AAAI Technical Report SS-12-02
Designing Intelligent Robots: Reintegrating AI

Autonomous Agents Research in Robotics: A Report from the Trenches

Gal A. Kaminka
MAVERICK Research Group
Computer Science Department and Gonda Brain Science Center
Bar Ilan University, Israel*

Abstract

This paper surveys research in robotics in the AAMAS (Autonomous Agents and Multi-Agent Systems) community. It argues that the autonomous agents community can, and has, impact on robotics. Moreover, it argues that agents researchers should proactively seek to impact the robotics community, to prevent independent re-discovery of known results, and to benefit autonomous agents science. To support these claims, I provide evidence from my own research into multi-robot teams, and from others'. This paper is a short version of a more detailed survey (Kaminka 2012).

Rise of the Machines

Today, there is a resurgent interest and recognition of the importance of robotics research framed within areas of research familiar to autonomous agents and multi-agent systems researchers. The AAMAS community is investing efforts to encourage robotics research within itself. An annual robotics special-track, an associated robotics workshop (*ARMS: Autonomous Robots and Multirobot Systems*), and a series of exciting AAMAS-sponsored plenary speakers and awards over a number of years are drawing roboticists in. There are fruitful interactions with the other communities within AAMAS, such as virtual agents , game theory, and machine learning. Robots are being used both to inspire AAMAS research as well as to conduct it.

I posit that this growing success is due not only to the nurturing efforts of the AAMAS community, but mainly to the increasing recognition of an important, deeper, truth: *Robots are Agents*. In other words, it is *scientifically useful* to think of robots as agents.

Research in agents raises challenges in integrated capabilities for intelligence, such as planning and execution, learning exploration and exploitation, strategic decision making in multi-agent settings, and more. It requires us to consider deeply and critically what we mean by calling a system "autonomous". It raises challenges in the software architectures needed for such capabilities.

Many roboticists share these challenges. Robotics, by nature of the research, requires its practitioners to evaluate not only a component, but also its use within the system. Moreover, many roboticists are increasingly setting their goals higher than what we in AI (sometimes arrogantly) refer to as "low level control". The stability of platforms have made it possible for roboticists to examine complex tasks, in which there is need for intelligence and knowledge, and for considering multiple robots. Agenticists have a wide variety of tools and techniques which can be brought to bear in facing both single- and multiple- robot challenges.

And similarly, agenticists increasingly realize that it is useful for them to think of robots as agent exemplars. To agents researchers, working with real robots (made of plastic, metal, electronics, and the sweat of graduate students) brings out important challenges to our current theory and practice. Robots make us fail in interesting ways, and give opportunity for gaining insights otherwise unattainable. They extend the system perspective to go beyond the conceptual perception and actuation to consider sensors and motors (with their uncertainties, faults, and latencies), imperfect communications, and multiple bodies (each with its two- or three- dimensional geometry). Roboticists know much about these challenges, and can greatly influence intellectual development within agents.

To support my argument, I report from the trenches of ongoing robotics work within the AAMAS community, highlighting success stories in which robotics research benefited from AAMAS research, and vice versa. I therefore admit in advance to a bias towards work appearing in AAMAS conferences and journals. This bias is intended to highlight robotics in the context of research areas appearing in AAMAS. However, the unfortunate result of this bias is that ground-breaking work in AI and robotics appearing elsewhere (e.g., Thrun, Burgard and Fox's game-changing work on probabilistic robotics (Thrun, Burgard, and Fox 2005)) will not receive proper treatment here. This, despite such work being excellent evidence for the generality of my argument as to the usefulness of AI to robotics, and vice versa.

Building Architectures

Robotics research today must address increasingly complex missions that the robots should carry out. Previously the problem of controlling a robot could be addressed by a carefully designed controllers. Relatively basic (and *very* useful) tasks such as finding a path, avoiding obstacles, and navi-

*Currently at the Radcliffe Institute of Advanced Study, Harvard University.

gating to a goal location, are now mature areas of research. Instead, robots are now expected to go beyond reaching a goal, to carrying out missions in which there are multiple (changing) goals, and multiple tasks, that should be carried out (sometimes concurrently, sometimes in sequence). Such complex missions require planning, managing resources, and in particular making decisions.

Well, this is something that AI researchers know about. Beginning with Gat's *ATLANTIS* architecture (Gat 1992), AI researchers have begun to integrate planners into the robot's control architecture. Following ATLANTIS three-tier design, a standard approach was to integrate a high-level planner together with controllers, mediated by an executive module whose role is to issue plan-requests, schedule and monitor the controller's successful execution. The key idea in such hierarchical layering is that *tasks are planning problems*, to be solved by a planner operating in discrete atomic steps. An executive module works at a different, finer, resolution to carry out the task. Guided by the plan, it translates its discrete steps into controller instantiations.

