TEAMS IN MULTI-AGENT SYSTEMS

Bevan Jarvis, Dennis Jarvis, Lakhmi Jain

Bevan.Jarvis@postgrads.unisa.edu.au
Jarvis.Dennis@gmail.com
Lakhmi.Jain@unisa.edu.au
KES Group, School of Electrical and Information Engineering, University of South Australia

Abstract: Multi-agent systems involve agents interacting with each other and the environment and working to achieve individual and group goals. The achievement of group goals requires that agents work together within teams. In this paper we first introduce three philosophical approaches that result from different answers to two key questions. Secondly we consider three theoretical frameworks for modelling team behaviour. Next we look at two agent implementation models. Finally, we consider one of those implementation models – JACK Teams – and place it in the context of the philosophical debate and the theoretical frameworks.

Key words: Intelligent agents, Teams, Multi-agent systems

1. INTRODUCTION

Multi-agent systems are of research interest in philosophy, artificial intelligence and cognitive science. There are two approaches to modelling MAS behaviour – by explicit specification of individual behaviours or by relying on emergent behaviours. In the latter case, collective activity may not always be easily derivable.

Two key issues emerge. First is the question of whether teams should be explicitly modelled, as constructs constraining individual behaviour. Koestler's description of holons [12] represents a positive answer to this question. Assuming an individual-oriented approach, a second issue arises of whether individual-oriented intention suffices to explain collective

Please use the following format when citing this chapter:

Jarvis, B., Jarvis, D., Jain, L., 2006, in IFIP International Federation for Information Processing, Volume 228, Intelligent Information Processing III, eds. Z. Shi, Shimohara K., Feng D., (Boston: Springer), pp. 1–10.

intentionality – as proposed by Bratman [4] – or, as argued by Searle [15], a separate type of intention is required that is oriented towards the group and is not reducible to individual-oriented intention.

The different approaches are reflected in different theoretical frameworks for multi-agent teaming. Holonics [6] is an interpretation of Koestler's ideas. Cohen and Levesque's Joint Intention theory [5] follows Searle by defining joint intentions that are held by the team as a whole. In the SharedPlans theory of Grosz et al. [9, 10], individual-oriented and collective-oriented intention are respectively represented by means of the mental attitudes "intend to" (perform an action) and "intend that" (a proposition becomes true).

Both the Joint Intention theory and the SharedPlans theory have provided the basis for a number of successful implementations. Perhaps the most noted of these is the team-oriented programming (TOP) framework, exhibited in the TEAMCORE system of Pynadath et al. [13], which combines elements of both theories.

JACK Teams [2] is here represented as an agent-based implementation of the holonics model. The defining concept in JACK Teams may be described as providing an agent with the capability to delegate roles and to accept role obligations. An agent can thus be at once part of a greater whole (a group serving another agent) and a self-contained entity, capable of coordinating its own groups. This is essentially the definition of a holon [6].

It will be useful to consider the sources of an individual agent's intentions, which we identify as desires, obligations and norms. Desires belong to the agent. Obligations arise from an agent's agreement with another agent to perform an action or role – they are the result of delegations or contracts. Norms represent the (in human terms often tacit) agreement of agents in a group to follow certain rules. Desires are thus individual-oriented, obligations are oriented to one other individual, and norms are group-oriented.

In sections 2 and 3 we look more closely at the philosophical viewpoints of Searle and Bratman and the related theoretical frameworks developed by Cohen and Levesque and by Grosz and her collaborators. In section 4 we selectively overview some implementations of team behaviour. We conclude by discussing JACK Teams, placing it in the context of the philosophical debate and the theoretical frameworks. While having been developed separately from the philosophical and theoretical models we discuss, JACK Teams still appears to find a natural place among them.

2. THE PHILOSOPHY OF TEAMS

As mentioned above, two key issues in describing team behaviour are whether the focus of attention is on teams or individuals, and (assuming the latter) how to represent collective intention. At the philosophical level this results in three distinct approaches. The team-centric view inspired by Koestler is that teams can be both parts of larger (or at least not smaller) teams and coordinate smaller teams. From the individual-centric viewpoint, there are two approaches, represented here by Bratman and Searle. Bratman's view is that collective intention can be described by referring to individual intention in combination with other mental attitudes. Searle's opposing view is that collective intention cannot be so reduced.

