structure is learned from experience.

SDYNA [6,7] is a family of systems that perform RL in the FMDP framework where the structure of the model of transitions is learned from experience—an approach that we call Factored Reinforcement Learning (FRL). Thus, like ALCSs, FRL systems are model-based RL systems endowed with a generalization capability.

In this contribution, we examine the conceptual similarities and differences between two ALCSs named MACS and XACS on the one hand, and one instance of SDYNA named SPITI on the other hand. Then we perform an empirical comparison between XACS and SPITI based on two benchmark problems, namely MAZE6 and BLOCKS WORLD. The comparison reveals a conceptual problem in the structured dynamic programming algorithm of SPITI, SVI, from which XACS does not suffer. As a consequence, we discuss the possibility of improving FRL systems based on XACS mechanisms.

The paper is organized as follows. In the next section, we give some background about LCSs, ALCSs, FMDPs, FRL and, in particular, SPITI. Then in Section 3, we highlight conceptual similarities and differences between SPITI, MACS and XACS. In Section 4, we present the experimental study. This comparison shows that XACS outperforms SPITI when the representation used to describe states of the problem can give rise to impossible combinations of values, which is discussed in Section 5 before concluding.

2 Background

2.1 Learning Classifier Systems

Learning Classifier Systems (LCSs) [8] were invented by Holland [9] in order to model the emergence of cognition based on adaptive mechanisms. In LCSs, knowledge is represented by a set of rules called *population of classifiers*, which is evolved by adaptive, usually evolutionary learning mechanisms. In Holland's original work, the cognitive part of the system was implemented by a list of internal messages that related the perception of an agent to its actions through an eventually complex message passing process.

Wilson published two radically simplified versions of the initial LCS architecture, named ZCS [10] and XCS [11], in which the list of internal messages was removed. These (now standard) LCSs use condition-action classifiers and combine RL methods with Genetic Algorithms (GAs) to learn a compact rule sets.

The [Condition] part of classifiers is a list of tests. There are as many tests as attributes in the problem description, each test being applied to a specific attribute. In the most common case where the test specifies a value that an attribute must take for the [Condition] to match, the test is represented just by this value. There exists a particular test, denoted "#" and called "don't care", which means that the [Condition] part of the classifier will match whatever the value of the corresponding attribute. At a more global level, the [Condition] matches if all its tests hold in the current situation. In the case of matching, the classifier may be used to determine current behavior.

2.2 Anticipatory Learning Classifier Systems

Riolo [12] was the first to publish an explicitly anticipatory LCS. His system, CFSC2, was directly inspired by the original LCS architecture of Holland [13] with internal messages.

The first ALCS designed after Wilson's simplifications of the original LCS architectures [10] was ACS [14,15]. Central to ACS, the ALP (*Anticipatory Learning Process*) algorithm is the formal counterpart of Hoffmann's psychological theory of *Anticipatory Behavioral Control* [16]. ACS was later extended by Butz to become ACS2 [17,18] and finally XACS [19]. In parallel, Gérard proposed YACS [20] and MACS [21].

The key difference between LCSs and ALCSs lies in the presence of an [Effect] part in the latter systems. In ACS, ACS2 and YACS, the [Effect] part of each classifier tells which attributes do change and which do not given a certain action is executed in a given situation. To represent this, the [Effect] part can contain a "=" symbol, which means that the corresponding attribute does not change. For instance, classifier `[#0#1] [0] [=10=]` predicts that situation `[1031]` changes into situation `[1101]` given action `[0]` is executed, while situation `[2011]` is predicted to change into `[2101]`. By contrast, MACS uses in the [Effect] part a "?" symbol, which denotes that the classifier cannot predict the value of the considered attribute. The addition of this new symbol results in the capacity to predict the value of each attribute separately at the next time step.

2.3 Factored Markov Decision Processes

The FMDP framework was invented independently from research on LCSs, but it is based on an equivalent formalism. Indeed, an FMDP is described by a set of state variables $S = \{X_i ...., X_n\}$, where each X_i takes value in a finite domain $Dom(X_i)$. A state $s \in S$ assigns a value $x_i \in Dom(X_i)$ to each state variable X_i. These variables are the formal counterpart of attributes in the LCS framework.

