Overview of All Papers

Tim Leys

October 2018

1 Introduction

In this document contains an overview of all papers that were considered in the literature review.

References

[1342925] A. El Fallah-Seghrouchni and A. Suna. Programming mobile intelligent agents: an operational semantics. In *Proceedings. IEEE/WIC/ACM International Conference on Intelligent Agent Technology, 2004. (IAT 2004).*, pages 65–71, Sept 2004. Amount of References: 8.

Abstract: This work presents the operational semantics of an agent oriented programming language called CLAIM. CLAIM allows the design of multiagent systems that support both stationary and mobile agents. Agents designed using CLAIM are endowed with cognitive capabilities, are able to communicate with other agents and are mobile. The primitives of mobility are inspired from the ambient calculus. The operational semantics reduction rules of CLAIM are illustrated on a practical example.

[1342988] W. Chainbi. Multi-agent systems: a petri net with objects based approach. In *Proceedings. IEEE/WIC/ACM International Conference on Intelligent Agent Technology, 2004. (IAT 2004).*, pages 429–432, Sept 2004. Amount of References: 9.

Abstract: Most of time, an agent theory is expressed by modal logic which is a good specification tool since it eases the description of intentional agents. However, this formalism cannot be easily refined into implementation even if some counterexamples exist. This work proposes another type

of formalism called cooperative objects which are based on Petri nets and objects. We show how it may be used with benefits to model multi-agent systems. More precisely, we argue that the presented formalism can bridge the gap between the world of formal theory and the world of system development. We illustrate our study with the well known prey/predators game.

[4441679] A. L. Dimeas and N. D. Hatziargyriou. Design of a mas for an island system. In *2007 International Conference on Intelligent Systems Applications to Power Systems*, pages 1–3, Nov 2007. Amount of References: 11.

Abstract: The introduction by the Greek state of a new law that regulates the numerous non interconnected Greek Islands creates an interesting area for using multi agent system technology. The scope of this paper is to analyze the new code and provide an formal UML description of the MAS. This is a critical issue in multi agent systems and in complex systems, since the complexity need to be organized. Furthermore a simplified version of the system is planned to be installed in the island of Kythnos.

[4725646] L. Yuefeng and Z. An. Multi-agent system and its application in combat simulation. In *2008 International Symposium on Computational Intelligence and Design*, volume 1, pages 448–452, Oct 2008. Amount of References: 7.

Abstract: Most conventional methods show their disadvantages in modern combat simulation. Multi-agent system (MAS) is an important research field of distributed artificial intelligence (DAI) and it proposes a new way of combat simulation. The focus of this paper is the MAS theory and its application in combat simulation. Definition of agent and MAS was proposed firstly and then the differences between MAS based and conventional combat simulation methods ware analyzed. After a discussion over the characteristics of combat, we found the combat was a complex adaptive system (CAS). MAS based CAS should be a new way of combat simulation. Finally, as an example, ISAAC, a famous MAS based combat simulation system, was analyzed to show the application of MAS in combat simulation. The analysis shows that MAS has many characteristics of mod-

ern warfare and is a promising way of combat simulation.

[5474446] R. Neruda. Ontology description of jade computational agents in owl-dl. In *2010 Second International Conference on Computer and Network Technology*, pages 514–518, April 2010. Amount of References: 1.

Abstract: This paper describes ontological description of computational agents, their properties and abilities. The goal of the work is to allow for autonomous behavior and semi-automatic composition of agents within a multi-agent system. The system has to be create foundation for the interchangeability of computational components, and emergence of new models. This paper focuses on ways of representing agents and systems in standard formalisms, such as description logics, OWL, and Jade.

[6341645] S. Getir, M. Challenger, S. Demirkol, and G. Kardas. The semantics of the interaction between agents and web services on the semantic web. In *2012 IEEE 36th Annual Computer Software and Applications Conference Workshops*, pages 619–624, July 2012. Amount of References: 4.

Abstract: Development of agent systems is naturally a complex task due to the fundamental characteristics of agents. In addition, agent internals and inter-agent behavior models inside Multi-agent Systems (MAS) may become even more difficult to implement when interactions of agents with web services on the Semantic Web are taken into account. Our approach consists of the utilization of a Domain-specific Modeling Language (DSML) during MAS development in order to cope with the abovementioned challenge. This paper describes how the formal semantics of this DSML can be defined by especially focusing on its viewpoint on agent-semantic service interactions and discusses the use of this semantics definition on MAS validation. Determined semantic rules are both defined and implemented by using Alloy specification language which has a strong description capability based on both relational and first-order logic.

[6481885] N. Hamrouni and Z. Abderrahim. Looking for verification and validation of a multi agent system using new formalism:

Apn. In *2012 6th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT)*, pages 30–35, March 2012. Amount of References: 9.

> **Abstract:** In this paper, we present a new formal Method for the Verification of a Multi agent Systems (MAS). We explain and applied a new method based on agents called Agents Petri Nets (APN). Our model describes each entity to model the system and verify the state of the agent and its behavior. This method use, mainly, the property of Petri Nets as deadline, state home, conflict, etc. Therefore we use a formalism having a mathematical definition in order to verify some property of the interactive systems. To validate our contribution, we will deal with real example of a Multi agents System.

[6521966] K. Addakiri and M. Bahaj. Applying knowledge model to multi-agent system. In *2013 International Conference on Computer Applications Technology (ICCAT)*, pages 1–6, Jan 2013. Amount of References: 0.

> **Abstract:** MAS is a powerful paradigm in nowadays, is become promising means for the development of distributed systems, however its disadvantage is that it lacks the interconnection with semantic web such as Ontology Web Language. In this article, we aim to present and to implement a semantic knowledge model of an agent suitable for discrete environments by using the Jena : is a Java framework for the creation of applications for the semantic web, the JADE : is a flexible agent platform that provides a middleware layer for the development of distributed multi-agent systems and the JESS is the rule engine for the JAVA platform used to enrich the ontology with logic and functionality. The use of these software enable interconnection of Agent and Semantic Web technologies can be used in an agent based application where such interconnection is needed.

[6618689] A. Hedhili, W. L. Chaari, and K. Ghédira. Explanation language syntax for multi-agent systems. In *2013 World Congress on Computer and Information Technology (WCCIT)*, pages 1–6, June 2013. Amount of References: 1.

> **Abstract:** For Multi-Agent Systems (MAS) considered as complex systems, it is important to make agent reasoning comprehensible to users. In fact,

MAS execution is uncontrollable; no one knows what effectively happens inside and how the solutions are given. In this context, our work aims to associate an explanation system with a multi-agent system. It consists first in intercepting significant events related to agents at runtime, thus an explanatory knowledge acquisition phase is performed where knowledge attributes are stored in an explanation structure named KAGR. Second, the semantic links between these attributes are expressed in an extended causal map model that constitutes knowledge representation formalism. Finally, an interpretation of this model is fulfilled using predicate logic. So, we get an acquisition, a representation, an interpretation of explanatory knowledge building up a knowledge based system for explanation. In this paper, we broach the interpretation phase where an explanation language is described to interpret the built causal maps.

[7321654] A. T. E. Dib and Z. Sahnoun. Model checking of multi agent system architectures using bigmc. In *2015 Federated Conference on Computer Science and Information Systems (FedCSIS)*, pages 1717–1722, Sept 2015. Amount of References: 11.

Abstract: Formal methods offer a great potential for early integration of verification in the design process. These are based on theories and mathematical notations that allow the formal specification of a program and check its implementation. They offer a global vision and a high-level structure and system organization. In addition, the software architecture plays a key role as a pivot point between the requirements of a system and its implementation. In this paper, we present a formal approach based on Bigraphical Reactive Systems for specifying and verifying the main features of the Multi Agent Systems (MAS) architectures based on the Belief-Desire-Intention (BDI) agent model. The proposed approach supports both the static and dynamic aspects of BDI-MAS architectures at different levels of abstraction. Further, we use automatic proof tool BigMc to analyze the specifications and verify system properties.

[7554553] B. Klebanov, T. Antropov, and E. Riabkina. The principles of multi-agent models of development based on the needs of the agents. In *2016 35th Chinese Control Conference (CCC)*, pages 7551–7555, July 2016. Amount of References: 6.

Abstract: Work is devoted to development of the principles of creation of the model of development of artificial society based on processes of emergence and realization of needs of specific agents. The propose structure of the objective, from the point of view of researchers, knowledge and mental knowledge of the particular agent. A basis of the knowledge base are: possible methods of application of object (relation) in recipes satisfaction of needs of active agents; recipes of creation, change of object (relation); counteraction to object (relation), protection against object, overcoming, destruction of object (relation); actualization - detection, identifications of object, definition of its relations with other objects. Classification of methods of strengthening of opportunities of active agents is offered. The model of process of a choice of strategy of realization of requirements based on gradual strengthening of opportunities of the agent is developed.

[7822181] T. Warnke, O. Reinhardt, and A. M. Uhrmacher. Population-based ctmcs and agent-based models. In *2016 Winter Simulation Conference (WSC)*, pages 1253–1264, Dec 2016. Amount of References: 4.

