Towards a Plan-based Learning Environment

Angelo E. M. Ciarlini
Departamento de Informática Aplicada
UNIRIO
angelo.ciarlini@terra.com.br

Antonio L. Furtado
Departamento de Informática
Pontifícia Universidade Católica do RJ
furtado@inf.puc-rio.br

Abstract

The use of the Plan Recognition/Plan Generation paradigm in the context of corporate training is discussed. The learning environment is grounded on three-level conceptual schemas of information systems, and offers a tool for simulating the behavior of agents with an adequate degree of realism. After arguing for the relevance of Plan-Based Learning, we stress the need of taking into account both cognitive and affective characteristics of the various agents operating in the specified multi-goal/multi-agent information system, as conveyed by their individual profiles and current internal states.

1. Introduction

The fundamental basis of our current research project is the development of more realistic methods for the *conceptual specification* of information systems, taking a broader perspective than their simple description as sets of software tools performing specific tasks. Information systems are recognized as complex structures composed of agents that can be either software agents, humans or organizations, which interact with each other. They may cover domains of practical applications, such as sales, banking, etc. Incorporating a temporal dimension, we can go beyond static descriptions to follow the *narratives* that arise in the mini-world delimited by the domain, consisting of events caused by the agents' interactions. Thus, in a banking application domain, one can usefully trace stories of clients handling their saving accounts and making investments, and their contacts with the management of the bank. But fiction also supplies domains, such as fairy-tales or detective stories, wherein descriptions and narratives are also amenable to computerized specification and simulation techniques ([1], [2]). The ability to handle domains belonging to literary genres seems particularly relevant to the growing area of entertainment applications ([3], [4]).

In our previous work [5], we showed how to elaborate formal specifications at three levels for information systems having a database component:

1. At the **static level**, *facts* are classified according to the Entity-Relationship model. Thus, a fact may refer either to the existence of an entity instance, or to the values of its attributes, or to its relationships with other entity instances. Entity classes may form an *is-a* hierarchy. All kinds of facts are denoted by *predicates*. The set of all facts holding at a given instant of time constitutes a *database state*.
2. The **dynamic level** covers the *events* happening in the mini-world represented in the database. Thus, a real world event is perceived as a *transition* between database states. Our dynamic level schemas specify a fixed repertoire of domain-specific *operations*, as the only way to bring about events and thus cause state transitions. Accordingly, we equate the notion of event with the execution of an operation. Operations are formally specified by the facts that should or should not hold as *pre-conditions* and by the facts added or deleted as the *effect* of execution.

PGLDB'2003, pp. 9-19, 2003.

3. The **behavioural level** models how agents are expected to act in the context of the system. To each individual agent (or agent class) A, we assign a set of *goal-inference rules*. A goal-inference rule A:S → G has, as antecedent, a *situation* S and, as consequent, a *goal* G, both of which are first-order logic expressions having database facts as terms. The meaning of the rule is that, if S is true at a database state, agent A will be motivated to act in order to bring about a state in which G holds. In addition, we indicate the *typical plans* (partially ordered sequences of operations) usually employed by the agents to achieve their goals.

The first two levels encompass an *object-oriented* view of information systems, whereas the third level extends this view to incorporate an *agent orientation*, with a stress on goal-driven requirement analysis ([6], [7]).

To experiment with our methods, we have been developing an executable prototype tool, using Logic Programming enhanced with Constraint Programming ([8], [9]). The tool, named **IPG** (Interactive Plot Generator), is based on a plan recognition/plan generation paradigm [10]. Plan recognition algorithms detect what plan an agent is trying to perform, by matching a few *observations* of the agent's behavior against a repertoire of typical plans contained in a previously constructed library of typical plans (TP-Library). Plan generation algorithms create plans as partially ordered sets of operations whose execution, starting from a given *state* of the mini-world, would lead to the satisfaction of an indicated *goal*. The plan recognition part of IPG implements Kautz's algorithm [11], whereas plan generation uses the techniques developed in the Abtweak project [12]. As will be seen later, plan modification is yet another useful function provided by IPG, whereby the plan generation algorithm starts with a still incomplete plan or with a plan that cannot reach the given goal due to unremovable obstacles.