FMDPs utilize dependencies between variables, defined using Dynamic Bayesian Networks (DBNs) [22], to compactly represent the transition and reward functions of structured MDPs.

The model of the transition of the FMDP is defined by a separate DBN model $T_a = \langle G_a, \{P^a_{X_1}, \ldots, P^a_{X_n}\}\rangle$ for each action a. G_a is a two-layer directed acyclic graph whose nodes are $\{X_1, \ldots, X_n, X'_1, \ldots, X'_n\}$ with X_i a variable at time t and X'_i the same variable at time $t+1$. The parents of X'_i are denoted $\text{Parents}_a(X'_i)$ with $\text{Parents}_a(X'_i) \subseteq X$. The transition model T_a is quantified by *Conditional Probability Distributions* (CPDs), denoted $P^a_{X_i}(X'_i|\text{Parents}_a(X'_i))$, associated to each node $X'_i \in G_a$. In practice, these CPDs can be represented as tables, as rules, as a set of decision trees, or as decision diagrams. In each case, the representation gives the probability distribution of each X'_i given the values of $\text{Parents}_a(X'_i)$. In the case of rules or tables, the generalization property comes from the fact that only the variables belonging to $\text{Parents}_a(X'_i)$ are used to represent the distribution over X'_i. This corresponds to using a "#" for the attributes that correspond to all other variables in the LCS representation.

Given this representation of the transition function and a similar compact representation of the reward function, different dynamic programming algorithms such as SVI and SPI for trees [23] and SPUDD for decision diagrams [24] were shown to converge to the optimal policy [23] while using a representation that is exponentially smaller than the tabular one.

2.4 Factored Reinforcement Learning and SPITI

In FMDPs, the transition function expressed as a set of CPDs is considered known. But for most complex problems, designing these probability distributions by hand is difficult, if not impossible. And to represent them compactly makes things even more difficult.

An alternative consists in learning from experience a model of the transition function under a compact form. If learning the model and dynamic programming backups are performed simultaneously, then this approach is the structured counterpart of *indirect* RL systems, whose prototype is the DYNA architecture.

This insight led to the design of SDYNA as a structured version of the DYNA architecture where the model of transitions and of the reward are learned from experience under a compact form [7]. SPITI is a particular instance of SDYNA. It uses an incremental version of SVI to perform dynamic programming and learns the model of transitions in the form of a collection of decision trees using the Incremental Tree Induction (ITI) algorithm [25].

3 Systems and Comparisons

3.1 Comparing SPITI with MACS

Both MACS and SPITI call upon a model-based RL process and are endowed with a generalization property that makes them able to address large MDPs without prior knowledge of the structure. Furthermore, their representations of the model of the transitions have a similar structure. Indeed, consider an agent in a grid world that perceives whether the eight surrounding cells (starting North and coding clockwise) contain a wall or not (see Figure 1). The formalism in ACS, ACS2, XACS and YACS is able to represent regularities such as *"when the agent perceives a wall to the north, whatever it perceives in any other direction, going north does not produce any sensory change"*, which may be represented by the following classifier if the first attribute corresponds to the value of the North sensor: `[1########] [North] [========]`. By contrast, MACS can represent regularities between different attributes with a classifier such as `[1#########] [Left] [??1??????]`, stating *"when the agent perceives a wall to the north, and turns left, it will perceive a wall on its right"*.

Thus, on the one hand, MACS can represent additional regularities since it can detect regularities between different attributes. However, on the other hand, it only predicts one attribute at a time, whereas the predictions of other ALCSs can be more compact.

Experimental results on model compactness and convergence speed of MACS in grid worlds have shown that it builds a slightly more compact model than YACS, which itself was building models four times more compact than an early version of ACS [21]. Furthermore, MACS was building this model three times faster than YACS, and nine times faster than the early version of ACS counting the number of iterations.

Interestingly, its unique representation of the [Effect] part makes MACS more similar to SPITI than any other ALCS. Indeed, in MACS, the value of each attribute is anticipated separately for each action as a function of a [Condition] part containing variables defining the previous state of the model whereas in SPITI the value of each state variable is anticipated separately for each action as a function of a tree representing the possible combinations of variables defining all previous states of the model. Thus, one classifier in MACS is similar to one branch in the decision tree in the model of transitions of SPITI.