Abstract: Currently, only few agent-based models are implemented with a continuous representation of time, although state-of-the-art agent-based modeling and simulation (ABMS) frameworks support continuous-time models and continuous time often allows for a more faithful capturing of reality. Intrigued by this discrepancy, we take a closer look at population-based Continuous-Time Markov Chains (CTMCs), their modeling and their simulation, on the one hand, and, how continuous-time agent-based models are currently realized in state-of-the-art ABMS frameworks, like Repast Simphony and Netlogo, on the other hand. Subsequently, we adopt and adapt concepts and algorithms of modeling and simulating population-based CTMCs. We propose a solution how to integrate those into contemporary ABMS frameworks which results in a more succinct description of continuous-time agent-based models.

[8270547] M. Pudane. Classification of agent-based models from the perspective of multi-agent systems. In *2017 5th IEEE Workshop*

on Advances in Information, Electronic and Electrical Engineering (AIEEE), pages 1–6, Nov 2017. Amount of References: 0.

Abstract: Agent based modelling is a paradigm that is widely used to simulate and research systems where each entity is autonomous and behaves independently. Currently, there are many diverse agent based models, however, aside from general guidelines there are no particular analysis of how such models should be designed and implemented. On the other hand, any agent based model can be broken into a single agent and a system level; these abstraction levels are used in the design of multi-agent systems, moreover, agent based models are often implemented using mechanisms and tools from multi-agent systems. Author proposes that one of the ways this problem could be solved is classification of agent-based models from the perspective of multi-agent systems which is then used to provide decision support for the design of agent-based models.

[8352646] A. Dorri, S. S. Kanhere, and R. Jurdak. Multi-agent systems: A survey. *IEEE Access*, 6:28573–28593, 2018. Amount of References: 0.

Abstract: Multi-agent systems (MASs) have received tremendous attention from scholars in different disciplines, including computer science and civil engineering, as a means to solve complex problems by subdividing them into smaller tasks. The individual tasks are allocated to autonomous entities, known as agents. Each agent decides on a proper action to solve the task using multiple inputs, e.g., history of actions, interactions with its neighboring agents, and its goal. The MAS has found multiple applications, including modeling complex systems, smart grids, and computer networks. Despite their wide applicability, there are still a number of challenges faced by MAS, including coordination between agents, security, and task allocation. This survey provides a comprehensive discussion of all aspects of MAS, starting from definitions, features, applications, challenges, and communications to evaluation. A classification on MAS applications and challenges is provided along with references for further studies. We expect this paper to serve as an insightful and comprehensive re-

source on the MAS for researchers and practitioners in the area.

[GARRO2019315] Alfredo Garro, Max Mühlhäuser, Andrea Tundis, Matteo Baldoni, Cristina Baroglio, Federico Bergenti, and Paolo Torroni. Intelligent agents: Multi-agent systems. In Shoba Ranganathan, Michael Gribskov, Kenta Nakai, and Christian Schönbach, editors, *Encyclopedia of Bioinformatics and Computational Biology*, pages 315 – 320. Academic Press, Oxford, 2019. Amount of References: 0.

Abstract: This article provides an overview of the fundamentals related to agent-oriented paradigm and Multi Agent Systems (MASs). Specifically, in the first part, the definition of agent and its main features are reported. The second section discusses multi-agent systems and their main features by focusing on the concepts of automation, coordination, norms and emerging behavior which characterize a MAS. Finally, in the sections third and forth, two main ways to approach and use such paradigm for developing respectively agent-based simulations and agent-based software, by breaking down the leading and most renowned reference software platforms, are discussed.

[Gutknecht:2000:MGM:336595.337048] Olivier Gutknecht and Jacques Ferber. Madkit: A generic multi-agent platform. In *Proceedings of the Fourth International Conference on Autonomous Agents*, AGENTS '00, pages 78–79, New York, NY, USA, 2000. ACM. Amount of References: 140.

Abstract: This paper describes brielfy MadKit, a generic multi-agent platform. This toolkit is based on a organizational model. It uses concepts of groups and roles for agents to manage different agent models and multi-agent systems at the same time, while keeping a global structure. The platform architecture is based on a minimalist agent kernel decoupled from specific agency models. Basic services like distributed message passing, migration or monitoring are provided by platform agents for maximal flexibility. A componential interface model allows variations in platform appearance and classes of usage.

[Heath:2011:CSM:2431518.2431850] Susan K. Heath, Arnold Buss, Sally C. Brailsford, and Charles M. Macal. Cross-paradigm simulation

modeling: Challenges and successes. In *Proceedings of the Winter Simulation Conference*, WSC '11, pages 2788–2802. Winter Simulation Conference, 2011. Amount of References: 66.

Abstract: This paper addresses the broad topic area of cross-paradigm simulation modeling with a focus on the discrete-event, system dynamics and agent-based paradigms. It incorporates contributions from four panel members with diverse perspectives and areas of expertise. First, each paradigm is described and definitions are presented. The difference between the process-oriented worldview and the event-oriented worldview within discrete-event simulation modeling, and the importance of this difference for cross-paradigm modeling, are discussed. Following the definitions, discussion of cross-paradigm modeling is given for each pair of these paradigms, highlighting current challenges and early successes in these areas. The basic time-advance mechanisms used in simulation modeling are also discussed, and the implications of these mechanisms for each paradigm is explored.

[JENNINGS2000277] Nicholas R. Jennings. On agent-based software engineering. *Artificial Intelligence*, 117(2):277 – 296, 2000. Amount of References: 3431.

Abstract: Agent-based computing represents an exciting new synthesis both for Artificial Intelligence (AI) and, more generally, Computer Science. It has the potential to significantly improve the theory and the practice of modeling, designing, and implementing computer systems. Yet, to date, there has been little systematic analysis of what makes the agent-based approach such an appealing and powerful computational model. Moreover, even less effort has been devoted to discussing the inherent disadvantages that stem from adopting an agent-oriented view. Here both sets of issues are explored. The standpoint of this analysis is the role of agent-based software in solving complex, real-world problems. In particular, it will be argued that the development of robust and scalable software systems requires autonomous agents that can complete their objectives while situated in a dynamic and uncertain environment, that can engage in rich, high-level social interactions, and

that can operate within flexible organisational structures.

[VANDERHOEK2008887] Wiebe van der Hoek and Michael Wooldridge. Chapter 24 multi-agent systems. In Frank van Harmelen, Vladimir Lifschitz, and Bruce Porter, editors, *Handbook of Knowledge Representation*, volume 3 of *Foundations of Artificial Intelligence*, pages 887 – 928. Elsevier, 2008. Amount of References: 151.

Abstract: Publisher Summary The discipline of knowledge representation focuses on the way to represent and reason about environments with various different properties, usually with the goal of making decisions, for example, about how best to act in this environment. This chapter discusses the state of the art in knowledge representation formalisms for multiagent systems. It divides the work in this area into two categories. In the first category, there are approaches that attempt to represent the cognitive state of rational agents and characterize logically the way such a state leads a rational agent to act. It begins by motivating this approach. It then describes four of the best-known such logical frameworks and discusses the possible roles that such logics can play in helping to engineer artificial agents. In the second category, there are approaches based on representing the strategic structure of a multiagent environment and, in particular, the powers that agents have, either individually or in coalitions. The chapter also discusses coalition logic, alternating-time temporal logic (ATL), and epistemic extensions.

[bayardo1997infosleuth] Roberto J Bayardo Jr, William Bohrer, Richard Brice, Andrzej Cichocki, Jerry Fowler, Abdelsalam Helal, Vipul Kashyap, Tomasz Ksiezyk, Gale Martin, Marian Nodine, et al. Infosleuth: agent-based semantic integration of information in open and dynamic environments. In *Acm Sigmod Record*, volume 26, pages 195–206. ACM, 1997. Amount of References: 725.

Abstract: The goal of the InfoSleuth project at MCC is to exploit and synthesize new technologies into a unified system that retrieves and processes information in an ever-changing network of information sources. InfoSleuth has its roots in the Carnot project at MCC, which specialized in integrating heterogeneous information bases. However, recent emerging

technologies such as internetworking and the World Wide Web have significantly expanded the types, availability, and volume of data available to an information management system. Furthermore, in these new environments, there is no formal control over the registration of new information sources, and applications tend to be developed without complete knowledge of the resources that will be available when they are run. Federated database projects such as Carnot that do static data integration do not scale up and do not cope well with this ever-changing environment. On the other hand, recent Web technologies, based on keyword search engines, are scalable but, unlike federated databases, are incapable of accessing information based on concepts. In this experience paper, we describe the architecture, design, and implementation of a working version of InfoSleuth. We show how InfoSleuth integrates new technological developments such as agent technology, domain ontologies, brokerage, and internet computing, in support of mediated interoperation of data and services in a dynamic and open environment. We demonstrate the use of information brokering and domain ontologies as key elements for scalability.

[borshchev2004system] Andrei Borshchev and Alexei Filippov. From system dynamics and discrete event to practical agent based modeling: reasons, techniques, tools. In *Proceedings of the 22nd international conference of the system dynamics society*, volume 22. Citeseer, 2004. Amount of References: 869.

Abstract: This paper may be considered as a practical reference for those who wish to add (now sufficiently matured) Agent Based modeling to their analysis toolkit and may or may not have some System Dynamics or Discrete Event modeling background. We focus on systems that contain large numbers of active objects (people, business units, animals, vehicles, or even things like projects, stocks, products, etc. that have timing, event ordering or other kind of individual behavior associated with them). We compare the three major paradigms in simulation modeling: System Dynamics, Discrete Event and Agent Based Modeling with respect to how they approach such systems. We show in detail how an Agent Based model can be built from an existing System Dynamics or a Discrete Event model and then show how eas-

ily it can be further enhanced to capture much more complicated behavior, dependencies and interactions thus providing for deeper insight in the system being modeled. Commonly understood examples are used throughout the paper; all models are specified in the visual language supported by AnyLogicTM tool. We view and present Agent Based modeling not as a substitution to older modeling paradigms but as a useful add-on that can be efficiently combined with System Dynamics and Discrete Event modeling. Several multi-paradigm model architectures are suggested.