Both the plan recognition and the plan generation algorithms in IPG proved to be powerful enough to handle not only plots of simple database narratives but also of fairly intricate folktales [13], allowing to perform *predictions* (via plan recognition) and *simulations* (through plan generation) over the real or fictional mini-worlds specified. In more detail, when using the three-level schemas for simulation, our prototype runs a multistage process in which the application of goal-inference rules alternates with planning phases. The execution of a plan brings about new situations, which may lead to new goals, and so forth; these iterations continue until either there is no new goal to be inferred or the user decides to stop the process.

Predictions and simulations serve a variety of purposes (e.g. to help decision-making). In special, as will be argued in this paper, both may prove most helpful for corporate training, as services to be offered in the context of **learning environments**, installed over information systems having a database or data warehouse component [14]. Section 2 introduces the Plan-Based Learning concept, and sections 3, 4 and 5 outline further methodological enhancements to be incorporated into our project. Section 6 contains concluding remarks.

2. The Plan-Based Learning concept

Plan-Based Learning (hereafter **PBL**), as an application of the plan recognition/plan generation paradigm to learning environments, can basically take one of the following modalities:

a. Learning by plan recognition
b. Learning by plan generation
c. Learning by plan modification

Consider the case of a sales information system, wherein the classes of agents are salesmen and clients, each class having available a repertoire of operations. Assume further that different sub-classes of clients have been identified in the past, according to their personality (borrowing from [15]): dominant, political, steady, and wary. Although the same operations are allowed to all clients, their pre-conditions and effects may be somewhat different for each sub-class. For instance, a wary client will not *buy* a product unless, among other requirements, he is sure that an unconditional "satisfaction guaranteed or your money back" policy is adopted by the Company. Accordingly, such clients may have been observed to resort to a typical plan including the request of a printed form stating this policy.

Besides modelling the possible operations and typical plans of each class of agent, we usually need to model goal-inference rules, stating the behaviour of the agents of our system, in particular of those who are not under our control. In this way, we are able to test and choose the best policies to achieve our goals. In our example, a rule could say that advertising very often by e-mail would make clients willing not to have any contact with the Company at all. In a simulation that takes such a rule into account, the possibility of being included in an anti-spam list would probably be considered by the Company. In order to prevent this from happening, different advertisement strategies could be tried.

Now, suppose that a tool such as IPG is used in salesmen training. A particular salesman *S* can be given different kinds of cues to initiate an interaction in this environment. If modality (a) of **PBL** is used, he may be given the *observation* that the client solicited the above form. Submitting the observation to IPG, *S* would learn that the observation is part of the typical plan mentioned before; as a consequence, *S* would also learn that he has to do with a wary client, since the typical plan is registered in the TP-Library in connection with this sub-class. Learning how to recognize what plan an agent is currently engaged in, by noting which few actions he has executed thus far, is tantamount to learning to anticipate his next actions and the goals motivating his behaviour. Identifying the sub-class to which he belongs leads to the ability to also anticipate other plans that he may try in the future. More importantly, the trainee can perform simulations to evaluate the efficacy of his own intended actions — a wary client, for instance, cannot be expected to react with blind enthusiasm to the salesman's offer of "special" prices. In such simulations, it might be possible to foresee the result of the intended actions both when the client continues with the execution of the detected plan and when he tries different alternatives to achieve the same goals.

