MULTIAGENT SYSTEMS
A MODERN APPROACH TO DISTRIBUTED ARTIFICIAL INTELLIGENCE

Gerhard Weiss
The MIT Press,
Cambridge, Massachusetts, London, England
ISBN 0-262-23203-0

Reviewed by:
Leonid Sheremetov,
Matías Alvarado Mentado
CIC-IPN

"Agents are the next major computing paradigm and will be pervasive in every market by the year 2000."
Janca, P. “Pragmatic Applications of Information Agents”,

Since the beginning of recorded history, people have been fascinated with the idea of non-human agencies. Wide spreading of computer technologies has permitted to shift from mechanic to digital, from hardware to software views on how to build agents and agencies o multiagent systems (MAS). Tracing the more recent roots of MAS, it should be mentioned that the idea of an agent originated with John McCarthy in the mid-1950’s, and the term was coined by Oliver G. Selfridge a few years later, when they were both at the Massachusetts Institute of Technology. Later, a new way of programming and understanding of complex systems by means of interaction between agents was realized by Hewitt. Since then, these ideas have been developed within a number of frameworks and now agent technology radically alters the way in which we conceptualize and build software systems. Today these systems are not simply a research topic, but are also beginning to become an important subject of academic teaching and industrial and commercial application.

Recently, there has been an explosion in the use of the term “agent” without a corresponding consensus on what it means. Some researchers, such as Nwana (1996) splits agent research into two main strands: the first beginning about 1977 with the works of Hewitt, whose roots are mainly in distributed artificial intelligence (DAI), and the second around 1990 is a recent, rapidly growing movement to study a much broader range of agent types.

Strand 1 has concentrated mainly on deliberative-type agents with symbolic internal models. Such work has contributed to an understanding of macro issues such as the interaction and
interaction, intelligent and adaptive interfaces, intelligent search and filtering, information retrieval, knowledge acquisition, and a growing list of other fields. The natural multi-disciplinary view on agents was one of the main reasons to bring together the leading researchers to cover the main themes of MAS. Careful coordination of the materials ensures the coherence of this book and contributed to the book’s success.

The book is divided into two parts. It reflects the state of the art in the field of MAS, and treats basic themes (Part I) giving a clear and careful presentation of the key concepts, methods, and algorithms that form the core of the field, as well as several closely related themes (Part II) in detail. Part I contains nine chapters, and in the following, the individual chapters and their themes are discussed in more detail.

Chapter 1, “Intelligent Agents” written by Michael Wooldridge, aims to introduce the reader to the basic issues surrounding the design and implementation of intelligent agents, considering four main agent architectures: agents based, reactive, belief-desire-intention, and, finally, layered architectures. In logic based architectures, decision-making is viewed as logical deduction; the process of deciding which action to perform is reduced to a theorem proving problem. Despite of semantic clarity, and all the power of the apparatus of logic and theorem proving brought to agents development, such architectures suffer from a number of drawbacks and do not seem well suited to domains that are subject to real time constraints. Reactive architectures, on the other hand, eschew symbolic representations and reasoning in favour of a closer relationship between agent perception and action. Such architectures seems to be more economical in computational terms, making them well-suited to episodic environments that require real-time performance. However, the process of engineering such architectures is not well understood. Third, in belief-desire-intention (BDI) architectures, decision making is viewed as practical reasoning from beliefs about how the world is and will be to the options available to an agent, and finally to intentions and actions. Belief-desire-intention architectures also have an attractive formalization, discussed also in the Chapter 8 of this book. Fourth, in layered agent architectures, decision making is partitioned into a number of different decision making layers, each dealing with the agent’s environment at a different level of abstraction, providing a natural way of decomposing agent functionality, and are currently a popular approach to agent design.

Chapter 2, “Multiagent Systems and Societies of Agents” written by Michael N. Huhns and Larry M. Stephens describes protocols for communication and interaction among groups of agents, based on social commitments, laws, and conventions. Usually, agents operate and exist in some computational and physical environment. The increasing interconnection and networking of computers is making the usual situations of interaction among agents. One of the key elements of the computational infrastructure for such interactions includes communication protocols, which enable information exchange between two agents. For instance, Agent 1 proposes a course of action to Agent 2; Agent 2 evaluates the proposal and sends a counterproposal to Agent 1; Agent 1 accepts the counterproposal. This chapter provides to theorist and developers a qualified overview about interaction protocols. These protocols enable agents to coordinate their activities, which can then be performed more efficiently and maintaining applicable safety conditions. Cooperation refers to coordination among nondisputant agents, while negotiation -to coordination among competitive or simply self-interested agents. The problems of cooperative behaviour are discussed in more details in Chapter 3.

