Università degli Studi di Bologna
DEIS

Computational Logic in Multi-Agent Systems: recent advances and future directions

Paolo Torroni

December 30, 2003

DEIS Technical Report no. DEIS-LIA-007-03 *LIA Series no. 68*

Computational Logic in Multi-Agent Systems: recent advances and future directions

Paolo Torroni

DEIS
Università di Bologna
Viale Risorgimento, 2 - 40136 Bologna, Italy
ptorroni@deis.unibo.it

December 30, 2003

Abstract. Starting from the early days of multi-agent systems research, considerable effort has been devoted to giving formal foundations to agent technologies. Work done in this direction, based on computational logic, is an attempt to bridge an existing gap, between theoretical frameworks and their practical implementations. In the last two editions of the workshop on Computational Logic in Multi-Agent Systems, CLIMA'01 and CLIMA'02, two discussion panels have been organized, aimed at bringing researchers together and exchanging ideas on a number of topics. In this article, we elaborate on the outcome of such panels, to draw some considerations about the recent advances and future directions of Computational Logic in Multi-Agent Systems.*

Keywords: *computational logic, logic programming, multi-agent systems*

1 Introduction

CLIMA has now been running for five years, as a satellite event of the main Logic Programming related events: in 1999, under the heading of MAS-LP (Workshop on Multi-Agent Systems in Logic Programming), organized by Stephen Rochefort, Fariba Sadri and Francesca Toni at New Mexico State University in Las Cruces, then

*This article will appear in J. Dix, J. Leite, and K. Satoh, Eds., *Computational Logic in Multi-Agent Systems*, Special Issue of *Annals of Mathematics and Artificial Intelligence*, Kluwer Academic Publishers (2004) subsequently in its current incarnation, CLIMA (Computational Logic in Multi-Agent Systems), organised by Ken Satoh and Fariba Sadri in London in 2000, by Ken Satoh and Jürgen Dix in 2001 in Paphos, and by Jürgen Dix, João Leite and Ken Satoh in 2002 in Copenhagen. CLIMA-IV is going to be held in Fort Lauderdale, Florida, in January 2004, in conjunction with LPNMR and AI & Math, and sponsored by CologNET.

CLIMA'01 ended with a panel session on the role of Computational Logic (CL) in Multi-Agent Systems (MAS). Two dimensions in MAS development were identified and discussed: on the one hand reactivity vs. rationality, and on the other hand individuals vs. societies. Most of the points discussed aimed at justifying and motivating the application of CL techniques to MAS development: how can be logics used to implement individual agents, societies, institutions? how can be logics used to model reactivity and rationality in agents? what is possible to achieve in terms of properties, openness to integration, etc.?

A most intuitive reply to these questions is that logic should be used for what logic is good at. For instance, logic programming-based techniques such as abductive logic programming seem suitable for modelling agent hypothetical reasoning in the presence of incomplete knowledge [15]. Modal logic operators such as those adopted by a BDI agent model [51] could be a powerful and synthetic way to describe the agent mental state and behaviour, and extensions have been proposed to it to make agents aware of each other and to reason with social categories, such as obligations [13] and cooperativity [5]. Model checking-based techniques can be applied to the verification of agent systems [8]. A combination of multiple approaches, like modal and temporal logics, or abduction and induction in a logic programming framework, could be the key to achieve a more comprehensive agent and agent system architecture. But in this case, to determine which properties of the chosen combinations hold is not an easy task.

At the time of CLIMA'02, while the debate about the role of CL in MAS is still open, we are witnessing an increasing interest in MAS from the CL community. This is due to many reasons, among which is the need to put "abstract" reasoning in the context of a "concrete" environment: the multi-agent metaphor of intelligent individuals that are situated into dynamic and unpredictable environments and that can interact with each other by updating their beliefs, can be regarded then as the basis for a new symbolic model of cognition.

The two CLIMA panels of 2001 and 2002 addressed some central issues in the area of logic-based multi-agent systems: the relationship between the internal reasoning of agents and the outer world, the semantics of a system composed of agents that reason on a private knowledge, and the practicability of BDI-inspired architectures.

Logic-based agents and the outer world. Some recent work in Logic Programming outlines this new concept of intelligent system. In [40], Kowalski says:

> "it is the objective perspective of multiagent systems that forces me to acknowledge the existence of a real environment, which exists independently of individual agents: As I see it now, if there is only one agent, then that agent's environment might only be virtual. But if there are several agents interacting with one another, and if all of them are equally real, then the environment of each agent must include the other agents, and therefore that environment itself must also be real. This real environment, shared by several agents, can be understood as a classical model-theoretic, semantic structure. It gives meaning to the agents' thoughts, making some thoughts true and other thoughts false. It grounds their thoughts in reality."

