INTELLIGENT MOBILE MULTI-ROBOTIC SYSTEMS: SOME CHALLENGES AND POSSIBLE SOLUTIONS

Flávio S. Corrêa da Silva, Renata Wassermann, Ana Cristina V. Melo,
Leliane N. Barros, Marcelo Finger
Dept. of Computer Science, University of São Paulo
Rua do Matão, 1010, São Paulo, BRAZIL

Keywords: Distributed multi-agent robotic systems, software for robotic agents, robotic surveillance and security.

Abstract: Intelligent mobile multi-robotic systems (IMMRSs) are coordinated systems of autonomous mobile robots endowed with reasoning capabilities. This sort of systems requires the integrated application of a variety of state-of-the-art techniques developed within the realm of Artificial Intelligence, as well as instigates the further development of different specialisations of Artificial Intelligence. In the present article we examine some of these techniques and specialisations, discuss some specific challenges proposed to the field of Artificial Intelligence by IMMRSs, and suggest possible solutions to these challenges. In order to make our presentation more concrete, we employ throughout the article a specific example of IMMRS application, namely security surveillance of an empty building by a team of robots.

1 INTRODUCTION

Intelligent mobile multi-robotic systems (IMMRSs) are coordinated systems of autonomous mobile robots endowed with reasoning capabilities. These systems require the integrated application of a variety of state-of-the-art techniques developed within the realm of Artificial Intelligence (AI), as well as instigates the further development of different specialisations of AI.

In the present article we examine some of these techniques and specialisations, discuss some specific challenges proposed to AI by IMMRSs, and suggest possible solutions to these challenges. In order to make our presentation more concrete, we employ throughout the article a specific example of IMMRS application, namely security surveillance of an empty building by a team of robots (Gerkey et al., 2004): *Given an* environment*, modelled as a connected polygonal free space, a fixed number of* searchers *– autonomous mobile robots equipped with cameras, each camera having a fixed angular aperture – and an unknown number of* evaders *– autonomous entities which can move arbitrarily fast – determine* trajectories *for each of the searchers so that the detection of all evaders is guaranteed.*

This problem has high computational complexity with respect to the complexity of the environment (e.g. characterised by the number of edges of the polygon that comprises it) and the number of searchers. Moreover, determining the optimal (i.e. minimum) quantity of searchers given an environment is NP-hard (Gerkey et al., 2004).

This problem, however, does not take into account a few parameters that can be important to improve its accuracy for practical applications. In the following sections we analyse this problem under the light of different specialisations of AI, which provide us with conceptual tools to refine its description. The present work can be thus regarded as a refinement of the work presented in (Gerkey et al., 2004), aiming at taking it from foundational research that provides some mathematically well founded guidelines for IMMRS to a *de facto* applied work that can effectively be used to build IMMRS.

In section 2 we add to the model the inherent uncertainties of the estimates of where a searcher and an evader are at a given moment. In section 3 we consider the fact that sometimes the sensed environment and its corresponding model may disagree, requiring a revision of the model of the environment. The refinements of multi-robotic models that result from taking into account the features discussed in these two sections can lead to high consumption of computational resources. In section 4 we focus on the efficiency of inference systems, to counterbalance this consumption of resources. In section 5 we consider the possibility of the searchers communicating with

S. Corrêa da Silva F., Wassermann R., Cristina V. Melo A., N. Barros L. and Finger M. (2005).
INTELLIGENT MOBILE MULTI-ROBOTIC SYSTEMS: SOME CHALLENGES AND POSSIBLE SOLUTIONS.
In *Proceedings of the Second International Conference on Informatics in Control, Automation and Robotics - Robotics and Automation*, pages 479-485
DOI: 10.5220/0001190904790485

each other, to act in a coordinated fashion, exchanging not only data but also functionalities. In section 6 we discuss research on high level languages to represent and reason about robotic actions, plans and capabilities. Finally, in section 7 we present some final discussion and proposed future work.

2 MANAGEMENT OF UNCERTAIN LOCATION ESTIMATES

In (Gerkey et al., 2004) the information about the positions of the searchers and the sensed positions of the evaders is assumed to be perfectly certain, thus not taking into account the imprecision of sensors. Among the many fields within AI to which uncertain reasoning is relevant, IMMRSs are perhaps the most prominent, due to the impossibility to idealise the environment in which the designed reasoning agents (autonomous robots in this case) shall inhabit (Gasos and Saffiotti, 1999; Saffiotti, 1999).