Moreover, MACS is the only ALCS that does not call upon a GA. Instead, to learn the model of transitions it relies on the combination of generalization and specialization heuristics that collaborate to converge towards a compact and accurate model of transitions. This mechanism can be compared more easily with the ITI algorithm used in SPITI, which relies on the χ^2 information metric to grow a decision tree incrementally.

However, beyond these similarities, MACS and SPITI differ in several points. First, MACS represents a deterministic transition model whereas SPITI models a stochastic process through a distribution of probabilities of transition. Secondly, as stated above, building a compact model of the transition function in MACS

relies on a complex combination of heuristics whereas SPITI calls upon the well established ITI algorithm. Thirdly, the model of transitions in MACS is represented as a set of classifiers and the value function is tabular, whereas SPITI implements the model of transitions, the value function and the policy as decision trees. This results in faster algorithmic information access. Finally, and most importantly, in MACS the dynamic programming component of model-based RL is applied to a tabular representation of states, whereas SPITI calls upon SVI to perform this computation compactly—with guarantees of convergence to optimality as far as the model of transitions is perfectly accurate. This computation is very efficient in practice.

All the differences above speak in favor of SPITI that seems mathematically better grounded than MACS and benefits from efficient algorithms. Thus an empirical comparison seems to be pointless. As a matter of fact, we did not perform any experiments comparing SPITI against MACS, since SPITI was shown to perform well on problems that are out of reach of MACS [6,7].

Among these differences, the most crucial one is the fact that MACS does not generalize the models of the reward and the value functions over states. Instead, these models are represented by a table giving a value for each encountered state, which prevents its usage for very large state space problems. Although it shares less similarities with SPITI, XACS is another ALCS that does not suffer from this crucial problem. And, quite interestingly, the experimental comparison that we perform after presenting XACS below reveals that it is endowed with a key property that makes it more efficient than SPITI in the context of large problems where a lot of combinations of state variable values cannot occur.

3.2 Presentation of XACS

The XACS system was developed to overcome the deficiency of not generalizing the value function estimates in MACS [19]. XACS combines two LCSs—the generalizing state transition learner ACS2 [18] and the generalizing function learner XCS, which learns generalized value function estimates in XACS. It was shown that XACS can be robustly applied to blocks world problems, in which previous overgeneralization issues in ACS2 were overcome [19].

Essentially, ACS2 learns a generalized representation of the encountered state-transition function of a problem. It has been shown to reliably learn in various discrete problem domains, being able to ignore irrelevant perceptual attributes, handling noisy inputs, or stochastic state transitions. Knowledge is represented in the aforementioned [Condition], [Action] → [Effect] rules. The rules are learned by a combination of a heuristic, which specializes the rule structures, and a genetic rule generalization mechanism.

The XCS system may be the most well-understood and used LCS to-date. It has been shown to be efficiently applicable in Boolean function problems, real-valued function problems, reinforcement learning problems, and mixed domains including datamining classification [11,26]. XCS learns based on a combination of gradient-based value approximation and genetic algorithm-based rule structure

learning. In combination with ACS2, XCS learns a generalized representation of the state-value function of the encountered reinforcement learning problem. In this case, value approximations are updated similar to the DYNA architecture [19,3].

During learning, XCS and ACS2 create their initial rules by means of a *covering* mechanism, which creates rules with matching conditions given no rules currently match the perceived problem state. For compaction purposes, both systems represent redundant identical rules in one *macro-classifier* rule [11]. The rule structuring mechanisms of either system basically assure that the whole perceived problem space is covered and rules that cover unsampled problem subspaces are forgotten (deleted) over time. Further details on the involved mechanisms as well as theoretic learning bounds can be found in the literature [19,26,27].

During goal-directed behavior, XACS predicts possible next problem states using the model from its ACS2 component, estimates the values of these anticipated states by means of its XCS component, and finally conducts its behavioral decision based on these estimates.