In modality (b) of **PBL**, the cue may be a *current state*, wherein a wary client already has received the form but is still not convinced that the policy is actually followed, and the *final goal* of the salesman to successfully complete the deal. Submitting both the state and the goal to IPG, *S* would learn from the tool a generated plan, which includes the delivery to the client of a list of previous clients in a position to confirm the consistent application of the policy. Learning to generate a plan exposes the trainee to the mechanics of planning: how to fulfill the pre-conditions of an action by previously executing other actions with the required effect, how to determine what must be done serially or may be done in different orders or in parallel, and how to find alternative sets of actions reaching the same objectives but possibly with significantly different side-effects. Moreover, if there is a behavioural model of the clients (i.e. goal-inference rules), it will be possible to simulate their reaction to the execution of plans, so that the trainee will be able to learn which plans render the best results in each situation.

In modality (c) of **PBL**, a possible cue would consist of a *current state* and a *typical plan* picked from the TP-Library. This would be the case, for example, when, after a preliminary modality (a) run, we detect that an agent is executing a plan to achieve a certain goal and we decide to help him, but the typical plan is not immediately executable. Suppose the agent's typical plan is the one we saw, adequate to wary clients and where the form is demanded, but assume that, at the current situation, there is an obstacle to the cooperation defined in the typical plan: the form is unavailable for the moment. Having these cues as input, IPG would supply to *S* a modified plan, in which the list of previous clients is delivered instead of the form. Learning to modify an existing typical plan is comparable to learning how to adapt for *reuse* stored software packages, a familiar concept from Software Engineering.

More generally, **PBL** is expected to prepare the trainee to work with, and choose between, two types of strategies to pursue a goal: (1) take a ready-made plan "from the shelf" (the typical plans) and use it as it is (or with only the changes that are absolutely necessary), noting that such plans correspond to the traditional — in some cases inadequate —practices of the corporation, or (2) create a solution from scratch, ideally trying to maximize the gains according to some criterion.

Not only executors, such as salesmen, but even those who design or are responsible for the operation or maintenance of an information system can benefit from **PBL** training. By experimenting with both kinds of plans, they can exercise a form of meta-level (also known as *double-loop* [16]) learning, wherein the trainee learns more about the meaning and implications of *business rules*, which must have been correctly captured in the conceptual specification. In particular, the trainee may fall upon unexpected situations reachable by plans, and sometimes detect loopholes, i.e. ways to violate constraints that the specification was supposed to enforce. Double-loop learning permits, besides checking whether the mini-world is functioning according to the specified model, to also question the quality of the model itself. Another possibility is considering the construction of the library of typical plans itself as a learning task in which the designers try to "learn" the typical behaviour of the agents. In a previous work [17], we presented a prototype we have implemented to semi-automatically help the construction of such a library based on an interpretation of events executed in the past and recorded in the system's log.

We are now proceeding to expand IPG, in an attempt to cope with more realistic scenarios. The current investigation mainly concentrates on how to model goal-inference rules and on how to use them during simulations. In its present implementation the tool relies on some rather common assumptions:

a. Omniscience – Agents (humans or organizations) cannot be expected to *know* all facts currently holding. An agent may well ignore a fact, and may even have an erroneous notion about it. So, an agent A may fail to behave as predicted by a goal-inference rule A:S $\rightarrow$ G, of which he is supposed to be aware, simply because he does not know that the motivating situation S holds.
b. Competence for logical reasoning – Similarly, human beings are not equally proficient to apply precise methods — logical inference, probabilities, etc. — to reach conclusions. For example, practical experiments [18] have demonstrated that people with training in statistics have been found to rate the occurrence of p $\wedge$ q as more probable than facts p or q alone! Likewise, even if A knows that S holds, he may fail to apply an apparently well-understood goal-inference rule leading to G as an implied consequence. This fact is also observed in software

agents, which may act according to very different patterns, varying from a purely reactive behaviour to complex reasoning mechanisms.

c. Rationality – Far more disturbing is to note how a person well provided with factual knowledge and reasoning skills, after duly concluding that a goal G corresponds to the best course of action available at the moment, can decide against it with no declared justification. An eloquent example is the episode having as protagonist the English philosopher Herbert Spencer, who chose not to move to New Zealand despite his conclusion that, according to his own evaluation — based on an astonishing early version of modern utility functions — his departure would be more advantageous than staying in England [19]. In the same way, software agents designed to act as similarly as possible to human beings can occasionally exhibit some kind of "irrational behaviour".