Efficient probabilistic reasoning has challenged the theorists for many years (Dix et al., 2000; Ng and Subrahmanian, 1992). The usual approach to control the computational complexity of probabilistic reasoning has been to look for special instances of reasoning problems which present useful probabilistic properties (Campos and Cozman, 2004; Ide and Cozman, 2004; Rocha and Cozman, 2003).

Let us consider the surveillance problem with a single searcher. Differently from the classical formulation of this problem, let us also consider that the accuracy of a sensor varies depending on the positions of the searcher and the target point in the environment (e.g. the reliability of the readings of the sensors can be inversely proportional to the distance separating the searcher and a point in the environment). Hence, assuming that positions are characterised by plane coordinates (x, y), if the searcher is at point (x_r, y_r) and sends a message stating that there is an evader at point (x, y), we should from this message infer a collection of probabilities related to points surrounding $(x1, y1)$ such that $(x_i, y_i) : \mu_i$ should be read as "there is a probability μ_i that point (x_i, y_i) is occupied by an evader".

We must rely on the ability of the searcher to locate itself in the environment, and the message sent by the searcher in this setting must contain *two* pairs of coordinates, one for the actual position of the searcher (x_r, y_r) and another for the position of the point possibly occupied by an evader (x, y). Both pairs of coordinates can be inaccurate, hence if we get a message from the searcher stating that it is at point (x_r, y_r), we should from this infer a collection of probabilities related to points surrounding (x_r, y_r) such that $(x_{ri}, y_{ri}) : \mu_{ri}$ should be read as "there is a probability μ_{ri} that the searcher is at point (x_{ri}, y_{ri})".

The composition of these two probability estimates can provide us with estimates for the probabilities that effectively there is an evader at certain points within the building. Assuming that the plane on which the searcher navigates is organised as a homogeneous square grid, the more accurate the sensors are, the smaller are the areas with significant probabilities given a position (x_r, y_r) and the areas with significant probabilities given a suspicious point (x, y). Thus, the accuracy of the sensors and the coarseness of the grid determine the amount of data needed to estimate probabilities of having evaders at different points. Accurate sensors and coarser grids make for smaller amounts of data. Hence, given the accuracy of the sensors of a searcher, one can control the size of the lists by making the grid squares larger or smaller.

This work is detailed in (Silva, 2005). An application to robot navigation for surveillance is also outlined in that reference.

3 EFFICIENT REVISION OF BELIEFS ABOUT THE ENVIRONMENT

In (Gerkey et al., 2004) it is assumed that the information about the environment is perfectly reliable. Suppose, however, that there are two searchers looking after an environment. Each searcher receives a map of the environment, and then the searchers build an internal model of the environment. As they go around the physical space, they may observe facts that contradict the information on the map. The robots must be able to deal with this new information. There are two main issues involved: (1) is the observation correct, so that it should be incorporated? (2) If a robot decides to accept the information, how does it accommodate it?

Belief revision (Gärdenfors, 1988; Hansson, 1999) is the sub-area of AI that deals with these problems. We are still far from having an acceptable framework to deal with multiple agents, dynamic worlds and real-time reasoning. In the present project, we have been working mainly on the last issue, i.e., how to employ logical models of belief revision to realistic problems in which computation time matters.

The main idea is to cope with the high complexity of logical reasoning by looking for approximate answers. We have extended Cadoli and Schaerf's framework for approximate reasoning (Sch95) in (Finger and Wassermann, 2004; Finger and Wassermann, 2005) as a possible solution. In previous work (Was99; Was01; Chopra et al., 2001), we had already proposed the use of approximate reasoning in the area of belief revision, but only for the clausal fragment of propositional logic. Our extension deals with full propositional logic and has a tableaux-based proof method that was implemented and tested. We presented families of *anytime reasoners* – reasoners that can be stopped anytime giving an approximate answer to a query. If the agent is given more time, the quality of the approximation is improved.

In (Was99), there was another proposal for approximate reasoning, based on the idea of relevance. The original version was for propositional logic, but has been extended to first-order logic (Ria04a; Ria04b).

