Robots are Agents, Too!

Gal A. Kaminka
Bar Ilan University
Israel
galk@macs.biu.ac.il

Recent years are seeing a worrisome decline in the activity of roboticists within the Autonomous Agents and Multi-Agent Systems (AAMAS) community. Robotics papers, that were abundant in the early Autonomous Agents conferences, have almost disappeared in recent editions of the International Joint Conference on Autonomous Agents and Multi-Agent Systems. To some, this trend seems natural. After all, what do physical robots, with their assortment of hardware issues, mechanical design challenges, and sensory uncertainty, have to do with web crawlers, interaction protocols, auctions, and game-theoretical results or logical proofs? What do they have to do with web services, and agent-oriented software engineering?

Yet, others – and I am among them – still argue that robots are agents, not only by definition, but also by use. In other words, treating robots as agents brings to the robotics world important useful results from the Agents world. The AAMAS community can have significant impact on the world of robotics, and vice versa; but to do so, it must be inclusive of robots.

I will first justify the claim that "robots are agents too" (i.e., that it is useful to think of robots as agents), and then draw several concrete implications. I offer two arguments in support of the claim that robots are agents. The first is that robots are agents because, at the appropriate level of abstraction, robots meet most common definitions of agents. Moreover, many challenges in developing robots (at this conceptual level) are also common to autonomous agents. The second argument is based on a case study. It examines a successful area of research within multi-agent systems – that of Teamwork – that is beginning to have an impact in robotics.

Let us begin with the conceptual argument. When robots are abstracted to their information processing components, they share characteristics of agents in general: they have sensors through which they perceive the world, they have actuators through which they act on their environment, they are situated in an environment, and they have a process that links their sensors and actuators, ideally in some goal-oriented fashion that maintains reactivity. This description has been applied to AI systems for many years, even while the field has fragmented into many sub-communities, such as vision, sensor fusion, planning, learning, etc. Yet, the Agents community has revived an important challenge hidden within that description, a challenge that was to some degree overlooked by pre-Agents AI. This is the challenge of integrating all these different components (sensing, acting, thinking, learning, etc.) into a single functioning system.

> Where once programmers had to hand-design coordination mechanisms for each possible interaction, automated teamwork models are now being used to make decisions that are automated, consistent, and reduce development time.

Indeed, a significant appeal to roboticists in the AAMAS community is that we treat integration of capabilities – the architecture of the agent – as an important scientific problem to be addressed. Many roboticists share this view, if only because the design and deployment of a physical robot necessarily requires integration of physical and computational components. A robot cannot just see, it must also act on what it sees, and it must do so in a way that fits within its physical and computational restrictions. Similarly, agents may communicate or otherwise perceive their environment (e.g., by accessing web-pages), but unless they can act on their perceptions (e.g., offer a recommendation, send a message with a specific content), they cannot truly be called agents. Roboticists deal with integration challenges on a daily basis, much like AAMAS researchers do. Their solutions take different things into consideration, but ultimately a scientific understanding of the integration problem will impact both communities.

The other argument in support of the "robots are agents" point of view is that techniques developed for autonomous agents and multi-agent systems are slowly but surely impacting work in robotics. I examine a specific case here, that of *teamwork* (i.e., collaboration).

Teamwork has been investigated within the multi-agent systems community for many years. Cohen and Levesque [1], and Grosz, Sidner, and Kraus [2,3] have published a series of articles on teamwork, developing logical models (*Joint Intentions Framework*, *SharedPlans*) to model and prescribe teamwork. Among other issues, these models describe the conditions under which an agent must inform its teammates of its own private beliefs, thus effectively maintaining synchronisation in the team as to specific propositions. The SharedPlans teamwork model also specifies conditions for proactive assistance to teammates, mutual support, etc.

The key message of these efforts has been that at least a subset of multi-agent interactions, collectively called teamwork, could be prescribed in a task-independent way. The key benefit of this approach is that generic (task- and domain-independent) algorithms and techniques could be built which would ensure that agents acted as team members, (almost) regardless of the task in which they were engaged. Unfortunately, in general, I think it is safe to say that roboticists took little notice of these theoretical frameworks, as groundbreaking as they were.

However, several autonomous agent researchers picked up on these logical frameworks, and begun investigations of how the frameworks might be applied in practice. Motivated by a seemingly endless stream of coordination failures in a distributed industrial system, Jennings [4] built on the Joint Intentions Framework to propose the Joint Responsibility Model, implemented in the GRATE* system. The system used this model to automate coordination messages between agents within the distributed system, thus reducing the number of coordination failures. A short while later, Tambe [5] significantly extended the techniques involved, allowing his system (called STEAM) to consider the cost of communications in its decisions. STEAM also provided mechanisms for recovery from failures, extended organisational hierarchies, etc. These were evaluated empirically, in a high-fidelity virtual environment, in which synthetic helicopter pilots used the system to automate their coordination decisions. This brought teamwork models close enough to robotics to get some attention from that community, and even more so when STEAM was shown to be applicable, with minor changes, to the domain of virtual RoboCup soccer.

The benefits of automated explicit teamwork have been developing steadily. Where once programmers had to hand-design coordination mechanisms for each possible interaction, automated teamwork models are now being used to make decisions that are automated, consistent, and reduce development time. These are concrete contributions that roboticists, in principle, should be very happy to adopt. At the end of the 1990s, the state of the art in robotic teamwork research did not differentiate teams that collaborate towards shared goals, from loosely-coordinating groups of essentially autistic robots. Relatively few roboticists (e.g., Parker [6]) have recognised that robot group interactions are an independent and specific object of study. Most others, instead, focused on investigating various mechanisms through which a group of robots would appear to act as a team, or in a coordinated manner, despite the robots not having any explicit notion of their teamwork (or even of other robots).