In the next sections some initial considerations on the enhancements envisaged for IPG in order to remove the need for these assumptions will be briefly outlined. They would certainly be relevant in several applications, and appear to be indispensable to effective **PBL** environments.

3. Internal states and profiles - cognitive elements

To avoid assuming omniscience, an *internal state* can be attributed to each agent A, registering the facts that A *believes*, correctly or not, as holding in the current global database state. We now establish that, for a goal-inference rule A:S $\rightarrow$ G to affect the behaviour of A, it is not enough that the facts denoting situation S be objectively true; in addition, such facts must be *believed* by A, and, accordingly, be part of A's internal state. One may even admit that, if the facts are believed by A, the rule is applicable even if they are not actually true.

Moreover, it is not enough that, believing S, A concludes that G is desirable. Except in cases where A's behaviour is purely *reactive*, he will still be free to decide, by some presumably objective criterion, whether or not he will actually *commit* to G as a goal and, consequently, adopt or develop a suitable plan Π to achieve it — which characterizes *deliberative* behaviour [20]. Individual beliefs, rather than global knowledge, and the concept of *intentions*, as the result of purposefully adding commitment to mere desires, are among the basic tenets of **BDI**-models ([21], [22], [23]).

Separate research has been applied to investigate what leads to this transition from desires to intentions. An approach that seems quite rational but is unfortunately hard to apply in domains of some complexity is based on the notion of *utility* [24]. Firstly, it requires that the desirability of a goal G be expressed by a numerical *utility value*. This would seem to be easy whenever a number is naturally attached to G; for instance, G may consist of the possession of an amount m of money. But the same amount m will have a different importance to people of different income levels, and so the utility value u of G, although depending on m, would not be necessarily identical to it. And, in general, the utility value may also be influenced by the internal state of the agent. If no quantitative attribute is attached to G, the determination of utility values becomes even harder. One should, at the very least, choose the values so as to ensure the ability to order situations according to their desirability; i.e. if G1 is intuitively more desirable than G2 then their respective utility values u1 and u2 should be determined so as to have u1 > u2.

An additional concern is that reaching a goal G by executing a plan Π is often an uncertain process. Instead of the purported G, the plan may achieve significantly different results G_1, G_2, ... , G_n, with probabilities p_1, p_2, ... , p_n, respectively. Of course, replacing G by the n possible results of Π requires that different utility values u_1, u_2, ... , un be assigned to each $_{Gi}$. The overall utility value of executing Π then becomes a statistical average, to be computed by a *utility function*:

$$U(\Pi) = \sum i \;\; pi \times ui, for\ i = 1, 2, \dots , n$$

Whenever there is more than a plan to reach a goal, the utility functions of all such plans have to be evaluated, and a "rational" agent should choose the plan of maximum utility. An analogous decision problem arises when an agent has to choose between two or more mutually exclusive goals (more about this in section 5), as in Spencer's dilemma. The difficulty of avoiding arbitrariness when determining utility values and the computational effort involved in the maximization calculations are drawbacks that must be recognized, since they can render unpractical the exhaustive comparison of all alternatives.

The adoption of internal states allows to consider what facts an agent A believes to be true at a given state, dropping thereby assumption (a) (omniscience). On the other hand, individual differences in logical reasoning competence, which underly assumption (b) (competence for logical reasoning), as well as other relatively stable (i.e. state-independent) personal characteristics of agents, should be captured in *profiles*, to be specified for each agent class with as small granularity as convenient, and even, if necessary, specialized for individual agents. For an initial design of profiles and their corrections and adaptations, as experience may demand, the methods and techniques in the *stereotype* approach to user-modelling ([25], [26]) look promising.