Belief revision involving multiple agents is also a topic which started to receive some attention recently (Roo03). The main problem here is that when each agent holds beliefs about other agents' beliefs, every new information received by some agent gives rise to a cascade of actualizations in every agents' beliefs. Recent work by Cantwell (Cantwell, 2005) tries to avoid this problem by defining belief states as primitives. While this works well on the formal side, it is still not clear how this can be applied to real world problems, chiefly due to computational complexity issues.

4 UNCERTAIN REASONING VIA APPROXIMATE REASONING

Attributing probability to the conclusion of a logical reasoner based on the a priori probabilities of the premises has been for a long time an active area of research (Nilsson, 1986).

The problem with this approach is its intractability, whose sources are both the intractability of logical inferences and the multiplicity of states that have to be considered to compute the interval of probabilities of conclusions. The latter problem may be linked to the fact that the attribution of probabilities to formulae is not truth functional, in the sense that one cannot always infer the probability of a formula simply by knowing the probability of its components. So one has to consider all the exponentially many possible valuations of a formula to compute probability intervals, even if the probability of all atomic facts are known.

Our goal is to solve some of these problems by using *approximations of classical inference*, as introduced by Schaerf and Cadoli (Sch95) and Dalal (Dalal, 1996). The idea here is to work in some subclassical logic that proves less theorems than classical logic, but in which inference is tractable. This approach was initially restricted to clausal form logic, without an established proof theory. These approximations have been extended to full classical logic with a well-defined proof theory (Massacci, 1997; Finger and Wassermann, 2002; Finger and Wassermann, 2005) and recently, a more refined control on the complexity of such approximations was established (Finger, 2004).

In this work, we expect to apply this latter approach to deal with some aspects of probabilistic logic. We assume that we are dealing with a propositional language based on a finite number of atoms $p_1, \ldots, p_n$ and the usual connectives $\neg, \wedge, \vee$ and $\rightarrow$.

The attribution of probabilities to a formula cannot be inferred from the probabilities of its subformulae. All we have is that, if $P(A)$ is the attribution of probabilities to a propositional formula A, such that $0 \leq P(A) \leq 1$, then the following must hold: (1) if $\vdash A$ then $P(A) = 1$; (2) if $\vdash \neg(A \wedge B)$ then $P(A \vee B) = P(A) + P(B)$. From this basic fact, it is possible to infer the following properties: if the symbol '$\vdash$' represents classical logical consequence, then: (1) $P(\neg A) = 1 - P(A)$; (2) if $A \vdash B$ then $P(A) \leq P(B)$; (3) if $\vdash A \leftrightarrow B$ then $P(A) = P(B)$; (4) $P(A \vee B) = P(A) + P(B) - P(A \wedge B)$.

The idea is now to consider $\vdash$ not as classical inference, but some tractable approximate inference, and assume the same definition for the attribution of probabilities. We want to investigate which of the properties above are preserved.

We present the tractable approximation here only in semantic terms; a full proof theory is discussed in (Finger, 2004). This semantics is called *Limited Bivaluation*, LB, and is paramererized by a set Σ of formulae.

The semantics of LB(Σ) is based on a three-level lattice, $L = (L, \sqcap, \sqcup, 0, 1)$, where L is a countable set of elements $L = \{0, 1, \epsilon_0, \epsilon_1, \epsilon_2, \ldots\}$ such that $0 \sqsubseteq \epsilon_i \sqsubseteq 1$ for every $i < \omega$ and $\epsilon_i \not\sqsubseteq \epsilon_j$ for $i \neq j$. The ϵ_i's are called *neutral* truth values. This lattice is enhanced with a *converse operation*, $\sim$, defined as: $\sim 0 = 1$, $\sim 1 = 0$ and $\sim \epsilon_i = \epsilon_i$ for all $i < \omega$.

A *limited valuation* is a function $v_\Sigma : \mathcal{P} \rightarrow L$ that maps formulae to elements of the lattice that are subject to a set of restrictions with regards to whether a formula is or is not in the parameter set Σ. Initially, the limited valuation v_Σ maps atoms to the elements of the lattice and is extended to all formulae in the following way: (1) $v_\Sigma(\neg A) = \sim v_\Sigma(A)$ (2) $v_\Sigma(A \wedge B) = v_\Sigma(A) \sqcap v_\Sigma(B)$ (3) $v_\Sigma(A \vee B) = v_\Sigma(A) \sqcup v_\Sigma(B)$ (4) $v_\Sigma(A \rightarrow B) = 1$ if $v(A) \sqsubseteq v(B)$; $\sim v_\Sigma(A) \sqcup v_\Sigma(B)$ otherwise.

