Motivations and declarative goals as cornerstones of autonomy

Felipe Rech Meneguzzi
frm05r@ecs.soton.ac.uk
PhD Candidate – University of Southampton

Abstract

Practical agent architectures have often forfeited true autonomy in favour of runtime efficiency by using simple triggering conditions for the activation of plans from a pre-compiled plan-library. This type of inflexible triggering rules is also used for the initiation of social behaviour, leaving little space for an agent to truly reason about the adoption of one type of behaviour over another. We propose the integration of declarative goals and motivations into an architecture as a means to create practical autonomous agents.

1 Introduction

As the demand for systems able to operate independently of human intervention increases, the concept of autonomous agents becomes more widely accepted as a suitable solution. Under the aegis of practical autonomous agent development a number of architectures have been developed. One of the most widely studied model of practical reasoning is based around the notions of beliefs, desires and intentions (BDI). Early practical implementations of BDI architectures include PRS and its descendants. These architectures have taken a particular interpretation of the BDI components aiming to maximise runtime efficiency, but sacrificing flexibility and autonomy. These architectures often base the adoption of plans on a set of reactive rules that do not account for the dynamics of the environment, nor do they provide a way for the agent to evolve through experience and favour the adoption of successful plans. Moreover, the plans chosen by the agent are often defined entirely by the designer prior to the agent being deployed, and cannot be changed by the agent at runtime. Even in non-BDI architectures it is often the case that behaviour selection occurs as a result of a set of fixed reactive rules, an approach that also applies to an agent's decision to adopt social behaviour to solve a problem. As a result, this type of agent is merely an abstraction for traditional software development, endowing the agent with no true autonomy.

We argue that truly autonomous agents must be able to adapt to a changing environment by being able to dynamically change it's goal selection strategy and the way in which goals are fulfilled. We believe that two recent research ares provide the solution to these requirements, namely: work on motivated agency provides the solution for evolving an agent's goal selection process; and work on declarative goals and plan formation to allow an agent to adapt it's goal achievement process. In this paper we try to enumerate key issues that must be investigated in order to create a sound and practical agent model from the integration of motivated agency and declarative goals. In Section 2, we provide an overview of the work in motivations and declarative goals; in Section 3 we describe the main issues arising from integrating these two technologies; finally, in Section 4 we outline future work we intend to undertake in order to address the issues of Section 3.

2 Related Work

2.1 Motivations

A psychology-inspired definition states that *a motivation represents an individual's orientation towards particular classes of goals* [6]. Such a definition, while broad, captures the fact that motivations are not necessarily tied to a specific set of goals, nor do they directly cause them to be adopted or dropped. That is to say that more than one specific motivation might be associated with a single or multiple goals, and they are not directly responsible for their adoption; rather, they create a setting in which adopting their associated goals is more likely.

The aspect of motivations most commonly sought to be captured by computational architectures is the continuous representation of priorities as a means to determine the *focus of attention* at any given time. This is important as it allows an agent with limited resources to concentrate its efforts on achieving goals that are relevant to it at specific moments, and to adapt such concentration of effort to the current reality. Contrasting with the traditional process of goal selection based solely on environmental state, real biological systems often generate different plans of action under the same environment. Hence, motivations provide a mechanism to model how internal *cues* explain goal generation in parallel with external factors [3]. An internal cue can be seen as a trigger condition that, when activated, causes an agent to consider the adoption of a set of associated goals. It differs from the simple logical preconditions discussed previously in that internal cues are a result of the dynamics of motivation strength, rather than a simple binary condition over the current state of the world.

Based on the concept of explicitly represented autonomy, research on *motivational* states to guide the reasoning process has been conducted by an increasing number of efforts. These efforts range from agent architectures specifically underpinned by motivational states [6] to the adaptation of existing architectures to cope with motivated behaviour [1].

2.2 Declarative Goals

The BDI model has been the focus of agents research for a significant time, and is still ongoing. Examples of recent research include improving the model through the construction of new theories to underpin it as a unified system [10], and extending pre-existing BDI theories to allow for more flexible BDI agents [5]. Among these efforts, many seek to address the fact that BDI architectures and models tended to avoid including many of the declarative aspects of desires/goals in support of practicality. More specifically, the first instances of complete BDI logics [9] assumed an agent able to foresee all of the future ramifications of its actions as part of the process of deciding which courses of action to take. This assumption was clearly too strong if computationally-bounded BDI architectures were to be constructed. Therefore, when designing practical architectures based on specific BDI logics, modifications were necessary to avoid unbounded computations. Since the agent cannot look directly into future world states and then select the sequence of actions that leads to the desired future (as this would imply omniscience), the inverse approach was taken; that is, an agent would select from a set of known courses of action, the one that would lead to the desired future. In practice, this means that the agent no longer selects directly what he wanted to achieve, but rather what he wants to perform under the assumption that his actions would ultimately bring about the desired state of affairs. This way of selecting agent goals was later dubbed goals *to do* [12].