Whereas both internal states and profiles might be restricted to diverse *cognitive* elements — basically related to awareness of facts and expertise to apply rules — in order to abolish assumptions (a) and (b), another type of elements must be brought in, if we propose to do without assumption (c) (rationality) as well. Conceivably, Herbert Spencer decided to stay in England because he "felt" better staying there than moving to a remote country. Now, *feeling* is not a rare determinant in human decision-making, and a recent trend in Artificial Intelligence research — on which our next section is based — is dedicated to what has been called *affective computing* [27].

4. Internal states and profiles - affective elements

One must recognize that behaviour is largely influenced, sometimes determined, by *drives* and *emotions*, among other affective elements (e.g. *moods*, not treated here) ([28], [29], [30]). There is already some recognition that believable agents, i.e. agents that provide the illusion of life, show emotions even when trying to behave rationally, and, to some extent, act under their influence; this remark is still more crucial in attempts to combine agent technologies with those of the entertainment industry, including cinema, interactive television, computer games, and virtual reality [3]. *Drives* are basic physical needs, such as hunger and thirst, to which it is legitimate to add social needs, such as the urge to acquire money or prestige. *Emotions* have been classified according to distinct criteria, depending on the purpose of the classification; one popular classification considers six primitive emotions, with the convenient feature that they can be easily mapped into sharply distinguishable facial traits [31]: anger, disgust, fear, joy, sadness, surprise. An emotion can, in general, be either taken by itself, e.g. "a person is angry", or with respect to an object, another person or an event [32], e.g. "a person is angry at the prospect of being cheated by a salesman".

Both drives and emotions are amenable to a numerical scale representation, showing their intensity within a lower and a higher limit. A drive or emotion is said to be present in an agent if its intensity measure exceeds an appointed threshold. With the passage of time, the intensity of an unsatisfied drive increases. The satisfaction of a drive is accompanied by an increase in "positive" emotions (e.g. joy), whereas leaving it unsatisfied or, on the contrary, reaching an overwhelming regime by going beyond saturation, can stimulate "negative" emotions (anger, sadness). The intensity of an emotion decays after some time. Certain emotions are able to excite or inhibit other emotions, e.g. fear may excite anger and inhibit joy. As would be expected, the assignment of numerical measures is a no less delicate process here, needing to be validated for its adequacy in actual practice. Curious experiments [30] to emulate a toddler with purely reactive behaviour have been conducted, dealing with mutually stimulating or inhibiting interactions among various drives and emotions.

The intensities of drives and emotions of an agent A must be recorded as part of the internal state of A. Likewise, *personality traits* of A (e.g. whether or not A is an introvert, or is aggressive, etc.) should be part of A's profile. Thus, both internal states and profiles should contain both a *cognitive component* and an *affective component* which, together, contribute to determine A's behaviour. Indeed, it seems clear that most people decide under the combined influence of rational and affective factors. Both kinds of factors should therefore be taken into consideration as constituents of pre-conditions and effects of operations and, hence, of situations and goals in goal-inference rules. Additionally, they should both be taken into account in the determination of utility values; in special, the satisfaction of fundamental physical and social drives tends to be at the root level in goal hierarchies.

Examples of the relevance of affective factors are easy to find. A client will buy an item from a salesman only if he is happy with the salesman's service. Delays in delivery will have the effect of increasing the client's anger against the salesman. The action of watching a movie aims mostly at procuring pleasurable emotions, not rationally determined profit. Some goals, like the purchase of food, owe their desirability to their contribution to satisfy a drive, such as hunger. Computer interfaces offering unrequested advice may cause anger in users with a high degree of expertise on the subject, as their profile should indicate — and preventively lead the system to turn down the advice-giving facility. For Herbert Spencer, the utility value of going to New Zealand may have been reduced by his fear of facing a new physical and human environment. Making or keeping a client happy has been defined as a "softgoal" in the requirements analysis literature ([6], [7]), due to the imprecision of the notion of "happy"; one may expect that, by numerically measuring emotions and the increasing or decreasing effect that operations can have on their level, it should be possible to contribute towards the treatment of softgoals as ordinary goals.