Around the same time, agents researchers have begun to advocate the idea that agents are not just planners (Pollack 1990). Rather, agents should reason about plans: generating them, adapting them, and contrasting them, to make decisions about carrying them out in service of various goals. Agents researchers developing robots with integrated capabilities have focused on integrating planning and execution (sometimes also with learning) in a way that reflects such reasoning about plans. Here, execution and planning work at the same temporal and task resolution. The approach, called *plan-based control* by Beetz et al. (Beetz et al. 2001), relies on utilizing a plan representation as a central, first-class object, which is reasoned about, generated, adapted, revised, and managed through the life-time of the robot. Planners are used not only to generate plans (and re-plan), but also to provide predictions (including of resource use and execution time). Separate processes estimate the state of the robot and the world, address perception (symbol grounding, sensor fusion, etc.), and make decisions as to alternative courses of action. Independently, roboticists have begun to consider similar notions, building on the hierarchical layering of planning and execution modules, in a way that allowed learning or considering which plans to execute, ((Haigh and Veloso 1997; Simmons et al. 1997; Beetz 2001; Thrun et al. 2000).

Thus the need to reconsider the design of the agent architecture was lead both by theorists, as well as robotics researchers working in the context of agents research. The challenge of how to integrate different capabilities was met, within the agents community, with an already-existing body of knowledge, and significant fascination with general agent architectures: with how, *in general*, agents should be built. Over the years, research into agent architectures that work across a wide variety of agent types and environments (including robots in various applications) has resulted in greater understanding of the architecture components and their operation. Some specific areas of research—still continuing today—are discussed below.

Beliefs, Desires, Intentions and Other Mental Attitudes First, it is by now understood that an agent operating in a dynamic environment (the settings for many robot applications) must manage the planning process. It must decide when to plan and when to avoid re-planning (as it is computationally infeasible to re-plan with every change). To do this, the construction of the agents must allow for explicit representation of beliefs, goals, and plans (whether pre-planned or dynamically generated). These will be revised, manipulated, contrasted, and reasoned about by the agents' action selection and perception processes. In other words, beliefs, goals, and plans are all first-class objects.

To a large degree, the huge literature on mental attitudes of agents, and in particular on BDI (Belief, Desire, Intention) theories and architectures (Rao and Georgeff 1995; Sardiña and Padgham 2011) is a response to this challenge. Recent years are seeing, side by side, developments in both the theory and practice of plan representations that are amenable to both planning and execution. A variety of academic and commercial BDI implementations exists, such as PRS (Lee et al. 1994), RPL (Beetz 2001), RAPs (Earl and Firby 1997), and CogniTAO (CogniTeam, Ltd. 2009)).

First-Class Plan Representations. In addition to these BDI languages that have been used in robots, there have been of course many plan representations (and sometimes programming languages) that have been tried and tested in robots, but that offer first-class status only to the plan, rather than also beliefs and goals. Nevertheless, they are useful in constructing robots that employ plan-based control. These include finite-state representations (Tousignant, Wyk, and Gini 2011; Lötzsch, Risler, and Jüngel 2006), Petri-Net representations (Ziparo et al. 2010), and temporal planning and scheduling languages (e.g., T-REX (Py, Rajan, and McGann 2010), which allows for multiple-resolution scheduling of tasks).

No single plan representation has emerged thus far as a clear de-facto standard, and in fact the comparison of these representations remains an open challenge. Many of the BDI languages have been developed to address reported failings in finite-state machine representations (such as their lack of a factored state and limited reactivity), but a clear theoretical contrast is still lacking.

Teams of Robots

Perhaps the area in which agents research has had the most impact on robotics research is in multi-robot systems. This is due to the principled domain-independent handling of the combinatorial complexity of multi-robot tasks. If multiple robots are to be coordinated in some fashion, the task of making decisions for them is more difficult than that of making decisions for a single robot, since in addition to the individual decisions, one must worry about the combinations of selected actions.

Most multi-robot research to date, within the robotics community, focuses on a single task at a time. Some examples of such canonical tasks include *formation maintenance* (Balch and Arkin 1998; Fredslund and Mataric 2002;