[bosse2011agent] Tibor Bosse, Charlotte Gerritsen, Mark Hoogendoorn, S Waqar Jaffry, and Jan Treur. Agent-based vs. population-based simulation of displacement of crime: A comparative study. *Web Intelligence and Agent Systems: An International Journal*, 9(2):147–160, 2011. Amount of References: 21.

Abstract: Central research questions addressed within Criminology are how the geographical displacement of crime can be understood, explained, and predicted. The process of crime displacement is usually explained by referring to the interaction of three types of agents: criminals, passers-by, and guardians. Most existing simulation models of this process take a 'local' perspective, i.e., they are agent-based. However, when the number of agents considered becomes large, more 'global' approaches, such as population-based simulation have computational advantages over agent-based simulation. This article presents both an agent-based and a population-based simulation model of crime displacement, and reports a comparative evaluation of the two models. In addition, an approach is put forward to analyse the behaviour of both models by means of formal techniques. The results suggest that under certain conditions, population-based models approximate agent-based models, at least in the domain under investigation.

[chan2010agent] Wai Kin Victor Chan, Young-Jun Son, and Charles M Macal. Agent-based simulation tutorial-simulation of emergent behavior and differences between agent-based simulation and discrete-event simulation. In *Simulation Conference (WSC), Proceedings of the 2010 Winter*, pages 135–150. IEEE, 2010. Amount of References: 112.

Abstract: This tutorial demonstrates the use of agent-based simulation (ABS) in modeling emergent behaviors. We first introduce key concepts of ABS by using two simple examples: the Game of Life and the Boids models. We illustrate agent-based modeling issues and simulation of emergent behaviors by using examples in social networks, auction-type markets, emergency evacuation, crowd behavior under normal situations, biology, material science, chemistry, and archaeology. Finally, we discuss the relationship between ABS and other simulation methodologies and outline some research challenges in ABS.

[christley2004ontology] Scott Christley, Xiaorong Xiang, and Greg Madey. An ontology for agent-based modeling and simulation. In *Proceedings of the agent 2004 conference*, 2004. Amount of References: 36.

Abstract: Ontologies are a formal methodology for establishing a common vocabulary, for defining the concepts and relationships between those concepts of a particular domain, and for reasoning about the objects, behaviors, and knowledge that comprises the domain. In this paper, we present an ontology for agent-based modeling and simulation. Agent-based modeling and simulation has become an important and popular paradigm for the computational social and natural sciences; however, the application of this paradigm tends to be performed in an ad-hoc fashion leading to questions about underlying assumptions in an agent-based model, verification of the software implementation as a representation of that model, and validation of hypothesized conclusions inferred from data produced by computer simulation experiments. An ontology provides a formal, logical knowledge representation that supports automated reasoning. Such reasoning capability provides for consistency checking of the concepts and relationships in an agent-based model, can infer the assumptions inherent in a model, can infer the assumptions and the parameters inherent in a simulation or software representation of a model, and can enforce adherence to formal methods or best practices for verification and validation testing. These reasoning tasks direct, or at least inform, the modeler about relevant techniques and methods in the agent-based paradigm. The reasoning capability also provides a framework for automated

generation of software code, automated design and execution of simulation experiments as well as automated generation and execution of validation tests for those experiments. Using the standard Ontology Web Language (OWL), we provide a complete, detailed ontology of agent-based modeling and simulation, and we show how the ontology is used as part of the modeling and simulation process.

[collier2003repast] Nick Collier. Repast: An extensible framework for agent simulation. *The University of Chicago's Social Science Research*, 36:2003, 2003. Amount of References: 305.

Abstract: RePast is a software framework for agent-based simulation created by Social Science Research Computing at the University of Chicago. It provides an integrated library of classes for creating, running, displaying, and collecting data from an agent-based simulation. This paper is an overview of RePast's design, features, and capabilities and describes the implementation of some of its key abstractions.

[corchado2004development] Juan M Corchado, Juan Pavón, Emilio S Corchado, and Luis F Castillo. Development of cbr-bdi agents: a tourist guide application. In *European Conference on Case-based Reasoning*, pages 547–559. Springer, 2004. Amount of References: 186.

Abstract: In this paper we present an agent-based application of a wireless tourist guide that combines the Beliefs-Desires-Intentions approach with learning capabilities of Case Base Reasoning techniques. This application shows how to develop adaptive agents with a goal driven design and a decision process built on a CBR architecture. The resulting agent architecture has been validated by real users who have used the tourist guide application, on a mobile device, and can be generalized for the development of other personalized services.

[davidsson2000multi] Paul Davidsson. Multi agent based simulation: beyond social simulation. In *International Workshop on Multi-Agent Systems and Agent-Based Simulation*, pages 97–107. Springer, 2000. Amount of References: 302.

Abstract: Multi Agent Based Simulation (MABS) has been used mostly in purely social contexts. However, compared to other approaches, e.g., traditional

discrete event simulation, object-oriented simulation and dynamic micro simulation, MABS has a number of interesting properties which makes it useful also for other domains. For instance, it supports structure preserving modeling of the simulated reality, simulation of pro-active behavior, parallel computations, and very dynamic simulation scenarios. It is argued that MABS is a useful technique for simulating scenarios also in more technical domains. In particular, this hold for the simulation of technical systems that are distributed and involve complex interaction between humans and machines. To illustrate the advantages of MABS, an application concerning the monitoring and control of intelligent buildings is described.

[drogoul2002multi] Alexis Drogoul, Diane Vanbergue, and Thomas Meurisse. Multi-agent based simulation: Where are the agents? In *International Workshop on Multi-Agent Systems and Agent-Based Simulation*, pages 1–15. Springer, 2002. Amount of References: 330.

Abstract: This paper is devoted to exploring the relationships between computational agents, as they can be found in multi-agent systems (MAS) or Distributed Artificial Intelligence (DAI), and the different techniques regrouped under the generic name "multi-agent based simulation" (MABS). Its main purpose is to show that MABS, despite its name, is in fact rarely based on computational agents. We base our demonstration on an innovative presentation of the methodological process used in the development of current MABS systems. This presentation relies on the definition of the different roles involved in the design process, and we are able to show that the notion of "agent", although shared at a conceptual level by the different participants, does not imply a systematic use of computational agents in the systems deployed. We then conclude by discussing what the use of computational agents, based on the most interesting research trends in DAI or MAS, might provide MABS with.

[ferber1998meta] Jacques Ferber and Olivier Gutknecht. A meta-model for the analysis and design of organizations in multi-agent systems. In *Multi Agent Systems, 1998. Proceedings. International Confer-*

ence on, pages 128–135. IEEE, 1998. Amount of References: 1154.

Abstract: This paper presents a generic meta-model of multi-agent systems based on organizational concepts such as groups, roles and structures. This model, called AALAADIN, defines a very simple description of coordination and negotiation schemes through multi-agent systems. Aalaadin is a meta-model of artificial organization by which one can build multi-agent systems with different forms of organizations such as market-like and hierarchical organizations. We show that this meta-model allows for agent heterogeneity in languages, applications and architectures. We also introduce the concept of organizational reflection which uses the same conceptual model to describe system level tasks such as remote communication and migration of agents. Finally, we briefly describe a platform, called MADKIT, based on this model. It relies on a minimal agent kernel with platform-level services implemented as agents, groups and roles.

[ferber1999multi] Jacques Ferber and Gerhard Weiss. *Multi-agent systems: an introduction to distributed artificial intelligence*, volume 1. Addison-Wesley Reading, 1999. Amount of References: 3893.

Abstract: This edition is a translation of the book formerly published in French in 1995 (Les systèmes multi-agents: Vers une intelligence collective, Inter Editions, Paris.) Even now, it is still the main reference for the French research community in multi-agent systems (MAS). The book is intended to be both a state of the art text and an introduction for people who are interested in capturing the main ideas of MAS. It deals mainly with the theoretical background, antecedents and applications. I will present a summary of the main ideas and the points that the author highlights as novel features arising from the multi-agent approach to computer science. When some details are very well developed in the book, I will just refer to them, since they are mainly useful for people who actually want to apply the techniques. The book begins with a long introduction that sketches the historical origins of MAS research (mainly stating that Decentralised Artificial Intelligence is a complement to Artificial Intelligence and

Artificial Life). Ferber gives a minimal definition of an agent and of an MAS, so that all branches of multi-agent research can accept it

[ferber2000organization] Jacques Ferber, Olivier Gutknecht, Catholijn M Jonker, J-P Muller, and Jan Treur. Organization models and behavioural requirements specification for multi-agent systems. In *MultiAgent Systems, 2000. Proceedings. Fourth International Conference on*, pages 387–388. IEEE, 2000. Amount of References: 91.