2.1 Holons and the Janus Effect

Koestler [6] coined the word 'holon' to denote an entity which is both a collection of parts and a part of a greater entity. For example, a human organ is an organised collection of cells and is also a part of the human body. Holons can be part of other holons, forming hierarchies – or heterarchies – called 'holarchies'. The 'Janus effect' denotes the two-sided nature of a holon within a holarchy: facing upwards it has the form of a dependent part, while facing downwards it appears to be a self-contained whole.

Teams are holons in that they are made up of individuals and are also part of a larger organisation. Teams can also be part of other teams, and so the team structure of an organisation is, in the general case, a holarchy.

Koestler in fact does not distinguish between individuals and teams. Rather, he seeks to capture the essence of system behaviour in terms of a holarchy.

2.2 Bratman and Shared Intention

In a series of papers (collected in [4]), Bratman develops his notion of shared intention. This is intention of the group, but comprises a public, interlocking web of intentions of individuals. The interlocking web aspect reflects the fact that an individual's intentions are achieved through hierarchies of plans and subplans that must be meshed with those of other cooperating individuals. The public nature of the web of intentions is established by invoking common knowledge. (Common knowledge is the knowledge by each individual in a group of an infinite set of propositions of the form "I know that X", "I know that you know that X", "I know that you know that I know that X", and so on. A detailed study is provided by Fagin, et al. [8]. It has a close analogue in mutual belief.)

2.3 Searle and Collective Intentionality

Searle contends that in addition to individual intentionality there is collective intentionality, which latter is expressed, by each individual, as "we intend" [15]. Collective intentionality is, he states, "a biologically primitive phenomenon that cannot be reduced to or eliminated in favor of something else." Searle further claims that individual intention plus mutual belief, or any alternative to mutual belief that he has seen, does not in fact result in collective intention. This claim appears incompatible, however, with the logical requirements of the theoretical frameworks discussed below, which rely on common knowledge.

2.4 Norms and Obligations

We follow Dignam et al. [7] in distinguishing between norms and obligations. Norms are held by a group or community, and no individual is identified as the instigator. Obligations involve just two parties (individuals or groups regarded as individuals). One party instigates the obligation, which is held by the other. Using Bratman's terminology [3], both norms and obligations are pro-attitudes (similar to desires), and only in the event that they are accepted and the individual in some way commits to them do they become conduct-controlling pro-attitudes (intentions).

Norms are rules of behaviour – prescriptions or proscriptions – that are understood and enforced within a group or community. Norms include mores, taboos, faux pas and commonly agreed ways of doing things. Some norms are codified as laws. Penalties for breaking a norm range from the extreme – execution or ostracism from the community or group – to minor or none at all – shame, or the knowledge that one has caused insult or injury. Punishments may even take subtle forms such as not being invited to receive some benefit, and it is quite possible for a person not to realise that he or she has been so penalised. These punishments are sanctioned by the group, implicitly or otherwise.

Obligations are most readily explained by referring to delegation. Consider that Adam asks or requires Belinda to act in a particular way (i.e. to perform a role or a task), and Belinda, through agreement or coercion, decides (commits, and therefore feels obligated) to so act. In the case of delegation, the behaviour asked of Belinda is part of a plan coordinated by Adam. Obligations per se allow for punishment only on a limited, individual level. If, for example, Belinda does not behave as agreed, Adam has the option not to rely on her in the future. Any further action that Adam might take is subject to social norms. Obligations in which one or both parties are groups or organisations, however, often involve contracts.

Contracts are one way to enable punishment for not fulfilling obligations. They require that a third person or entity (representing the community or group) may be called upon to arbitrate and decide punishment if the agreement is broken. Contractual obligations exist in the context of a norm (codified in a set of laws) that prescribes that people in general should adhere to contracts.