Much of the work presented at CLIMA'01 [21] and CLIMA'02 [20] well reflected this concept. Speculative computation and planning together with action execution [33, 36] reconcile the agent reasoning with the effect of actions made on an external world. Techniques proposed to deal with message loss or modification [56] clearly picture the idea of an environment where logic based agents are situated that could indeed be very different from the model that they have of it. The introduction of hierarchies, roles, protocols and norms [39] puts the agent in a context that sometimes we can call society, or institution, which is at a higher level than that of the single individuals, and which gives a meaning to the agents' thoughts and behaviour.

Why is this new model interesting from a CL perspective? Taking into account an environment with its own semantics means to accept destructive assignment [40]. If we consider MAS as a distributed and concurrent computational system, considering multiple autonomous agents could imply imposing a committed choice at every step. If we want to adopt this new paradigm for Computational Logic, what are the choices that we ought to do? What new assumptions should we make, and on the other hand, how could we accommodate these new features in our background?

If we consider an agent's viewpoint, the other agents in the system could be seen as a part of the environment. Therefore, an aspect that is fundamental in this new cognitive model is that of communication, since it is one of the ways agents become aware of each other.

In general, the question could be put in the following way: how to accommodate in an agent knowledge representation and reasoning activity the external inputs given by a dynamically evolving environment?

Reconciling individual and collective agent semantics. The problem raised above is tightly related to another central issue: that of defining a semantics for the collective behaviour of a multi-agent system. Depending on what semantics we want to give to the overall system, by putting together several agents, each with a different knowledge and with different properties to enforce (expressed for instance by way of integrity constraints), we run the risk to obtain a system which is inconsistent: both in terms of contradictory views of the world, and in terms of properties that are not any more guaranteed.

Let us mention a couple of concrete application scenarios. In distributed systems management there might be agents that provide services (resource managers), and they may have some policies about the access to such resources. Client agents that request services to the resource managers may also have their own policies that constrain and rule their requests. Integrity constraints on the agent's behaviour can be used to describe such policies. In a similar way it could be possible to model, for instance, electronic institutions, where a "governor" agent could encode the institution's norms in the form of integrity constraints.

We have shown two examples of systems where integrity constraints play a key role, both at the individual's level and at the system's level. In the first case, they implicitly define interaction protocols, as resulting from the composition of individuals; in the second case they explicitly encode behaviour rules and norms with which everyone in an institution must comply. Consistency is only one (very important) aspect of a more general problem: what semantics could we give to a system of logic-based agents? What does it mean to preserve the individual's and the system's integrity? How to enforce properties in the overall system emerging from the "composition", or better, interaction, of multiple and independent interpretations of the world?

Computationally practical models or agent rationality. A last important issue was addressed in the panels. It is often the case that new proposals are made with the ultimate purpose to have a system that is implementable, and which at the same time has properties that can be formally defined and proven (which is arguably an advantage of a declarative, logic-based approach). But in fact, the problem of bridging the gap between theory and practice often remains. Which agent architectures could we adopt to this purpose?

Indeed, a reference architecture for the agent behaviour is the well known BDI model, based on Beliefs, Desires, and Intentions, and its variations and evolutions. After more than a decade from its introduction, there is still a considerable amount of work being done and to be done, aimed at establishing links between operational models and abstract architectures for BDI, at verifying whether practical implementations of BDI do actually meet the theoretical requirements or whether a BDI agent adapts itself to a particular BDI-strategy, at recasting the foundations of BDI into a logic programming framework, at providing proof methods to establish the consistency of classes of formulas to represent introspective beliefs. BDI seems then a powerful way to model the agent behaviour and the evolution of a society of agents. The idea of possible worlds is indeed appealing as a possible representation of an evolving environment. But to what extent are applications of full BDI proof-theoretic? Will we get to a comprehensive implementation of a BDI agent or is this a utopia? What simplifications to this model can be considered acceptable in a realistic application?

In the remainder of this article, we will elaborate on the outcome of the CLIMA panels, and draw some considerations about recent advances and future directions of Computational Logic in Multi-Agent Systems.

2 Towards a computationally tractable rationality

In the early nineties, some very influential work proposed a model of agent rationality based on combinations of modal operators, such as beliefs and goals [16], intentions [51], obligations and capabilities [57]. One of the best-known such models was the one proposed by Georgeff and Rao, based on Beliefs, Desires, and Intentions (BDI), consisting of an axiomatization based on modal logics, an interpreter and cycle theory. The main purpose of such an axiomatization, in the context of agents, was to be able to build *formally verifiable* and *practical* systems, through the introduction of intentions as first class modalities.