One possible reason for the apparent lag in taking ideas from agents into the robotics world is that robot limitations are different from, and possibly severer than, those of agents described above. Robotics involves challenges for teamwork which are not sufficiently addressed in previous work within the AAMAS community. For instance, many techniques in multi-agent teamwork assume that sensors are reliable: if information X is sensed by an agent, than X is indeed true. However, most robotics work must avoid such an assumption, since physical sensors suffer from inherent uncertainty, which can be mitigated, but not eliminated. Thus, teamwork techniques must be adapted for them to be useful to robotics.

One example of such an attempt is the BITE (Bar Ilan Teamwork Engine) system [7], which is a behaviour-based control architecture that incorporates a teamwork model (see Figure 1). Its closest direct ancestor is STEAM, but BITE contains various features that target robots in particular, such as:

- emphasised dependence on a world-model, in which sensor readings are pre-processed to address uncertainty;
- capabilities for human-robot interaction (specifically, human operation and command), which are of vital important to multi-robot team applications; and
- flexible interaction protocols, to match the conditions of the environment.

What we have here is an Agents success story. Arguably, the Agents community has produced significant insights into the nature of collaboration. It is my conjecture that the fact that, around that time, robotics researchers attended joint meetings (Autonomous Agents conferences, RoboCup Symposia, ICMAS conferences) led, to some extent, to the recognition of such insights by roboticists. Indeed, additional multi-agent systems techniques (e.g., the use of auctions to allocate tasks [8]) have begun to make visible contributions within work in robotics.

To continue this impact on the robotics world, we must be inclusive of robotics work within our community – indeed, we must counter the trend of recent years. For instance, it would make sense to insist on including roboticists in organising and programme committees of Agents conferences. Another important step to take would be to form links with the EURON network of excellence, which serves, for roboticists, a similar role to that of AgentLink. We need to evangelise both within and outside of AAMAS, to send a clear message: robots are agents too!

References

[1] Cohen, P. R. and Levesque H. J. 1991. Teamwork, Nous 35.

[2] Grosz, B. J. and Sidner, C. L. 1990. Plans for discourse. Intentions in Communications, Cohen, P. R., Morgan, J. & Pollack, M. (Eds.), 417-445.

[3] Grosz, B. J. and Kraus, S. 1996. Collaborative Plans for Complex Group Actions, Artificial Intelligence, 86, 269-358.

[4] Jennings, N. R. 1995. Controlling cooperative problem solving in industrial multi-agent systems using joint intentions. Artificial Intelligence, 75(2) 195-240.

[5] Tambe, M. 1997. Towards flexible teamwork. Journal of Artificial Intelligence Research, 7, 83-124.

[6] Parker, L. E. 1998. ALLIANCE: An architecture for fault-tolerant multirobot cooperation. IEEE Transactions on Robotics and Automation 14(2) 220-240.

[7] Kaminka, G. A., Elmaliach, Y., Frenkel, I., Glick, R., Kalech, M. and Shpigelman, T. 2004. Towards a Comprehensive Framework for Teamwork in Behavior-Based Robots. Proceedings of the Eighth Conference on Intelligent Autonomous Systems (IAS-8).

[8] Dias, M. B., Zlot, R., Zinck, M., Gonzalez, J. P., Stentz, A. 2004. A Versatile Implementation of the TraderBots Approach for Multirobot Coordination. Proceedings of the Eighth Conference on Intelligent Autonomous Systems (IAS-8).

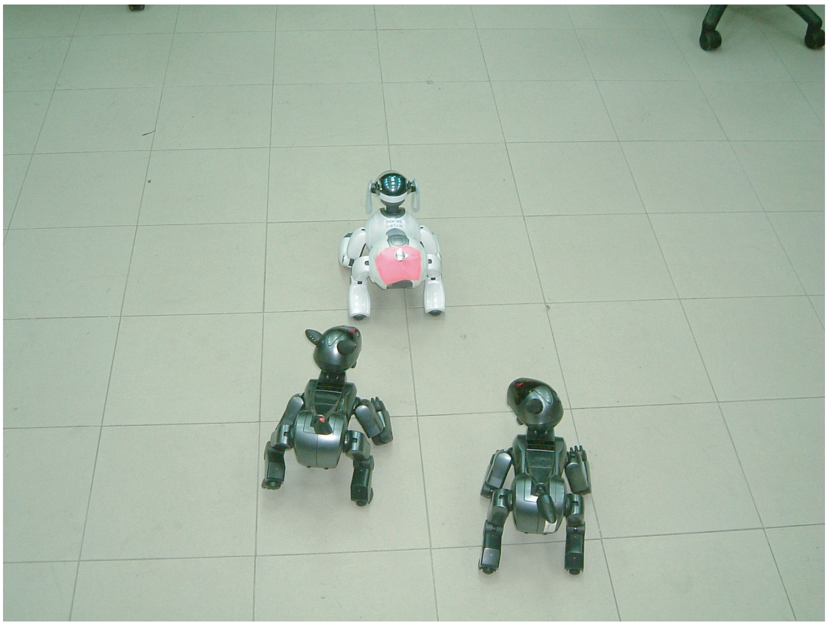

Figure 1: Sony AIBO Robots moving in formation at Bar Ilan University.
The robots are executing BITE, a teamwork architecture which automates their teamwork.