3. THEORETICAL FRAMEWORKS FOR TEAMS

3.1 Holonics

In general terms, holonics is the application of Koestler's ideas to the design of multi-agent systems. The objective is to attain in designed systems the benefits that holonic organisation provides to living organisms and societies. These benefits are: stability in the face of disturbance; adaptability and flexibility to change; and efficient use of resources [6].

A noted application is Holonic Manufacturing Systems (HMS). In HMS it is desired that behaviour be explicitly specified: unpredicted emergent behaviour is generally unwelcome in a manufacturing environment.

In the holonics model, a holon has behaviours that are coordinated from above and also specifies behaviours of subsidiary holons. Additionally, it may have behaviours as an individual entity.

3.2 Joint Intentions

In [5], Cohen and Levesque establish that joint intention cannot be defined simply as individual intention with the team regarded as an individual. This is because after the initial formation of an intention, team members may diverge in their beliefs and hence in their attitudes towards the intention. Instead, Cohen and Levesque generalise their own definition of intention. First they present a definition of individual persistent goal and, in terms of this, individual intention. Both definitions use the notion of individual belief. Next, they define precise analogues of these concepts – joint persistent goal and joint intention – by invoking mutual belief in place of individual belief. The definition of joint persistent goal additionally requires each team member to commit to informing other members – to the extent of the team's mutual belief – if it comes to believe that the common goal has been achieved, becomes impossible or is no longer relevant. The result is that, while a team is not an individual, nevertheless joint intention is – at least in definition – similar to individual intention.

In Cohen and Levesque's theory, then, a team with a joint intention is a group that shares a common objective and a certain shared mental state [5]. In particular, joint intentions are held by the team as a whole.

3.3 SharedPlans

In Grosz and Sidner's SharedPlans model [10], two intentional attitudes are employed: "intending to" (do an action) and "intending that" (a proposition will hold). The former is individual-oriented intention, while the latter represents intention directed toward group activity. Additionally, shared intentions are described along with mutually known partial plans to achieve those intentions. Agents are said to have a SharedPlan to do α just in case they hold: (1) individual intentions that the group perform α; (2) mutual belief of a (partial) plan to do α; (3) beliefs about individual or group plans for the sub-acts in the plan to do α; (4) intentions that the selected agents or subgroups succeed in performing their designated sub-acts; and (5) subsidiary commitments to group decision-making aimed at completing the plan to do α.

Grosz and Hunsberger [9] claim to reconcile the two approaches to teams that we have ascribed to Bratman and Searle (to the extent of the disagreement about whether or not group-oriented intention is reducible to individual intention). They provide the "Coordinated Cultivation of SharedPlans" (CCSP) model, which, while relying solely on individual intention, captures the essential properties argued for in accounts that require group-oriented intention [9]. CCSP also provides a general architecture for collaboration-capable agents.

4. IMPLEMENTATIONS OF TEAMS IN MULTI-AGENT SYSTEMS

Two important architectures for building intelligent agents are Production Systems and the Belief-Desire-Intention (BDI) model [17].

In the Production System model, agent behaviour is coded by specifying rules that are invoked through variable binding and forward chaining.

In the BDI model, individual agents are specified which each have their own beliefs, desires, intentions (desires to which the agent has committed), and plans to carry out their intentions. In practice, commitment to intentions is handled internally to the execution engine, while desires (or goals) are implicit in the events that the agent declares it has plans to handle. Thus, coding a BDI agent consists predominantly in specifying plans and initial beliefs.

In this section we consider two implementation models of team behaviour. Team Oriented Programming (TOP) is an agent-based model that combines ideas from the Joint Intentions and SharedPlans theoretical frameworks described in the previous section. It is implemented in the Production System model. JACK Teams is also agent-based, through extending BDI, but represents an implementation of the Holonics framework.

4.1 JACK Teams

JACK Teams [2] is an extension of JACK [1], which is an implementation of the BDI model of intelligent agency. JACK itself is implemented as an extension of Java, giving it the power of a complete (and well known) computer language.

