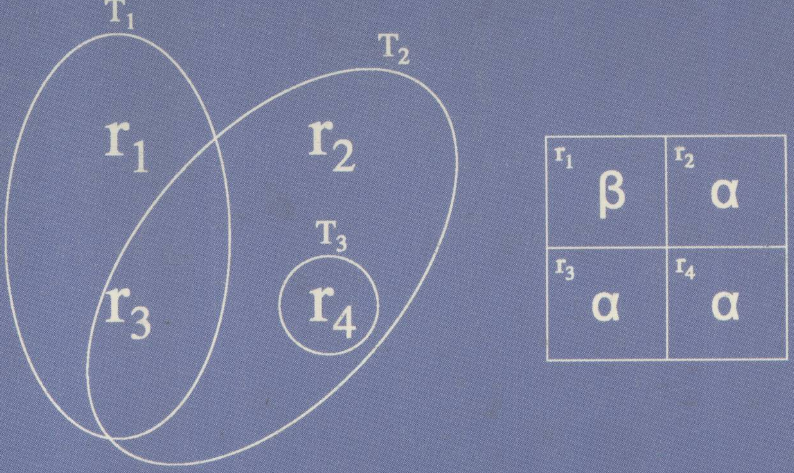
Hot Topics

LNAI 2969

Matthias Nickles
Michael Rovatsos
Gerhard Weiss (Eds.)

Agents and Computational Autonomy

Potential, Risks, and Solutions



Springer

Matthias Nickles Michael Rovatsos
Gerhard Weiss (Eds.)

Agents and Computational Autonomy

Potential, Risks, and Solutions

Springer

Series Editors

Jaime G. Carbonell, Carnegie Mellon University, Pittsburgh, PA, USA
Jörg Siekmann, University of Saarland, Saarbrücken, Germany

Volume Editors

Matthias Nickles
Michael Rovatsos
Gerhard Weiss
Technical University of Munich
Department of Informatics
Boltzmannstr. 3, 85748 Garching bei München, Germany
E-mail: {nickles, rovatsos, weissg}@cs.tum.edu

Library of Congress Control Number: 2004109920

CR Subject Classification (1998): I.2.11, C.2.4, I.2, D.2

ISSN 0302-9743
ISBN 3-540-22477-7 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springeronline.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 11301097 06/3142 5 4 3 2 1 0

Lecture Notes in Artificial Intelligence 2969

Edited by J. G. Carbonell and J. Siekmann

Subseries of Lecture Notes in Computer Science

Preface

This volume contains the postproceedings of the 1st International Workshop on Computational Autonomy – Potential, Risks, Solutions (AUTONOMY 2003), held at the 2nd International Joint Conference on Autonomous Agents and Multi-agent Systems (AAMAS 2003), July 14, 2003, Melbourne, Australia. Apart from revised versions of the accepted workshop papers, we have included invited contributions from leading experts in the field. With this, the present volume represents the first comprehensive survey of the state-of-the-art of research on autonomy, capturing different theories of autonomy, perspectives on autonomy in different kinds of agent-based systems, and practical approaches to dealing with agent autonomy.

Agent orientation refers to a software development perspective that has evolved in the past 25 years in the fields of computational agents and multiagent systems. The basic notion underlying this perspective is that of a computational agent, that is, an entity whose behavior deserves to be called flexible, social, and autonomous. As an autonomous entity, an agent possesses action choice and is at least to some extent capable of deciding and acting under self-control. Through its emphasis on autonomy, agent orientation significantly differs from traditional engineering perspectives such as structure orientation or object orientation. These perspectives are targeted on the development of systems whose behavior is fully determined and controlled by external units (e.g., by a programmer at design time and/or a user at run time), and thus inherently fail to capture the notion of autonomy.

To date autonomy is still a poorly understood property of computational systems, both in theoretical and practical terms, and among all properties usually associated with agent orientation it is this property that is being most controversially discussed. On the one hand, it is argued that there is a broad range of applications in complex domains such as e/m-commerce, ubiquitous computing, and supply chain management which can hardly be realized without taking autonomy as a key ingredient, and that it is first of all agent autonomy which enables the decisive features of agent-oriented software, namely robustness, flexibility and the emergence of novel solutions of problems at run time. On the other hand, it is argued that autonomy is mainly a source of undesirable and chaotic system behavior. Obviously, without a clarification of these two positions, it is unlikely that agent orientation and agent-oriented systems (having "autonomy" as a real property and not just as a catchy label) will become broadly accepted in real-world, industrial and commercial applications.

