ELSEVIER

Robotics and Autonomous Systems 27 (1999) 91–106

Robotics and Autonomous Systems

Multi-agent systems: which research for which applications

Eugénio Oliveira [a,1], Klaus Fischer [b,*], Olga Stepankova [c,2]

[a] *Faculdade de Engenharia, Universidade do Porto, Rua dos Bragas, 4099 Porto codex, Portugal*
[b] *DFKI, Stuhlsatzenhausweg 3 (Bau 43), D-63123 Saarbrücken, Germany*
[c] *Czech Technical University, Technicka 2, 166 27, Prague 6, Czech Republic*

Received 26 August 1998; received in revised form 6 November 1998

Abstract

For sometime now agent-based and multi-agent systems (MASs) have attracted the interest of researchers far beyond traditional computer science and artificial intelligence (AI). In this article we try to identify focal points of interest for researchers working in the area of distributed AI (DAI) and MAS as well as application-oriented researchers coming from related disciplines, e.g. electrical and mechanical engineering. We do this by presenting key research topics in DAI and MAS research and by identifying application domains in which the DAI and MAS technologies are most suitable. The research topics we discuss are separated into agent architectures and organisations, negotiation among agents, and self-adaptation of MAS using learning techniques. Regarding the application domains for these techniques we distinguish the application domains according to whether the agents control a physical or virtual body (Gestalt) or not. This separation of the application domains is not strict; it represents two ends of a continuum. On the one end of this continuum we have autonomous robot systems which act in a physical environment (sometimes referred to as hardware agents), and on the other end, we have abstract environments, such as in workflow systems, which rarely display the geometrical and physical aspects of the environment we are used to living in.

Keywords: Multi-agent system; Architecture; Learning; Application; Co-ordination

1. Introduction

Multi-agent systems (MASs) emerged, as a scientific area, from the previous research efforts in distributed artificial intelligence (DAI) started in the early eighties. MAS is now seen as a major trend in R & D, mainly related to artificial intelligence and distributed computing techniques. This research has attracted attention in many application domains where difficult and inherently distributed problems have to be tackled. The principal aim of this paper is to suggest possible answers to the following questions:
- Which research topics seem to be the most promising relevant in the MAS research community?
- Which application domains seem to be the most suitable for the use of MAS?

Moreover, the paper tries to find out the existing inter-relationship between both treated issues: research in the MAS area and MAS potential needs for applications. The structure of the paper is as follows: We first give, in Section 2, definitions of agents and multi-agent systems, and at the same time, some MAS related problems are also referred to. Section 3.1

* Corresponding author. Tel.: +49-681-302-3917; fax: +49-681-302-2235; e-mail: klaus.fischer@dfki.de
[1] E-mail: eco@fe.up.pt
[2] E-mail: step@labe.felk.cvut.cz

0921-8890/99/$ – see front matter
PII: S0921-8890(98)00085-2

introduces the agent architecture debate and Section 3.2 the co-ordination aspects. This includes negotiation protocols, market-based mechanisms, and communication issues. Section 3.3 is devoted to possible impact of learning mechanisms on MAS design. Section 4 presents some important classes of MAS applications and finally, Section 5 summarises efforts being (and to be) done on research in terms of models of agency.

2. Definition and characterisation of multi-agent systems

2.1. Agent

The concept of an agency is now being broadly used not only as a model for computer programming units displaying certain kind of characteristics but also in a more abstract and general way, as a new metaphor for the analysis, specification, and implementation of complex software systems.

Many authors have given definitions for the notion of agency [28,47]. What has to be emphasised is that an agent, unlike other programs, must simultaneously have at least the following main features:

- It perceives the world in which it is situated.
- It has the capability of interacting with other agents.
- It is pro-active in the sense that it may take the initiative and persistently pursues its own goals.

Therefore, an agent is supposed to act spontaneously (with initiative), executing pre-emptive and independent actions that eventually benefit the user through accomplishment of the assigned goals.

Different kinds of agents can be characterised, and other sophisticated features are usually needed to make them act intelligently in their respective environment. Nevertheless, it is usually assumed that the core of an agent includes the three characteristics mentioned above.

2.2. Multi-agent systems

Although, in many cases, agents can act separately to solve a particular problem, it often happens that a complete system made of several different agents has to be designed to cope with a complex problem involving either distributed data, knowledge, or control. A multi-agent system can therefore be defined as a collection of, possibly heterogeneous, computational entities, having their own problem-solving capabilities and which are able to interact in order to reach an overall goal. It may also be the case that a MAS is seen as a system revealing a kind of synergy that would not be expected from the simple sum of its component agents. This synergy is an emergent property of the system as a whole. Following Bond and Gasser [6] and also [47], we can explicitly point out existing and challenging problems that researchers are facing in designing and implementing multi-agent systems.

- *The domain specification problem.* How can we formulate in a non-ambiguous way the problem at hand? Is it possible to adapt existing software and knowledge engineering methodologies, like object-oriented approaches, to be applied to agent-based systems?
- *The communication problem.* What are the most suitable protocols and languages enabling a possibly sophisticated and meaningful interaction between agents in a MAS?
- *The co-ordination problem.* How can we enforce the necessary teamwork, leading to coherent and effective results according to the overall system's goals and preventing agents from being "autistic" by giving them the possibility to reason about other agents plans, strategies, beliefs, and actions?
- *The computational problem.* Can we design and implement a system in a way that avoids computational overload by means of load balancing strategies?
- *The implementation problem.* What techniques and tools are needed to support multi-agent system design and implementation in a safe, easy, and productive way? and finally,
- *The verification problem.* What formal and practical approaches will allow us to verify, diagnose, and easily correct multi-agent systems applications?

We are not going to analyse all these problems in this paper. Instead, we shall pick out some important points, which are critical in order to make MAS useful and applicable at present.

3. Research themes for MAS

From the broad spectrum of possible research themes, we have selected a few topics we assume to

be important and promising. They are related to
- current controversy around agent architectures and organisational structures in multi-agent systems,
- policies and strategies for agent co-ordination, including needed interaction and possible negotiation,
- inter-agent communication,
- agent adaptability and learning capabilities in a MAS framework.

3.1. Architecture and organisation of agents

Coming from the seventies, the fundamental physical symbol system hypothesis of Simon and Newell [63] was for more than a decade the only accepted paradigm supporting really intelligent systems. Thus, agents that maintain their own internal representation of the world and are able to perform symbolic reasoning were the only ones recognised as displaying some intelligence and they were called deliberative agents.

It was only a decade after Newell and Simon's statement that, at MIT, Brooks [7] launched a new idea, strongly influenced by behaviourist psychology, denying the need for internal world models ("the world is its best model"). The main thesis associated with this new paradigm claims that intelligence need not be situated within each agent but, on the contrary, intelligence can emerge as a consequence of the interaction among very simple computational units. Since then, many researchers have been following the same line of thought, mainly influenced by theories related to the study of animal societies [1] and other theories of self-organisation [28]. Agents based on this paradigm are often called reactive agents.

This controversy between what can be seen as the proposal of a vertical layered agent architecture (inputs, coming from the "right-hand side" cross all the layers in order to produce an output, making agents more cognitive), and on the other extreme, a horizontal layered architecture (agents can activate just one, or some of the several possible horizontal layers in parallel, where behaviours reside, to produce a fast reaction), gave way to the need for a clearer separation and characterisation of both respective application fields.