Abstract: The main question addressed in this paper is how requirements on the dynamics of a multi-agent systems and individual agents can be related to the dynamics of high level concepts given by an organisation model, such as groups, roles within groups, and role interaction. In terms of these organisational concepts, which abstract from the specific agents assigned to roles, different types of requirement are introduced and formalised, which can be used as templates or patterns. Moreover, logical relationships between these different types of requirements are analysed: proof patterns. It turns out that the organisational concepts and their logical relations, form a very useful level of requirements specification mediating between specification of single agent behaviour and the overall multi-agent system dynamics. Thus the gap between single agent behaviour requirements specification and requirements specifications of the emerging overall dynamics is reduced. A contribution of this work to the area of organisation modelling is that it is shown how the dynamics of an organisation model can be specified in a structured manner using different types of behavioural requirements based on a formal temporal trace language.

[georgeffbelief] Michael Georgeff, Barney Pell, Martha Pollack, Milind Tambe, and Michael Wooldridge. The belief-desire-intention model of agency. In Jörg P. Müller, Anand S. Rao, and Munindar P. Singh, editors, *Intelligent Agents V: Agents Theories, Architectures, and Languages*, pages 1–10, Berlin, Heidelberg, 1999. Springer Berlin Heidelberg. Amount of References: 719.

Abstract: Within the ATAL community, the belief-desire-intention (BDI) model has come to be possibly the best known and best studied model of practical reasoning agents. There are several reasons for its

success, but perhaps the most compelling are that the BDI model combines a respectable philosophical model of human practical reasoning, (originally developed by Michael Bratman [1]), a number of implementations (in the IRMA architecture [2] and the various PRS-like systems currently available [7]), several successful applications (including the now-famous fault diagnosis system for the space shuttle, as well as factory process control systems and business process management [8]), and finally, an elegant abstract logical semantics, which have been taken up and elaborated upon widely within the agent research community [14, 16].

[gilbert2002platforms] Nigel Gilbert and Steven Bankes. Platforms and methods for agent-based modeling. *Proceedings of the National Academy of Sciences*, 99(suppl 3):7197–7198, 2002. Amount of References: 234.

Abstract: The range of tools designed to help build agent-based models is briefly reviewed. It is suggested that although progress has been made, there is much further design and development work to be done. Modelers have an important part to play, because the creation of tools and models using those tools proceed in a dialectical relationship.

[gutknecht2001integrating] Olivier Gutknecht, Jacques Ferber, and Fabien Michel. Integrating tools and infrastructures for generic multi-agent systems. In *Proceedings of the fifth international conference on Autonomous agents*, pages 441–448. ACM, 2001. Amount of References: 81.

Abstract: In this paper, we present MadKit/SEdit, an agent infrastructure combined with a generic design tool for multi-agent systems. This toolkit is based on a organizational metaphor to integrate highly heterogeneous agent systems. We explain the principles of MadKit, the underlying agent platform, and show how it can integrate various agent architectures and provides structuration for multiple simultaneous systems and semantics. The architecture, based on a minimal agent runtime, agentified platform services and modular application host, is presented. The SEdit design tool, built itself as a MAS is also discussed. We present its key points in terms of multi-model support, and integration with the infrastructure, from design to maintenance. We illustrate

our approach by discussing some consequences of this architecture, and describe our motivation for this design: integration and reuse, organizational patterns, and overall versatility. A summary is given of some key MadKit-based applications to date.

[hilaire2000formal] Vincent Hilaire, Abder Koukam, Pablo Gruer, and Jean-Pierre Müller. Formal specification and prototyping of multi-agent systems. In *International Workshop on Engineering Societies in the Agents World*, pages 114–127. Springer, 2000. Amount of References: 135.

Abstract: This paper presents a multi agent-oriented prototyping approach. It is a generic approach, applicable to a wide range of multi-agent systems. This approach relies on a few assumptions, the most important is that MAS must be described by an organizational model which semantics is given in term of a formal framework. This model allows for a simple description of both individual and collective multi-agent system aspects. The framework we use to give a formal description of this model is based on a multi-formalism approach. We illustrate this approach through a case study.

[hilaire2005formal] Vincent Hilaire, Olivier Simonin, Abder Koukam, and Jacques Ferber. A formal approach to design and reuse agent and multiagent models. In James Odell, Paolo Giorgini, and Jörg P. Müller, editors, *Agent-Oriented Software Engineering V*, pages 142–157, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg. Amount of References: 28.

Abstract: While there are many useful models of agents and multi-agent systems, they are typically defined in an informal way and applied in an ad-hoc fashion. Consequently, multi-agent system designers have been unable to fully exploit these models commonalities and specialise or reuse them for specific problems. In order to fully exploit these models and facilitate their reuse we propose a formal approach based upon organisational concepts. The formal notation is the result of the composition of Object-Z and statecharts. The semantics of this multi-formalisms is defined by transition systems. This operational semantics enables validation and verification of specifications. We present this approach through the specification of the satisfaction-

altruism model which has been used to design situated multi-agent systems. We put the emphasis on the specification of a mobile robot architecture based on the refinement of this model. The availability of such generic models is a fundamental basis for reuse. We also show how to analyse the specification by validation and verification.

[hindriks1999agent] Koen V Hindriks, Frank S De Boer, Wiebe Van der Hoek, and John-Jules Ch Meyer. Agent programming in 3apl. *Autonomous Agents and Multi-Agent Systems*, 2(4):357–401, 1999. Amount of References: 460.

Abstract: An intriguing and relatively new metaphor in the programming community is that of an intelligent agent. The idea is to view programs as intelligent agents acting on our behalf. By using the metaphor of intelligent agents the programmer views programs as entities which have a mental state consisting of beliefs and goals. The computational behaviour of an agent is explained in terms of the decisions the agent makes on the basis of its mental state. It is assumed that this way of looking at programs may enhance the design and development of complex computational systems. To support this new style of programming, we propose the agent programming language 3APL. 3APL has a clear and formally defined semantics. The operational semantics of the language is defined by means of transition systems. 3APL is a combination of imperative and logic programming. From imperative programming the language inherits the full range of regular programming constructs, including recursive procedures, and a notion of state-based computation. States of agents, however, are belief or knowledge bases, which are different from the usual variable assignments of imperative programming. From logic programming, the language inherits the proof as computation model as a basic means of computation for querying the belief base of an agent. These features are well-understood and provide a solid basis for a structured agent programming language. Moreover, on top of that 3APL agents use so-called practical reasoning rules which extend the familiar recursive rules of imperative programming in several ways. Practical reasoning rules can be used to monitor and revise the goals of an

agent, and provide an agent with reflective capabilities. Applying the metaphor of intelligent agents means taking a design stance. From this perspective, a program is taken as an entity with a mental state, which acts pro-actively and reactively, and has reflective capabilities. We illustrate how the metaphor of intelligent agents is supported by the programming language. We also discuss the design of control structures for rule-based agent languages. A control structure provides a solution to the problem of which goals and which rules an agent should select. We provide a concrete and intuitive ordering on the practical reasoning rules on which such a selection mechanism can be based. The ordering is based on the metaphor of intelligent agents. Furthermore, we provide a language with a formal semantics for programming control structures. The main idea is not to integrate this language into the agent language itself, but to provide the facilities for programming control structures at a meta level. The operational semantics is accordingly specified at the meta level, by means of a meta transition system.

[janssen2005agent] Marco A Janssen. Agent-based modelling. *Modelling in ecological economics*, pages 155–172, 2005. Amount of References: 86.

Abstract: Agent-based modelling (ABM) is the computational study of social agents as evolving systems of autonomous interacting agents. ABM is a tool for the study of social systems from the complex adaptive system perspective. From this perspective, the researcher is interested in how macro phenomena are emerging from micro level behaviour among a heterogeneous set of interacting agents (Holland, 1992). By using ABM as computational laboratories, one may test in a systematic way different hypotheses related to attributes of the agents, their behavioural rules, and the types of interactions, and their effect on macro level stylized facts of the system. Since the early 1990s ABM has increasingly been used in most of the social sciences. I shall focus on the applications of ABM related to ecological economics (Janssen, 2002, 2005; Janssen and Ostrom, 2005). ABM of ecological economic systems can be defined as systems that are populated with heterogeneous population of agents, who determine their interactions with other

agents and with their environment, on the basis of internalized social norms and mental models, internal behavioural rules and cognitive abilities, formal and informal institutional rules that affect how agents interact, individual and social learning, etc.

[janssen2008towards] Marco A Janssen, Lilian Na'ia Alessa, Michael Barton, Sean Bergin, and Allen Lee. Towards a community framework for agent-based modelling. *Journal of Artificial Societies and Social Simulation*, 11(2):6, 2008. Amount of References: 80.

Abstract: Agent-based modelling has become an increasingly important tool for scholars studying social and social-ecological systems, but there are no community standards on describing, implementing, testing and teaching these tools. This paper reports on the establishment of the Open Agent-Based Modelling Consortium, www.openabm.org, a community effort to foster the agent-based modelling development, communication, and dissemination for research, practice and education.

[jennings1998agent] Nick Jennings, Nicholas R Jennings, and Michael J Wooldridge. *Agent technology: foundations, applications, and markets*. Springer Science & Business Media, 1998. Amount of References: 1135.

Abstract: Agents represent an exciting and promising new approach to building a wide range of software applications. Agents are autonomous problem-solving entities that are able to flexibly solve problems in complex, dynamic environments, without receiving permanent guidance from the user. Agent Technology: Foundations, Applications and Markets is the first book to provide an integrative presentation of the issues, challenges and success of designing, building and using agent applications.