The AUTONOMY 2003 workshop was the first workshop organized to discuss different definitions and views of autonomy, to analyze the potential and the risks of computational autonomy, and to suggest solutions for the various issues raised by computational autonomy.

Acknowledgements

We would like to express our gratitude to the authors of this volume for their contributions and to the workshop participants for inspiring discussions, as well as to the members of the Steering Committee, the Programme Committee, and the additional reviewers of AUTONOMY 2003 for their reviews and for their overall support for the workshop.

Special thanks also go to Simon Parsons, the AAMAS 2003 Workshops Chair, for his great support.

April 2004 Matthias Nickles, Michael Rovatsos, Gerhard Weiß

Co-chairs

Matthias Nickles (Technical University Munich, Germany)
Michael Rovatsos (Technical University Munich, Germany)

Steering Committee

Frank Dignum (Utrecht University, The Netherlands)
Michael Luck (University of Southampton, UK)
Carles Sierra (IIIA-CSIC, Barcelona, Spain)
Milind Tambe (University of Southern California, Marina del Rey, USA)
Gerhard Weiß (Technical University Munich, Germany)
Mike Wooldridge (University of Liverpool, Liverpool, UK)

Programme Committee

Eduardo Alonso (City University, London, UK)
Richard Alterman (Brandeis University, Waltham, USA)
K. Suzanne Barber (University of Texas, Austin, USA)
Bernard Burg (HP Research, Palo Alto, USA)
Cristiano Castelfranchi (National Research Council, Rome, Italy)
Yves Demazeau (Laboratoire Leibniz – Institut IMAG, Grenoble, France)
Piotr J. Gmytrasiewicz (University of Illinois, Chicago, USA)
Mark d'Inverno (University of Westminster, London, UK)
Catholijn Jonker (Vrije Universiteit Amsterdam, The Netherlands)
Barbara Keplicz (Warsaw University, Poland)
Matthias Klusch (DFKI, Saarbrücken, Germany)
Jiming Liu (Hong Kong Baptist University, Hong Kong)
Jörg P. Müller (Siemens AG, München, Germany)
Olga Pacheco (University of Minho, Portugal)
Lynne Parker (University of Tennessee, Knoxville, USA)
Van Parunak (ERIM, Ann Arbor, USA)
Michael Schillo (DFKI, Saarbrücken, Germany)
Frank Schweitzer (Fraunhofer Institute, Sankt Augustin, Germany)
Munindar P. Singh (North Carolina State University, Raleigh, USA)
Liz Sonenberg (University of Melbourne, Australia)
Leon Sterling (University of Melbourne, Australia)
Harko Verhagen (Royal Institute of Technology, Stockholm, Sweden)
Hans Weigand (Tilburg University, The Netherlands)
Steven Willmott (University of Catalonia, Barcelona, Spain)
Franco Zambonelli (University of Modena and Reggio Emilia, Italy)

Additional Reviewers

Jisun Park
Josh Introne
Alex Feinman

Lecture Notes in Artificial Intelligence (LNAI)

Vol. 3157: C. Zhang, H. W. Guesgen, W.K. Yeap (Eds.), PRICAI 2004: Trends in Artificial Intelligence. XX, 1023 pages. 2004.

Vol. 3139: F. Iida, R. Pfeifer, L. Steels, Y. Kuniyoshi (Eds.), Embodied Artificial Intelligence. IX, 331 pages. 2004.

Vol. 3131: V. Torra, Y. Narukawa (Eds.), Modeling Decisions for Artificial Intelligence. XI, 327 pages. 2004.

Vol. 3127: K.E. Wolff, H.D. Pfeiffer, H.S. Delugach (Eds.), Conceptual Structures at Work. XI, 403 pages. 2004.

Vol. 3123: A. Belz, R. Evans, P. Piwek (Eds.), Natural Language Generation. X, 219 pages. 2004.

Vol. 3120: J. Shawe-Taylor, Y. Singer (Eds.), Learning Theory. X, 648 pages. 2004.