Research needed on hybrid architectures and MAS tools. There are of course a number of hybrid architectures rising from the explicit need of particular applications, to which both former approaches are not adequate, as it is the case with robotics. Here, in the robotics field, the robot is either seen as an agent itself or it is its control system that is envisaged as a set of agents of several different types (reactive, adaptive, deliberative) working together [31,62,65].

An important open issue in hybrid multi-agent systems (as well as for the agent itself) is what we can call the problem related with the "schizoid" syndrome [65]. The source of this syndrome is the fact that several different agents with different capabilities, architectures and response times, may be competing for control of a system (as it is the case in the control architecture of a mobile robot). Since we do not always want to establish an a priori hierarchy to resolve conflicts, reactive agents may be proposing all the time actions that contradict or supersede more elaborate deliberations coming from more cognitive agents working in parallel.

A flexible, hybrid architecture which is able to decide when to pay attention to one or another kind of the proposed actions, or whether they should or should not be combined, leading to a kind of fuzzy control, is still a matter of investigation [62]. This problem of conflict resolution is somehow avoided, in most hybrid architectures, by using layering as an agent organisation principle. This principle has some advantages with respect to conceptualisation (modularity), robustness (fault tolerance, easy debugging), efficiency (agents on different levels run in parallel), and clearness (clear separation between reactive and deliberative agents). This is the case with the RAP system [31] for robot control and with InteRRaP [60]. InteRRaP can be seen as a general agent-based layered system where control is shifted bottom up from one layer to the other (from reactive to more cognitive agents that propose plans and co-operation) whenever one layer is not competent to deal with the situation. This solution is not always acceptable due to its inflexibility coming from the fact that a kind of serialisation of competencies has to be provided from the beginning by the designer.

The question "what kind of knowledge has to be included in each agent's mental state", as well as how it has to be represented, has led to the, now very popular, BDI (believes, desires, intentions) architecture. These mental categories, which have already been enhanced by other concepts, like goals, commitments, and plans, are still far from being well-defined for generic situations. To indicate current and future directions of research concerned with knowledge for agent architectures, let us point out a more advanced and well-founded modal logic approach that allows to capture and express all the richness of those mental state concepts. Singh [79] developed a theory providing a framework for specifying multi-agent systems that includes descriptions of intentions, know-how, and communications. However, the author himself considers it in most situations not sufficient for the design of MAS. A more elaborate design theory is thus urgently needed. Such a theory would also encompass algorithms for checking the satisfaction of the MAS specifications using the proposed formal language (model checking). Another aspect that deserves some attention and research efforts is the development of safe and easy to use tools both for specifying agents and relationships between agents, as well as to translate a complete and coherent (multi-)agent system specification into an efficient, correct, and robust implementation.

On the one hand, we may consider Shoam's Agent0 [77], Congolog [52] among others as preliminary high-level tools to specify agents in a well-founded way. On the other hand, at a lower level, platforms for the distribution of and inter-communication among agents have been in use for sometime. JATLite [45], which already enforces KQML [30] as a communication language is such a tool. We are still waiting for a good visual programming tool enabling agents as well as complete multi-agent systems specification using graphical design tools (as is already the case for object-oriented programming). Such a tool should include clear as well as simple logic formalisms for knowledge specification.

3.2. Co-ordination of autonomous agents

Whenever agents have to work in a group setting, interactions take place to find a suitable organisation (who does what) as well as to enable communication of results (when and to whom). All these interactive activities imply the need for a clear policy for co-ordination. Following Lesser [53], reasoning about communication has to take into account the amount of interaction, performance and needed resources, future activities and loads, resources loading, imposed deadlines, and also knowledge representing an agent's desires, intentions, and beliefs. The larger the number of different possibilities and the set of the identified constraints for agents' joint work are, the richer the co-ordination policies will be.

In the case where agents are self-interested, interaction aims at maximising an agent's utility. In the case where agents share an overall goal, the objective is to maintain global coherence without violating autonomy, thus avoiding explicit global control. Task decomposition, task distribution, task monitoring, and task evaluation in the context of a multi-agent system, all could be considered in the scope of co-ordination of the agent's activities in a dynamic environment where resources may be scarce.

While task decomposition clearly is out of the scope of this paper, task distribution has been considered as one of the main goals for co-ordination among agents in a multi-agent system. According to Durfee [25], task distribution can be done according to the following criteria:
- to avoid overloading of critical resources,
- to assign tasks according to appropriate agents competencies,
- to enable possible sub-decomposition by some important agents, and
- to minimise communications through appropriate clustering of agents.

A variety of different mechanisms for co-ordinating agents in a multi-agent system is already available:
- Use of the contract net protocol [17] which proposes episodic rounds of inter-communication acts (announcements, bids, award messages). It is a simple and widely used protocol that, on one hand, does not affect too much the system responsiveness but, on the other hand, neglects possible strategic reasoning capabilities of the responding agents. The contract net protocol is mainly applicable to well-defined coarse-grained task decomposition.
- Multi-agent planning implies that all agents have planning capabilities, and each one of them takes into account the other agent's actions and constraints. Local plans can be communicated to other nodes in the network in order to reach a global plan accepted by all agents (that are willing to collaborate) before the actual action execution. This is the case with partial global planning (PGP), first proposed by Durfee [24] and then enhanced further by Decker in TAEMS [18], to ensure more general

applicability and ability to deal with real-time problems.
- Computational market-based mechanisms can be designed to enhance the adaptivity, robustness, and flexibility of multi-agent systems. Within a market, agents representing modular services and/or competencies available in the multi-agent system compete to perform tasks to serve their individual objectives.

Market-based co-ordination strategies usually use auction-inspired protocols not only to empower, through computers and Internet, the real market activities, but also to facilitate distribution of task and resource allocation. Wellman [96] advocates market-based mechanisms for flexible management of information systems, claiming that a market's ability to rapidly disseminate (through price variation) changes in the scarcity value of remaining resources minimises communication and avoids global control and synchronisation. Clearly, a key feature of a market-based multi-agent system is that each agent makes local decisions based on its individual knowledge plus prices (or other parameters) announced in the market.

Uncertainty about the future conditions of the market is difficult to be mastered. This uncertainty about the future situations is inherent to the market framework and therefore, the market-based mechanism to be used should include some risk assessment and risk management features.

Also the influence of non-competitive agents behaviour, in contrast to selfish agents behaviour, in the overall multi-agent system performance has to be studied. Still lacking are also appropriate software interfaces (APIs) consisting of libraries of routines implementing several different auction protocol types. More general bidding strategies could then be implemented on top of these lower-level operations. This could constitute a kind of agentware for auctions.

3.2.1. Negotiation among agents

The practical reasoning paradigm that can be found in BDI-based agent architectures [37,72] assumes a joint intention mental state enforcing the agents to commitments to specific (eventually persistent) goals. Commitments are usually related to the execution of tasks, allocation of resources or exchanges of partial or final results. Commitments may or even should be established through negotiation. Negotiation is the process for self-interested agents to solve conflicts under bounded rationality. The main approaches for agent decision making while negotiating can be based on game theory [73]. Agents using payoff matrices to represent common knowledge, exchange their offers until an acceptable deal is reached. Variations of this principle can be found, e.g., in the PERSUADER system [83] where agents' utilities, involving several other dimensions than prices, are kept private. In the above-mentioned system, a simple argumentation strategy can take place between agents involved in the negotiation process. The same idea about argumentation deserves some research attention and may also be very useful for the sake of maintaining global belief consistency in a multi-agent system. In the unified negotiation protocol of Rosenschein and Zlotkin, a deal, i.e. a joint plan, is negotiated such as some utility is distributed among the interested agents.