[juan2003meta] Thomas Juan and Leon Sterling. A meta-model for intelligent adaptive multi-agent systems in open environments. In *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, pages 1024–1025. ACM, 2003. Amount of References: 25.

Abstract: In this paper, we introduce the ROADMAP meta-model, designed to describe intelligent adaptive multi-agent systems in open environments. The meta-model captures our understanding

of the properties of such systems and our perspective on organizational concepts such as roles. It defines agent constructs such as agents and roles, and their inter-relationships such as aggregation, in a way that best facilitates the development of intelligent adaptive systems in open environments.We intend to use the meta-model for knowledge sharing. We expect developers of AOSE methodologies, tools, programming languages and frameworks to benefit from understanding the design and structure of the ROADMAP meta-model. By adopting the meta-model, the resulting methodologies, tools and languages should inherit its desirable characteristics and better support the development of intelligent adaptive systems in open environments.

[lin2010modeling] Jing Lin, Sahra Sedigh, and Ann Miller. Modeling cyber-physical systems with semantic agents. In *Computer Software and Applications Conference Workshops (COMPSACW), 2010 IEEE 34th Annual*, pages 13–18. IEEE, 2010. Amount of References: 66.

Abstract: The development of accurate models for cyber-physical systems (CPSs) is hampered by the complexity of these systems, fundamental differences in the operation of cyber and physical components, and significant interdependencies among these components. Agent-based modeling shows promise in overcoming these challenges, due to the flexibility of software agents as autonomous and intelligent decision-making components. Semantic agent systems are even more capable, as the structure they provide facilitates the extraction of meaningful content from the data provided to the software agents. In this paper, we present a multi-agent model for a CPS, where the semantic capabilities are underpinned by sensor networks that provide information about the physical operation to the cyber infrastructure. This model is used to represent the static structure and dynamic behavior of an intelligent water distribution network as a CPS case study.

[luke2005mason] Sean Luke, Claudio Cioffi-Revilla, Liviu Panait, Keith Sullivan, and Gabriel Balan. Mason: A multiagent simulation environment. *Simulation*, 81(7):517–527, 2005. Amount of References: 964.

Abstract: MASON is a fast, easily extensible, discrete-event multi-agent simulation toolkit in Java, designed to serve as the basis for a wide range of multi-agent simulation tasks ranging from swarm robotics to machine learning to social complexity environments. MASON carefully delineates between model and visualization, allowing models to be dynamically detached from or attached to visualizers, and to change platforms mid-run. This paper describes the MASON system, its motivation, and its basic architectural design. It then compares MASON to related multi-agent libraries in the public domain, and discusses six applications of the system built over the past year which suggest its breadth of utility.

[macal2007agent] Charles M Macal and Michael J North. Agent-based modeling and simulation: Desktop abms. In *Simulation Conference, 2007 Winter*, pages 95–106. IEEE, 2007. Amount of References: 498.

Abstract: Agent-based modeling and simulation (ABMS) is a new approach to modeling systems comprised of autonomous, interacting agents. ABMS promises to have far-reaching effects on the way that businesses use computers to support decision-making and researchers use electronic laboratories to support their research. Some have gone so far as to contend that ABMS "is a third way of doing science," in addition to traditional deductive and inductive reasoning (Axelrod 1997b). Computational advances have made possible a growing number of agent-based models across a variety of application domains. Applications range from modeling agent behavior in the stock market, supply chains, and consumer markets, to predicting the spread of epidemics, the threat of bio-warfare, and the factors responsible for the fall of ancient civilizations. This tutorial describes the theoretical and practical foundations of ABMS, identifies toolkits and methods for developing agent models, and illustrates the development of a simple agent-based model of shopper behavior using spreadsheets.

[macal2010agent] Charles M Macal. To agent-based simulation from system dynamics. In *proceedings of the winter simulation conference*, pages 371–382. Winter Simulation Conference, 2010. Amount of References: 82.

Abstract: Agent-based simulation (ABS) is a recent modeling technique that is being widely used in

modeling complex social systems. Forrester's System Dynamics (SD) is another longstanding technique for modeling social systems. Several classical models of systems, such as the Kermack-McKendrick model of epidemiology, the Lotka-Volterra equations for modeling predator-prey relationships, and the Bass model for innovation diffusion are formulated as systems of differential equations and have corresponding System Dynamics representations as difference equations. The ABS and SD modeling approaches take fundamentally different perspectives when modeling a system, which can be characterized as bottom-up (ABS) versus top-down (SD). Yet many systems can be equivalently modeled by either approach. In this paper, we present a formal specification for SD and ABS models, use the specification to derive equivalent ABS representations, and present an example of an SIR epidemic model having SD and ABS counterparts.

[macal2010tutorial] Charles M Macal and Michael J North. Tutorial on agent-based modelling and simulation. *Journal of simulation*, 4(3):151–162, 2010. Amount of References: 1235.

Abstract: Agent-based modelling and simulation (ABMS) is a relatively new approach to modelling systems composed of autonomous, interacting agents. Agent-based modelling is a way to model the dynamics of complex systems and complex adaptive systems. Such systems often self-organize themselves and create emergent order. Agent-based models also include models of behaviour (human or otherwise) and are used to observe the collective effects of agent behaviours and interactions. The development of agent modelling tools, the availability of microdata, and advances in computation have made possible a growing number of agent-based applications across a variety of domains and disciplines. This article provides a brief introduction to ABMS, illustrates the main concepts and foundations, discusses some recent applications across a variety of disciplines, and identifies methods and toolkits for developing agent models.

[macal2014introductory] Charles Macal and Michael North. Introductory tutorial: Agent-based modeling and simulation. In *Proceedings of*

the 2014 winter simulation conference, pages 6–20. IEEE Press, 2014. Amount of References: 163.

Abstract: Agent-based simulation (ABS) is an approach to modeling systems comprised of individual, autonomous, interacting "agents." Agent-based modeling offers ways to more easily model individual behaviors and how behaviors affect others in ways that have not been available before. There is much interest in developing agent-based models for many application problem domains. Applications range from modeling agent behavior in supply chains and the stock market, to predicting the success of marketing campaigns and the spread of epidemics, to projecting the future needs of the healthcare system. Progress in the area suggests that ABS promises to have far-reaching effects on the way that businesses use computers to support decision-making and researchers use agent-based models as electronic laboratories to aid in discovery. This brief tutorial introduces agent-based modeling and simulation by describing the basic ideas of ABS, discussing some applications, and addressing methods for developing agent-based models.

[macal2016everything] Charles M Macal. Everything you need to know about agent-based modelling and simulation. *Journal of Simulation*, 10(2):144–156, 2016. Amount of References: 70.

Abstract: This paper addresses the background and current state of the field of agent-based modelling and simulation (ABMS). It revisits the issue of ABMS represents as a new development, considering the extremes of being an overhyped fad, doomed to disappear, or a revolutionary development, shifting fundamental paradigms of how research is conducted. This paper identifies key ABMS resources, publications, and communities. It also proposes several complementary definitions for ABMS, based on practice, intended to establish a common vocabulary for understanding ABMS, which seems to be lacking. It concludes by suggesting research challenges for ABMS to advance and realize its potential in the coming years.

[michel2001generic] Fabien Michel, Jacques Ferber, and Olivier Gutknecht. Generic simulation tools based on mas organization. In *10th*

European Workshop on Modelling Autonomous Agents in a Multi Agent World MAMAAW, volume 1, 2001. Amount of References: 37.

Abstract: This paper presents generic simulation tools which rely on an original methodological approach of designing multi-agent simulators. We will see that these tools are generic specially because they are not related to a particular scheduling method. On the contrary they aim at providing facilities that allow to design complex activation structures that remain comprehensible, analyzable and moreover modifiable, thanks to a problem division. To achieve this, the main idea of this methodology is to express the multi-agent system (MAS) simulator as a particular MAS itself and use explicitly its subjacent organizational structure. We will show how the Aalaadin organizational model enables us to finely apply such a methodology. Precisely, we will present a particular agent of MadKit (the platform that relies on Aalaadin): the Scheduler agent and its tool called Activator

[minar1996swarm] Nelson Minar, Roger Burkhart, Chris Langton, Manor Askenazi, et al. The swarm simulation system: A toolkit for building multi-agent simulations. 1996. Amount of References: 1019.

Abstract: Swarm is a multiagent software platform for the simulation of complex adaptive systems In the Swarm system the basic unit of simulation is the swarm a collec tion of agents executing a schedule of actions Swarm supports hierarchical model ing approaches whereby agents can be composed of swarms of other agents in nested structures Swarm provides ob ject oriented libraries of reusable components for build ing models and analyzing displaying and controlling experiments on those models Swarm is currently available as a beta version in full free source code form It requires the GNU C Compiler Unix and X Windows More information about Swarm can be obtained from our web pages httpwwwsantafeedupro jectsswarm Santa Fe Ins

[niazi2011agent] Muaz Niazi and Amir Hussain. Agent-based computing from multi-agent systems to agent-based models: a visual survey. *Scientometrics*, 89(2):479, 2011. Amount of References: 239.