A further constraint, called the Limited Bivalence Restriction, is imposed on v_Σ for formulae in Σ: formulae in Σ are bivalent, that is, if $A \in \Sigma$ then $v_\Sigma(A)$ must be *bivalent*, that is, $v_\Sigma(A)$ must satisfy the rules above for unlimited valuations and be such that $v_\Sigma(A) = 0$ or $v_\Sigma(A) = 1$.

When $A \notin \Sigma$, $v_\Sigma(A)$ is not always *compositional*, which means that a neutral value may be assigned to A independently of the truth value of its components.

This is the case so that the bivalence of $A \in \Sigma$ can always be satisfied without forcing all A's subformulae to be bivalent.

If $A \in \Sigma$ it is always possible to have $v_\Sigma(A) \in \{0,1\}$ by making for every atom p in A, $v_\Sigma(p) \in \{0,1\}$. However, this is not the only possibility. For example, if $B, C \notin \Sigma$ then we can make $v_\Sigma(B) = \epsilon_i \neq \epsilon_j = v_\Sigma(C)$, so that $v_\Sigma(B \wedge C) = 0$; similarly, we obtain $v_\Sigma(B \vee C) = 1$ and $v_\Sigma(B \rightarrow C) = 1$.

A valuation v_Σ *satisfies* A if $v_\Sigma(A) = 1$, and A is called *satisfiable*; a set of formulae Γ is satisfied by v_Σ if all its formulae are satisfied by v_Σ. A valuation v_Σ *contradicts* A if $v_\Sigma(A) = 0$; if A is neither satisfied nor contradicted by v_Σ, we say that v_Σ is *neutral* with respect to A. A valuation is *classical* if it assigns only 0 or 1 to all proposition symbols, and hence to all formulae.

The notion of a *parameterised* LB-*Entailment*, $\models^{\mathsf{LB}}_\Sigma$, is obtained by defining, for a set of formulae Γ and a formula A, $\Gamma \models^{\mathsf{LB}}_\Sigma A$ if no valuation v_Σ such that $v_\Sigma(\Gamma) = 1$ also makes $v_\Sigma(A) = 0$. This consequence relation is not classic, for if $\Gamma \models^{\mathsf{LB}}_\Sigma A$ and $v_\Sigma(\Gamma) = 1$ it is possible that A is either neutral or satisfied by v_Σ.

We now define the tractable entailment $\models^{\mathsf{LB}}_k$, parameterized by an integer k. For that, let $\mathbb{S}$ be a set of sets of atoms and, for every $\Pi \in \mathbb{S}$, let Π^+ be the closure of Π under formula formation. We define $\Gamma \models^{\mathsf{LB}}_{\mathbb{S}} A$ iff there exists a set $\Pi \in \mathbb{S}$ such that $\Gamma \models^{\mathsf{LB}}_{\Pi^+} A$. We define $\mathbb{S}_k = \{\Pi \subseteq \mathcal{P} |\ |\Pi| = k\}$. That is, $\mathbb{S}_k$ is a set of sets of atoms of size k. Note that if we restrict our attention to n atoms, $|\mathbb{S}_k| = \begin{pmatrix} n \\ k \end{pmatrix} = O(n^k)$ sets of k atoms.

We write $\models^{\mathsf{LB}}_k$ to mean $\models^{\mathsf{LB}}_{\mathbb{S}_k}$. We have focused on the used of a quadratic decision procedure, $\models^{\mathsf{LB}}_2$. We then apply the definition of probability attibution using such sub-classical entailment, as follows: (i) if $\models^{\mathsf{LB}}_2 A$ then $P(A) = 1$; (ii) if $\models^{\mathsf{LB}}_2 \neg(A \wedge B)$ then $P(A \vee B) = P(A) + P(B)$. From that, we can prove that the following classical properties are preserved: if the symbol '$\vdash$' represents classical logical consequence, then (1) $P(\neg A) = 1 - P(A)$; (2) if $A \models^{\mathsf{LB}}_2 B$ then $P(A) \leq P(B)$; (3) if $\models^{\mathsf{LB}}_2 A \leftrightarrow B$ then $P(A) = P(B)$.