This utility is a positive figure if what an agent is willing to pay is greater than the cost that he actually has to pay to establish the deal. Through the negotiation process, one deal is selected from the non-empty set of possible deals.

An interesting topic of research that is related with this subject is how to develop negotiation protocols that are able to recognise and to avoid inappropriate agents attitudes like deception and misleading, and accordingly, award honest agent behaviour.

Agent theory and economics cross-fertilisation has motivated fruitful research leading to economics-based approaches of agent rationality. Wellman [96] and Sandholm [75] are among those researchers who propose interesting formulas for price calculation. The main idea behind this approach is that equilibrium in the market (including consumers and producers) reflects a good allocation of resources in a corresponding multi-agent system.

Economically motivated agents may also negotiate to enter the best agent coalition which will execute tasks for a price that guarantees (in the present but also in the near future) good profits (or at least avoiding loosing too much). Those coalitions map good agents' dynamic organisation and agents' coalitions have been used to find out an optimal allocation of the available distributed resources in a multi-agent system [66].

Finally, another interesting topic to look at is the development of multi-criteria functions for agent utility calculation.

3.2.2. Agent communication

Big efforts are being made to develop agent communication facilities. Questions like "How can an agent meet other agents?" which is being dealt with through the use of middle agents (matchmakers, yellow pages, brokers), and "How to interoperate?", have been the main concerns of MAS developers regarding agents communication. Agents appearing and disappearing in an open environment, accessible through the Internet, need both those facilitators to meet others and specialised languages to understand each other. Languages using performatives based on speech act theory like KQML [30] and ACL recently proposed by FIPA (Foundation for Intelligent Physical Agents) clearly distinguish different message levels and identify the unambiguity of the semantics of the message content as a difficult and crucial point to focus on. Therefore, methods of building up well-structured and acceptable ontologies are now under intensive research.

Finally, it is not surprising to recognise that one of the important features that can improve multi-agent system dynamic co-ordination is agent adaptability. Learning about other agents and learning the effects of the agent's own actions can be useful for adapting old strategies (or building up new ones) in order to improve agents and/or the overall multi-agent system performance.

3.3. Design options to be changed by learning

For most application tasks, even in environments appearing simple, it is difficult (or even impossible) to determine the behaviour of a multi-agent system a priori – that is, at the time of its design and prior to its application. This would require, for instance, that the designer knows in advance which environmental requirements will occur in the future, which agents will be available at the considered time, and plan their interaction in response to these requirements. Enumeration of all possible states of a multi-agent system would inevitably cause a combinatorial explosion. Construction of rational agents points to some fundamental questions including, e.g. the concept of *bounded optimality* for rational agents [74]. Some problems resulting from complexity of the multi-agent systems can be avoided, or at least reduced, by providing the agents with the ability to *adapt* and to *learn* [94,95].

Most types of classical learning algorithms [58] can be applied within a single agent when this agent does not rely on the presence of multiple agents to achieve *isolated (single-agent) learning*. Results of isolated learning in a distributed environment can be further combined on a meta level by new techniques. This approach has achieved many promising results. One of its recent successes is that of data-mining [71] in inherently distributed databases which can appear, e.g. in banking institutions. Development of software information agents, which learn how to better meet the demands of their human users [38], is based on these principles. Case-based reasoning works well in a distributed context, too [65,70].

Interactive (multi-agent) learning relies on the presence of multiple agents and their interaction. The concept of interactive learning itself can be applied in two different ways. In the stronger and more specific meaning, interactive or multi-agent learning refers only to situations in which several agents learn how to pursue a common learning goal [4]. In the weaker and less specific meaning, it additionally refers to situations in which an agent pursues its own learning goal, but is affected in its learning by other agents [94]. Research in interactive learning can shed light on the origins of language [81] communication and co-ordination [20] as well as on the emergence of social conventions [78].

Study of interactive learning in MAS has started only recently. Its aim differs radically from that of classification or diagnosis. The goal of interactive learning is to influence characteristic features of the multi-agent system's design, namely its *structure* (e.g. number of agents and their interconnection), type of *communication* between agents (e.g. choice between task announcement or direct addressing), and *co-ordination* (policy of interaction of agents). Ways of communication and *co-ordination* can be modified by changing:

- the initial design of *task decomposition* and *allocation*, e.g. using knowledge, agent specialisation or identification of clusters of similar knowledge,
- the original *properties of individual agents* (e.g. specialisation, capacity), and
- *knowledge* the agents have *about the abilities of the others and about the environment* (e.g. agent reliability).

New methods of learning have to be developed to reach such goals.

Classical learning methods can be classified according to the type of *feedback* they employ. This point of view retains its importance in the interactive learning, too. The following forms of interactive learning can be distinguished with respect to the type of feedback available to a learning entity to indicate the performance level achieved so far:

- *supervised learning* – the feedback specifies the desired activity of the learner; the goal of learning is to match this desired action as closely as possible;
- *reinforcement learning* – the feedback only specifies the utility of the actual activity of the learner and the goal is to maximise this utility;
- *unsupervised learning* – no explicit feedback is provided and the goal is to find out useful and desired activities on the basis of a trial-and-error process.

There are other, more specific, criteria related to a multi-agent environment which offer additional means for structuring the learning methods. They refer, e.g. to the *number of agents* involved in the learning steps, *the purpose and goal of learning* (improvement with respect to one single agent or with respect to the group of agents as a unit, their coherence and co-ordination) or the *time of communication among agents* with respect to the learning process. Davies and Edwards [16] use the last criterion to distinguish methods with respect to localisation of the training data:

- data are gathered into one place before the learning process starts,
- individual agents learn on local data, partial results are shared by means of communication during the learning process,
- agents learn locally and later they share their results, which are then refined and integrated by other agents in the light of their own data and knowledge.

So far, the pioneering attempts in interactive learning, suggest methods which are tailored for specific and simplified problems (e.g. stable price for communication within the system, strictly limited set of the agent's actions) and which result in a change of one (at most two) of the design features characterising the MAS system. Learning can influence communication patterns which shift from broadcasting to direct addressing [64] and it can improve knowledge about competence of the peer agents in the community [85]. Reinforcement Q-learning [93] has been successfully applied in multi-agent systems to improve interaction scenarios. Modular Q-learning [68] helps to create new patterns of co-operation in the MAS community – this process leads to the *emergence* of surprising social phenomena, e.g. altruistic behaviour in the predator and prey problem [67], as well as to emergence of specialised agents in the society of agents all of which have been originally equipped with the same abilities.

Multi-agent learning is a rich source of challenging problems. The methods mentioned above show promising directions but they have to be scaled up to real-life cases. New ways on how to combine the developed methods to gain complex change of a MAS have to be invented. In real situations, it can easily happen that the complex system design can hide some unintended loops or bottlenecks, which should be attended as soon as they are identified. That is why it is necessary to search for learning methods that are able to suggest changes to the architecture of the considered MAS, e.g. add a new specific agent (or switch the role of one of the existing agents), if a bottleneck in the original MAS system is detected [82].