Abstract: Agent-based computing is a diverse research domain concerned with the building of intelligent software based on the concept of "agents". In this paper, we use Scientometric analysis to analyze all sub-domains of agent-based computing. Our data consists of 1,064 journal articles indexed in the ISI web of knowledge published during a 20 year period: 1990–2010. These were retrieved using a topic search with various keywords commonly used in sub-domains of agent-based computing. In our proposed approach, we have employed a combination of two applications for analysis, namely Network Workbench and CiteSpace—wherein Network Workbench allowed for the analysis of complex network aspects of the domain, detailed visualization-based analysis of the bibliographic data was performed using CiteSpace. Our results include the identification of the largest cluster based on keywords, the timeline of publication of index terms, the core journals and key subject categories. We also identify the core authors, top countries of origin of the manuscripts along with core research institutes. Finally, our results have interestingly revealed the strong presence of agent-based computing in a number of non-computing related scientific domains including Life Sciences, Ecological Sciences and Social Sciences.

[nikolai2009tools] Cynthia Nikolai and Gregory Madey. Tools of the trade: A survey of various agent based modeling platforms. *Journal of Artificial Societies and Social Simulation*, 12(2):2, 2009. Amount of References: 309.

Abstract: Agent Based Modeling (ABM) toolkits are as diverse as the community of people who use them. With so many toolkits available, the choice of which one is best suited for a project is left to word of mouth, past experiences in using particular toolkits and toolkit publicity. This is especially troublesome for projects that require specialization. Rather than using toolkits that are the most publicized but are designed for general projects, using this paper, one will be able to choose a toolkit that already exists and that may be built especially for one's particular domain and specialized needs. In this paper, we examine the entire continuum of agent based toolkits. We characterize each based on 5 important characteristics users consider when choosing a toolkit,

and then we categorize the characteristics into user-friendly taxonomies that aid in rapid indexing and easy reference.

[nwana1999zeus] Hyacinth S Nwana, Divine T Ndumu, Lyndon C Lee, and Jaron C Collis. Zeus: a toolkit for building distributed multi-agent systems. *Applied Artificial Intelligence*, 13(1-2):129–185, 1999. Amount of References: 456.

Abstract: The multi-agent systems approach of knowledge level co-operation between autonomous agents promises significant benefits to distributed systems engineering, such as enhanced interoperability, scalability and reconfigurability. However, thus far, because of the innate difficulty of constructing multi-agent systems, this promise has been largely unrealised. Hence, there is an emerging desire amongst agent developers to move away from developing point solutions to point problems in favour of developing methodologies and toolkits for building distributed multi-agent systems. This philosophy led to the development of the ZEUS Agent Building Toolkit, which facilitates the rapid development of collaborative agent applications through the provision of a library of agent-level components and an environment to support the agent building process. The ZEUS toolkit is a synthesis of established agent technologies with some novel solutions to provide an integrated collaborative agent building environment.

[odell2002modeling] James J Odell, H Van Dyke Parunak, Mitch Fleischer, and Sven Brueckner. Modeling agents and their environment. In *International Workshop on Agent-Oriented Software Engineering*, pages 16–31. Springer, 2002. Amount of References: 187.

Abstract: Without an environment, an agent is effectively useless. Cut off from the rest of its world, the agent can neither sense nor act. An environment provides the conditions under which an entity (agent or object) can exist. It defines the properties of the world in which an agent will function. Designing effective agents requires careful consideration of both the physical and communicational aspects of their environment.

[oliveira1999multi] Eugenio Oliveira, Klaus Fischer, and Olga Stepankova. Multi-agent systems: which research for which applications.

Robotics and Autonomous Systems, 27(1-2):91–106, 1999. Amount of References: 195.

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.

[omicini2008artifacts] Andrea Omicini, Alessandro Ricci, and Mirko Viroli. Artifacts in the a&a meta-model for multi-agent systems. *Autonomous agents and multi-agent systems*, 17(3):432–456, 2008. Amount of References: 357.

Abstract: In this article we focus on the notion of artifact for agents in multi-agent systems (MAS) as a basis for a new meta-model promoting the modelling and engineering of agent societies and MAS environment as first-class entities. Its conceptual foundations lay upon theories and results coming from computational sciences as well as from organisational and cognitive sciences, psychology, computer supported cooperative work (CSCW), anthropology and ethology. In the resulting agents artifacts (AA) meta-

model, agents are the (pro-)active entities in charge of the goals/tasks that altogether build up the whole MAS behaviour, whereas artifacts are the reactive entities providing the services and functions that make individual agents work together in a MAS, and that shape agent environment according to the MAS needs. After presenting the scientific background, we define the notions of artifact in the AA meta-model, discuss how it affects the notion of intelligence in MAS, and show its application to a number of agent-related research fields.

[parunak1998agent] H Van Dyke Parunak, Robert Savit, and Rick L Riolo. Agent-based modeling vs. equation-based modeling: A case study and users' guide. In *International Workshop on Multi-Agent Systems and Agent-Based Simulation*, pages 10–25. Springer, 1998. Amount of References: 783.

Abstract: In many domains, agent-based system modeling competes with equation-based approaches that identify system variables and evaluate or integrate sets of equations relating these variables. The distinction has been of great interest in a project that applies agent-based modeling to industrial supply networks, since virtually all computer-based modeling of such networks up to this point has used system dynamics, an approach based on ordinary differential equations (ODE's). This paper summarizes the domain of supply networks and illustrates how they can be modeled both with agents and with equations. It summarizes the similarities and differences of these two classes of models, and develops criteria for selecting one or the other approach.

[parunak2003design] H Van Dyke Parunak, Sven Brueckner, Mitch Fleischer, and James Odell. A design taxonomy of multi-agent interactions. In *International Workshop on Agent-Oriented Software Engineering*, pages 123–137. Springer, 2003. Amount of References: 60.

Abstract: Agent interactions are frequently characterized as "coherent," "collaborative," "cooperative," "competitive," or "coordinated." These terms specialize the more foundational category of "correlation," which can be measured by the joint information of a system. "Congruence"is orthogonal to the others, reflecting the degree to which correlation and its specializations satisfy user requirements.

A taxonomy of these mechanisms can guide the design of multi-agent interaction. Lack of correlation is sometimes necessary, and requires the use of formal stochasticity.

[parunak2004universality] H Parunak, Sven Brueckner, and Robert Savit. Universality in multi-agent systems. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems-Volume 2*, pages 930–937. IEEE Computer Society, 2004. Amount of References: 41.

Abstract: Much research in MAS explores how refinements to one agentýs reasoning can improve system performance. Sometimes, aspects of a systemýs behavior are independent of individual agentsý algorithms. Inspired by statistical physics, we term this phenomenon "universality": systems whose elements differ widely may have common emergent features. We develop a notion of universality in MAS based on the conceptýs use in its original (physics) setting. We give examples of the phenomenon, and discuss its implications for the theory and practice of MAS. We speculate that there exists a hierarchy of types of universality. The statistical mechanics sense refers to the most refined, simplest, and quantitative, while commonalities among MAS systems are associated with somewhat more general and qualitative levels of universality. Such a hierarchy would be an important integrating principle across systems of interacting components, including human societies, animal ecologies, multi-agent systems, and atoms and molecules.

[payne2008web] Terry R Payne. Web services from an agent perspective. *IEEE Intelligent Systems*, 23(2), 2008. Amount of References: 59.

Abstract: Multiagent systems evolved from a need for knowledge-aware, distributed, problem-solving mechanisms. These systems are formally grounded using theoretical approaches, including those that assume mentalistic notions. As a result, much of this research into multiagent systems has provided formal proofs or proof-of-concept demonstrators (such as example systems or prototypes). It has provided only limited, pragmatic support (systems, software, and tools) for the user community. Research into Web services, in contrast, has focused on the user community, resulting in a pragmatic, bottom-up enabling technology that readily facilitates the robust

construction of service-oriented systems. Much of the focus of Web services research has been on developing declarative descriptions that application developers can share and that their tools can use to construct and develop large-scale distributed software.

[railsback2006agent] Steven F Railsback, Steven L Lytinen, and Stephen K Jackson. Agent-based simulation platforms: Review and development recommendations. *Simulation*, 82(9):609–623, 2006. Amount of References: 793.

Abstract: Five software platforms for scientific agent-based models (ABMs) were reviewed by implementing example models in each. NetLogo is the highest-level platform, providing a simple yet powerful programming language, built-in graphical interfaces, and comprehensive documentation. It is designed primarily for ABMs of mobile individuals with local interactions in a grid space, but not necessarily clumsy for others. NetLogo is highly recommended, even for prototyping complex models. MASON, Repast, and Swarm are "framework and library" platforms, providing a conceptual framework for organizing and designing ABMs and corresponding software libraries. MASON is least mature and designed with execution speed a high priority. The Objective-C version of Swarm is the most mature library platform and is stable and well organized. Objective-C seems more natural than Java for ABMs but weak error-handling and the lack of developer tools are drawbacks. Java Swarm allows Swarm's Objective-C libraries to be called from Java; it does not seem to combine the advantages of the two languages well. Repast provides Swarm-like functions in a Java library and is a good choice for many, but parts of its organization and design could be improved. A rough comparison of execution speed found MASON and Repast usually fastest (MASON 1-35% faster than Repast), Swarm (including Objective-C) fastest for simple models but slowest for complex ones, and NetLogo intermediate. Recommendations include completing the documentation (for all platforms except NetLogo), strengthening conceptual frameworks, providing better tools for statistical output and automating simulation experiments, simplifying common tasks, and researching technologies for understanding how simulation results arise.