The Procedural Reasoning System, PRS [50], has been one attempt to implement the abstract BDI architecture, resulting in applications such as the control of autonomous robots, telecommunication systems, and air combat modelling. By programming agents using PRS, one is able to specify their behaviour by means of categories such as commitments and beliefs. Unfortunately, the correspondence between the abstract BDI architecture and its practical implementation, as this stage, is still left to the intuition. Only the beliefs about the current state of the world are explicitly represented; only ground sets of literals are allowed, with no disjunctions or implications; the information about the means of achieving certain future world state is coded in a plan library; intentions are represented implicitly using a conventional run-time stack of hierarchically related plans. After PRS, some other BDI-inspired agent platforms have been implemented, such as dMARS [18] and JACK [37], but the problem still remains.

In 1996, following a bottom-up approach, Rao proposes an agent programming language, AgentSpeak(L) [52], which is an abstraction of an implemented BDI systems, such as PRS, and allows agent programs to be written and interpreted in a manner similar to that of Horn-clause logic programs. This attempt to bridge the gap between theory and practice of BDI theories represents in fact a solid formalization of but a restricted fragment of the original BDI abstract model.

Building on Rao's work, other authors have proposed extensions of AgentSpeak(L) [10], and investigated properties of AgentSpeak(L) agents, contributing towards giving firm theoretical grounds for BDI agent programming [11]. Among other directions of research aiming at bridging the gap between theory and practice, we mention that by Parsons, Sierra, and Jennings [49], based on Giunchiglia and Serafini's multi-context systems [29], and 3APL, a combination of declarative and imperative programming, primarily concerned with the dynamics of an agent's mental state [35]. Despite the conspicuous amount of work done on extensions of the BDI architecture and implementations of subsets of it, to date, to the best of our knowledge, the problem of modal logic based approaches not being proof-theoretic remains still open.

Other logic-based approaches, probably less expressive than the ones cited above, present a more direct link between theory and implementation. Linear logic has been recently proposed as a suitable formalism for specifying, verifying, and prototyping agent systems [48]. Linear logics, as a logic of occurrences, is suitable for agents because resources are usually bounded: An agent can be represented by a sequent like $\mathcal{E}, \mathcal{A}, \mathcal{B}, !\mathcal{P} \vdash \mathcal{G}$, where $\mathcal{E}$, $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{G}$, representing events, actions, beliefs and goals (intentions) are modelled as linear formulae since they change, and $\mathcal{P}$, representing program clauses, are modelled like classical formulae, since they are persistent. This representation has been introduced in previous editions of CLIMA by Harland and Winikoff [30, 31] and is discussed in more detail in an article of this volume [32]. Recent work has been devoted to applying the linear logic approach to negotiation in agents by Küngas and Matskin [42, 43].

In this landscape, several logic programming approaches have been proposed to model agent rationality. Probably the first milestone in this respect was Kowalski and Sadri's work on the Observe-Think-Act cycle and their proposal to adopt an abductive logic programming proof procedure to reconcile agent rationality with reactivity [41]. The Kowalski-Sadri architecture has been extended in several directions, towards accommodating updates [17], and dialogues for negotiation and resource sharing [54]. In such work, the logic-based approach allows to define and prove some interesting properties of agent systems. For example, the adoption of an abductive proof procedure for which termination, correctness and completeness results hold, allows to extend such general results to specific problems, to which the agent architecture is applied: this is the case, for example, of the resource reallocation problem tackled by [54]. This is made possible by an organic agent architecture, where declarative specification and operational model are tightly related.

An effort to extend the results of [54] is undergoing within the EU funded SOCS project [58]. The idea is to provide a computational logic model for the description, analysis and verification of global and open societies of heterogeneous *computees*. Computees are abstractions of entities situated in global and open computing environments. The architecture of a computee is composed by several reasoning functionalities, like planning, temporal reasoning, observation and reaction to stimuli, whose activation is governed by a cycle theory based on Kakas and Moraïtis' work on reasoning with preference [39], presented at CLIMA'02, and grounded on Logic Programming without Negation as Failure (LPwNF). Planning is based on Abductive

Event Calculus, and uses an extended version of the IFF abductive proof procedure [27] able to deal with Constraints (C-IFF). The computee model [38] is proposed as a full-fledged agent model, based on extended logic programming, allowing to define and study properties that can be enforced by its operational model.

Another noteworthy work in this sense is the $\mathcal{MINERVA}$ agent architecture [46], based on dynamic logic programming [4]. $\mathcal{MINERVA}$ is in fact a modular architecture, where every agent is composed of specialized sub-agents that execute special tasks, e.g., reactivity, planning, scheduling, belief revision, action execution. A common internal knowledge base, represented as one or more Multi-dimensional Dynamic Logic Programs (MDLP) [47], is concurrently manipulated by its specialized sub-agents. The MDLPs may encode object level knowledge, or knowledge about goals, plans, intentions, etc. KABUL [45] is the language used to encode specification and evolution of the epistemic state of each sub-agent MDLP.