Inalhan, Busse, and How 2000; Kaminka, Schechter-Glick, and Sadov 2008), *multi-robot coverage* (Rekleitis et al. 2004; Zheng et al. 2005; Rekleitis, Dudek, and Milios 2001; Batalin and Sukhatme 2002; Butler, Rizzi, and Hollis 2000; Agmon, Hazon, and Kaminka 2008), foraging (Goldberg and Matarić 2001; Rybski et al. 1998; Rosenfeld et al. 2008; Zuluaga and Vaughan 2005; Schneider-Fontan and Matarić 1996; Jager and Nebel 2002; Ostergaard, Sukhatme, and Matarić 2001; Kaminka, Erusalimchik, and Kraus 2010), and patrolling or surveillance (Elmaliach, Shiloni, and Kaminka 2008; Agmon et al. 2008; Jensen et al. 2011; Basilico, Gatti, and Amigoni 2009; Smith, Schwager, and Rus 2011; Agmon, Urieli, and Stone 2011; Marino et al. 2009; Delle Fave et al. 2009). Many of these are approached from the perspective of a distributed control problem. In other words, a controller is devised such that when it is operating in each individual robot, the total sum behavior is as required. Such controllers are built a new for each task. But as future robot applications grow in complexity, such controllers would need to take into account allocating and scheduling the execution of multiple tasks, taking place concurrently or in sequence. For instance, urban search and rescue (Murphy et al. 2008) applications require elements of both coverage and foraging, and introduces additional novel tasks.

A key insight gained in the AAMAS field in the last 15 years is that in fact, multi-agent tasks can be decomposed—conceptually, as well as technically—into two components. The first, called *taskwork*, includes domain-dependent individual capabilities. The second, called *teamwork* in teams, and *socialwork* in general, includes the capabilities for collaboration (in teams), or maintaining other social relations. This *socialwork* component includes social choice mechanisms. For instance, protocols for allocating tasks to different team-members (e.g., by bidding), or protocols for reaching joint decisions (e.g., by voting). The combination of taskwork and socialwork creates a working multi-agent system for a given domain.

This insight has manifested itself in several different ways in robotics research. I will briefly discuss some of these areas of cross-fertilization between agents and robotics research, and then dive in detail into one specific area (teamwork).

Market-Based Task Allocation. In terms of impact on robotics, the use of market-based methods for allocating tasks to robots enjoys widespread popularity. It is now being adopted and investigated by roboticists outside of the AAMAS community, certainly a positive sign. Starting with Dias and Stentz's work (Dias and Stentz 2000) on the use of market-mechanisms for coordinating robots in exploration and mapping tasks, there has been much work in t his area, addressing challenges that are raised when working with robots (see, for example, (Lin and Zheng 2005; Gerkey and Mataric 2002; Zlot and Stentz 2006; Lagoudakis et al. 2004; Vig and Adams 2006; Köse et al. 2003; Michael et al. 2008; Tang and Parker 2007; Lagoudakis et al. 2005; Bererton, Gordon, and Thrun 2003)). Dias et al. (Dias et al. 2006) provides a comprehensive survey, and Xu et al. (Xu et al. 2006) provides a comparison with other methods

Reaching Joint Decisions in Teamwork. More generally, AAMAS researchers have long discovered that teamwork involves more than task allocation. It also involves agreement on a common goal, agreement on a plan to reach the common goal, assisting teammates as necessary, etc.

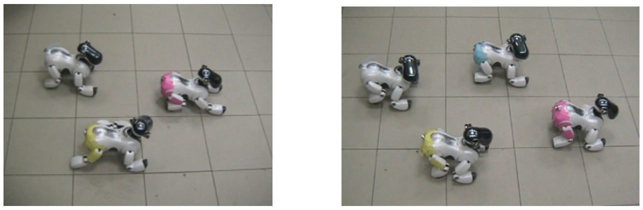
Teamwork has been investigated within the multi-agent systems community for many years (Grosz and Kraus 1996; Cohen and Levesque 1991) have published a series of articles on teamwork, using logic to model and prescribe teamwork. Among other issues, these models describe the conditions under which an agent must inform its teammates of its own private beliefs, thus effectively maintaining synchronization in the team as to specific propositions.

The key benefit of this approach is that much of such teamwork can be algorithmitized. It can be described by a set of behavioral rules, which, if followed, would cause the agent to act appropriately in the context of a team, regardless of the task it was assigned, or the application domain.

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. One of the unique features of the AAMAS conference is that it is a rare forum in which both researchers of virtual humans (virtual agents), and roboticists can meet to exchange ideas. The demonstration of automated teamwork in software agents brought teamwork models close enough to robotics to get some attention from that community.

To illustrate the contribution of teamwork—as understood in state-of-the-art AAMAS—to robotics, I will describe my groups' utilization of teamwork software as part of an technology-transfer project, intended to implement a canonical multi-robot task—formation maintenance—familiar to many roboticists. Given the space constraints, I settle here for a relatively high-level description; details are in (Traub 2011).