A MAS composed of learning agents represents a dynamic system, the behaviour of which should be analysed and understood. Most often, MAS learning is evaluated experimentally. Vidal and Durfee [89,90] use concepts from computational learning theory to suggest a theoretical framework for study of the dynamics of learning MAS. The learning task to be solved by each of their agents is to change its decision function so that it matches the target function. Their theoretical results correspond well with the experimental findings. This approach represents the first step towards the development of a theoretical alternative to experimental approaches to evaluate results of learning in MAS. Nevertheless, experimental verification maintains its importance when complex heterogeneous systems are concerned.

Despite the fact that learning in multi-agent systems is a relatively new field of study, there is a large body of references to work already done in this field. A recent volume [95] edited by Weiss offers an up-to-date reader's guide and a valuable overview of the major challenges for machine learning in MAS. Learning in distributed AI systems has been chosen as the topic of a special issue (4/97) of the Journal of Experimental and Theoretical Artificial Intelligence (JETAI). The Machine Learning Journal plans a special issue on multi-agent learning to be edited by Huhns and Weiss.

A general comparison of distributed problem-solving and multi-agent systems can be found in [26], while special problems of the agent-based approach to data-mining are analysed in [16].

4. MAS applications

The general notion of agency is so universally applicable that it is actually hard to tell an application domain in which agent methodologies are not a useful concept. Even if we are just interested in the problem-solving capabilities of a system, it is in many cases reasonable to take a multi-agent perspective. However, what we can do is to identify properties, which are typical for application domains in which multi-agent system technologies are most appropriate. Properties like this are:

Distribution. In the application domain geographically and/or logical distributed and heterogeneous entities, data, or information are identifiable; e.g., distributed entities that have to make decisions or distributed knowledge bases which have been developed independent of each other have to be integrated.

Complexity. The overall problem, which has to be solved, is in its computational complexity only tractable with heuristic strategies which use local chunks of data or knowledge and which can be easily separated into autonomous problem-solving entities.

Flexible interaction. There is no a priori assignment of tasks to problem solvers and there are no fixed problem-solving processes.

(Highly) Dynamic environments. Such environments require *responsive* and *adaptive* problem-solving entities (e.g., autonomous robots acting in a shop floor environment or softbots acting in virtual reality worlds on the World Wide Web (WWW)).

Openness. In these settings it is not even possible to give a complete specification of the problem which has to be solved. An example of such a setting is an electronic marketplace, in which a large number of users with differing interests interact with each other. We can neither define a global utility function nor can we declare any user's objective function as the one the overall system should use. In such settings users can interact in a collaborative or competitive manner.

If we investigate specifically one of these properties in a given application domain, we might come up with theories and solutions that are not necessarily exclusively agent-related. MAS theories and technologies are most appropriate when we investigate application domains which reveal several of these properties. However, even if we restrict ourselves to these kinds of application domains, we end-up with a list that is much too long for the paper. We therefore rather concentrate on a classification of application domains in which the use of a strong notion of agency is beneficial. From an intuitive point of view this means that we assume the agents in these application domains to be intelligent at least in a technical sense. More concretely it is likely that the agents in the application examples we discuss Sections 4.1 and 4.2 will at least need the following abilities: (1) to represent local knowledge about the outside environment, (2) to do local problem solving, (3) to perform actions in the environment, and (4) to communicate with other agents and/or human users.

In the following we give a brief description of several application domains for which multi-agent technologies are especially useful. However, we present these application domains from the point of view of multi-agent system research and stress that electrical engineering, mechanical engineering, and traditional computer science have also made contributions to these application domains.

4.1. Software agents

In our classification of application domains for agent technologies and MAS we follow the argumentation of Müller [61] who distinguishes software and hardware agents. Müller requires that a hardware agent has a physical *Gestalt* and that such an agent acts in a physical environment. We extend this definition to agents which control a physical or a virtual body in a physical or virtual reality environment. Of course this is no longer a strict distinction but rather a scale spectrum with two extremes. On the one end of this spectrum we have physical robot systems acting in a physical world, and on the other end of the scale we have pure software entities living in a virtual environment which does not necessarily have anything in common with the environment we are used to living in.

4.1.1. Intelligent manufacturing systems

From the very beginning of research on *intelligent manufacturing systems* (IMSs) IT technologies played an important part in the design and implementation of an IMS. About 20 years ago the idea of computer-integrated manufacturing was proposed by Harrington [42]. While the first approaches to realise these ideas were based on a centralised model, more recently the design models shifted to decentralised paradigms. Fractal [92] and holonic [10,13,19] approaches to the design of manufacturing systems are the architectures which have the largest impact on current research projects. Especially the concepts for holonic manufacturing were deeply influenced by DAI and MAS research [9,19]. [35] advocates that agent technologies are the ideal means to design and implement such systems. Within a CIM system we can identify different layers of abstraction: workflow management, shop floor control, and autonomous control systems. Because workflow management and autonomous control systems have a broader context, we shall describe these two application domains in separate paragraphs. Parunak [88] presented YAMS as one of the first approach models to design a *flexible manufacturing system* (FMS) with a DAI approach. The main idea of YAMS is to use the contract net protocol [17] for task allocation in the FMS. Ow et al. [69] and Butler and Ohtsubo [11] used the contract net model for job shop scheduling and Bussmann [9] investigated the application of the contract net model to the control of material flow. Fischer [32] proposed an agent-based approach to the design of an FMS, which suggests a hierarchical architecture, which is built up of autonomously acting agents. How task allocation can be done using a reactive scheduling approach was described in [33,41] and proposed a completely decentralised model for job shop scheduling which is able to produce better results than the pure contract net protocol because planning is done with some look-ahead. The bottom line of these results is that multi-agent system technologies will play an important role in the design of future control systems for FMS, because of the inherent distribution of such systems, the complex system architecture, and the need to adapt to changing market situations. Self-organisation and problem-solving based on negotiation protocols are a crucial part in the design of such systems. Auction protocols, such as, e.g. the contract net protocol, are a good approach to tackle some of these problems. However, from [31,41] we can see that the quality of the solutions provided by a system based on these approaches can be significantly improved when they are integrated with other problem-solving paradigms, e.g., market mechanisms [34].

4.1.2. Workflow management and virtual enterprises

Workflow management deals with the specification and execution of business processes. General process definitions include activities to be performed, their control flow, and data exchange. They also comprise organisational roles of persons and software components that are permitted to perform activities. Policies, which describe the organisational environment, complete a process definition [57]. Workflow management tools seem to be the ideal means to realise new organisational structures in enterprises of the future.

The term virtual enterprise, which is generally attributed to Mowshowitz [59], characterises such an organisation structure, which takes an IT-driven view to the organisation of future enterprises. The term obtained its current importance for business economics from Davidow and Malone's landmark book [15]. A virtual enterprise is a temporal co-operation of legally independent enterprises, institutions, or individuals, which provide a service on the basis of a common understanding of business. The co-operating units mainly contribute their core competence, whilst sharing skills, costs, and access to each other's market. To external partners they act, however, as a single corporation. The corporation refuses an institutionalisation, e.g., by central offices; instead, the co-operation is managed by using information and communication technologies [2,34].