Finally, IMPACT [6] is an agent platform where programs may be used to specify what an agent is obliged to do, what an agent may do, and what an agent cannot do on the basis of deontic operators of permission, obligation and prohibition. IMPACT is grounded on a solid semantic framework based on the concept of feasible status set [23]. Agent programs define integrity constraints, which are to be satisfied in order to provide a feasible status set, and their violation has the principal meaning of a corrupted state. In this case, agents must be able to recover from being corrupted to uncorrupted [22]. The adoption of a logic programming based formalism, and the use of integrity constraints to define a feasible status set, guarantee agents to behave in a way that some properties hold. Recent significant improvements in the efficiency of Logic Programming implementations for Non-monotonic Reasoning allow architectures such as IMPACT to effectively work on real world applications, where agent programs can be used as a wrapper for the "agentification" of legacy systems (à la Shoham).

3 Modelling interaction in logic-based multi-agent systems

As we stressed in the introduction, the problem of reconciling individual and collective semantics in a system of interacting agents has attracted the attention of several groups of research in the past years.

Starting from a model of agency inspired to Georgeff and Rao's BDI architecture, around 1993, a group of researchers involved in the Knowledge Sharing Effort proposed a language for manipulating the agent knowledge through interaction, called KQML [24]. A few years later, the Foundation for Intelligent Physical Agents (FIPA) issued a series of proposals (directives) towards standardizing Agent Communication Languages [25]. The idea was to define a number of communicative actions (*performatives*) having a semantics based on mental states. In Labrou and Finin's proposal [44], the semantics of a communicative act is given in terms of pre- and post-conditions, and completion conditions. Such conditions are expressed as formulae, whose elements are the communicating agents' mental states. Similarly, in [25] the semantics of a communicative act is given in terms of feasibility preconditions (ability and context-relevant), and rational effect of the act.

Indeed, this "mentalistic" approach could be an interesting way to prove properties of systems of agents, where it is possible to specify the agents themselves in terms of mental categories, provided that the agents' internals are accessible and verifiable. But this is not always the case. In open and heterogeneous systems, it might be impossible to define agents in terms of mental states, or their mental state can be invisible to the outside. Moreover, different agents could use the same communicative act based on different rationales (they could intend two different rational effects for it).

For these reasons, both KQML and FIPA failed to impose themselves as a standard in agent communication, for what goes beyond mere a syntactic convention in the format of communicative acts.

In recent years, an alternative approach to the semantics of ACLs has been proposed by Singh and Yolum [61] and by Fornara and Colombetti [26], where a social semantics of agent interaction is exemplified by using a commitment-based approach. The use of commitments [14] is a way to facilitate autonomy, heterogeneity, and ability to exploit opportunities and exceptions in open and heterogenous agent societies [61]. Along this line, interesting work on computational societies has been presented and developed in the context of the ALFEBIITE project [3] and of the SOCS project [58].

Within the ALFEBIITE project, Artikis et al. [7] present a theoretical framework for providing executable specifications of particular kinds of multi-agent systems, called open computational societies, and present a formal framework for specifying, animating and ultimately reasoning about and verifying the properties of systems where the behaviour of the society members and their interactions cannot be predicted in advance.

Within the SOCS project, Alberti et al. [1, 2] propose an abductive logic programming based framework for the social semantics of agent communication languages and protocols. Such work is especially oriented to computational aspects, and it has been developed with the purpose of providing a computational framework that can be directly used for automatic verification of properties such as compliance. The idea is to model interaction protocols as *social* integrity constraints, which generate expectations about the (compliant) behaviour of agents in the system. The semantics of agent interaction is given in terms of expectations, and their fulfillment or violation.

The use of abductive logic programming to model and give semantics to agent interaction is something that can be found also in other bodies of work. In work by Kowalski and Sadri [41] on agent reasoning and by Sadri et al. [53] on agent dialogues, communicative acts are the outcome of an abductive proof procedure, and they are modelled as abducible predicates.

Ciampolini et al. [15] propose a framework (ALIAS) and a language (LAILA) for the coordination of reasoning of abductive logic agents, where interaction properties can be defined in terms of "global" consistency with respect to sets of integrity constraints, which are part of the agent programs. In [28], Gavanelli et al. propose an abductive logic programming framework where information is exchanged among agents in terms of variable binding; in [12], Bracciali and Torroni introduce a framework for knowledge exchange, again based on abductive reasoning. Hindriks et al. [34] propose for an agent programming language such as 3ALP a semantics of communication based on deduction and abduction. Finally, Satoh [55] proposes an agent architecture for a combination of speculative computation and abduction.

Abduction seems to be a natural way to model interaction, since it implements the process of extending an agent's knowledge with additional elements, let them be incoming communicative acts, social expectations, or any other kind of external input. The interaction among multiple reasoning agents operating in the same virtual environment gives raise to a distributed abductive reasoning process, where integrity constraints "code" the properties of individuals, and global properties can be derived by suitable coordination architectures.