Vol. 3097: D. Basin, M. Rusinowitch (Eds.), Automated Reasoning. XII, 493 pages. 2004.

Vol. 3071: A. Omicini, P. Petta, J. Pitt (Eds.), Engineering Societies in the Agents World. XIII, 409 pages. 2004.

Vol. 3070: L. Rutkowski, J. Siekmann, R. Tadeusiewicz, L.A. Zadeh (Eds.), Artificial Intelligence and Soft Computing - ICAISC 2004. XXV, 1208 pages. 2004.

Vol. 3068: E. André, L. Dybkjær, W. Minker, P. Heisterkamp (Eds.), Affective Dialogue Systems. XII, 324 pages. 2004.

Vol. 3067: M. Dastani, J. Dix, A. El Fallah-Seghrouchni (Eds.), Programming Multi-Agent Systems. X, 221 pages. 2004.

Vol. 3066: S. Tsumoto, R. Słowiński, J. Komorowski, J.W. Grzymała-Busse (Eds.), Rough Sets and Current Trends in Computing. XX, 853 pages. 2004.

Vol. 3065: A. Lomuscio, D. Nute (Eds.), Deontic Logic in Computer Science. X, 275 pages. 2004.

Vol. 3060: A.Y. Tawfik, S.D. Goodwin (Eds.), Advances in Artificial Intelligence. XIII, 582 pages. 2004.

Vol. 3056: H. Dai, R. Srikant, C. Zhang (Eds.), Advances in Knowledge Discovery and Data Mining. XIX, 713 pages. 2004.

Vol. 3055: H. Christiansen, M.-S. Hacid, T. Andreasen, H.L. Larsen (Eds.), Flexible Query Answering Systems. X, 500 pages. 2004.

Vol. 3040: R. Conejo, M. Urretavizcaya, J.-L. Pérez-de-la-Cruz (Eds.), Current Topics in Artificial Intelligence. XIV, 689 pages. 2004.

Vol. 3035: M.A. Wimmer (Ed.), Knowledge Management in Electronic Government. XII, 326 pages. 2004.

Vol. 3034: J. Favela, E. Menasalvas, E. Chávez (Eds.), Advances in Web Intelligence. XIII, 227 pages. 2004.

Vol. 3030: P. Giorgini, B. Henderson-Sellers, M. Winikoff (Eds.), Agent-Oriented Information Systems. XIV, 207 pages. 2004.

Vol. 3029: B. Orchard, C. Yang, M. Ali (Eds.), Innovations in Applied Artificial Intelligence. XXI, 1272 pages. 2004.

Vol. 3025: G.A. Vouros, T. Panayiotopoulos (Eds.), Methods and Applications of Artificial Intelligence. XV, 546 pages. 2004.

Vol. 3020: D. Polani, B. Browning, A. Bonarini, K. Yoshida (Eds.), RoboCup 2003: Robot Soccer World Cup VII. XVI, 767 pages. 2004.

Vol. 3012: K. Kurumatani, S.-H. Chen, A. Ohuchi (Eds.), Multi-Agnets for Mass User Support. X, 217 pages. 2004.

Vol. 3010: K.R. Apt, F. Fages, F. Rossi, P. Szeredi, J. Váncza (Eds.), Recent Advances in Constraints. VIII, 285 pages. 2004.

Vol. 2990: J. Leite, A. Omicini, L. Sterling, P. Torroni (Eds.), Declarative Agent Languages and Technologies. XII, 281 pages. 2004.

Vol. 2980: A. Blackwell, K. Marriott, A. Shimojima (Eds.), Diagrammatic Representation and Inference. XV, 448 pages. 2004.

Vol. 2977: G. Di Marzo Serugendo, A. Karageorgos, O.F. Rana, F. Zambonelli (Eds.), Engineering Self-Organising Systems. X, 299 pages. 2004.

Vol. 2972: R. Monroy, G. Arroyo-Figueroa, L.E. Sucar, H. Sossa (Eds.), MICAI 2004: Advances in Artificial Intelligence. XVII, 923 pages. 2004.

Vol. 2969: M. Nickles, M. Rovatsos, G. Weiss (Eds.), Agents and Computational Autonomy. X, 275 pages. 2004.

Vol. 2961: P. Eklund (Ed.), Concept Lattices. IX, 411 pages. 2004.