Concurrently with goals *to do* are what have been termed goals *to be* [12]; the difference being that here, an agent selects the desired state of affairs directly. As a consequence, the actions required by the agent to reach such a state of affairs are decoupled from the ultimate goal. The most widely known BDI agent implementations bypass this problem through the use of plan libraries where the courses of action for every possible objective an agent might have are stored [2] (which we have seen are associated with *to do* agents). The near absence of pragmatic architectures that implement the notion of *to be* goals represents a gap that current research is trying to address.

3 Open Issues

Efforts in developing agent architectures yielded two main models of agent interpreter: procedural and declarative. Each of these two models of agent architecture is not necessarily better than the other. In some applications where predictability is more important than autonomy, a procedural approach is more desirable than a declarative one. Autonomy on the other hand is facilitated by a declarative approach. Though a large number of procedural agent architectures exist, there are few efforts towards specifying practical declarative agent architectures. The most recent efforts focus on important logical properties and semantic possibilities for mental states, but their solution to how an agent chooses a course of action tends to fall back into either a procedural approach or delegating the definition of plan libraries to the programmer. In these efforts, the declarative nature of goals is explored only to the extent of verifying goal achievement after the agent has executed a plan, and upon goal failure, deciding whether or not to try other plans in an agent's plan library.

Such an approach to the implementation of declarative goals merely provides a higher level method for organising *procedure calls* since the agent is not truly reasoning about the steps it is taking to achieve its declarative goals. This is also true for multi-agent interactions in the sense that agents are often bound by their design to participate in a joint problem-solving effort without any consideration of the reasons for doing so or the actual benefit of joining other agents rather than remaining on its own. In these situations the analysis of when to perform a particular plan or when to enter multiagent *mode* is done by the designer prior to the agent being deployed, so when such an agent is operating without supervision the reasons for its behaviour amounts more to the *dogmas* imposed by the designer than to actual autonomy. If an agent is to behave outside the boundaries of hard-coded rules it must be able to assess:

- when to forgo established sets of behaviours and construct new plans; and
- when to delegate tasks to, and when to accept tasks from, other agents.

Assessing when to use one strategy over another is not a simple task, as it involves weighing the effort required by these strategies against their perceived benefits. Attempts to model such an assessment as utility maximisation have lead to decision procedures that assume an omniscient agent [9], resulting in models that are unsuitable for practical applications. On the other hand, research on motivational states focuses on the utility that an individual agent *expects*, as opposed to absolute knowledge about utility, allowing the agent to determine a goals outcome based only on present and past world-states. Clearly, an agent designer wants an agent to satisfy its design objectives in a predictable way under ideal circumstances, but he also wants the agent to be able to fend for itself in unforeseen situations. Truly autonomous agents are able to generate their own goals, and a model of motivations can underpin the process of goal generation [6]. Using this strategy we propose *motivation* as a *control* mechanism for autonomous declarative agents.

By using a quantitative model of motivations along with a compatible representation of the cost of an agent's capabilities and resources, an agent should be able to quickly assess the reward of a given strategic choice. Moreover, if we assume that other agents within an environment operate using a similar model of motivational control, an agent should also be capable of querying its neighbours and discover their respective motivational level towards specific goals. Given such an assessment of the motivations of others, an agent should be able to decide when to delegate the achievement of certain goals to (as well as when to accept tasks delegated by) others. For such a model of control to work, a number of issues must first be addressed, these are related to:

- modelling motivations versus the cost of resources and planning;
- modelling the motivations of others;
- assessing the motivations of others;
- evaluating the reasons for interacting with others; and
- assessing when delegation should be attempted.

4 Conclusions and Future Work

If motivations are to be used to tune the operation of a planning component, then it is necessary to define which parameters of the planning process are to be affected by motivational intensity. We currently envision a model in which the intensity of a motivation will determine the amount of processing time an agent should dedicate to the planning of the goal associated to that motivation. When such processing time is consumed, the agent stops the planning process and assumes that the goal is not *worthy* of being achieved at that time. We consider this to be a weaker failure mode for a given goal, since the agent is not proving that such goal is impossible. We believe that this kind of reason for a goal to be dropped constitutes one of the reasons for an agent to attempt cooperation with other agents (that may be motivated to spend more processing power planning for the achievement of a joint goal).