5. Cognitive and affective factors in multi-goal/multi-agent environments

At a given state, some of the goals resulting from the application of goal-inference rules may form one or more sets of mutually interfering goals. Robert Willensky [33] notes that interferences can be separately characterized, on the one hand, as *negative* or *positive*, and, on the other hand, as *internal* (involving goals of the same agent) or *external* (goals of different agents). On the basis of these two dimensions, he proposes the following classification of goal interferences:

a. *goal conflict*: negative, internal;
b. *goal competition*: negative, external;
c. *goal overlap*: positive, internal;
d. *goal concord*: positive, external.

A major cognitive requirement in multi-agent environments has to do with the need to establish communication in order to adapt interfering goals and the corresponding plans. We saw that each agent perceives the external world in terms of beliefs, which are part of his internal state. Communication [34] between agents then means the ability of one agent to act on the other agent's internal state, changing his beliefs, typically by an exchange of information. *Speech acts* [35] thus provide an additional repertoire of operations — such as *inform* and *request* — noting that the latter is essential whenever an agent A1 wants another agent A2 to perform an operation which A2, but not A1, is authorized to execute. Operations corresponding to speech acts, besides being included in plans, intermingled with the domain-specific operations, can serve as a basis for *agent communication languages* [36].

But speech actions go beyond their cognitive effect. They are associated with emotions, which, in turn, may be manifested by facial expressions [37].

More generally, affective considerations certainly influence the choice of strategies for handling the various cases of interfering goals. Temperament traits, which we propose to model as part of the agents' profiles, may establish a preference for either goal abandonment or for aggressive outdo, or even undo (i.e. obstruct the competitor's plan), competitive acting. A prototype reported in [15] (and cited here at the beginning of section 2) has been developed to help training salesmen by simulating their interaction with clients with four different personalities; the same actions of a salesman were expected to elicit different reactions in each case.

A study of emotions that stresses interpersonal relationships [38], and was used in the above-mentioned training prototype, attempts to formally characterize what is meant by a number of words and phrases expressing emotions closely related to behaviour, grouped as follows:

Well-being: joy, distress;
Fortunes-of-others: happy-for, gloating, resentment, sorry-for;
Prospect-based: hope, satisfaction, relief, fear, fears-confirmed, disappointment;
Attribution: pride, admiration, shame, reproach;
Attraction: love, hate;
Well-being & attribution: anger, remorse.

Gloating, for example, as analysed in the corresponding expression in the authors' situation calculus formalism, means to be pleased about an event undesirable for another agent. Reproach is disapproving of the action of another agent, assuming that the action is considered blameworthy. Love and hate (or like and dislike) are not decomposed into simpler terms, being considered primitive and hence unexplainable.

Such kinds of emotions may well play a role in the choice of strategies. In a pair of salesmen competing to win a client, one of them may find that an undo strategy is justified if he feels reproach for past actions of the other salesman. On the contrary, he may spontaneously abandon his attempt, especially if he has a benevolent personality, in view of his admiration for the competitor. Individual agents may reconsider their goals to better suit the needs of a group to which they belong; in [38], for instance, an agent can demonstrate pride or shame for, respectively, a praiseworthy or blameworthy act attributed to a "cognitive unit" of which he is a member.

Going further, if the agents involved are not individual persons or groups, but rather industrial firms or some other kind of *organization*, it becomes far more difficult to characterize their activity in cognitive and affective terms. For human agents, computer scientists seek the orientation of Cognitive Psychology ([39], [40]). For organizations, fortunately, some clues are provided by Management Science, in particular from studies on Theories of Organization. Showing that the various proposed theories can be classified according to the metaphor through which they visualize what the concept of "organization" signifies, Gareth Morgan [16] argues convincingly that all classes of theories have important contributions to offer; for instance, whereas mechanistic theories stress a rational concern with efficiency and profit, other theories detect practices inherent in the company's traditional "culture", or the pressure of hidden agendas emerging from political struggles for power, etc.