Vol. 2953: K. Konrad, Model Generation for Natural Language Interpretation and Analysis. XIII, 166 pages. 2004.

Vol. 2934: G. Lindemann, D. Moldt, M. Paolucci (Eds.), Regulated Agent-Based Social Systems. X, 301 pages. 2004.

Vol. 2930: F. Winkler (Ed.), Automated Deduction in Geometry. VII, 231 pages. 2004.

Vol. 2926: L. van Elst, V. Dignum, A. Abecker (Eds.), Agent-Mediated Knowledge Management. XI, 428 pages. 2004.

Vol. 2923: V. Lifschitz, I. Niemelä (Eds.), Logic Programming and Nonmonotonic Reasoning. IX, 365 pages. 2004.

Vol. 2915: A. Camurri, G. Volpe (Eds.), Gesture-Based Communication in Human-Computer Interaction. XIII, 558 pages. 2004.

Vol. 2913: T.M. Pinkston, V.K. Prasanna (Eds.), High Performance Computing - HiPC 2003. XX, 512 pages. 2003.

Vol. 2903: T.D. Gedeon, L.C.C. Fung (Eds.), AI 2003: Advances in Artificial Intelligence. XVI, 1075 pages. 2003.

Vol. 2902: F.M. Pires, S.P. Abreu (Eds.), Progress in Artificial Intelligence. XV, 504 pages. 2003.

Vol. 2892: F. Dau, The Logic System of Concept Graphs with Negation. XI, 213 pages. 2003.

Vol. 2891: J. Lee, M. Barley (Eds.), Intelligent Agents and Multi-Agent Systems. X, 215 pages. 2003.

Vol. 2882: D. Veit, Matchmaking in Electronic Markets. XV, 180 pages. 2003.

Vol. 2871: N. Zhong, Z.W. Raś, S. Tsumoto, E. Suzuki (Eds.), Foundations of Intelligent Systems. XV, 697 pages. 2003.

Vol. 2854: J. Hoffmann, Utilizing Problem Structure in Planing. XIII, 251 pages. 2003.

Vol. 2843: G. Grieser, Y. Tanaka, A. Yamamoto (Eds.), Discovery Science. XII, 504 pages. 2003.

Vol. 2842: R. Gavaldá, K.P. Jantke, E. Takimoto (Eds.), Algorithmic Learning Theory. XI, 313 pages. 2003.

Vol. 2838: N. Lavrač, D. Gamberger, L. Todorovski, H. Blockeel (Eds.), Knowledge Discovery in Databases: PKDD 2003. XVI, 508 pages. 2003.

Vol. 2837: N. Lavrač, D. Gamberger, L. Todorovski, H. Blockeel (Eds.), Machine Learning: ECML 2003. XVI, 504 pages. 2003.

Vol. 2835: T. Horváth, A. Yamamoto (Eds.), Inductive Logic Programming. X, 401 pages. 2003.

Vol. 2821: A. Günter, R. Kruse, B. Neumann (Eds.), KI 2003: Advances in Artificial Intelligence. XII, 662 pages. 2003.

Vol. 2807: V. Matoušek, P. Mautner (Eds.), Text, Speech and Dialogue. XIII, 426 pages. 2003.

Vol. 2801: W. Banzhaf, J. Ziegler, T. Christaller, P. Dittrich, J.T. Kim (Eds.), Advances in Artificial Life. XVI, 905 pages. 2003.

Vol. 2797: O.R. Zaïane, S.J. Simoff, C. Djeraba (Eds.), Mining Multimedia and Complex Data. XII, 281 pages. 2003.

Vol. 2792: T. Rist, R.S. Aylett, D. Ballin, J. Rickel (Eds.), Intelligent Virtual Agents. XV, 364 pages. 2003.

Vol. 2782: M. Klusch, A. Omicini, S. Ossowski, H. Laamanen (Eds.), Cooperative Information Agents VII. XI, 345 pages. 2003.

Vol. 2780: M. Dojat, E. Keravnou, P. Barahona (Eds.), Artificial Intelligence in Medicine. XIII, 388 pages. 2003.

Vol. 2777: B. Schölkopf, M.K. Warmuth (Eds.), Learning Theory and Kernel Machines. XIV, 746 pages. 2003.