JACK Teams extends JACK by allowing the definition of agent plans in terms of roles that unspecified agents may perform, and by providing a mechanism by which roles can be matched to agents that have plans to handle them. Importantly, the delegating agent does not require the details of those plans. By way of example, Adam may ask Belinda to buy some anchovies for the pizza he wants to make, but does not need to know whether she will buy them from the local grocer or at the supermarket. All he requires (in the JACK Teams model) is for Belinda to say she has a plan for buying anchovies. Of course, Adam is free to tighten the role specification.

JACK Teams allows for belief propagation, through the notion of team belief connections. The connection is strictly one-to-one, between the coordinating agent and the agent performing a role. The flow of beliefs may be directed either upwards – synthesising the beliefs of an agent into those of the agent whose role it is performing – or downwards – allowing an agent performing a role to inherit beliefs from the coordinating agent.

JACK Teams also separates team structure (the structure needed to perform a plan) from organisational structure. In fact, it has nothing to say about the latter – although appropriate restrictions may be specified if desired.

An agent in JACK Teams is best interpreted as an extension of an ordinary agent such that it can communicate with other (similarly enhanced) agents about the roles that it requires or can fulfil. JACK Teams thus provides a mechanism whereby an individual agent can establish a group that is to some extent committed to the obligations (roles) that it prescribes. The fact that all individuals are so enhanced establishes a powerful mechanism for describing team structures.

A JACK Teams agent is clearly a holon. It performs plans at the behest of other agents, and it coordinates groups to perform its own plans. Since it as extension of an ordinary BDI agent, it also has its own private plans.

4.2 Team Oriented Programming

The Team Oriented Programming framework (TOP) is an attempt to simplify the process of building robust, flexible agent teams [13]. Each potential team member is required to have a functional interface that describes its capabilities, specifying the tasks it can perform, input and output parameters for each task, and constraints on input parameters. TOP has an explicit "team layer", a level of abstraction at which the programmer specifies: the organisational hierarchy of agents for achieving team goals; the team goals; the team procedures for achieving team goals (including initiation and termination conditions); and coordination constraints between agents executing joint activities.

The TEAMCORE [13] (and the more recent Machinetta [14]) implementation of TOP is implemented as wrappers or proxies for agents defined using the production system-based architecture Soar. TEAMCORE is an extension of STEAM [16]. Teamwork knowledge in STEAM consists of three classes of domain-independent rules: coherence preserving, monitor and repair, and selectivity-in-communication. In TEAMCORE, this domain-independent knowledge is encapsulated within wrapper agents, separating it from the possibly heterogeneous domain-level agents.

5. JACK TEAMS IN CONTEXT

We now look more closely at and seek to place JACK Teams in context with the philosophical positions and theoretical frameworks mentioned in this paper. We focus on JACK Teams because it occupies a unique philosophical position. As an extension of JACK it is based on previous work done by Bratman that provided the philosophical basis for BDI. At the same time it presents an implementation of Koestler's notion of holon.

JACK Teams was developed in response to a requirement to model team structures within organisations – specifically, military organisations. In this context the holonic approach, with its emphasis on explicit specification of behaviour, provides a mechanism for specifying standard military procedures.

The contribution of JACK Teams may be summarised by saying that it gives an agent the capability to reason about and coordinate the delegated behaviour of other agents. If a normal BDI agent represents a member of the species *homo actor* ("man the doer"), then a JACK Teams agent is *homo delegator*. From this apparently simple extension there emerges a powerful device for specifying heterarchies of delegation and obligation.

JACK Teams is firmly on the side of explicit representation of MAS behaviour. Moreover, the engine that matches roles with agents willing to perform them may be said to provide a form of mutual belief for the group, making cooperation possible. This puts it on the side of Bratman.

There is a clear distinction in JACK Teams between private intentions and intentions that are expressed by specifying roles to be filled. These respectively mirror the formulations "intend to" and "intend that" of Grosz et al. Also, there is a clear recognition of the importance of plans in mutual activity. These considerations indicate a concordance with the SharedPlans framework.