6. Concluding remarks

The concept of Plan-Based Learning (**PBL**) opens a variety of possibilities for corporate training. By implementing both plan recognition and plan generation, our IPG tool seems to represent the right kind of engine needed to operate in **PBL** environments. However, we believe that the enhancements planned for the tool, to better support decision making and for entertainment applications, should be introduced before testing IPG in practical experiments.

It is widely recognized, in fact, that characters of computer-generated stories and games are expected to display lively personality traits, within the conventions of the chosen literary genre ([41], [1], [2]). Another example is provided by the cooperative interfaces [42] that emulate the behaviour of human beings, to provide useful responses in a friendly fashion. Similarly, **PBL** environments, modelling real people and corporations, cannot do without these relatively complex cognitive and affective features, both to mimic real situations as closely as possible, and to offer an attractive interface able to keep the attention of trainees.

References

[1] J. Carroll - The deep structure of literary representations. *Evolution and Human Behavior* 20 (1999) 159-173.

[2] N. M. Sgouros - Dynamic generation, management and resolution of interactive plots. *Artificial Intelligence* 107 (1999) 29-62.

[3] G. Davenport, S. Agamanolis, B. Barry, B. Bradley and K. Brooks - Synergistic storyscapes and constructionist cinematic sharing. *IBM System Journal*, 39, 3 & 4 (2000) 456-469.

[4] Special issue on digital entertainment - Scientific American, 283, 5, November (2000).

[5] A. E. M. Ciarlini and A. L. Furtado - Understanding and Simulating Narratives in the Context of Information Systems. Proc. of the 21th International Conference on Conceptual Modeling (2002).

[6] A. Dardenne, A. v. Lamsweerde and S. Fickas - Goal-directed requirements acquisition. *Science of Computer Programming* 20 (1993) 3-50.

[7] J. Mylopoulos, L. Chung and E. Yu - From object-oriented to goal-oriented requirements analysis. Communications of the ACM 42, 1 (1999) 31-37.

[8] M. Carlsson and J. Widen - *Sicstus Prolog Users Manual, Release 3.0.* Swedish Institute of Computer Science (1995).

[9] P. C. Kanelakis, G. M. Kuper and P. Z. Revesz - Constraint query languages. Journal of Computer and System Sciences 51 (1995) 26-52.

[10] A. L. Furtado and A. E. M. Ciarlini - The Plan Recognition/ Plan Generation Paradigm. In Information Systems Engineering. S. Brinkkemper, E. Lindencrona and A. Solvberg (eds.). Springer (2000) 223-235.

[11] H. A. Kautz. “A formal theory of plan recognition and its implementation”, in *Reasoning about Plans*. J. F. Allen et al (eds.). San Mateo: Morgan Kaufmann (1991).

[12] Q. Yang, J. Tenenberg. and S. Woods. “On the Implementation and Evaluation of Abtweak”. In *Computational Intelligence Journal*, Vol. 12, Number 2, pages 295-318, Blackwell Publishers (1996).

[13] V. Propp. *Morphology of the Folktale*. Laurence Scott (trans.). Austin: University of Texas Press (1968).

[14] S. W. M. Siqueira, M. H. Braz, R. N. Melo - E-learning content warehouse architecture. Proc. of the IADIS International Conference - Lisboa (2002).

[15] C. Elliott: Using the Affective Reasoner to Support Social Simulations. Proc. of the 13th International Joint Conference on Artificial Intelligence (1993) 194-201.

[16] G. Morgan - *Images of Organization: the Executive Edition*. Berrett-Koehler Publishers (1998).

[17] A. L. Furtado and A. E. M. Ciarlini. “Constructing Libraries of Typical Plans”. Proc. CAiSE´01, The Thirteenth International Conference on Computer Advanced Information System Engineering, Interlaken, Switzerland (2001).

[18] A. Tversky and D. Kahneman - Extensional versus intuitive reasoning: the conjunction fallacy in probability judgement. Psychological Review, 90 (1983) 293-315.