Vol. 2752: G.A. Kaminka, P.U. Lima, R. Rojas (Eds.), RoboCup 2002: Robot Soccer World Cup VI. XVI, 498 pages. 2003.

Vol. 2741: F. Baader (Ed.), Automated Deduction – CADE-19. XII, 503 pages. 2003.

Vol. 2705: S. Renals, G. Grefenstette (Eds.), Text- and Speech-Triggered Information Access. VII, 197 pages. 2003.

Vol. 2703: O.R. Zaïane, J. Srivastava, M. Spiliopoulou, B. Masand (Eds.), WEBKDD 2002 - MiningWeb Data for Discovering Usage Patterns and Profiles. IX, 181 pages. 2003.

Vol. 2700: M.T. Pazienza (Ed.), Extraction in the Web Era. XIII, 163 pages. 2003.

Vol. 2699: M.G. Hinchey, J.L. Rash, W.F. Truszkowski, C.A. Rouff, D.F. Gordon-Spears (Eds.), Formal Approaches to Agent-Based Systems. IX, 297 pages. 2002.

Vol. 2691: V. Mařík, J.P. Müller, M. Pechoucek (Eds.), Multi-Agent Systems and Applications III. XIV, 660 pages. 2003.

Vol. 2684: M.V. Butz, O. Sigaud, P. Gérard (Eds.), Anticipatory Behavior in Adaptive Learning Systems. X, 303 pages. 2003.

Vol. 2682: R. Meo, P.L. Lanzi, M. Klemettinen (Eds.), Database Support for Data Mining Applications. XII, 325 pages. 2004.

Vol. 2671: Y. Xiang, B. Chaib-draa (Eds.), Advances in Artificial Intelligence. XIV, 642 pages. 2003.

Vol. 2663: E. Menasalvas, J. Segovia, P.S. Szczepaniak (Eds.), Advances in Web Intelligence. XII, 350 pages. 2003.

Vol. 2661: P.L. Lanzi, W. Stolzmann, S.W. Wilson (Eds.), Learning Classifier Systems. VII, 231 pages. 2003.

Vol. 2654: U. Schmid, Inductive Synthesis of Functional Programs. XXII, 398 pages. 2003.

Vol. 2650: M.-P. Huget (Ed.), Communications in Multiagent Systems. VIII, 323 pages. 2003.

Vol. 2645: M.A. Wimmer (Ed.), Knowledge Management in Electronic Government. XI, 320 pages. 2003.

Vol. 2639: G. Wang, Q. Liu, Y. Yao, A. Skowron (Eds.), Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing. XVII, 741 pages. 2003.

Vol. 2637: K.-Y. Whang, J. Jeon, K. Shim, J. Srivastava, Advances in Knowledge Discovery and Data Mining. XVIII, 610 pages. 2003.

Vol. 2636: E. Alonso, D. Kudenko, D. Kazakov (Eds.), Adaptive Agents and Multi-Agent Systems. XIV, 323 pages. 2003.

Vol. 2627: B. O'Sullivan (Ed.), Recent Advances in Constraints. X, 201 pages. 2003.

Vol. 2600: S. Mendelson, A.J. Smola (Eds.), Advanced Lectures on Machine Learning. IX, 259 pages. 2003.

Vol. 2592: R. Kowalczyk, J.P. Müller, H. Tianfield, R. Unland (Eds.), Agent Technologies, Infrastructures, Tools, and Applications for E-Services. XVII, 371 pages. 2003.

Vol. 2586: M. Klusch, S. Bergamaschi, P. Edwards, P. Petta (Eds.), Intelligent Information Agents. VI, 275 pages. 2003.

Vol. 2583: S. Matwin, C. Sammut (Eds.), Inductive Logic Programming. X, 351 pages. 2003.

Vol. 2581: J.S. Sichman, F. Bousquet, P. Davidsson (Eds.), Multi-Agent-Based Simulation. X, 195 pages. 2003.

Vol. 2577: P. Petta, R. Tolksdorf, F. Zambonelli (Eds.), Engineering Societies in the Agents World III. X, 285 pages. 2003.

Vol. 2569: D. Karagiannis, U. Reimer (Eds.), Practical Aspects of Knowledge Management. XIII, 648 pages. 2002.