4 Experimental Study

4.1 Maze6

The maze environments are classical LCS benchmark problems. They are represented by a two-dimensional grid. Each cell can be occupied by an obstacle, denoted as attribute value by the character `'O'`, a food item, denoted by `'F'`, or can be empty, denoted by `'.'`. The animat perceives its immediate surrounding starting with the cell to the north and coding clockwise. Thus, the perceptual space in the maze environment $\mathcal{I}_{maze} \subseteq \{., O, F\}^L$ where $L = 8$, the eight adjacent cells. Figure 1 shows MAZE6, one of such standard mazes. For example, an animat located one position below the food perceives `'FOOO..OO'` whereas an animat located at the lower left corner perceives `'.O.OOOOO'`. The simulated animat possesses eight primitive actions, the movements to the eight adjacent cells (i.e. $\mathcal{A}_{maze} = \{N, NE, E, SE, S, SW, W, NW\}$). If a movement leads to a position that is blocked by an obstacle, the action has no effect. Once the food position is entered, the environment provides a reinforcement of 1000 and one

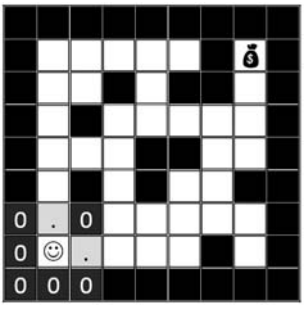


Fig. 1. Maze6

trial ends. In that case, the animat is repositioned to a randomly chosen empty spot in the maze and tries again. Note that, while the state is observed through attributes or random variables, giving rise to an FMDP representation, MAZE6 still obeys the Markov property, that is, the current state perceptions suffice to uniquely identify the current state of the agent, and the knowledge of that state and the action suffice to determine the distribution over next states, by contrast with what would happen in a Partially Observable MDP.

4.2 Blocks World Problem

Our second benchmark is a blocks world scenario introduced in [2]. In this problem, b blocks are distributed over a certain number of stacks s. The agent can manipulate the stacks by the means of a gripper that can either grip or release a block on a certain stack. It perceives the current block distribution coding each stack with b attributes. One additional attribute indicates if the gripper is currently holding a block. Thus, the perceivable situations are a subset of $\mathcal{I} \subset \{*, b\}^{bs+1}$. Additionally, the problem is defined by a particular goal state. We define the goal by putting a particular number y of blocks on the first stack. Figure 2 (right-hand side) shows the goal in the problem with $b = 4, s = 3, y = 3$.

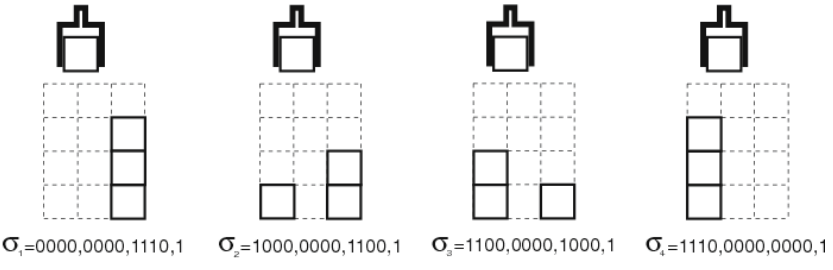
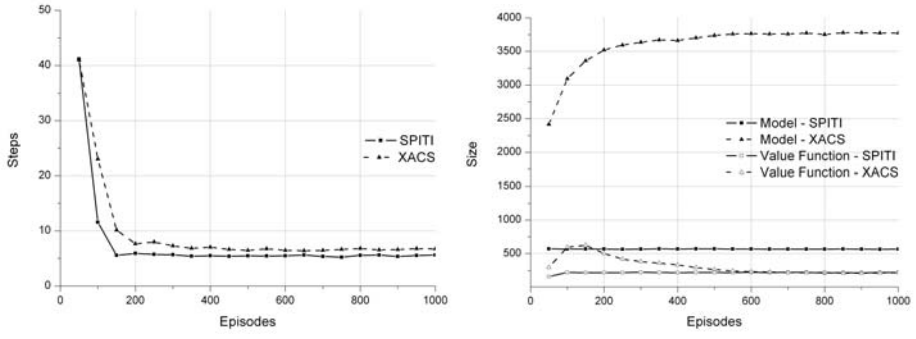


Fig. 2. A blocks world scenario, from a random initial position (left-hand side) to the goal position (right-hand side)