Merz et al. [57] argue in contrast to this that current workflow management tools reduce the local autonomy of organisational units involved in co-operative workflow. While the required adaptation may be carried through within an organisation, several co-ordination problems may arise between them: (1) lack of communication infrastructure, (2) lack of central management, and (3) high co-ordination costs of the workflow management systems. Open network infrastructure and intelligent multi-agent systems provide a promising perspective to solving these problems. Intelligent user interfaces [91] and human computer co-operative work [8] are other threads of research, which will contribute to this development.

4.1.3. Electronic commerce

Nowadays commerce is almost entirely driven by human interactions. Humans decide when to buy goods and how much they are willing to pay. With the incredible success of the Internet and the WWW, this picture seems to change. Even today it is possible to order a computer hard and software, books, and CDs on the WWW and this is quite widely done. However, there is no reason why commerce cannot be automated to a larger degree. By this is meant that some commercial decision making can be placed in the hands of agents [46]. Although widespread electronic commerce is likely to lie some distance in the future, an increasing amount of trade is being undertaken by agents. Examples for this are electronic market places like Tete a Tete and Kasbah [12], which were developed at MIT's media lab. Kasbah implements a marketplace allowing users to create buying and selling agents for each good (books or music) to be purchased or sold, respectively. Commercial transactions then take place by means of the interactions of these agents. Other examples are general sites for private open auctions on the WWW like the auction bot, which was developed at the University of Michigan. More commercially oriented applications include BargainFinder [50] an agent which discovers the cheapest CDs, Jango [22] a personal shopping assistant able to search on-line stores for product availability and price information, MAGMA [87] a virtual marketplace for electronic commerce, and several agent-based interactive catalogues [76,84].

4.1.4. Intelligent information agents

Information agents are computational software systems that have access to multiple, heterogeneous and geographically distributed information sources as in the Internet or corporate Intranets. The main task of information agents is to perform active searches for relevant information in non-local domains on behalf of their users or other agents. This includes retrieving, analysing, manipulating, and integrating information available from multiple autonomous information sources. Intelligent information agents [49] may have different characteristics dependent on the concrete application domain, e.g. they can show adaptive, self-interested rational, or co-operative behaviour. Artificial intelligence, database systems, and information retrieval provide basic techniques, which can be applied to information discovery by or groups of information agents in the Internet and the World Wide Web. These techniques include, e.g. interoperability among database systems, efficient techniques from machine learning, evolutionary computing, and approaches for reasoning about uncertainty as well as information retrieval in sources with semi-structured or multi-media data. Especially relevant for information agents is the use of and the reasoning about ontologies [40] which allow the agents to interpret symbols with respect to a specific application domain. Applications of intelligent information agents range form relatively simple in-house information systems, through large-scale multi-database systems, to the visionary Infosphere ("Cyberspace") in the Internet. Commercial aspects of information gathering on the Internet are becoming more and more relevant: e.g., agents are paid and have to pay for services (electronic commerce and virtual agent marketplaces). The need for human–agent interaction in such environments, e.g., via synthetic characters, believable avatars or multi-media-based representation of the partly 'fuzzy' information space available for individual users on the Internet, is a challenging research topic [27].

4.2. Applications for agents with physical or virtual bodies

As already mentioned above this second class of application scenarios features agents that have either a physical or a virtual *Gestalt*. This means they have a physical or virtual body they are able to autonomously control in an environment that is closely related to the physical environment we actually live in. We do not make any difference between the physical and the virtual setting because, after all, it is impossible for a software program to actually verify the existence of a physical outside world in the same degree as this is impossible for the human mind. The difference of pure software agents and agents that have a physical or virtual *Gestalt* is that these agents have to do geometrical reasoning and that the perceptions and actions of such agents are often much more fine grained than that of pure software agents.

4.2.1. Autonomous control systems

There is a long history of research in engineering and AI on the design of autonomous systems, e.g. autonomous robots and machine tools in an FMS or autonomous robots exploring unknown environments. The traditional AI point of view to these kind of systems was that within these systems we have a control loop in which the system (1) gets perception, (2) integrates this perception into its internal world model, (3) reasons about this internal world model to find out which action it should perform next, and (4) eventually starts to actually execute the selected action. Brooks [7] radically criticised this approach and proposed the subsumption architecture for the design of autonomous systems. This approach almost completely denies the need for the representation of an internal world model in an autonomous agent. Instead the agents behaviour is directly triggered by the sensor data it perceives from the outside environment. Within an autonomous system several behaviours can be active in parallel and they interact in a predefined manner according to the rules of the subsumption architecture. Within AI and MAS research Brook's ideas led to a thread of research which investigates hybrid architectures for the design of autonomous systems which try to integrate reactive (Brook's style) and deliberative (AI style) behaviour within a uniform architecture. Most of these architectures propose three layers where the lowest layer deals with the reactive abilities, the second layer is responsible for deliberative problem solving, and the third layer tends to deal with the social abilities of the system [5,29,60]. For a long time researchers were involved with the investigation of an individual autonomous system acting in a moderately complex environment. It was only a few years ago that researchers actually started to investigate groups of such agents [32,48,60,80].

4.2.2. Entertainment

There is a large number of games available in which animated characters face challenges in a virtual world. Fighting and shooting games are the most prominent examples. However, there are also adventures available in which the actions of the characters are not so cruel. Grand and Cliff [39] brought together agent technologies and concepts from biology and designed the marvellous game Creatures. Creatures provide a rich, simulated environment containing a number of synthetic agents that a user can interact with. The agents are intended to be sophisticated pets whose development is shaped by their experience during their lifetime. Interactive theatre and cinema are other application examples, which are particularly demanding with respect to the abilities of the participating agents. In these settings, a user is able to play out a role analogous to those played by human actors in plays or films and in doing so the user is able to interact with artificial computer characters that should ideally be able to act like real people. Such agents that play the part of humans in theatre-style applications are often called *believable agents* – software programs *that provide the illusion of life, thus permitting [an] audience's suspension of disbelief* [3]. A number of projects have been set up to investigate the development of such agents [36,43,54,86].

4.2.3. Traffic telematics

Traffic telematics brings together two ideas: physical entities (normally vehicles, but in some cases even humans) and IT infrastructure to form a new class of applications. Three basic technologies form the basis for traffic telematics applications: (1) intelligent agents and multi-agent systems, (2) satellite and mobile phone communication, and (3) global positioning systems, which allow one to geographically locate a physical entity with a precision of 10–100 m. Examples of such applications are: fleet management, handling of an emergency situation, theft protection of vehicles, road pricing, and mobile office applications [35]. In an industrial or commercial context logistics applications – like, e.g. fleet management [34] – are most interesting. They allow one to extend the workflow of a production plant, a wholesaler, or a retailer and integrate it with the delivery of goods. This leads us to a situation in which the border line of where one company starts and another company ends becomes vague. We then have a situation in which there is a web of interacting companies which form virtual organisations to reach their individual goals (see Sections 4.1.2 and 4.1.3). Because we have in most traffic telematics applications a link between the physical entities and the workflow of an organisation, we have in most of these settings actually both types of agents pure software agents and agents with Gestalt.

Traffic telematics is therefore a typical example of an application domain with a hybrid agent structure.

5. Conclusions

MAS specific applications have been developed most of the time in an ad hoc manner. However, MAS is now envisaged as a new metaphor for (information or control) systems specification and implementation. This implies an urgent need for formal and systematic methods of problem analysis and agent-based specification of programs, through which distributed problem-solving capabilities and rich, efficient, and well co-ordinated interactions can be designed. We may say that there is a need for several simultaneous complementary (and sometimes overlapping) models of agency [44]. Theories behind the models take into account not only the agents' desired behaviour but also "subjective" motivations internal to the agents motivations based on their explicit intentions and commitments.