In fact, the results above hold for any integer k, and not only for $k = 2$. However, the following property fails: $P(A \vee B) = P(A) + P(B) - P(A \wedge B)$. This is due to the fact that a classical theorem does not hold, in general, for any k: $\not\models^{\mathsf{LB}}_k A \vee B \leftrightarrow (A \wedge \neg B) \vee (A \wedge B) \vee (\neg A \wedge B)$. In fact, to obtain an expression for $P(A \vee B)$ we have to consider the cases where A and B can have neutral values ϵ_i.

That is, although it may be less complex to compute an inference in $\models^{\mathsf{LB}}_k$ than classical inference, the computation of probabilities has to take in consideration many more terms.

Future work consider the investigation of the complexity of using probabilistic logic based on $\models^{\mathsf{LB}}_2$ and the effects of loss of expressivity on this inference in the computation of probabilities.

5 FORMAL SPECIFICATION AND VERIFICATION OF IMMRSS

As usual, it is assumed in (Gerkey et al., 2004) that the embedded software in each robot is *fixed* within that robot. Indeed, it is usually accepted that each robot is the physical embodiment of an agent in a multiagent system. To our understanding, this constraint in the design of IMMRSs is unnecessary and restrictive. If robots can exchange messages to coordinate their actions, there is no reason why they cannot also exchange lines of code that implement actions themselves. This amounts to a decoupling of the notions of robot and agent: a single robot, in this framework, can accomodate more than one agent simultaneously, and agents can migrate between robots.

The decoupling of the notions of robot and agent greatly extends the flexibility in the specification, design and implementation of IMMRSs. It also makes these activities far more complex. In order to keep such systems under control, it can be important to employ formal techniques for the specification and verification of mobile agents for robotic systems.

The π-calculus (Milner, 1999) is a theory for mobile agents based on an algebra for concurrent processes, the CCS (Milner, 1989). In a very simplified way, mobile agents can be regarded as concurrent processes endowed with the capability of dynamic reconfiguration. The π-calculus has added to the CCS, among other things, features of mobility and dynamic reconfiguration.

It is relevant for the formal specification and verification of mobile agents the development of tools for automatic verification of mobile agents, based on novel verification techniques or the combination of existing techniques. Our work has focused on (1) techniques to get rid of useless code in the specification of mobile agents and their formal specification; (2) novel techniques for the formal verification of mobile agents; (3) combination of formal verifiers of mobile agents; (4) ontologies for mobile agents and mobility features, to support the combination of systems for specification and verification of mobile agents; (5) ontologies about the capabilities of formal verifiers; (6) formal models of autonomous and mobile agents; (7) automatic code generation for the communication between mobile agents; and (8) tools for the specification and integration of formal verifiers for mobile agents.

The behaviour of mobile agents must be tested, based on explicitly defined criteria (Wey88; McGregor and Korson, 1994; Chen and Kao, 1999; Kung et al., 1995) that determine what should be tested. Our work on this field has focused on (1) the study of what sets of properties can be verified in programs, and how the verification of these properties simplifies the tests requirements; (2) tools to support the combination of formal verification and tests, e.g. the automation of object control graphs; (3) the extension of model based formal methods to the specification of exception handling; (4) the study of how to use formal verifiers to improve the testing of software components with exception handling.

Some preliminary results in these topics were published in (Melo, 2005; Melo, 2004; Nunes and Melo, 2004; Andrade et al., 2004; Mel04b; Melo, 2003).

6 HIGH LEVEL REASONING ABOUT ROBOTIC ACTIONS

The problem solved in (Gerkey et al., 2004) is how to determine the plans (i.e. trajectories in the environment) for a collection of searchers, based on primitive actions such as *move forward* and *turn 90 degrees to the left*. This formulation of the problem scales badly as the number of searchers increases. Another possible planning strategy is a higher-level set of actions, possibly involving teams of robots, specified in terms of the above primitive actions. This is an AI planning approach called *Hierarquical Task Network Planning* (HTN), that has been proved to be more powerful then the search in the state/action space solution using only primitive actions (Kutluhan et al., 1995).