[19] W. Durant - *The Story of Philosophy*. Simon and Schuster (1961).

[20] A. Sloman and B. Logan - Architectures and tools for human-like agents. *Communications of the ACM* 42, 3 (1999) 71-77.

[21] P. R. Cohen, H. J. Levesque - Intention is Choice with Commitment. *Artificial Intelligence* 42, 2-3 (1990) 213-261.

[22] H. J. Levesque and G. Lakemeyer - *The logic of knowledge bases*. MIT Press (2000).

[23] A. S. Rao and M. P. Georgeff - Modeling rational agents within a BDI-architecture. Proc. of the International Conference on Principles of Knowledge Representation and Reasoning (1991) 473-484.

[24] S. Russell and P. Norvig - *Artificial Intelligence - a Modern Approach*. Prentice-Hall (1995).

[25] P. Persson, J. Laaksolahti and P. Lönnqvist - Proc. of Stereotyping characters: a way of triggering anthropomorphism? Socially Intelligent Agents - The Human in the Loop - AAAI Fall Symposium (2000).

[26] E. Rich - User modeling via stereotypes. *Cognitive Science* 3 (1979) 329-354.

[27] R. W. Picard, E. Vyzas and J. Healey - Toward machine emotional intelligence: analysis of affective physiological state. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 23, 10 (2001) 1175-1191.

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

[29] C. Breazeal - A motivational system for regulating human-robot interaction. Proc. of the Fifteenth National Conference on Artificial Intelligence, Madison (1998) 54-61.

[30] J. D. Velásquez - Modeling emotions and other motivations in synthetic agents. Proc. of the Fourteenth National Conference on Artificial Intelligence, Providence (1997) 10-15.

[31] G. Donato, M. S. Bartlett, J. C. Hager, P. Ekman and T. J. Sejnowski - Classifying facial actions. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 21, 10 (1999) 974-989.

[32] A. Ortony, G. L. Clore and M. A. Foss - The referential structure of the affective lexicon. *Cognitive Science* 11, 3 (1987) 341-364.

[33] R. Willensky - *Planning and Understanding - a Computational Approach to Human Reasoning*. Addison-Wesley (1983).

[34] G. Mantovani - Social context in HCL: a new framework for mental models, cooperation and communication. *Cognitive Science* 20, 2 (1996) 237-269.

[35] P. R. Cohen and C. R. Perrault - Elements of a plan-based theory of speech acts. In *Readings in Natural Language Processing*. B. J. Grosz, K. S. Jones and B. L. Webber (eds.). Morgan Kaufmann (1986) 423-440.

[36] T. Finin, R. Fritzson, D. McKay and R. McEntire - KQML as an agent communication language. Proc. of the Third International Conference on Information and Knowledge Management (1994).

[37] C. Pelachaud, N. I. Badler and M. Steedman - Generating facial expressions for speech. *Cognitive Science* 20 (1996) 1-46.

[38] P. O'Rorke and A. Ortony - Explaining Emotions. *Cognitive Science* 18, 2 (1994) 283-323.

[39] S. Kaiser and T. Wehrle - Emotion research and AI: some theoretical and technical issues. *Geneva Studies in Emotion and Communication* 8, 2 (1994).1-16.

[40] D. Rousseau, B. Hayes-Roth - A social-psychological model for synthetic actors. Knowledge Systems Laboratory of the Department of Computer Science, Report KSL 97-07. Stanford University (1997).

[41] M. A. Alberti, D. Maggiorin, P. Trapani - NARTOO: a tool based on semiotics to support the manipulation of a narrative. Proc. of Computational Semiotics for Games and New Media (COSIGN02). Augsburg (2002).

[42] J. J. Perez Alcazar, A. E. M. Ciarlini and A. L. Furtado - Cooperative interfaces based on plan recognition and generation. XXVII Conferencia Latinoamericana de Informática (CLEI) - Mérida (2001).