4 Future directions

In this article, we have been considering work done on multiple facets of agent technology, such as reasoning, interaction, coordination, semantics. Some of this work provides a partial answer to the questions that we put forward in the introduction. We would like to conclude by outlining some future directions for computational logics in multi-agent systems.

By proposing multi-agent systems in association with the main logic programming events, CLIMA was born as a forum to present and discuss mainly theoretical work. This is a very important perspective, since formal correctness must be a main research focus for supporting the development of reliable, stable, verifiable, and predictable systems. Nevertheless, in the same way as agent theory helps agent implementation, in some cases also the analysis of implemented agent systems could help the theory, and interesting directions of research could be suggested by considering the state of applied agent-based technology.

At the time of the writing of this article, many agent-based solutions can be already found in areas such as information and knowledge integration (see the Sage and Find Future projects by Fujitsu), Business Process Management (Agentis Software), the Oracle Intelligent Agents, not to mention decentralized control and scheduling, and e-procurement (Rockwell Automation, Living Systems AG, Lost Wax, iSOCO), just to cite some.[1]

Despite these existing applications, some problems related to agent technology are still open, which limit its diffusion to a large scale. Among the main problems that remain to be sorted out we can list: trust and security issues, related to monitoring, delegation, verification, and control; the lack of communication standards and of general purpose development frameworks, which makes building agent applications costly and of limited portability; the lack of benchmarks, which makes it difficult to evaluate and compare agent frameworks with each other; the objective difficulty in designing interfaces which can make agents easily understood and controlled by potential users.

In our opinion, computational logics could be the key to propose solutions to several of these problems. Most approaches to agent trust, based on reputation, adopt a quantitative approach. This, by privileging efficiency, does not solve the problems of enforcing properties and helping the users understanding the agent behaviour. A qualitative, symbolic approach based on some logic formalism seems to be a promising - but so far little explored - way to approach this issue.

In previous parts of this article, we have shown how logics can be used to model and give semantics to agent interaction. Computational logic-based technologies could be used to help verification of agent interaction, whose semantics is specified in terms of logic formulae.

Benchmarks have been so far proposed in restricted and closed domain, such as the Trading Agent Competition [9]. Again, such benchmarks are based on quantitative parameters, such as the profit scored during the competition. Another generation of benchmarks could be based on properties, specified in a logic formalism.

Finally, considerable work has been devoted to proposing user interfaces which are human-like or which mimic human attitudes, as in the case of user agents and avatars. A different approach to interfaces could be aimed at providing the user with an understandable and semantically well specified internal status of an agent or of an agent system. A declarative representation of the agent internals and of social categories could be a way to help providing this kind of information.

5 Conclusion

In the last two editions of the workshop on Computational Logic in Multi-Agent Systems, CLIMA'01 and CLIMA'02, two discussion panels have been organized, aimed at bringing researchers together and exchanging ideas on a number of topics.

[1] A collection of excerpts of papers and web pages about industrial applications of agent technology, including references to the above mentioned projects and applications, can be downloaded from the address: `http://lia.deis.unibo.it/~pt/misc/AIIA03-review.pdf`. An interesting review of industrial agent applications can be found in [60].

Among such topics, the role of computational logics in multi-agent systems, the computational practicality of abstract agent architectures, and the semantics of systems of multiple independent agents.

Along with the work done on the theoretical side, many agent oriented applications already exist which consider the agent metaphor a natural way to model numerous application domains. Nevertheless, some problems remain to date unsolved, which have contributed restraining a potential break-through of agent technology.

By considering some recent work produced in the intersection of computational logic and multi-agent systems, and drawing inspiration from the outcome of the CLIMA panels, in this article we have tried to draw some considerations about the recent advances and future directions of a lively research area, which has the potential to help solving many open problems and promoting the use of agent technology at a larger scale. Among future directions of work we considered trust and security issues, standardization, user interfaces, and evaluation methodologies.

Acknowledgments

This work has been supported by the SOCS project, funded by CEC, contract IST-2001-32530. The information provided is the sole responsibility of the author and does not reflect the Community's opinion. The Community is not responsible for any use that might be made of the data appearing in this publication.

I would like to thank Paola Mello and Evelina Lamma for their precious comments on this article. Thanks also to Francesca Toni, Ken Satoh, João Leite and Jürgen Dix for fruitful discussions about a previous version of this article [59] and for the material that they provided me with for its preparation.

References

[1] Alberti, M., M. Gavanelli, E. Lamma, P. Mello, and P. Torroni: 2003a, 'An Abductive Interpretation for Open Societies'. In: A. Cappelli and F. Turini (eds.): *AI*IA 2003: Advances in Artificial Intelligence, Proceedings of the 8th Congress of the Italian Association for Artificial Intelligence, Pisa*, *Lecture Notes in Artificial Intelligence*, Vol. 2829: Springer-Verlag, pp. 287–299.