Vol. 2560: S. Goronzy, Robust Adaptation to Non-Native Accents in Automatic Speech Recognition. XI, 144 pages. 2002.

Table of Contents

Models and Typologies

Design and Applications

INVITED CONTRIBUTIONS

STANDARD CONTRIBUTIONS

Agency, Learning and Animal-Based Reinforcement Learning

Eduardo Alonso[1] and Esther Mondragón[2]

[1] Department of Computing, City University
London EC1V 0HB, United Kingdom
eduardo@soi.city.ac.uk
[2] Department of Psychology, University College London
London WC1H 0AP, United Kingdom
e.mondragon@ucl.ac.uk

Abstract. In this paper we contend that adaptation and learning are essential in designing and building autonomous software systems for real-life applications. In particular, we will argue that in dynamic, complex domains autonomy and adaptability go hand by hand, that is, that agents cannot make their own decisions if they are not provided with the ability to adapt to the changes occurring in the environment they are situated. In the second part, we maintain the need for taking up animal learning models and theories to overcome some serious problems in reinforcement learning.

1 Agency and Learning

Agents (and thus agency) have been defined in many different ways according to various research interests. It is universally accepted though that an agent is a software system capable of flexible, autonomous behaviour in dynamic, unpredictable, typically multi-agent domains. We can build up on this fundamental definition and state which characteristics an agent should display following the traditional distinction between strong and weak agency. The later prescribes autonomy, social ability, reactivity, and pro-activeness, to which the former adds various mental states, emotions, and rationality. How these features are reflected in the systems architecture will depend on the nature of the environment in which it is embedded and the degree of control the designer has over this environment, the state of the agent, and the effect of its actions on the environment.

It can be said, therefore, that, at first glance, learning does not seem to be an essential part of agency. Quoting Michael Wooldridge, "learning is an important agent capability, but it is not central to agency" [11]. Following the same line of argumentation, agent research has moved from investigating agents components to multi-agent systems organization and performance. For instance, the CfP for the International Joint Conference on Autonomous Agents and Multi-Agent Systems has discouraged papers that address isolated agent capabilities *per se* such as learning. In addition, AgentLinkII's (Europe's Network of Excellence in Agent-Based Computing) Special Interest Group on Agents that Adapt, Learn

M. Nickles, M. Rovatsos, and G. Weiss (Eds.): AUTONOMY 2003, LNAI 2969, pp. 1–6, 2004.

and Discover (ALAD) may disappear under the Sixth EU Framework. However generous AgentLinkII was in supporting activities directed towards the increase awareness and interest in adaptive and learning agent research (sponsoring, for example, the Symposia Series on Adaptive Agents and Multi-Agent Systems), and although there are plenty of recent references to learning algorithms and techniques (*e.g.*,[2],[10]) and that a considerable effort has been done in providing a suitable infrastructure for the development of close collaboration between machine learning experts and agent systems experts with the creation of the Adaptive and Learning Agents and Multi-Agent Systems (ALAMAS) Consortium, the truth is that learning agents does not seem to be a priority any longer.

It is our understanding that one of the main arguments against considering learning as a requisite for agency is that there are scenarios in which agents can be used and learning is not needed. For example, little can be learned in accessible domains where agents can obtain complete, accurate, up-to-date information about the environment's states, or in deterministic domains where any action has a single guaranteed effect, or in static domains where the environment remains unchanged unless an action is executed.

It is our contention though that in such domains agents are not strictly necessary anyway and that applying object-oriented technology would suit best the requirements and constraints designers must meet. Put roughly, if you can use objects, do not use agents. Unlike agent-oriented technology, object-oriented technology is well-established and understood, with clear modeling and specification languages (UML) and programming languages (Java, C++). On the other hand, as denounced in [1], an Unified Agent Modeling Language is still under development, and although some object-oriented features such as abstraction, inheritance and modularity make it easier to manage increasingly more complex systems, JAVA (or its distributed extensions JINI and RMI) and other OO programming languages cannot provide a direct solution to agent development.