The decomposition of a goal task adds other goal tasks to be decomposed and this is very similar to what it is done in a regressive partial-order planner that adds actions to satisfy the goal making the preconditions of these actions to become new goals (*subgoaling*) to plan for. In fact, it has been proved (Kutluhan et al., 1995) that planning with goal tasks allows the HTN planners to inherit the efficiency of the partial-order planners. HTN planners have been applied in many applications, such as a system for integrated product design and manufacturing planning, and computer games like the winer of the 1997 world championship of computer bridge.

Golog (Levesque et al., 1997) is a logical language, implemented as a Prolog meta-interpreter, that allows the definition of procedures to describe the behavior of an agent, using the *Situation Calculus* (McCarthy, 1963) to represent knowledge, actions and states of the world. A programmer can specify a robot high-level control program as a set of high-level procedures, which are decomposed by Golog into low-level procedures (or primitive actions) during execution time, very similar to HTN planning. Golog is a high-level agent programming language, in which standard programming constructs (*e.g.* *sequence*, *choice* and *iteration*) are used to write the agent control program. It can effectively represent and reason about the actions performed by agents in dynamic environments. The emerging success of Golog has shown that, by using a logical approach, it is possible to solve complex robotic tasks efficiently, despite contrary belief. However, Golog uses a planning strategy based on situation calculus, a logical formalism in which plans are represented as a totally ordered sequence of actions and, therefore, it inherits the well known deficiencies of this approach (Kutluhan et al., 1995).

Since Golog decomposes procedures in a very similiar way that it is done by HTN planning (Gabaldon, 2002), we have been working on the proposal of a number of extensions in the Golog meta-interpreter (Barros and Iamamoto, 2003; Iamamoto, 2005), namely (1) we have introduced the idea of *goal tasks* into the Golog language; (2) based on the formalisation of HTN planning (Kutluhan et al., 1995), we have proposed a special kind of Golog procedures, called *Goal Procedures,* that can be decomposed by the Golog meta-interpreter to solve *goals of attainment problems*; (3) we have presented a way to interleave Goal Procedures to solve action interactions (conflicting subgoals); (4) we have also compared the efficiency of using Goal Procedures with an example of a Situation Calculus planner proposed by Reiter, the *Wspbf* planner, encoded in Golog (Rei01) and showed that our proposal, besides being more compatible with Golog way of planning through decompositions, can be also more efficient; (5) we have argued that Goal Procedures can be very useful in an on-line execution to replanning when action's execution fails.

We are currently working on the use of Goal Procedures to replanning in *IndiGolog*, an extention of GOLOG to perform *on-line* execution of programs.

We have also proposed a new high-level robot programming language, called ABGOLOG (Per04b; Per04c; Per04a). This language is based on Golog, but it uses Event Calculus as the formalism to describe actions and abductive reasoning to synthesize plans, which corresponds to partial order planning. So, based on our previous work on implementation and analysis of abductive event calculus planning systems (Per04b), we have shown how it is possible to modify AbGolog's implementation to improve its efficiency, according to specific domain characteristics.

In (Tre05) we have shown an example of a robot application using a Lego® MindStorms™robot. The implementation was done in *Legolog*, a package composed by the Indigolog language and communication protocols for the Lego® MindStorms™robot.

7 CONCLUSION

The design and implementation of IMMRSs is a great challenge to AI and related areas. Chief among the reasons to make this sort of systems so challenging is the fact that one cannot idealise the environments upon which the systems shall act, and hence they require refined methods to ensure the coupling between idealised models (based on which the systems are programmed) and physical environments.

In the present article we listed some areas of Artificial Intelligence that we believe are most relevant to multi-robotic systems, and that at the same time are more strikingly challenged by those systems. We suggested some solutions to specific challenges, which are the ones on which we are working at the moment.

REFERENCES

A. G. Andrade, A. C. V. Melo, and M. M. Amorim. Da especificao verificao de agentes mveis – um ambiente grfico. In Mauricio Solar and David Fernandez Baca, editors, *Anales de 30th Conferencia Latino Americana de Informtica*, Arequipa, Peru. 2004.

L. N. de Barros and E. Iamamoto. Planning for tasks in cognitive robotics. Proceedings of VI Simpsio Brasileiro de Automao Inteligente, Bauru, SP, Brazil. 2003.