Chapters 3 to 6 address elementary “intelligent activities” and their realization in multiagent systems, namely, distributed problem solving and planning, search, decision making, and learning.

Chapter 3, “Distributed Problem Solving (DPS) and Planning” written by Edmund H. Durfee focuses on strategies for using protocols and reasoning capabilities to realize the benefits of cooperation. It is worth to mention that research work in DPS was one of the main areas where the agent approach was born. Though, this chapter considers only one particular type of agents, the cooperative ones, which are aiming to reach a common goal, a number of crucial questions for the agent development are discussed. The necessity of cooperation raises from the fact that the problem can’t be solved by one particular agent and mutual expertise is necessary. Of particular interest are strategies for moving tasks or results among agents to realize the benefits of cooperative problem solving. The main threads of work is the development of task-passing techniques to decide where to allocate subtasks and the study of result-sharing strategies to decide how agents that might be working on pieces of larger task can discover the relationships among their activities and coordinate them. The specific requirements and representations of planning problems, considered in the chapter, allow us to identify techniques that are particularly suited for distributed planning. An interesting issue arises as to whether the coordination process should precede or succeed the planning processes of the agents; different decisions lead to different flavors of distributed planning, and a perspective is presented that allows these approaches to be seen as extremes of a more general process.
Chapter 4, "Search Algorithms for Agents" written by Makoto Yokoo and Torn Ishida applies for search problem solving techniques, where the sequence of actions required solving a problem are determined by a trial-and-error exploration of alternatives. Throughout this chapter, classical search problem solving techniques in AI as well as the emergent MAS themselves are introduced in a smart way. Search problems are divided into path-finding problems, constraint satisfaction problems, and two-player games such as chess and checkers. *Asynchronous search* algorithms, useful for cooperative problem solving by multiple agents are described. In these algorithms, by accumulating local computations for each agent, a problem is solved. The execution order of these local computations can be arbitrary or highly flexible. With respect to constraint satisfaction problems, the filtering algorithm, the hyperresolution-based consistency algorithm, the asynchronous backtracking algorithm, and the asynchronous weak-commitment search algorithm are presented. Concerning path-finding problems, the Learning Real-time A* algorithm, the Real-time A* algorithm, the Moving Target Search algorithm, Real-time Bidirectional Search algorithms, and real-time multiagent search algorithms, as special cases of asynchronous dynamic programming are described. With respect to two-player games, the basic minimax procedure and the alpha-beta pruning method to speed up the minimax procedure are presented.

Chapter 5, "Distributed Rational Decision Making" written by Thomas W. Sandholm is a complementary to the Chapter 3 and deals with another aspect of MAS, systems consisting of self-interested agents, which not necessarily are cooperative. Many applications, such as electronic commerce on the Internet for purchasing goods and information, or virtual enterprises: dynamic alliances of small, agile enterprises which together can take advantage of economies of scale when available - are based on the strong need for computer support for negotiation at the operative decision making level. Automated negotiation can save labor time of human negotiators, but in addition, other savings are possible because computational agents can be more effective at finding beneficial short-term contracts than humans are in strategically and combinatorially complex settings. This chapter discusses methods for making socially desirable decisions among rational agents that only care of their own good, and may act insincerely to promote it, considering voting, auctions, bargaining, contracting, and other techniques. Special emphasis is placed on the implications of limited computation on the classic results. This is one area where game theory and computer science fruitfully blend within the field of DAI. It should also be mentioned that different negotiation techniques considered in the chapter can be especially useful for conflict resolution between cooperative and competing agents.