Despite its recognised importance in the development of autonomous agents, architectures of declarative agents are scarce, and declarative architectures suitable for application in real domains are non-existent. Existing architectures implement only parts of what we believe to be truly declarative operation. Moreover, motivation models are mostly used in toy-examples rather than applied to fully-fledged agent architectures. Therefore, our main contribution is an investigation of how these concepts can be integrated to create a practical agent architecture. Underlying the integration of motivations with a declarative semantics are the various issues of the correlation of motivational states with planning processes and resource allocation, as well as issues regarding the assessment of potential interaction partners in a multiagent system. We believe that the analysis of these underlying issues constitutes another important contribution to be achieved by our work.

Efforts towards integrating planning algorithms to BDI agents are not new, and several systems have been developed that take advantage of the similarity between BDI mental states and the components of planning formalisms. $X^2 - BDI$ translates mental states into STRIPS problems in order to use an external planner to generate plans [4], while Walczak [11] integrates a customised planning algorithm to JADEX [8]. However, planning problems are inherently complex [7], and even the most efficient planning algorithm may take a significant amount of time to solve certain classes of problems. Agents must therefore have a way of controlling the amount of time spent planning in order to avoid losing reaction time. Though Walczak's system limits the amount of time spent in planning through a user-defined timeout value, we believe that this curtails the agent's flexibility, as an ill-defined timeout value can permanently cripple an agent's ability to deal with certain goals. In order to address this shortcoming, we intend to model a suitable set of motivations and their dynamics so as to provide an adaptable control mechanism for prioritising goals and handling failure.

References

[1] A. M. Coddington and M. Luck. A motivation-based planning and execution framework. *International Journal on Artificial Intelligence Tools.*, 10(1):5–25, 2004.

[2] M. d'Inverno and M. Luck. *Understanding Agent Systems.* Springer Series on Agent Technology. Springer Verlag, Berlin, 2nd edition, 2004.

[3] N. R. Jennings, A. G. Cohn, M. Fox, D. Long, M. Luck, D. T. Michaelides, S. Munroe, and M. J. Weal. *Cognitive Systems: Information Processing Meets Brain Science*, chapter 8. Motivation, Planning and Interaction, pages 163–188. Queen's Printer and Controller of HMSO, 2006.

[4] F. R. Meneguzzi, A. F. Zorzo, and M. da Costa Móra. Mapping mental states into propositional planning. In *Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multiagent Systems.* ACM Press, 2004. Submetido.

[5] F. R. Meneguzzi, A. F. Zorzo, and M. D. C. Móra. Propositional planning in BDI agents.

In *Proceedings of the 2004 ACM Symposium on Applied Computing*, pages 58–63, Nicosia, Cyprus, 2004. ACM Press.

[6] S. Munroe, M. Luck, and M. d'Inverno. Motivation-based selection of negotiation partners. In *AAMAS '04: Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 1520–1521, Washington, DC, USA, 2004. IEEE Computer Society.

[7] B. Nebel. On the compilability and expressive power of propositional planning formalisms. *Journal of Artificial Intelligence Research (JAIR)*, 12:271–315, 2000.

[8] A. Pokahr, L. Braubach, and W. Lamersdorf. Jadex: A bdi reasoning engine. In *Multi-Agent Programming*, pages 149–174. 2005.

[9] A. S. Rao and M. P. Georgeff. Formal models and decision procedures for multi-agent systems. Technical Report 61, Australian Artificial Intelligence Institute, 171 La Trobe Street, Melbourne, Australia, 1995. Technical Note.

[10] M. B. van Riemsdijk, M. Dastani, and J.-J. C. Meyer. Semantics of declarative goals in agent programming. In *AAMAS '05: Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*, pages 133–140, New York, NY, USA, 2005. ACM Press.

[11] A. Walczak, L. Braubach, A. Pokahr, and W. Lamersdorf. Augmenting BDI Agents with Deliberative Planning Techniques. In *The Fifth International Workshop on Programming Multiagent Systems (PROMAS-2006)*, 2006.

[12] M. Winikoff, L. Padgham, J. Harland, and J. Thangarajah. Declarative & Procedural Goals in Intelligent Agent Systems. In D. Fensel, F. Giunchiglia, D. L. McGuinness, and M.-A. Williams, editors, *Proceedings of the Eights International Conference on Principles and Knowledge Representation and Reasoning (KR-02)*, pages 470–481, Toulouse, France, April 2002. Morgan Kaufmann.