Current and future research is (should be) directed towards the enhancement of such models of agency which are responsible for agents (and multi-agent) rationality (logical and economic), sociability, interactivity and adaptability. Rational agency encompasses both a logical and economic rationality. As far as logics is concerned, mechanisms for both local as well as global coherence of an agent's knowledge and beliefs are being investigated. They can not only be based on previous single systems approaches (JTMS [23], ATMS [21]) since they have to meet the specific requirements of the distributed multi-agent system environment [55,56].

The logical rational model is also related to the representation of the agent's mental state behind its behaviour backing its decisions. How to describe the precise meaning of concepts like beliefs, know-how, intention, desires, goals, commitments and their respective relationships, is being deeply investigated. A sound definition of these concepts comes from [14,72]. However, these approaches take a logic-based perspective still and therefore poses difficulties like the omniscience problem (if an agent believes in proposition X then it believes in all equivalent propositions including all logical consequences of X) [97]. This problem is a strong argument against a purely logic-based approach, when we look at real systems that are necessarily resource-bounded.

Economic rational agency uses a simple principle. The best action to be selected by an agent is that one that maximises an appropriate scalar. The problem that is being addressed is how to reduce all different kinds of preferences, some of which may even be contradictory, to an appropriate utility function. Once agent preferences are well represented through utilities, either some negotiation process or elaborated market-based mechanism can be used for decision making [96].

Social agency is particularly important in the context of multi-agent systems. Agents have to co-operate through fair, useful and efficient interactions. Co-ordination policies in order to guarantee convergence as well as coherence, enhancement of agent models to make them aware of the environment and committed to other agents, all this implies the need for important laws to be included in such asocial agent models.

Interactive agency is one of the multi-agent facets that has been more studied and simple suitable agents communication languages are already available. Nevertheless, all the meaning of the communicative act is under the perspective of the sender agent. Recent proposals [51] also look at the post-conditions of communication, thus taking into account the receiver's perspective. A multi-agent system where agents share a social perspective of any communicative act would be much more aware of the entire global context embedding the interaction taking place.

Finally, adaptive agency is crucial for multi-agent systems in open environments. As it was discussed in Section 3.3, there are many challenges related to learning for MAS.

References

[1] P. Agre, D. Chapman, Pengi: an implementation of a theory of activity, in: Proceedings of the AAAI-97, Morgan Kaufmann, Los Altos, CA.

[2] O. Arnold, W. Faisst, M. Härtling, P. Sieber, Virtuelle Unternehmen als Unternehmenstyp der Zukunft?, in: Handbuch der Modernen Datenverarbeitung Theorie und Praxis der Wirtschaftsinformatik, vol. 185, Hüthig-Verlag, Heidelberg, 1995.

[3] J. Bates, The role of emotion in believable agents, Communications of the ACM 37 (7) (1994) 122–125.

[4] A. Bazzan, Evolution of coordination as a metaphor for learning in multi-agent systems, in: G. Weiss (Ed.), DAI Meets ML, LNAI 1221, Springer, Berlin, 1997, pp. 117–136.

[5] R.P. Bonasso, D. Kortenkamp, D.P. Miller, M. Slack, Experiences with an architecture for intelligent, reactive agents, in: M.J. Wooldridge, N.R. Jennings (Eds.), Intelligent Agents II – Theories, Architectures, and Languages, LNAI 890, Springer, Berlin, 1996, pp. 187–202.

[6] A. Bond, L. Gasser, Readings of Distributed Artificial Intelligence, Morgan Kaufman, Los Altos, CA, 1988.

[7] R. Brooks, A robust layered control system for a mobile robot, IEEE Journal of Robotics and Automation, V.RA-2 (1) (1986).

[8] H. Bunt, R.-J. Beun, T. Borghuis, Multimodal Human Computer Communication Systems, Techniques and Experiments, LNAI, Springer, Berlin, 1998.

[9] S. Bussmann. A multi-agent approach to dynamic, adaptive scheduling of material flow, in: J.W. Perram, J.-P. Müller (Eds.), Distributed Software Agents and Applications (MAAMAW'94), LNAI 1069, Springer, Berlin, 1996, pp. 191–205.

[10] S. Bussmann, An agent-oriented architecture for holonic manufacturing control, in: Proceedings of the First International Workshop on IMS, EPFL Europe, Lausanne, Switzerland, 1998.

[11] J. Butler, H. Ohtsubo, ADDYMS: architecture for distributed dynamic manufacturing scheduling, in: A. Famili, D.S. Nau, S.H. Kim (Eds.), Artificial Intelligence Applications in Manufacturing, AAAI Press/MIT Press, 1992, Cambridge, MA, pp. 199–213.

[12] A. Chavez, P. Maes, Kasbah: an agent marketplace for buying and selling goods, in: Proceedings of the First International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology (PAAM-96), London, UK, 1996, pp. 75–90.

[13] J. Christensen, Holonic manufacturing systems: initial architecture and standard directions, in: Proceedings of the First European Conference on Holonic Manufacturing Systems, European HMS Consortium, Hannover, Germany, 1994.

[14] P.R. Cohen, H.J. Levesque, Persistence, intention and commitment, in: Cohen, Morgan, Pollack (Eds.), Intentions and Communications, MIT Press, Cambridge, MA, 1990.

[15] W.H. Davidow, M.S. Malone, The Virtual Corporation: Structuring and Revitalizing the Corporation for the 21st Century, Harper, New York, 1992.

[16] W. Davies, P. Edwards, Distributed learning: an agent based approach to data-mining, University of Aberdeen, 1996.

[17] R. Davis, R.G. Smith, Negotiation as a metaphor for distributed problem solving, Artificial Intelligence 20 (1983) 63–109.

[18] K. Decker, V. Lesser, Designing a family of coordination algorithms, Proceedings of the First ICMAS, San Francisco, 1995.

[19] S.M. Deen, A cooperation framework for holonic interactions in manufacturing, in: Proceedings of the Second International Working Conference on Cooperating Knowledge Based Systems, DAKE Centre, Universtity of Keele, June 1994, pp. 103–124.

[20] E.D. De Jong, The development of a lexicon based on behaviour, in: Proceedings of the Tenth Dutch Conference on Artificial Intelligence NAIC'98.

[21] J. De Kleer, An assumption-based truth maintenance system, Artificial Intelligence 28 (1986).

[22] R. Doorenbos, O. Etzioni, D. Weld, A scalable comparison-shopping agent for the world-wide web, in: Proceedings of the First International Conference on Autonomous Agents (Agents 97), Marina del Rey, CA, 1997, pp. 39–48.

[23] J. Doyle, A truth maintenance system, Artificial Intelligence 12 (1979).

[24] E. Durfee, V. Lesser, Using partial global plans to coordinate problem solvers, in: Proceedings of the IJCAI, Los Angeles, 1985.

[25] E. Durfee, Coordination of Distributed Problem Solvers, Kluwer Academic Publishers, Boston, 1988.

[26] E. Durfee, J. Rosenschein, Distributed problem solving and multi-agent systems: comparisons and examples, University of Michigan, 1994.