Chapter 6, "Learning in Multiagent Systems" written by Sandip Sen and Gerhard Weiss considers the problems of how really autonomous and adaptive agents can be developed. Most application of multiagent systems are extremely difficult (sometimes impossible) to correctly specify the behavioral repertoires and concrete activities of them at design time. That is in spite of the fact that the environments appear to be more or less simple at a first glance. This would require to known in advance, which environmental requirements will emerge in the future, which agents will be available at the time of emergence, and how the available agents have to interact in response to these requirements, among others issues to be considered. Often the only feasible way to cope with this kind of problems is to endow the agents themselves with the ability to learn appropriate activity and interaction patterns. *Centralized and decentralized* learning in multiagent systems are the principle learning categories considered. Alongside, an overview of *differencing features* structuring the kinds of learning that may occur in multiagent systems is given, joined with a description of the basic learning problem known as the *credit-assignment problem*, from the point of view of MAS. Then typical learning approaches are described and illustrated according to *i*) learning and activity coordination, *ii*) learning about and from other agents; and *iii*) learning and communication. A summary of the major related work from machine learning, psychology, and economics is mentioned. Directions of relevant future research concern with parallel and distributed inductive learning, multistrategy learning and theory of team learning. Our opinion is that this chapter is an opportune state of the art about MAS learning.

Chapter 7, "Computational Organization Theory" written by Kathleen M. Carley and Les Gasser, provides an overview of the emergent field of Computational Organization Theory (COT). In most of this work, organizations are characterized as multiagent systems in which agents are embedded in particular social roles, have particular cognitive capabilities, and are engaged in specific organizationally-relevant tasks. In general, the aim of research in this area is to build new concepts, theories, and knowledge about organizing and organization in the abstract, to develop tools and procedures for the validation and analysis of computational organizational models, and to reflect these computational abstractions back to actual organizational practice through both tools and knowledge. Some issues addressed in this chapter include the dominant approaches and models in this area, potential toolkits, new findings, directions, and trends. Some computational frameworks, such as MACE, SDML, Multiagent Soar are discussed from the point of view of social modeling of multiagent organizations. There is no doubt that the problems of multiagent organizations are of great importance for the MAS developing.
Nevertheless this chapter seems more to be a resume of the current state of the art in the field then a basic didactic material. We think that it could be better placed in the Part II of the book.

Chapter 8, “Formal Methods in DAI” written by Munindar P. Singh, Anand Rao, and Michael P. Georgoff focuses on the development of formal bases for the abstractions and constructions for agents and multiagent systems. Importance to develop techniques to ensure that DAI systems behave safe and correct is due to the fact that DAI moves into larger and more critical applications each day. As in traditional software engineering this can be achieved developing methodologies founded upon rigorous characterizations of the architecture and systems behavior. Thus, modal, dynamic and temporal logics that are commonly used in traditional computer science (concurrent programs verification), are included. In addition the concepts of knowledge, beliefs, desires, goals, intentions, and know-how, that offer high-level specifications of the agents’ design and behavior based on those logics and practical reasoning are introduced. It allows the independence of most implementation details. Complementary, how those concepts may be realized in a practical interpreter is described. Discussion about coordination, teamwork, interagent communications, and social primitives together with concepts concerning joint and group intentions, which lift single-agent primitives to multiagent systems, are presented. All these topics provide the essential conceptual basis for multiagent systems and the chapter overview is a straight way to get in.

Chapter 9, “Industrial and Practical Applications of DAI” written by H. Van Dyke Parunak, attempts to bridge the gap between the perspectives of a researcher seeking practical problems to demonstrate the usefulness of the particular capabilities of MAS (considered in details in Chapters 1 to 8) and an industrial practitioner who has a practical problem to solve. It offers an overview of the kinds of problems that industrialists face, and some examples of agent technologies that have made their way into practical application. On the other hand, it explains why agents are not just the latest technical fad, but a natural match to the characteristics of a broad class of real problems. Up to date, agent technology have been applied in a great number of application. Chapter 9 emphasizes agent applications in manufacturing and physical control because good examples from these areas are available, the problems of interfacing agents to the environment are more challenging than in all-electronic domains, and the evidence of success or failure is clearer when a system must directly confront the laws of physics. For two sets of case studies, it shows first where in the life cycle agent-based systems are used, and second, discusses the design and construction of an agent-based system in terms of the life cycle. The chapter also includes a review of some development tools that will hasten deployment of agent technology in industry.