4.3 Experiments

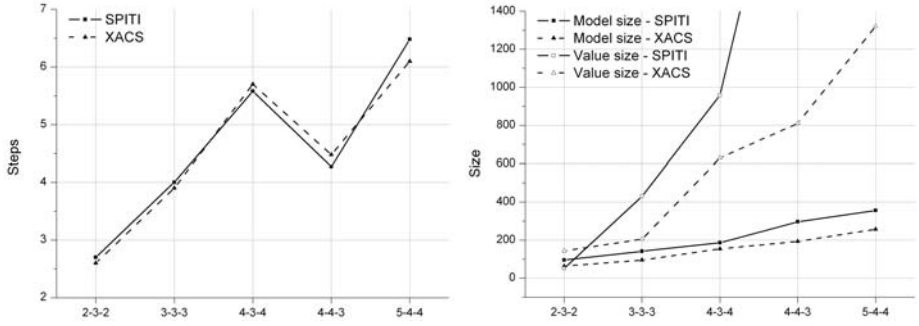
Our experimental protocol is the following. In all runs, we alternate one episode of pure exploration with a random policy and one episode of pure exploitation based on the learned policy. Learning is turned off during exploitation runs. In both benchmark problems, each episode is limited to 50 steps. All results presented below are averaged over 10 runs.

We compare the performance and size of the models in XACS and SPITI. In XACS, the size of the model corresponds to the number of macro-classifiers, for the value function as well as for the model of transitions. In SPITI, it corresponds to the number of branches in the value tree and trees representing the model of transitions.



(a) *(b)*

Fig. 3. *(a)*:Performance in MAZE6 *(b)*:Size of the models



(a) *(b)*

Fig. 4. *(a)*:Performance in BLOCKS WORLD *(b)*:Size of the models

In the case of MAZE6, Figure 3 shows the averaged performance and size of the models from episode to episode.

In the case of BLOCKS WORLD, Figure 4 shows the averaged performance and size of the models in problems with an increasing size, so as to compare the scaling capabilities of both algorithms. In that case, the performance and size for each problem is measured after 200 episodes (alternating 100 exploration episodes and 100 exploitation episodes).

An analysis of Figure 3 shows that, in the case of MAZE6, SPITI slightly outperforms XACS while building a much more compact model of transitions and a model of the value function of similar size after convergence.

By contrast, the analysis of Figure 4 shows that, even if SPITI performs comparably to XACS for small BLOCKS WORLD problems, its model size scales much worse. Thus, XACS can deal with much larger problems than SPITI. The explosion of the size of the model in SPITI also resulted in a much slower computation of the optimal policy.

5 Discussion

The results show that SPITI outperforms XACS on MAZE6 and it performs similar to XACS on small BLOCKS WORLD problems. This suggests that given their clearer mathematical background the basic algorithms in SPITI are intrinsically at least as efficient as the combination of heuristics in XACS. However, the fact that XACS outperforms SPITI on larger BLOCKS WORLD problems and scales much better on these problems reveals a conceptual problem in SPITI that XACS does not suffer from.

The problem is about the representation of "impossible states". In the case of BLOCKS WORLD with the binary representation we used, many arbitrary combinations of attribute values correspond to states that can be represented by both formalisms but that do not occur in practice: all states where a block is lying "in the air" (that is, neither on another block nor on the table) and all states that denote the presence of more or less than b blocks are impossible. The more empty cells in the problem, the more such impossible states. In SPITI, the model of the value function ends with representing explicitly a lot of these impossible states. A closer examination of the algorithms reveals that this undesirable property is inherited from SVI itself, the structured dynamic programming algorithm used in SPITI.

Indeed, in SVI the probabilities of transitions over each variables are computed separately. Thus, the information about the possible or impossible co-occurrences of values of such variables is lost. In the structured Bellman regression algorithm used in SVI, nothing prevents the expression of impossible states in the value function, although these states do not occur in practice. To our knowledge, this fact has never been noticed or made explicit in the literature—seeing also that the benchmark problems used to present structured dynamic programming algorithms are free of such impossible states.

By contrast, XACS benefits from several generalization biases that restrain a possible tendency to represent such impossible states:

- the classifier population is limited in size, resulting in a compactness pressure;
- the covering operator favors the creation of classifier that correspond to actually encountered states rather than impossible ones;
- the genetic-based generalization assures coverage of sufficiently frequently sampled states but also enforces the deletion of rules that cover unsampled subspaces.