[2] Alberti, M., M. Gavanelli, E. Lamma, P. Mello, and P. Torroni: 2003b, 'Specification and Verification of Agent Interactions using Social Integrity Constraints'. *Electronic Notes in Theoretical Computer Science* **85**(2).

[3] ALFEBIITE: A Logical Framework for Ethical Behaviour between Infohabitants in the Information Trading Economy of the Universal Information Ecosystem. IST-1999-10298. `http://www.iis.ee.ic.ac.uk/~alfebiite/`.

[4] Alferes, J. J., J. A. Leite, L. M. Pereira, H. Przymusinska, and T. C. Przymusinski: 2000, 'Dynamic Updates of Non-Monotonic Knowledge Bases'. *The Journal of Logic Programming* **45**(1–3), 43–70.

[5] Ancona, D. and V. Mascardi: 2003, 'Coo-BDI: Extending the BDI Model with Cooperativity'. In: J. A. Leite, A. Omicini, L. Sterling, and P. Torroni (eds.): *Declarative Agent Languages and Technologies, First International Workshop, DALT 2003. Melbourne, Victoria, July 15th, 2003. Workshop Notes*. pp. 146–161.

[6] Arisha, K. A., F. Ozcan, R. Ross, V. S. Subrahmanian, T. Eiter, and S. Kraus: 1999, 'IMPACT: a Platform for Collaborating Agents'. *IEEE Intelligent Systems* **14**(2), 64–72.

[7] Artikis, A., J. Pitt, and M. Sergot: 2002, 'Animated specifications of computational societies'. In: C. Castelfranchi and W. Lewis Johnson (eds.): *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems, Bologna, Italy (AAMAS-2002), Part III*: ACM Press, pp. 1053–1061.

[8] Benerecetti, M. and A. Cimatti: 2001, 'Symbolic Model Checking for Multi-Agent Systems'. in [21]. Electronic proceedings.

[9] Boman, M.: 2001, 'Trading Agents'. *AgentLink News Newsletter* **6**, pp. 15–16.

[10] Bordini, R., A. L. C. Bazzan, R. O. Jannone, D. M. Basso, R. M. Vicari, and V. R. Lesser: 2002, 'AgentSpeak(XL): Efficient intention selection in BDI agents via decision-theoretic task scheduling.'. In: C. Castelfranchi and W. Lewis Johnson (eds.): *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems, Bologna, Italy (AAMAS-2002), Part III*: ACM Press, pp. 1294–1302.

[11] Bordini, R. H. and Á. F. Moreira: 2002, 'Proving the Asymmetry Thesis Principles for a BDI Agent-Oriented Programming Language'. in [20], pp. 94–108.

[12] Bracciali, A. and P. Torroni: 2004, 'A New Framework for Knowledge Revision of Abductive Agents through their Interaction (preliminary report)'. in [19].

[13] Broersen, J., M. Dastani, J. Hulstijn, Z. Huang, and L. van der Torre: 2001, 'The BOID architecture: conficts between beliefs, obligations, intentions and desires'. In: *Proceedings of Fifth International Conference on Autonomous Agents (Agents2001)*. Montreal, Canada: ACM Press, pp. 9–16.

[14] Castelfranchi, C.: 1995, 'Commitments: From individual intentions to groups and organizations'. In: *Proceedings of the 1st International Conference on Multiagent Systems, San Francisco, California*. pp. 41–48.

[15] Ciampolini, A., E. Lamma, P. Mello, F. Toni, and P. Torroni: 2003, 'Co-operation and Competition in *ALIAS*: a logic framework for agents that negotiate'. *Computational Logic in Multi-Agent Systems. Annals of Mathematics and Artificial Intelligence* **37**(1-2), 65–91.

[16] Cohen, P. and H. Levesque: 1990, 'Intention is choice with commitment'. *Artificial Intelligence* **42**(2-3), 213–261.

[17] Dell'Acqua, P., U. Nilsson, and L. M. Pereira: 2002, 'A Logic Based Asynchronous Multi-Agent System'. in [20], pp. 63–78.

[18] d'Inverno, M., D. Kinny, M. Luck, and M. Wooldridge: 1998, 'A formal specification of dMARS'. In: M. Singh, A. Rao, and M. Wooldridge (eds.): *Intelligent Agents V, Agent Theories, Architectures, and Languages, 5th International Workshop, ATAL '98, Paris, France, Proceedings*, *Lecture Notes in Artificial Intelligence*, Vol. 1365: Springer-Verlag, pp. 155–176.

[19] Dix, J. and J. Leite (eds.): 2004, 'CLIMA-IV: Computational Logic in Multi-Agent Systems, Fourth International Workshop. Proceedings. Fort Lauderdale, FL, USA.' To Appear.