[railsback2011agent] Steven F Railsback and Volker Grimm. *Agent-based and individual-based modeling: a practical introduction.* Princeton university press, 2011. Amount of References: 850.

> **Abstract:** Agent-based modeling is a new technique for understanding how the dynamics of biological, social, and other complex systems arise from the characteristics and behaviors of the agents making up these systems. This innovative textbook gives students and scientists the skills to design, implement, and analyze agent-based models. It starts with the fundamentals of modeling and provides an introduction to NetLogo, an easy-to-use, free, and powerful software platform. Nine chapters then each introduce an important modeling ...

[schelfthout2002agent] Kurt Schelfthout, Tim Coninx, Alexander Helleboogh, Tom Holvoet, Elke Steegmans, and Danny Weyns. Agent implementation patterns. In *Proceedings of the OOPSLA 2002 Workshop on Agent-oriented Methodologies*, pages 119–130, 2002. Amount of References: 46.

> **Abstract:** Our own experience in developing agents and multiagent systems painfully taught us that there are no instruments to support the implementation of such systems. A thorough literature study confirms the fact that implementation issues, which are definitely non-trivial but clearly underestimated, are mostly achieved in an ad hoc fashion. This obviously results in people reinventing solutions for common problems. In this paper, we propose agent implementation patterns as an instrument to describe problems and generic solutions for implementing agents and multiagent systems. Agent implementation patterns contrast with other proposals of agent patterns in that the latter typically deal with architectural and conceptual models, and not with the actual implementation of agent characteristics. If we eventually want to come to a best practice for the development and implementation of agent systems, system developers should share their knowledge and experience gained during the implementation of their systems. Agent implementation patterns can contribute to formalize and share such knowledge.

[siebers2010discrete] Peer-Olaf Siebers, Charles M Macal, Jeremy Garnett, David Buxton, and Michael Pidd. Discrete-event simulation

is dead, long live agent-based simulation! *Journal of Simulation*, 4(3):204–210, 2010. Amount of References: 283.

Abstract: There has been much discussion about why agent-based simulation (ABS) is not as widely used as discrete-event simulation in Operational Research (OR) as it is in neighbouring disciplines such as Computer Science, the Social Sciences or Economics. To consider this issue, a plenary panel was organised at the UK Operational Research Society's Simulation Workshop 2010 (SW10). This paper captures the discussion that took place and addresses the key questions and opportunities regarding ABS that will face the OR community in the future.

[suna2007programming] Alexandru Suna and Amal El Fallah Seghrouchni. Programming mobile intelligent agents: An operational semantics. *Web Intelligence and Agent Systems: An International Journal*, 5(1):47–67, 2007. Amount of References: 21.

Abstract: This paper presents the operational semantics of an agent-oriented programming language called CLAIM. CLAIM allows to design multi-agent systems that support both stationary and mobile agents, endowed with cognitive capabilities and able to communicate. The semantics takes into account agents' reasoning, mobility, communication and concurrence, and contains a set of reduction rules between coherent states of a program. The operational semantics is a first step towards the formal verification of agent-based programs written in CLAIM. The reduction rules are illustrated on a case study.

[tisue2004netlogo] Seth Tisue and Uri Wilensky. Netlogo: Design and implementation of a multi-agent modeling environment. In *Proceedings of agent*, volume 2004, pages 7–9, 2004. Amount of References: 265.

Abstract: NetLogo is a multi-agent programming language and modeling environment for simulating complex phenomena. It is designed for both research and education and is used across a wide range of disciplines and education levels. In this paper, though, we focus on NetLogo as a tool for research and for teaching at the undergraduate level and higher. We outline the principles behind our design and describe recent and planned enhancements

[tisue2004netlogoenvironment] Seth Tisue and Uri Wilensky. Netlogo: A simple environment for modeling complexity. In *International conference on complex systems*, volume 21, pages 16–21. Boston, MA, 2004. Amount of References: 566.

Abstract: NetLogo is a multi-agent programming language and modeling environment for simulating complex phenomena. It is designed for both research and education and is used across a wide range of disciplines and education levels. In this paper we focus on NetLogo as a tool for research and for teaching at the undergraduate level and higher. We outline the principles behind our design and describe recent and planned enhancements.

[tolk2009agents] Andreas Tolk and Adelinde M Uhrmacher. Agents: agenthood, agent architectures, and agent taxonomies. *Agent-directed simulation and systems engineering*, pages 75–109, 2009. Amount of References: 27.

Abstract: he agent metaphor is based on developments in diverse computer science areas such as distributed systems, software engineering, and artificial intelligence. It has been strongly influenced by the research results of other disciplines as well, in particular sociology, biology, systems and decision science, and many others. Those diverse areas of research are reflected in multiple facets that characterize agents. This shared parenthood also explains the similarity with certain approachesas agents cannot deny their roots-and consequently the need for distinction. For example, agents have been characterized as reflective and concurrent objects (Gasser and Briot, 1992) and as embedded artificial intelligence (Russell and Norvig, 1995), migrating agents have been described as mobile code plus state plus autonomy (Kotz and Gray, 1999), agents should represent human behavior as placeholders in applications (Silverman, 2001), and so on. Thus, the properties of being" autonomous" and" flexible"-often referred to as" adaptive"-are perhaps the most prominent characteristics of agents (Wooldridge and Jennings, 1995) and are also considered to hold the most promise for future development (Jennings et al., 1998). In this chapter, we will look at some typical agent properties and describe some methods and architectures that have proven helpful in bringing about these desirable

agent characteristics. In addition, we will explore the relation to simulation and systems engineering and, finally, discuss some possibilities for structuring the realm of agents, starting with taxonomies. We will also develop a three-dimensional space, taking into account ...

[uhrmacher1997concepts] Adelinde M Uhrmacher, P Tyschler, and D Tyschler. Concepts of object-and agent-oriented simulation. *Transactions of the Society for Computer Simulation*, 14(2):59–67, 1997. Amount of References: 53.

Abstract: Object-oriented concepts are widely employed in simulation. The increasing activity in developing agent-oriented simulation systems calls for exploring the relationship between object-oriented and agent-oriented modeling and simulation more closely. The paper constitutes one step to that enterprise.

[uhrmacher1998agents] Adelinde M Uhrmacher and Bernd Schattenberg. Agents in discrete event simulation. In *European Simulation Symposium-ESS*, volume 98, pages 129–136, 1998. Amount of References: 55.

Abstract: Test beds for multi-agent systems provide the means for experimenting with multiple agents that act and interact concurrently in their environment. Mostly, test beds constitute specialized tools which provide specific scenarios for testing specific agentarchitectures. To facilitate reuse and a exible compositional construction of experimental frames for multiagent systems, JAMES, a Java Based Agent Modeling Environment for Simulation, explores the integration of agents within a general modeling and simulation formalism for discrete event systems. Therefore, a compositional and hierarchical model design which supports variable structure models is complemented with a distributed, concurrent execution. Two experiments with planning agents illustrate our approach.

[van2007real] H Van Dyke Parunak, Sven Brueckner, Robert Matthews, John Sauter, and Steve Brophy. Real-time agent characterization and prediction. In *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*, page 278. ACM, 2007. Amount of References: 32.

Abstract: Reasoning about agents that we observe in the world is challenging. Our available information is often limited to observations of the agent's external behavior in the past and present. To understand these actions, we need to deduce the agent's internal state, which includes not only rational elements (such as intentions and plans), but also emotive ones (such as fear). In addition, we often want to predict the agent's future actions, which are constrained not only by these inward characteristics, but also by the dynamics of the agent's interaction with its environment. BEE (Behavior Evolution and Extrapolation) uses a faster-than-real-time agent-based model of the environment to characterize agents' internal state by evolution against observed behavior, and then predict their future behavior, taking into account the dynamics of their interaction with the environment.

[vidal2001inside] José M Vidal, Paul A Buhler, and Michael N Huhns. Inside an agent. *IEEE Internet Computing*, 5(1):82–86, 2001. Amount of References: 69.

Abstract: When we discuss agent-based system construction with software developers or ask students to implement common agent architectures using object-oriented techniques, we find that it is not trivial for them to create an elegant system design from the standard presentation of these architectures in textbooks or research papers. To better communicate our interpretation of popular agent architectures, we draw UML (Unified Modeling Language) diagrams to guide an implementer's design. However, before we describe these diagrams, we need to review some basic features of agents. The paper considers an architecture showing a simple agent interacting with an environment. The agent senses its environment, uses what it senses to choose an action, and then performs the action through its effectors. Sensory input can include received messages, and action can be the sending of messages. To construct an agent, we need a more detailed understanding of how it functions. In particular, if we are to build one using conventional object-oriented analysis and design techniques, we should know in what ways an agent is more than just a simple object.

[weiss1999multiagent] Gerhard Weiss. *Multiagent systems: a modern approach to distributed artificial intelligence.* MIT press, 1999. Amount of References: 5368.