Agents are ideal for uncertain, dynamic systems. Lehman and Belady's Laws of Software Evolution [6], particularly, those referring to continuing change and increasing complexity have proven true with the growth of the Internet and the arrival of the Grid computing. Certainly, it has become increasingly complicated to model and control the way software systems interact and get co-ordinated. Perhaps, many claim, we should focus on observing their emergent behaviours. Perhaps we should move from software engineering to software phenomenology and study the performance of multi-agent systems the same way we study biological or chemical systems (for example, by using minimalist Multi-Agent System platforms such as BTExact's DIET [4]). On the one hand, by distributing the tasks among different autonomous entities we gain in both speed and quality. On the other hand, such systems seem to work as if guided by an "invisible hand". Designers cannot foresee in which situations the systems will encounter themselves or with whom they will interact. Consequently, such systems must adapt to and learn from the environment so that they can make their own decisions when information comes. To sum up, agents need to learn in real-life domains. Therefore, in real-life domains, learning is essential to agency.

Several initiatives have been launched recently to investigate adaptive intelligent systems. In an attempt to improve our understanding of the mechanisms and organisational principles involved in the generation of adaptive behaviour in intelligent machines or adaptive systems inspired by the study of animals the BBSRC and EPSRC have announced a call for Adaptive and Interactive Behaviour of Animal and Computational Systems (AIBACS). Separately, a Special Interest Group on Computation and Animal Learning has been formed. Moreover, the Society for the Study of Artificial Intelligence and the Simulation of Behaviour (SSAISB) and the International Society for Adaptive Behaviour (ISAB) are organising symposia and workshops on this research area in 2004.

Our contribution to this trend of thought is briefly explained in the next section. We propose to use animal learning theories to overcome some drawbacks in the quintessential machine learning technique for agents, namely, reinforcement learning.

2 Animal-Based Reinforcement Learning

Reinforcement learning has been defined as learning what to do —how to map situations to actions— so as to maximise a numerical reward signal. Unlike supervised learning such as pattern recognition or artificial neural networks, the learner is not told which actions to take, but instead must discover which actions yield the most reward by trying them. Typically, actions may affect not only the immediate reward but also the next situation and, through that, all subsequent rewards. These two characteristics, trial-and-error search and delayed reward, are the two most important features of reinforcement learning.

The reinforcement learning problem is the problem of finding an optimal policy, *i.e.*, the optimal way of behaving at a given time defined as a mapping from perceived states of the environment to actions.

Among the different techniques used to solve the reinforcement learning problem (see [7] for a detailed account) temporal-difference methods are the most widely used due to their great simplicity: They can be applied on-line, with minimal computation, to experience generated from interaction with an environment; besides, they can be expressed nearly completely by single equations that can be implemented with small computer programs.

Despite the success of these methods, for most practical tasks, reinforcement learning does fail to converge even if a generalising function approximation is introduced. It turns out to generate extreme computational cost when not dealing with small state-action pairs, which are, in practice, very rare in any real learning scenarios. For example, since all state-action pairs must be repeatedly visited, Q-learning does not address generalisation over large state-action spaces. In theory, it may converge quite slowly to a good policy. In practice however, Q-learning can require many thousands of training iterations to converge even in modest-sized problems.

To overcome these problems we propose to go back to animal learning psychology and update reinforcement learning algorithms with contemporary associative and instrumental learning models.

One historical thread of reinforcement learning concerns learning by trial and error and started in the psychology of animal learning. In particular, Thorndike [8] described the effect of reinforcing events on the tendency to select actions in what he called the Law of Effect. This Law includes the two most important aspects of trial-and-error learning and, in turn, of reinforcement learning, namely, it is selectional (it involves trying alternatives and selecting among them by comparing the consequences) and associative (the alternatives are associated with particular situations).

There is however plenty of evidence that suggest that Thorndike's Law is wrong. For example, it has been established experimentally that rewards are not essential for learning and that the assumption that learning consists of the gradual strengthening of a connection between neural centres concerned with the perception of a stimulus and the performance of a response is far too simple.

Reinforcement learning has tried to keep abreast with new animal learning theories by, for example, incorporating Tolman's findings on instrumental learning [9]. Yet, these new paradigms have, in turn, proved to be inaccurate. As a consequence, reinforcement learning techniques based on out-of-date animal learning models may be conceptually incorrect. Indeed, it could be argued that this failure to understand recent developments in animal psychology is of no consequence for reinforcement learning. On the contrary, we claim that the poor results so far encountered may be at least in part due to this gap between theory and practice.