In addition, failure by an agent in a role will be detected by the coordinating agent. The latter will either handle the failure or cascade it up the delegation hierarchy (or heterarchy). It is in this manner that JACK Teams implements the communication of plan outcomes to interested parties, analogously to the communication requirement of the Joint Intentions framework.

The mechanism of belief propagation, mentioned above, provides to JACK Teams a form of mutual belief. Although this mutuality is strictly between the agent performing a role and the coordinating agent, the beliefs could be further propagated by either party.

In itself, JACK Teams does nothing towards implementing norms. However, if needed, norms could be implemented in the design of particular systems. One approach would be to implement norms as beliefs, which could be propagated throughout the group.

6. CONCLUSION AND FUTURE DIRECTIONS

In this paper we have contextualised research into the modelling of team behaviour in multi-agent systems, by considering philosophical and theoretical issues and by briefly describing two implementations.

We have also categorised JACK Teams, which, while closely connected to the holonics model, yet appears to correspond well with the work of Grosz et al., and includes an important feature of Cohen and Levesque's model.

One direction for future research would be to investigate the implementation of norms in JACK Teams. One suggested approach [11] uses explicit team contracts that specify required behaviour of each member of a task team as well as synchronisation requirements between members.

Team contracts can be said to provide agents with an expression of joint intention.

Also of interest is the problem of incomplete information. By way of example, in the operation of agent-controlled unmanned aerial vehicles (UAVs) there may be periods when communication is impossible or is deliberately not used. All the theoretical frameworks discussed assume perfect communication. It would thus become necessary to revisit this assumption.

REFERENCES

1. AOS: JACK Intelligent Agents: JACK Manual, Release 5.0, Agent Oriented Software, Pty Ltd (2005)
2. AOS: JACK Intelligent Agents: JACK Teams Manual, Release 5.0, Agent Oriented Software, Pty Ltd (2005)
3. Bratman, M. E.: Intention, Plans, and Practical Reason, Harvard University Press (1987)
4. Bratman, M.E.: Faces of Intention, Cambridge University Press (1999)
5. Cohen, P.R. and Levesque, H.J.: Teamwork, Technical Report 503, SRI International (1991)
6. Deen, S.M. (ed.): Agent-Based Manufacturing: Advances in the Holonic Approach, Springer-Verlag (2003)
7. Dignum, F., Kinny, D., and Sonenburg, L.: From Desires, Obligations and Norms to Goals, in Cognitive Science Quarterly, **2**(3-4) (2002) 407-430
8. Fagin, R., Halpern, J.Y., Moses, Y., and Vardi, M.Y.: Reasoning About Knowledge, The MIT Press (1995)
9. Grosz, B.J. and Hunsberger, L.: The Dynamics of Intention, in Collaborative Activity, Conference on Collective Intentionality IV, Siena (2004)
10. Grosz, B.J. and Sidner, C.L.: Plans for Discourse, in Cohen, P.R., Morgan, J. and Pollack, M.E, eds, Intentions in Communication, The MIT Press (1990)
11. Jarvis, D., Jarvis, J., Rönnquist, R. and Fletcher, M.: Provision of Robust Behaviour in Teams of UAVs – A Conceptual Model, paper to be presented at the 1st SEAS DTC Technical Conference, Edinburgh (2006)
12. Koestler, A.: The Ghost in the Machine, Arkana (1967)
13. Pynadath, D.V., Tambe, M., Chauvat, N. and Cavedon, L.: Toward Team-Oriented Programming, in Intelligent Agents VI: Agent Theories, Architectures and Languages, Springer-Verlag (1999)
14. Scerri, P., Pynadath, D.V., Schurr, N., Farinelli, A., Gandhe, S. and Tambe, M.: Team Oriented Programming and Proxy Agents, in Proceedings of the 1st International Workshop on Programming Multiagent Systems (2004)
15. Searle, J.R.: The Construction of Social Reality, Free Press (1997)
16. Tambe, M.:Towards flexible teamwork, in Journal of Artificial Intelligence Research, **7** (1997) 83-124
17. Wooldridge, M.: Reasoning About Rational Agents, The MIT Press (2000)