In this way, XACS tends to cover the encountered subspace manifold of the full representational space as compactly as possible based on several occurrence and validity signals that are received by means of (random) problem space sampling.

Moreover, due to its interactive specialization and generalization mechanism, XACS identifies the action-dependent state transitions with maximally compact representations. While the specialization mechanism includes seemingly relevant state attributes heuristically, the genetic generalization mechanism deletes over-specializations. While researches might hesitate to utilize the evolutionary-inspired mechanisms used in XACS, comparisons to statistical approaches show

similar functionality and scalability [28,29,30]. Thus, drawing inspiration from XACS suggests to include state occurrence estimates and state relevance estimates, where the latter are based on prediction accuracy estimates, in order to approximate the FMDP model more compactly and efficiently while still sufficiently accurately.

Nevertheless, one must not forget that the model of transitions built by XACS differs from the model in SPITI since XACS calls upon the "=" symbol and predicts several attributes simultaneously whereas MACS uses the "?" symbol and anticipates one attribute at a time, like SPITI. In that respect, on the one hand, a more straightforward SPITI and an ALCS should be the comparison of SPITI with an ideal combination of XACS and MACS– which does not exist so far. On the other hand, trying to figure out whether it is possible to anticipate several attributes at a time within the FRL framework might result in interesting insights.

6 Conclusion

The goal of this contribution was to show that, although ALCSs and FRL systems such as SPITI are conceptually very similar and share interesting properties, they also show some important differences that have major consequences on their algorithmic properties and their performance. By means of an empirical comparison, we have discovered that the structured dynamic programming algorithm, which lies at the heart of one of the main FRL systems, SPITI, suffers from a conceptual problem that prevents it from scaling as efficiently as XACS does—the most efficient ALCS currently available. Future work will need to fix this conceptual problem possibly drawing inspiration from the mechanisms employed in XACS as discussed above.

References

1. Butz, M.V., Sigaud, O., Gérard, P.: Anticipatory behavior: Exploiting knowledge about the future to improve current behavior. In: Butz, M.V., Sigaud, O., Gérard, P. (eds.) Anticipatory Behavior in Adaptive Learning Systems. LNCS, vol. 2684, pp. 1–10. Springer, Heidelberg (2003)
2. Butz, M.V.: Anticipatory Learning Classifier Systems. Kluwer Academic Publishers, Boston (2002)
3. Sutton, R.S.: Planning by incremental dynamic programming. In: Proceedings of the Eighth International Conference on Machine Learning, pp. 353–357. Morgan Kaufmann, San Mateo (1990)
4. Gérard, P., Sigaud, O.: Designing efficient exploration with MACS: Modules and function approximation. In: Cantú-Paz, E., Foster, J.A., Deb, K., Davis, L., Roy, R., O'Reilly, U.-M., Beyer, H.-G., Kendall, G., Wilson, S.W., Harman, M., Wegener, J., Dasgupta, D., Potter, M.A., Schultz, A., Dowsland, K.A., Jonoska, N., Miller, J., Standish, R.K. (eds.) GECCO 2003. LNCS, vol. 2723, pp. 1882–1893. Springer, Heidelberg (2003)
5. Boutilier, C., Dearden, R., Goldszmidt, M.: Exploiting structure in policy construction. In: Proceedings of the 14th International Joint Conference in Artificial Intelligence, pp. 1104–1111 (1995)