C. P. Campos, and F. G. Cozman. Inference in credal networs using multilinear programming. Second Starting AI Researcher Symposium (STAIRS), 50-61, IOS Press. 2004.

J. Cantwell. A formal model of multi-agent belief-interaction. *Journal of Logic, Language and Information*. 2005. to appear.

M.-H. Chen, and H. M. Kao. Testing object-oriented programs - an integrated approach. In *Proceedings of the Tenth International Symposium on Software Reliability Engineering*. IEEE Computer Society Press. 1999.

S. Chopra, R. Parikh and R. Wassermann. Approximate belief revision. *Logic Journal of the IGPL*, 9(6):755–768. 2001.

F. S. Correa da Silva. Where am I? Where are you? Technical Report 2005-03. Department of Computer Science, University of Sao Paulo. 2005.

M. Dalal. Anytime families of tractable propositional reasoners. In *International Symposium of Artificial Intelligence and Mathematics AI/MATH-96*, pages 42–45. 1996.

J. Dix, M. Nanni, and V. S. Subrahmanian. Probabilistic agent programs. ACM Transactions on Computational Logic, v. 1(2), 207–245. 2000.

M. Finger. Polynomial approximations of full propositional logic via limited bivalence. In *9th European Conference on Logics in Artificial Intelligence (JELIA 2004)*, LNAI vol. 3229, pages 526–538. 2004.

M. Finger and R. Wassermann. Expressivity and control in limited reasoning. In Frank van Harmelen, editor, *15th European Conference on Artificial Intelligence (ECAI02)*, pages 272–276, Lyon, France. 2002.

M. Finger and R. Wassermann. Approximate and limited reasoning: Semantics, proof theory, expressivity and control. *Journal of Logic And Computation*, 14(2):179 – 204. 2004.

M. Finger and R. Wassermann. The universe of propositional approximations. *Theoretical Computer Science*. 2005. to appear.

A. Gabaldon. Programming hierarchical task networks in the situation calculus. AIPS-2002 Workshop on On-line Planning and Scheduling. Toulouse, France. 2002.

P. Gärdenfors. *Knowledge in Flux - Modeling the Dynamics of Epistemic States*. MIT Press. 1988.

J. Gasos, and A. Saffiotti. Using fuzzy sets to represent uncertain spatial knowledge in autonomous robots. Spatial Cognition and Computation, v. 1, 205-226, Kluwer. 1999.

B. P. Gerkey, S. Thrun, and G. Gordon. Visibility-based pursuit-evasion with limited field of view. In *Proceedings of the National Conference on Artificial Intelligence (AAAI 2004)*. AAAI Press. 2004.

S. O. Hansson. *A Textbook of Belief Dynamics*. Kluwer Academic Press. 1999.

E. Iamamoto. HTN Planning in Golog. Msc Dissertation at the Departament of Computer Science of the University of So Paulo, Brazil. 2005.

J. S. Ide, and F. G. Cozman. IPE and L2U: Approximate algorithms for credal networks. Second Starting AI Researcher Symposium (STAIRS), 118-127, IOS Press. 2004.

D. Kung, N. Suchak, P. Hsia, Y. Toyoshima, and C. Chen. Object state testing for object-oriented programs. In *Proc. of COMPSAC'95*. IEEE Computer Society Press. 1995.

E. Kutluhan, J. Hendler, and D. Nau. Complexity, decidability and undecidability results for domain-independent planning. *Artificial Intelligence* 76(1-2):75-88. 1995.

H. J. Levesque, R. Reiter, Y. Lesprance, F. Lin, and R. B. Scherl. Golog: A logic programming language for dynamic domains. 1997.

F. Massacci. A proof theory for tractable approximations of propositional reasoning. In Maurizio Lenzerini, editor, *AI*IA-97*, volume 1321 of *LNAI*, pages 219–230. SV. 1997.

J. McCarthy. Situations, actions and causal laws, *Technical report, Stanford University. Reprinted in Semantic Information Processing (M. Minsky ed.), MIT Press, Cambridge, MAss., 1968, pp. 410-417*. 1963.

J. D. McGregor, and T. D. Korson. Integrated object-oriented testing and development processes. *Communications of ACM*, 37(9). 1994.