[20] Dix, J., J. A. Leite, and K. Satoh (eds.): 2002, 'Computational Logic in Multi-Agent Systems: 3rd International Workshop, CLIMA'02, Copenhagen, Denmark, August 1, 2002, Proceedings', No. 93 in Datalogiske Skrifter (Writings on Computer Science). Roskilde University, Denmark.

[21] Dix, J. and K. Satoh (eds.): 2001, *Proceedings CLIMA'01, 2nd International Workshop on Computational Logic and Multi-Agent Systems*. Electronic proceedings. `http://research.nii.ac.jp/~ksatoh/clima01.html`.

[22] Eiter, T., V. Mascardi, and V. S. Subrahmanian: 2002, 'Error-Tolerant Agents'. In: A. C. Kakas and F. Sadri (eds.): *Computational Logic. Logic Programming and Beyond*, *Lecture Notes in Computer Science*, Vol. 2407: Springer-Verlag, pp. 586–625.

[23] Eiter, T., V. Subrahmanian, and G. Pick: 1999, 'Heterogeneous active agents, I: Semantics'. *Artificial Intelligence* **108**(1-2), 179–255.

[24] Finin, T., Y. Labrou, and J. Mayfield: 1997, 'KQML as an agent communication language'. In: J. Bradshaw (ed.): *Software Agents*. MIT Press, Cambridge.

[25] FIPA Communicative Act Library Specification: 2001, 'FIPA Communicative Act Library Specification'. Published on August 10th, 2001, available for download from the FIPA website, `http://www.fipa.org`.

[26] Fornara, N. and M. Colombetti: 2003, 'Defining Interaction Protocols using a Commitment-based Agent Communication Language'. In: J. S. Rosenschein, T. Sandholm, M. Wooldridge, and M. Yokoo (eds.): *Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems, Melbourne, Victoria (AAMAS-2003)*: ACM Press, pp. 520–527.

[27] Fung, T. H. and R. A. Kowalski: 1997, 'The IFF proof procedure for abductive logic programming'. *Journal of Logic Programming* **33**(2), 151–165.

[28] Gavanelli, M., E. Lamma, P. Mello, and P. Torroni: 2004, 'An Abductive Framework for Information Sharing in Multi-Agent systems'. in [19].

[29] Giunchiglia, F. and L. Serafini: 1994, 'Multilanguage hierarchical logics (or: How we can do without modal logics)'. *Artificial Intelligence* **65**, 29–70.

[30] Harland, J. and M. Winikoff: 2001, 'Agents via Mixed-Mode Computation: a Proposal'. in [21]. Electronic proceedings.

[31] Harland, J. and M. Winikoff: 2002, 'Language Design Issues for Agents based on Linear Logic (Extended Abstract)'. in [20], pp. 79–93.

[32] Harland, J. and M. Winikoff: 2004, 'Agents via Mixed-mode Computation in Linear Logic'. *In this volume.*

[33] Hayashi, H., K. Cho, and A. Ohsuga: 2002, 'Speculative Computation and Action Execution in Multi-Agent Systems'. in [20], pp. 136–148.

[34] Hindriks, K., F. de Boer, W. van der Hoek, and J. J. Meyer: 1999a, 'Semantics of Communicating Agents based on Deduction and Abduction'. In: *Foundations And Applications Of Collective Agent Based Systems (CABS).*

[35] Hindriks, K. V., F. S. de Boer, W. van der Hoek, and J. C. Meyer: 1999b, 'Agent Programming in 3APL'. *Autonomous Agents and Multi-Agent Systems* **2(4)**, 357–401.

[36] Iwanuma, K. and K. Inoue: 2002, 'Conditional Answer Computation in SOL as Speculative Computation in Multi-Agent Environments'. in [20], pp. 149–162.

[37] JACK Intelligent Agents User Guide: 2001. Agent Oriented Software (AOS), Carlton, Victoria. Version 3.0. `http://www.jackagents.com`.

[38] Kakas, A., P. Mancarella, F. Sadri, K. Stathis, and F. Toni: 2003, 'A logic-based approach to model computees'. Technical report, SOCS Consortium. Deliverable D4.

[39] Kakas, A. C. and P. Moraïtis: 2002, 'Argumentative Agent Deliberation, Roles and Context'. in [20], pp. 35–48.

[40] Kowalski, R. A.: 2001, 'Logic Programming and the Real World'. *Logic Programming Newsletter* **14**(1).

[41] Kowalski, R. A. and F. Sadri: 1999, 'From Logic Programming Towards Multi-Agent Systems'. *Annals of Mathematics and Artificial Intelligence* **25**(3/4), 391–419.

[42] Küngas, P. and M. Matskin: 2003, 'Linear Logic, Partial Deduction and Cooperative Problem Solving'. In: J. A. Leite, A. Omicini, L. Sterling, and P. Torroni (eds.): *Declarative Agent Languages and Technologies, First International Workshop, DALT 2003. Melbourne, Victoria, July 15th, 2003. Workshop Notes*. pp. 97–112.