Abstract: This is the first comprehensive introduction to multiagent systems and contemporary distributed artificial intelligence that is suitable as a textbook. The book provides detailed coverage of basic topics as well as several closely related ones. Unlike traditional textbooks, the book brings together many leading experts, guaranteeing a broad and diverse base of knowledge and expertise. It emphasizes aspects of both theory and application, and provides many illustrations and examples. Also included are thought-provoking exercises of varying degrees of difficulty and a twenty-page glossary of terms found in the study of agents, multiagent systems, and distributed artificial intelligence. The book can be used for teaching as well as self-study, and is designed to meet the needs of both researchers and practitioners. In view of the interdisciplinary nature of the field, it will be a useful reference not only for computer scientists and engineers, but for social scientists and management and organization scientists as well. Contributors: Gul A. Agha, Kathleen M. Carley, Jose Cuena, Edmund H. Durfee, Clarence Ellis, Les Gasser, Michael P. Georgeff, Michael N. Huhns, Toru Ishida, Nadeem Jamali, Sascha Ossowski, H. Van Dyke Parunak, Anand S. Rao, Tuomas W. Sandholm, Sandip Sen, Munindar P. Singh, Larry M. Stephens, Gerard Tel, Jacques Wainer, Gerhard Weiss, Michael J. Wooldridge, Makoto Yokoo.

[weyns2002colored] Danny Weyns and Tom Holvoet. A colored petri-net for a multi-agent application. In *Proceedings of MOCA'02*, volume 561, pages 121–141, 2002. Amount of References: 39.

Abstract: In this paper we present a Colored Petri Net (CPN) for a multi-agent application. In particular we modeled the Packet-World. In our research we use the packet-world as a case to study the fundamentals of agents' social behavior. Our approach is to combine experiments with conceptual modeling. We start from a very basic model and then add social skills in a modular way. Integrating new social skills by means of adding new modules offers us a clear conceptual view on the evolution of agents and the

environment. With a conceptual view we mean: (i) which concepts does an agent need in order to acquire a new kind of social ability, (ii) which infrastructure is necessary in the environment to support these abilities, (iii) how do these concepts relate to each other? With the insights we learn from the case study, we gradually develop a generic conceptual model for social agents situated in a MAS. In this paper we first present a CPN for a basic model of the packet-world. This model consists of agents that can only interact through passive objects in the environment. Because interaction is the central issue of multi-agent systems, we have incorporated basic infrastructure for agent coordination straight away into our basic model. Then we extend the model, making it possible for the agents to communicate information with each other. Communication is the basis for social organization. Besides the concrete realization of a CPN for a multiagent application, the model we present in this paper has the potential to support our future research of agents' social behavior. Our major motives for using CPNs as modeling tool are (i) CPNs gives a clear conceptual view on agents and the environment wherein they live, and (ii) CPNs support neat verification and formalization.

[weyns2003regional] Danny Weyns and Tom Holvoet. Regional synchronization for simultaneous actions in situated multi-agent systems. In *International Central and Eastern European Conference on Multi-Agent Systems*, pages 497–510. Springer, 2003. Amount of References: 26.

Abstract: Agents of a multi-agent system (MAS) must synchronize whenever they want to perform simultaneous actions. In situated MASs, typically, the control over such synchronization is centralized, i.e. one synchronizer has the supervision on all agents of the MAS. As a consequence, all agents are forced to act at a global pace and that does not fit with autonomy of agents. Besides, global synchronization implies centralized control, in general an undesirable property of MASs. In this paper we present an algorithm that allows agents to synchronize with other agents within their perceptual range. The result of the algorithm is the formation of independent groups of synchronized agents. The composition of these groups depends on the locality of the agents

and dynamically changes when agents enter or leave each others perceptual range. Since in this approach agents are only synchronized with colleagues in their region, the pace on which they act only depends on the acting speed of potential collaborating agents. The price for decentralization of synchronization is the communication overhead to set up the groups. In the paper, we discuss experimental results and compare the benefits of regional synchronization with its costs.

[weyns2004formal] Danny Weyns and Tom Holvoet. A formal model for situated multi-agent systems. *Fundamenta Informaticae*, 63(2-3):125–158, 2004. Amount of References: 119.

Abstract: Contrary to cognitive approaches of agency where a lot of effort is devoted to the formalization of agent concepts, little work has been done on the formalization of situated multi-agent systems (situated MASs). In this paper we present a generic model for situated MASs. This model formally describes an abstract architecture for situated MASs. In this architecture each agent is situated in its local context that it is able to perceive and in which it can act. Since intelligence in situated MASs results from the interactions of agents with the environment rather than from their individual capabilities, the model takes an action-centric approach. The model deals with (1) the actions of agents in the environment, (2) ongoing activities in the environment, such as moving objects, and (3) the interactions between agents and ongoing activities through the environment. One model for situated MASs was described by J.Ferber and J.P. Mller. In this model all agents of the MAS act at one global pace, i.e.the agents are globally synchronized. Drawbacks of global synchronization are centralized control and poor scalability. We present a model that allows agents to synchronize locally. In this model there is no centralized entity that imposes all agents to act at one global pace, but instead agents themselves decide when they perform their next actions. The model supports simultaneous actions through regional synchronization. With regional synchronization agents form synchronized groups on the basis of their actual locality. Different groups can act asynchronously, while agents within one group act synchronously. The result is a model

that does not suffer from the drawbacks of global synchronization while it preserves the properties for handling simultaneous actions. In the paper we apply the model to a simple MAS application. We show how the abstract model can be instantiated for a practical application. Then we follow a trace in the evolution of the application and demonstrate how the model deals with each particular step.

[wilensky2015introduction] Uri Wilensky and William Rand. *An introduction to agent-based modeling: modeling natural, social, and engineered complex systems with NetLogo*. MIT Press, 2015. Amount of References: 302.

Abstract: A comprehensive and hands-on introduction to the core concepts, methods, and applications of agent-based modeling, including detailed NetLogo examples. The advent of widespread fast computing has enabled us to work on more complex problems and to build and analyze more complex models. This book provides an introduction to one of the primary methodologies for research in this new field of knowledge. Agent-based modeling (ABM) offers a new way of doing science: by conducting computer-based experiments. ABM is ...

[wooldridge1992logical] Michael John Wooldridge. *The logical modelling of computational multi-agent systems*. PhD thesis, Citeseer, 1992. Amount of References: 249.

Abstract: The aim of this thesis is to investigate logical formalisms for describing, reasoning about, specifying, and perhaps ultimately verifying the properties of systems composed of multiple intelligent computational agents. There are two obvious resources available for this task. The first is the (largely AI) tradition of reasoning about the intentional notions (belief, desire, etc.). The second is the (mainstream computer science) tradition of temporal logics for reasoning about reactive systems. Unfortunately, neither resource is ideally suited to the task: most intentional logics have little to say on the subject of agent architecture, and tend to assume that agents are perfect reasoners, whereas models of concurrent systems from mainstream computer science typically deal with the execution of individual program instructions. This thesis proposes a solution which draws upon both resources. It defines a

model of agents and multi-agent systems, and then defines two execution models, which describe how agents may act and interact. The execution models define what constitutes an acceptable run of a system. A run may then act as a model for a temporal logic; this logic can subsequently be used to describe and reason about multi-agent systems. A number of logics, with various properties, are developed in this way. Several detailed examples are presented, showing how the logics may be used for specifying and reasoning about multi-agent systems. The thesis includes a detailed literature survey.

[wooldridge2000gaia] Michael Wooldridge, Nicholas R Jennings, and David Kinny. The gaia methodology for agent-oriented analysis and design. *Autonomous Agents and multi-agent systems*, 3(3):285–312, 2000. Amount of References: 2766.

Abstract: This article presents Gaia: a methodology for agent-oriented analysis and design. The Gaia methodology is both general, in that it is applicable to a wide range of multi-agent systems, and comprehensive, in that it deals with both the macro-level (societal) and the micro-level (agent) aspects of systems. Gaia is founded on the view of a multi-agent system as a computational organisation consisting of various interacting roles. We illustrate Gaia through a case study (an agent-based business process management system).

[wooldridgejennings1995] Michael Wooldridge and Nicholas R. Jennings. Intelligent agents: theory and practice. *The Knowledge Engineering Review*, 10(2):115–152, 1995. Amount of References: 11123.

Abstract: The concept of an agent has become important in both artificial intelligence (AT) and mainstream computer science. Our aim in this paper is to point the reader at what we perceive to be the most important theoretical and practical issues associated with the design and construction of intelligent agents. For convenience, we divide these issues into three areas (though as the reader will see, the divisions are at times somewhat arbitrary). Agent theory is concerned with the question of what an agent is, and the use of mathematical formalisms for representing and reasoning about the properties of agents. Agent architectures can be thought of as software engineering

models of agents; researchers in this area are primarily concerned with the problem of designing software or hardware systems that will satisfy the properties specified by agent theorists. Finally, agent languages are software systems for programming and experimenting with agents; these languages may embody principles proposed by theorists. The paper is not intended to serve as a tutorial introduction to all the issues mentioned; we hope instead simply to identify the most important issues, and point to work that elaborates on them. The article includes a short review of current and potential applications of agent technology.

[woolridge1995agent] Michael Wooldridge and Nicholas R. Jennings. Agent theories, architectures, and languages: A survey. In Michael J. Wooldridge and Nicholas R. Jennings, editors, *Intelligent Agents*, pages 1–39, Berlin, Heidelberg, 1995. Springer Berlin Heidelberg. Amount of References: 1976.

Abstract: The concept of an agent has become important in both Artificial Intelligence (AI) and mainstream computer science. In this article, we present a survey of what we perceive to be the most important theoretical and practical issues associated with the design and construction of intelligent agents. The article also includes a short review of current and potential applications of agent technology, and closes with a glossary of key terms, an annotated list of systems, and a detailed bibliography. Pointers to further reading are provided throughout.