[27] European network for intelligent information interfaces, HYPERLINK http://www.i3net.org/ http://www.i3net.org/, Major Esprit Long-Term, Research Pro-Active Initiative.

[28] J. Ferber, Reactive distributed artificial intelligence: principles and applications, in: G. O'Hare, N. Jennings (Eds.), Foundations of Distributed Artificial Intelligence, Wiley, New York, 1995.

[29] I.A. Ferguson, Integrated control and coordinated behaviour, in: M.J. Wooldridge, N.R. Jennings (Eds.), Intelligent Agents Theories, Architectures, and Languages, LNAI 890, Springer, Berlin, 1995, pp. 203–218.

[30] T. Finin, R. Fritzson, D. Mckey, R. McEntire, KQML as an agent communication language, in: Proceedings of the International Conference on Information and Knowledge Management, ACM Press, New York, 1994.

[31] R.J. Firby, Building symbolic primitives with continuous control routines, in: Proceedings of the First International Conference on AI and Planning Systems, Morgan Kaufmann, Los Altos, CA, 1992.

[32] K. Fischer, Concepts for an agent system in a flexible manufacturing system, in: Proceedings of the 23rd International Symposium on Automative Technology and Automation, Vienna, December 1990.

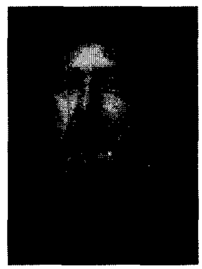
[33] K. Fischer, Knowledge-based reactive scheduling in a flexible manufacturing system, in: E. Szelke, R.M. Kerr (Eds.), Knowledge-Based Reactive Scheduling, North-Holland, Amsterdam, 1994.

[34] K. Fischer, J.P. Müller, M. Pischel, A pragmatic BDI architecture, in: M. Wooldridge, J.P. Müller, M. Tambe (Eds.), Intelligent Agents II: Agent Theories, Architectures and Languages, LNAI 1037, Springer, Berlin, 1996.

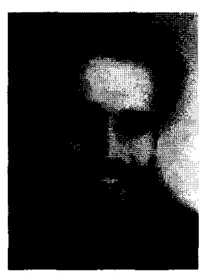
[35] K. Fischer, An Agent-based approach to holonic manufacturing system, in: Intelligent Systems for Manufacturing-Multiagent Systems and Virtual Organisations, Kluwer Academic Publishers, Dordrecht 1998.

[36] L.N. Foner, Entertaining agents: A sociological case study, in: Proceedings of the First International Conference on Autonomous Agents (Agents 97), Marina del Rey, CA, 1997, pp. 122–129.
[37] M. Georgeff, A. Rao, The semantis of intention maintenance for rational agents, in: Proceedings of IJCAI, 1995.
[38] C.V. Goldman, A. Langer, J.S. Rosenschein, Musag: an agent that learns what you mean, in: Proceedings of International Conference on Practical Application of Intelligent Agents and Multi-agent Technology, pp. 311–319.
[39] S. Grand, D. Cliff, Creatures: entertainment software agents with artificial life, Autonomous Agents and Multi-Agent Systems 1 (2) (1998).
[40] T.R. Gruber, A translation approach to portable ontology specifications, Knowledge Acquisition 5(2) (1993).
[41] S. Hahndel, P. Levi, Optimizing distributed production planning, in: Proceedings of the Second International Conference on Intelligent Systems Engineering'94, Hamburg, September 1994, pp. 419–424.
[42] J. Harrington, Computer integrated manufacturing, Reprint, Malabar, FL, 1993.
[43] B. Hayes-Roth, Agents on stage: advancing the state of the art in AI, in: Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI-95), Montréal, Que., Canada, August 1995, pp. 967–971.
[44] M. Huhns, M. Singh, Readings in Agents, Morgan Kaufmann, Los Altos, CA, 1997.
[45] JATLite, URL: htt://java.stanford.edu/java_agent/html/.
[46] N. Jennings, Coordination: commitment and convention, the foundation of coordination in multiagent systems, Knowledge Engineering Review 8 (3) (1993).
[47] N. Jennings, K. Sycara, M. Wooldridge, A road map of agent research and development, Agents and Multi-Agent Systems 1 (1) (1998).
[48] K. Kanazawa (Ed.), Working Notes of the IJCAI'93 Workshop on Dynamically Interacting Robots, Chamberz, 1993.
[49] M. Klusch (Ed.), Intelligent Information Agents – Cooperative, Rational and Adaptive Information Gathering on the Internet, Springer, Berlin, March 1999.
[50] B. Kurlwich, The bargain finder agent: comparison price shopping on the internet, in: J. Williams, (Ed.), Bots, other Internet Beasties, Macmillan, Indianapolis, 1996, pp. 257–263.
[51] Y. Labrou, T. Finin, Semantics and conversations for an Agent Communication Language, in: Proceedings of the IJCAI, 1997.
[52] Y. Lesperance, H. Levesque, F. Lin, D. Marcu, R. Reiter, R. Scherl, Foundations of a logical approach to agent programming, in: M. Wooldridge, J. P. Müller, M. Tambe (Eds.), Intelligent Agents II: Agent Theories, Architectures and Languages, LNAI 1037, Springer, Berlin, 1996.
[53] V. Lesser, Autonomous Agents and Multiagent Systems 1(1) (1998).
[54] J.C. Lester, B.A. Stone, Increasing believability in animated pedagogical agents, in: Proceedings of the First International Conference on Autonomous Agents (Agents 97), Marina del Rey, CA, 1997, pp. 16–21.
[55] B. Malheiro, E. Oliveira, Consistency, context management in a multi-agent belief revision testbed, in: M. Wooldridge, J.P. Müller, M. Tambe (Eds.), Agent Theories, Architectures and Languages, LNAI 1037, Springer, Berlin, 1996.
[56] C. Mason, R. Johnson, DATMS: A framework for distributed assumption based reasoning, in: L. Gasser, M. Huhns (Eds.), Distributed Artificial Intelligence II, Morgan Kaufmann, Los Altos, CA, 1989.
[57] M. Merz, B. Liebermann, W. Lamersdorf, Using mobile agents to support inter-organisational workflow management, Applied Artificial Intelligence 11 (6) (1997) 551–572.
[58] T.M. Mitchell, Machine Learning, Morgan Kaufman, Los Altos, CA, 1997.
[59] A. Moshowitz, Social dimensions of office automation, Advances in Computers 25 (1986) 335–404.
[60] J.P. Müller, The Design of Intelligent Agents: A Layered Approach, LNAI 1177, Springer, Berlin, 1996.
[61] J.P. Müller, The right agent (architecture) to do the right thing, in: Proceedings of the ATAL'98 Workshop, Paris, 1998.
[62] M. Neves, E. Oliveira, A control architecture for an autonomous mobile robot, in Proceedings of First International Conference on Autonomous Agents, Marina del Rey, CA, 1997.
[63] A. Newell, H. Simon, Computer science an empirical inquiry: symbols and search, Communications of the Association for Computing Machinery 19 (4) (1976) 113–126.
[64] T. Ohko, K. Hiraki, Y. Anzai, Addressee learning and message interception for communication load reduction in multiple robot environments, in: G. Weiss (Ed.), DAI Meets ML, LNAI 1221, Springer, Berlin, 1997, pp. 242–258.
[65] E. Oliveira, Robots as responsible agents, in: Proceedings of the IEEE Systems Man and Cybernetics, Orlando, 1997.
[66] E. Oliveira, J.M. Fonseca , A. Garço, International Journal on Applied Artificial Intelligence 11 (6) (1997).
[67] N. Ono, K. Fukumoto, A modular approach to multi-agent reinforcement learning, in: G. Weiss (Ed.), DAI Meets ML, LNAI 1221, Springer, Berlin, 1997, pp. 25–39.
[68] N. Ono, Y. Fukuta, Learning coordinated behaviour in a continuous environment, in: G. Weiss (Ed.), DAI Meets ML, LNAI 1221, Springer, Berlin, 1997, pp. 73–81.
[69] P.S. Ow, S.F. Smith, R. Howie, A cooperative scheduling system, in: M.D. Oliff (Ed.), Expert System and Intelligent Manufacturing, 1988, pp. 43–56.
[70] N. Prasad, V. Lesser, S. Lander, Retrieval and reasoning in distributed case bases, Journal of Visual Communication and Image Representation (1996) 74–87.
[71] A.L. Prodromidis, S. Stofo, Mining databases with differnt schemas: integrating incompatible classifiers, in: Proceedings of the Knowledge Discovery in Databases'98.
[72] A. Rao, M. Georgeff, Modeling rational agents within a BDI architecture, in: Proceedings of the 13th IJCAI, Chambery, 1991.