[43] Küngas, P. and M. Matskin: 2004, 'Symbolic Negotiation with Linear Logic'. in [19].

[44] Labrou, Y. and T. Finin: 1998, 'Semantics and Conversations for an Agent Communication Language'. In: M. Huhns and M. Singh (eds.): *Readings in Agents*. Morgan Kaufmann Publishers.

[45] Leite, J. A.: 2003, *Evolving Knowledge Bases*. IOS Press.

[46] Leite, J. A., J. J. Alferes, and L. M. Pereira: 2002a, '$\mathcal{MINERVA}$: A Dynamic Logic Programming Agent Architecture'. In: *Intelligent Agents VIII: 8th International Workshop, ATAL 2001, Seattle, WA, USA, Revised Papers*, *Lecture Notes in Artificial Intelligence*, Vol. 2333: Springer-Verlag, pp. 141–157.

[47] Leite, J. A., J. J. Alferes, L. M. Pereira, H. Przymusinska, and T. C. Przymusinski: 2002b, 'A Language for Multi-dimensional Updates'. in [20], pp. 19–34.

[48] Mascardi, V., M. Martelli, F. Zini, and R. Degl'Innocenti: 2004, 'CaseLP: a Prototyping Environment for Heterogeneous Multi-Agent Systems'. *Journal of Autonomous Agents and Multi-Agent Systems*. to appear.

[49] Parsons, S., C. Sierra, and N. R. Jennings: 1998, 'Agents that reason and negotiate by arguing'. *Journal of Logic and Computation* **8**(3), 261–292.

[50] Procedural Reasoning System Users's Guide: 2001. Artificial Intelligence Center, SRI Internetional, Menlo Park, CA. Version 2.0. `http://www.sri.com`.

[51] Rao, A. and M. Georgeff: 1992, 'An Abstract Architecture for Rational Agents'. In: C. Rich, W. Swartout, and B. Nebel (eds.): *Proceedings of the International Workshop on Knowledge Representation & Reasoning, KR'92*: Morgan-Kaufmann, pp. 439–449.

[52] Rao, A. S.: 1996, 'AgentSpeak(L): BDI Agents Speak Out in a Logical Computable Language'. In: R. van Hoe (ed.): *Agents Breaking Away, 7th European Workshop on Modelling Autonomous Agents in a Multi-Agent World, MAAMAW'96, Eindhoven, The Netherlands, January 22-25, 1996, Proceedings*, *Lecture Notes in Computer Science*, Vol. 1038: Springer-Verlag, pp. 42–55.

[53] Sadri, F., F. Toni, and P. Torroni: 2002a, 'An abductive logic programming architecture for negotiating agents'. In: S. Greco and N. Leone (eds.): *Proceedings of the 8th European Conference on Logics in Artificial Intelligence (JELIA)*, *Lecture Notes in Computer Science*, Vol. 2424: Springer-Verlag, pp. 419–431.

[54] Sadri, F., F. Toni, and P. Torroni: 2002b, 'Dialogues for negotiation: agent varieties and dialogue sequences'. In: *Intelligent Agents VIII: 8th International Workshop, ATAL 2001, Seattle, WA, USA, Revised Papers*, *Lecture Notes in Artificial Intelligence*, Vol. 2333: Springer-Verlag, pp. 405–421.

[55] Satoh, K.: 2002, 'Speculative Computation and Abduction for an Autonomous Agent'. In: *Proceedings of the 9th International Workshop on Non-Monotonic Reasoning, NMR'02, Toulouse, France*. pp. 191–199.

[56] Schroeder, M. and R. Schweimeier: 2002, 'Arguments and Misunderstandings: Fuzzy Unification for Negotiating Agents'. in [20], pp. 1–18.

[57] Shoham, Y.: 1993, 'Agent-oriented programming'. *Artificial Intelligence* **60**(1), 51–92.

[58] Societies Of ComputeeS (SOCS): a computational logic model for the description, analysis and verification of global and open societies of heterogeneous computees. IST–2001–32530. `http://lia.deis.unibo.it/Research/SOCS/`.

[59] Torroni, P.: 2002, 'Logics and Multi-agents: towards a new symbolic model of cognition'. in [20], pp. 177–180.

[60] van Dike Parunak, H.: 2000, 'A Practitioner's Review of Industrial Agent Applications'. *Journal of Autonomous Agents and Multi-Agent Systems* **3**, 389–407.

[61] Yolum, P. and M. Singh: 2002, 'Flexible protocol specification and execution: applying event calculus planning using commitments'. In: C. Castelfranchi and

W. Lewis Johnson (eds.): *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems, Bologna, Italy (AAMAS-2002), Part II*: ACM Press, pp. 527–534.