A. C. V. Melo. From active names to pi-calculus rewriting rules. *ENTCS (Electronic Notes in Theoretical Computer Science)*, to appear. 2005.

A. C. V. Melo. π-calculus rewriting rules based on active names. In Alexandre Mota and Arnaldo Moura, editors, *SBMF2004: 7th Brazilian Symposium on Formal Methods*, Recife, PE, Brazil. 2004.

A. C. V. Melo. A study on the potential active names of π-agents. *ENTCS (Electronic Notes in Theoretical Computer Science)*, 95(C):269–286. 2004.

A. C. V. Melo. A study on the potential active names of π-agents. In A. L. C. Cavalcanti and Patrcia Machado, editors, *WMF2003: 6th Brazilian Workshop on Formal Methods*, Campina Grande, PB, Brazil. 2003.

R. Milner. *Communication and Concurrency*. Prentice–Hall, first edition. 1989.

R. Milner. *Communicating and Mobile Systems: the π-Calculus*. Cambridge University Press. 1999.

R. Ng, and V. S. Subrahmanian. Probabilistic logic programming. Information and Computation, v. 101(2), 150–201. 1992.

N. Nilsson. Probabilistic logic. *Artificial Intelligence*, 28(1):71-87. 1986.

P. R. Nunes and A. C. V. Melo. Ocongra - uma ferramenta para gerao de grafos de controle de fluxo de objetos. In Murilo Camargo and Jaelson Castro, editors, *Anais do Simpsio Brasileiro de Engenharia de Software*, Braslia, Brasil. 2004.

S. L. Pereira and L. N. Barros. High-Level Robot Programs Based on Abductive Event Calculus. International Workshop on Cognitive Robotics, CogRob-2002. 2002.

S. L. Pereira and L. N. Barros. Formalizing planning algorithms: a logical framework for the research on extending the classical planning approach. International Conference on Planning Systems Workshop: Connecting Planning Theory with Practice. 2004.

S. L. Pereira and L. N. Barros. Planning with Abduction: a logical framework to explore extensions to classical planning. Simpsio Brasileiro de Inteligncia Artificial, *LNAI*. 2004.

S. L. Pereira and L. N. Barros. High-level Robot Programming: an abductive approach using event calculus. Simpsio Brasileiro de Inteligncia Artificial, *LNAI*. 2004.

R. Reiter. Knowledge in Action: Logical Foundations for Specifying and Implementing Dynamical Systems. MIT Press. 2001.

J. Riani. Towards an efficient inference procedure through syntax based relevance. Master's thesis, Department of Computer Science, University of São Paulo. 2004. Available at http://www.ime.usp.br/˜joselyto/mestrado.

J. Riani and R. Wassermann. Using relevance to speed up inference – some empirical results. In *Proceedings of the Brazilian Artificial Intelligence Symposium*, Lecture Notes on Artificial Intelligence. Springer-Verlag. 2004.

J. C. F. Rocha, and F. G. Cozman. Inference in credal networks with branch-and-bound algorithms. Third International Symposium on Imprecise Probabilities and Their Applications, 482–495, Carleton Scientific. 2003.

J. W. Roorda, W. Van der Hoek, and J. J. Meyer. Iterated belief change in multi agent systems. *Logic Journal of the IGPL*, 11(2):223–246. 2003.

A. Saffiotti. Handling uncertainty in control of autonomous robots. in Wooldridge, M., Veloso, M. (eds.) Artificial Intelligence Today, 381–408, Springer-Verlag. 1999.

M. Schaerf and M. Cadoli. Tractable reasoning via approximation. *Artificial Intelligence*, 74(2):249–310. 1995.

F. W. Trevizan and L. N. de Barros. Planejamento e Execuo em Golog para Robs Lego Mindstorm. Encontro Nacional de Inteligncia Artificial. 2005.

R. Wassermann. *Resource-Bounded Belief Revision*. PhD thesis, Institute for Logic, Language and Computation — University of Amsterdam. 1999.

R. Wassermann. On structured belief bases. In Hans Rott and Mary-Anne Williams, editors, *Frontiers in Belief Revision*. Kluwer. 2001.

E. J. Weyuker. The evaluation of program-based software test data adequacy. *Communications of ACM*, 31(6). 1988.