Part II includes chapters on closely related, selected themes from computer science and software engineering. These themes are usually called “agent enabling technologies”. Chapter 10, written by Clarence Ellis and Jacques Wainer, focuses on groupware and computer supported cooperative work. Chapter 11, written by Jose Cuena and Sascha Ossowski, concentrates on distributed decision support systems. Chapter 12 written by Gul A. Agha and Nadeem Jamali, discusses various issues of concurrent programming. And finally, Chapter 13 written by Gerard Tel, describes distributed control algorithms. The relevance of these themes for the field can be easily seen. Agents in a multiagent system often have to coordinate their activities, and so there is a need for technologies that support them in acting coherently as a group; additionally, groupware and computer supported cooperative work constitute an important application domain for multiagent systems. Agents in a multiagent system often have to jointly make decisions, and so there is a need for technologies that support them in their distributed decision processes; moreover, distributed decision making is another obvious application domain for multiagent systems. As is was seen from the Part I, there is a need for powerful computation techniques, such as concurrent programming techniques that allow to efficiently implement multiagent systems as parallel and distributed systems, and mechanisms and methods that enable agents to control their distributed computations.

We consider that this book will be of particular interest to the professors and students who require an up-to-date, in-depth source of material for their courses on multiagent systems and DAI. The book is good enough as a text book since many illustrations and examples are provided that illustrate the main theoretic concepts. On the other hand, the emphasis is not only on theory, but also on practice. In particular, the book includes a number of thought-provoking exercises of varying degrees of difficulty at the end of each chapter that allow the reader to gain practical experience.

Part I of the book can be used for the didactic support of the complete graduate course on DAI. The book can also be useful as a supplementary text for a general AI course; for instance, within which the considerations on “classical” AI topics like problem solving and search could be enriched by Chapter 3 and Chapter 4, respectively. Moreover, most chapters could be also used as the material for specialized courses and seminars; for instance, Chapter 5, Chapter 6, and Chapter 7 could be used for courses devoted to distributed decision
make decisions, and so there is a need for technologies that support them in their distributed decision processes; moreover, distributed decision making is another obvious application domain for multiagent systems. As is was seen from the Part I, there is a need for powerful computation techniques, such as concurrent programming techniques that allow to efficiently implement multiagent systems as parallel and distributed systems, and mechanisms and methods that enable agents to control their distributed computations.

We consider that this book will be of particular interest to the professors and students who require an up-to-date, in-depth source of material for their courses on multiagent systems and DAI. The book is good enough as a text book since many illustrations and examples are provided that illustrate the main theoretic concepts. On the other hand, the emphasis is not only on theory, but also on practice. In particular, the book includes a number of thought-provoking exercises of varying degrees of difficulty at the end of each chapter that allow the reader to gain practical experience.

Part I of the book can be used for the didactic support of the complete graduate course on DAI. The book can also be useful as a supplementary text for a general AI course; for instance, within which the considerations on “classical” AI topics like problem solving and search could be enriched by Chapter 3 and Chapter 4, respectively. Moreover, most chapters could be also used as the material for specialized courses and seminars; for instance, Chapter 5, Chapter 6, and Chapter 7 could be used for courses devoted to distributed decision making, distributed machine learning, and computational organization theory, respectively.

The exercises allow the reader to further deepen her or his knowledge, and course instructors might use them for putting more emphasis on practical aspects. Some exercises are fairly simple and are intended to make sure that the material provided in the chapters is mastered. Others are much more difficult and may serve as a subject of class discussions or advanced team work.

The book will be very useful as well for the active researchers in the field who wish to branch out beyond the area in which they are specialized to better understand the field as a whole, to investigate relationships between their own work and work by others, and to obtain valuable stimuli for their future research activities.

Software practitioners and professionals from industry who want to find out whether and how the technologies available in the field can be usefuly applied in their working domains, can find some difficulties because some of the very important points are beyond the scope of the book. We speak about namely the implementation and application points of view.

Such topics as the methodology of software development with agents, would be of particular interest for this category of readers. Sure, one book can’t cover all the topics so we would recommend to these readers the recent books by Bradshow (1998), by Knapik and Johnson (1998), and by Wooldridge and Rao (1999).

The book that we are presenting here is a global and without any doubt will convert into the one of the texts of obliged reference in the scientific community working on agents. It gives us a global view on the research work which has been done in the area and how much is to be done yet. We strongly recommend to those interested in the latest advances in the AI and advanced methodology of software development, read this book. Agents are coming, they have already come.

**References:**