6. Degris, T., Sigaud, O., Wuillemin, P.H.: Chi-square tests driven method for learning the structure of factored MDPs. In: Proceedings of the 22nd Conference on Uncertainty in Artificial Intelligence, Massachusetts Institute of Technology, Cambridge, pp. 122–129. AUAI Press (2006)
7. Degris, T., Sigaud, O., Wuillemin, P.H.: Learning the structure of factored markov decision processes in reinforcement learning problems. In: Proceedings of the 23rd International Conference in Machine Learning, pp. 257–264. ACM, Pittsburgh (2006)
8. Sigaud, O., Wilson, S.W.: Learning Classifier Systems: a survey. Journal of Soft Computing 11(11), 1065–1078 (2007)
9. Holland, J.H.: Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence. University of Michigan Press, Ann Arbor (1975)
10. Wilson, S.W.: ZCS, a Zeroth level Classifier System. Evolutionary Computation 2(1), 1–18 (1994)
11. Wilson, S.W.: Classifier Fitness Based on Accuracy. Evolutionary Computation 3(2), 149–175 (1995)
12. Riolo, R.L.: Lookahead planning and latent learning in a Classifier System. In: Meyer, J.A., Wilson, S.W. (eds.) From animals to animats: Proceedings of the First International Conference on Simulation of Adaptative Behavior, pp. 316–326. MIT Press, Cambridge (1991)
13. Holland, J.H., Reitman, J.S.: Cognitive Systems based on Adaptive Algorithms. Pattern Directed Inference Systems 7(2), 125–149 (1978)
14. Stolzmann, W.: Anticipatory Classifier Systems. In: Koza, J., Banzhaf, W., Chellapilla, K., Deb, K., Dorigo, M., Fogel, D.B., Garzon, M.H., Goldberg, D.E., Iba, H., Riolo, R. (eds.) Proceedings of the 1998 Genetic and Evolutionary Computation Conference, pp. 658–664. Morgan Kaufmann Publishers, Inc., San Francisco (1998)
15. Butz, M.V., Goldberg, D.E., Stolzmann, W.: Introducing a genetic generalization pressure to the Anticipatory Classifier Systems part I: Theoretical approach. In: Proceedings of the 2000 Genetic and Evolutionary Computation Conference (GECCO 2000), pp. 34–41 (2000)
16. Hoffmann, J.: Vorhersage und Erkenntnis [Anticipation and Cognition]. Hogrefe, Göttingen (1993)
17. Butz, M.V.: An Algorithmic Description of ACS2. In: Lanzi, P.L., Stolzmann, W., Wilson, S.W. (eds.) IWLCS 2001. LNCS, vol. 2321, pp. 211–229. Springer, Heidelberg (2002)
18. Butz, M.V., Goldberg, D.E., Stolzmann, W.: The Anticipatory Classifier System and Genetic Generalization. Natural Computing 1(4), 427–467 (2002)
19. Butz, M.V., Goldberg, D.E.: Generalized state values in an anticipatory Learning Classifier System. In: Butz, M.V., Sigaud, O., Gérard, P. (eds.) Anticipatory Behavior in Adaptive Learning Systems. LNCS (LNAI), vol. 2684, pp. 282–301. Springer, Heidelberg (2003)
20. Gérard, P., Stolzmann, W., Sigaud, O.: YACS: a new Learning Classifier System with Anticipation. Journal of Soft Computing: Special Issue on Learning Classifier Systems 6(3-4), 216–228 (2002)
21. Gérard, P., Meyer, J.A., Sigaud, O.: Combining latent learning with dynamic programming in MACS. European Journal of Operational Research 160, 614–637 (2005)
22. Dean, T., Kanazawa, K.: A Model for Reasoning about Persistence and Causation. Computational Intelligence 5, 142–150 (1989)

23. Boutilier, C., Dearden, R., Goldszmidt, M.: Stochastic dynamic programming with factored representations. Artificial Intelligence 121(1-2), 10–49 (2000)
24. Hoey, J., St-Aubin, R., Hu, A., Boutilier, C.: SPUDD: Stochastic Planning using Decision Diagrams. In: Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence, pp. 279–288. Morgan Kaufmann, San Francisco (1999)
25. Utgoff, P.E.: Incremental induction of decision trees. Machine Learning 4, 161–186 (1989)
26. Butz, M.V.: Rule-Based Evolutionary Online Learning Systems: A Principled Approach to LCS Analysis and Design. Springer, Heidelberg (2006)
27. Butz, M., Kovacs, T., Lanzi, P.L., Wilson, S.W.: Toward a theory of generalization and learning in XCS. IEEE Transactions on Evolutionary Computation 8(1), 28–46 (2004)
28. Butz, M.V., Lanzi, P.L., Wilson, S.W.: Function approximation with XCS: Hyperellipsoidal conditions, recursive least squares, and compaction. IEEE Transactions on Evolutionary Computation 12, 355–376 (2008)
29. Potts, D.: Incremental learning of linear model trees. In: Proceedings of the Twenty-First International Conference on Machine Learning (ICML 2004), pp. 663–670 (2004)
30. Schaal, S., Atkeson, C.G.: Constructive incremental learning from only local information. Neural Computation 10, 2047–2084 (1998)

Author Index