[73] J. Rosenshein, G. Zlotkin, Rules of Encounter, MIT Press, Cambridge, MA, 1994.
[74] S.J. Russell, Rationality and Intelligence, Artificial Intelligence 94 (1997) 57–77.
[75] T. Sandholm, V. Lesser, Issues in automated negotiation and electronic commerce: extending the contract net framework, in: Proceedings of the First ICMAS, San Francisco, 1995.
[76] R. Schrooten, W. van de Velde, Software agent foundation for dynamic interactive electronic catalogues, Applied Artificial Intelligence 11 (5) (1997) 459–482.
[77] Y. Shoam, Agent-oriented programming, Artificial Intelligence 60 (1) (1993).
[78] Y. Shoham, M. Tennenholz, On emergence of social conventions: modeling, analysis, and simulations, Artificial Intelligence 94 (1997) 139–166.
[79] M. Singh, Multiagent Systems: A Theoretical Framework for Interaction, Know-how and Communication, LNAI 799, Springer, Berlin, 1994.
[80] L. Steels, Cooperation between distributed agents through self-organisation, in: Y. Demazeau, J.-P. Müller (Eds.), Decentralized AI, North-Holland, Amsterdam, 1990, pp. 175–196.
[81] L. Steels, P. Vogt, Grounding adaptive language games in robotic agents, in: P. Harvey, P. Husbands (Eds.), Proceedings of the Fourth European Conference on Artificial Life 97.
[82] O. Stepankova, V. Marik, J. Lazansky, Improving cooperative behaviour in a multi-agent system, in: Proceedings of the Fourth World Computer Congress IFIP'96, Canberra, 1996, Chapmann & Hall, London, pp. 293–302.
[83] K. Sycara, Negotiation planning: an AI approach, European Journal of Operational Research 46 (1990).
[84] K. Takahashi, Y. Nishibe, I. Morihara, F. Hattori, Intelligent pages: collecting shop and service information with software agents, Applied Artificial Intelligence 11 (6) (1997) 489–500.
[85] M. Terabe et al., A study of organizational learning in multiagents systems, in: G. Weiss (Ed.), DAI Meets ML, LNAI 1221, Springer, Berlin, 1997, pp. 168–179.
[86] R. Trappl, P. Petta, Creating Personalities for Synthetic Actors, Springer, Berlin, 1997.
[87] M. Tsvetovatyy, M. Gini, B. Mobasher, Z. Wieckowski, MAGMA: an agent-base virtual marketplace for electronic commerce, Applied Artificial Intelligence 11 (6) (1997) 501–524.
[88] H. Van Dyke Parunak, Manufacturing experience with the contract net, in: M.N. Huhns (Ed.), Distributed Artificial Intelligence, Morgan Kaufmann, Los Altos, CA, 1987, pp. 285–310.
[89] J.M. Vidal, E.H. Durfee, Agents learning about agents: a framework and analysis, Workshop on Learning Multi-Agent Systems, AAAI'97.
[90] J.M. Vidal, E.H. Durfee, Learning nested models in an infomation economy, Journal of Experimental and Theoretical Artificial Intelligence, Special Issue Learning in DAI Systems (1998) http://ai.eecs.umich.edu/people/jmvidal/papers.
[91] W. Wahlster (Ed.), Readings in Intelligent User Interfaces, Morgan Kaufmann, Los Altos, CA, 1997.
[92] H.-J. Warnecke, M. Hüser, The Fractal Company – A Revolution in Corporate Culture, Springer, Berlin, 1995.
[93] Ch. Watkins, P. Dayan, Technical note Q-learning, Machine Learning 8 (1992) 279–292.
[94] G. Weiss, J. Sen (Eds.), Adaptation and Learning in Multi-Agent Systems, LNAI 1042, Springer, Berlin, 1996.
[95] G. Weiss (Ed.), DAI Meets ML, LNAI 1221, Springer, Berlin, 1997.
[96] M. Wellman, A computational market model for distributed configuration design, AI for Engineering, Design and Manufacturing 9 (1995).
[97] M. Wooldridge, N. Jennings, Intelligent agents: theory and practice, The Knowledge Engineering Review 10 (2) (1995) 115–152.



Eugenio C. Oliveira is the head of the Artificial Intelligence & Robotics Group in the Faculty of Engineering of the University of Porto. He is also co-director of LIACC the Artificial Intelligence and Computer Science Laboratory at the same University. His main research interests are in the multi-agent systems area, especially on agents' architecture, negotiation and cooperation issues. He also carries out research in intelligent robotics (planning and hybrid multi-layer architecture). He is (and has been) member of several Conferences Programme Committees and co-chair of several workshops on multi-agent systems. He graduated in Electronics in 1972 from Porto University; he was R & D Engineering at Brown Boveri (now ABB) in Switzerland (1972–1974). He received a Ph.D. from the New University of Lisbon (1984) on Artificial Intelligence. He was "guest scientist" at IBM IEC in La Hulpe, Belgium in 1985. He is Associate Professor with habilitation at the University of Porto.



Klaus Fischer studied computer science at the Technische Universität (TU) in Munich. During 1986–1991 he worked in a joint research project SFB 331 *Information Processing in Autonomous Mobile Robot Systems* at the Department of Computer Science, TU, Munich. In 1992 he finished his doctoral degree with his thesis on *Distributed and Co-operative Planning in a Flexible Manufacturing System.* In 1992 he joined the Multi-Agent System Research Group at DFKI GmbH in Saarbrücken in the department of Deduction and Multi-Agent Systems headed by Prof. J.H. Siekmann and assumed the responsibility of group leader in November 1993.

Olga Stepankova graduated in 1972 from the Faculty of Mathematics and Physics of the Charles University in Prague and she received the doctoral degree in 1981. Since 1988, She is with the Department of Control Engineering, Faculty of Electrical Engineering, Czech Technical University in Prague. There she became associate professor in 1992 and she is a vice-head of this department at the moment. Her research is bound to topics on the border between mathematical logics and artificial intelligence starting with situation calculus in 1970s, logic programming in 1980s and distributed AI as well as inductive logic programming, now.