To start with, an analysis of the terminology allegedly taken from animal psychology shows that most of the concepts used in reinforcement learning do not match with their counterparts in animal learning. States (or events) are compounds of stimuli, actions are responses, and rewards are values associated to actions that can be understood as reinforcers on stimulus-response associations. Of course, changing our vocabulary should not be a problem. The real problem arises when such changes are ontological and thus prevent reinforcement learning from studying the nature of reinforcers, the sort of associations formed and the conditions for their formation.

What else are we missing? First of all, associations are formed among different stimuli (classical learning) and between the response and the reinforcer (instrumental learning), not ony between a stimulus and a response. Secondly, reinforcers are stimuli and, therefore, elements of the associations, not mere values assigned to stimulus-response associations. Moreover, reinforcers have not only specific but also affective characteristics, that is, characteristics that reflect their motivational quality. Finally, learning depends on contiguity, contingency, associative competition, attention and surprisingness, none of which is considered in reinforcement learning.

One more example. Unlike other machine learning paradigms, reinforcement learning assumes that, for optimal performance, animals do explore (state-action pairs which outcomes are unknown) and exploit (those state-action pairs which rewards are known to be high). Numerous functions have been presented in the literature to control the balance between these two opposite processes (*e.g.*, soft-max action selection). Either way, behaviour as such is neutral in reinforcement learning, *i.e.*, behaviours acquire values only when rewarded. On the contrary, animals explore by default: Exploratory behaviours act as (internal) reinforcers *per se*. The strength of the association between these behaviours and other reinforcers will, thus, depend on the behaviours intrinsic value. Indeed, Kaelbling *et al.* [5] have already suggested that, in order to solve highly complex problems, we must give up *tabula rasa* learning techniques and begin to incorporate bias that will give leverage to the learning process. The very nature of animal learning may well be such a bias.

Of course, there have been several attempts to bring together the machine learning community and the animal learning community. For example, Peter Dayan and L.F. Abbott [3] have successfully modelled several phenomenon in classical conditioning and instrumental learning. This trend has been directed towards the identification of mathematical techniques, mainly temporal difference algorithms, that psychologists might use to analyse and predict animal behaviour.

Our goal is complementary. We are not trying to bridge gaps in the analytical skeleton of animal learning using reinforcement learning techniques. Instead, we intend to use animal learning models to improve reinforcement learning performance. Our contention is that convergence and generalization problems so common to reinforcement learning can be corrected by re-defining some of its more fundamental basis according to animal learning models.

References

[1] E. Alonso. AI and Agents: State of the Art. *AI Magazine,* 23(3):25–29, 2002.

[2] E. Alonso, D. Kudenko, and D. Kazakov, editors. *Adaptive Agents and Multi-Agent Systems.* LNAI 2636, Springer-Verlag, Berlin, 2003.

[3] P. Dayan and L.F. Abbott, *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems.* The MIT Press, Cambridge: MA, 2001.

[4] C. Hoile, F. Wang, and P. Marrow. Core specification and experiments in DIET: A decentralised ecosystem-inspired mobile agent system. In C. Castelfranchi and W. Lewis Johnson, editors, *Proc. 1st Int. Conf. on Autonomous Agents and Multi-Agent Systems (AAMAS-2002),* pages 623–630, ACM Press, New York, 2002.

[5] L.P. Kaelbling, M.L. Littman, and A.W. Moore. Reinforcement Learning: A Survey. *Journal of Artificial Intelligence Research* 4:237–285, 1996.

[6] M.M. Lehman, and L. Belady. *Program Evolution: Processes of Software Change.* Academic Press, London, 1985.

[7] R.S. Sutton and A.G. Barto. *Reinforcement Learning: An Introduction.* The MIT Press, Cambridge: MA, 2002.

[8] E.L. Thorndike. *Animal Intelligence: Experimental Studies.* Macmillan, New York, 1911.

[9] E.C. Tolman. *Purposive Behavior in Animals and Men.* Century, New York, 1932.

[10] G. Weiss, editor. *Distributed Artificial Intelligence Meets Machine Learning.* LNAI 1221, Springer-Verlag, 1997.

[11] M. Wooldridge. *An Introduction to Multiagent Systems.* John Wiley & Sons, Chichester, 2002.