In formation maintenance, robots must move in unison along a given path, while maintaining a given geometric shape. Various formation maintenance methods have been investigated (e.g., (Balch and Arkin 1998; Desai 2002; Fredslund and Mataric 2002; Balch and Hybinette 2000; Desai, Ostrowski, and Kumar 2001; Carpin and Parker 2002; Inalhan, Busse, and How 2000; Tabuada, Pappas, and Lima 2005; Kaminka, Schechter-Glick, and Sadov 2008; Elmaliach and Kaminka 2008)). All of these schemes are distributed; all require each robot to run a local control process, which executes the controller that fits the role of the robot. For instance, a left-following robot in a equilateral triangle formation would keep the leader in a fixed distance (matching the distance kept by the right-following robot), such that the leader robot is at bearing 30 degrees to the right. A right-following robot would do the same, but its controller would maintain the leader at a bearing of 330 degrees (i.e., 30 de-



(a) Triangular AIBO formation, in (Elmaliach and Kaminka 2008; Kaminka and Frenkel 2005).

(b) Diamond AIBO formation, in (Kaminka, Schechter-Glick, and Sadov 2008).

(c) Column Shrimps-III formation, in (Traub 2011).

(d) Triangular Shrimps-III formation, in (Traub 2011).

Figure 1: Robots moving in formation.

grees to the left). Figure 1 shows a number of formations, the basis for the work in (Kaminka and Frenkel 2005; 2007; Elmaliach and Kaminka 2008; Kaminka, Schechter-Glick, and Sadov 2008; Traub 2011).

The various control schemes differ in the type of operating conditions they assume, as well as in the type of performance they provide. For instance, some control schemes (called SBC for *Separation-Bearing Control*) require each follower robot to be able to identify the distance and angle to a leader robot in the formation (Fredslund and Mataric 2002), based on sensor readings. In contrast, communication-based formation maintenance can be used to eliminate the need for perception, by relying on dead-reckoning and communications from the leader robots (Elmaliach and Kaminka 2008). Others still use more robust schemes that allow robots to switch which robots are to be followed (Desai, Ostrowski, and Kumar 2001).

The goal of the project was to create a robust controller, by tying these different control schemes together, switching between them as necessary. This creates a formation-maintenance scheme that is robust to intermittent perception and communications failures, as long as they do not coincide. The key is to switch between the different schemes, based on availability of the perception and communication processes.

Now suppose we adopt a standard robotics approach to this problem. This would entail writing a switching controller which switches between the different modes. Each such switching controller would operate on a different robot, and thus we immediately face a challenge: We need to make sure that when one robot switches, the others do as well (since mixing up formation maintenance schemes is not, in general, likely to work well). This means that we need to add code that manages communications between robots, so that when one robot finds it necessary to switch, it automatically lets the other ones know, and awaits confirmation of their switching, too. Of course, the conditions underwhich a robot may want to switch are not necessary those that another robot senses, and so we also need code for them to negotiate and agree as to which control scheme the team should use. Now we just need to get all of this working for more than two robots, and more than two schemes, and across potential communication errors. And all of this still not taking into account issues such as changing roles in the formations, etc. *Just a simple matter of programming*, as the expression goes.

Agent researchers have long recognized that the challenges above are general. Teamwork architectures offer a general solution to cases where agents must decide on (1) when to communicate (and to some degree, what to communicate about), (2) how to come to a joint agreement (in this case, which mode mode should be used by the robots), and (3) how to allocate (and reallocate) tasks to different robots. The needs for such decisions comes up again and again.

Thus my research group used a teamwork architecture to manage the joint switching of controllers, and allocation of roles and tasks. We utilized the CogniTAO (CogniTeam, Ltd. 2009) commercial teamwork architecture to integrate together robust SBC (Kaminka, Schechter-Glick, and Sadov 2008) and communication-based formation-maintenance controllers (Elmaliach and Kaminka 2008). The details of the integration are well beyond the scope of this article, but the lessons are not.

Traub (Traub 2011) has carried out an analysis of the benefits of using a teamwork architecture, by using a standard software engineering model (CoCoMo) (Boehm 1981)) to measure its impact in automating the coordination processes described above, contrasting it with conservative and optimistic estimates of the size of the project given a standard robotics approach. The results show 50% to 68% savings in programming effort within the project, which of course translate into significant savings in both development time and number of programmers. These numbers are compatible with earlier reported results (Tambe 1997) (in fact, they are more conservative). Teamwork in robots is a success story for AAMAS research, with measurable effects.

A Call to Arms

My argument in this article is that AAMAS has a lot to offer robotics, and also a lot to benefit from robotics; that as a field of science goes, working with robots is a useful endeavor. To show this, I have reported from the trenches of on-going work in two specific ares of contribution, where past contributions and on-going work are showing significant promise, both in robotics as well as in AAMAS. But there's quite a bit more; I've described the tip of the iceberg, hoping to convince you, the reader, to look at the iceberg underneath. There's a lot going on: In swarms, traffic control, surveillance, learning, and much more.

Indeed, this article is also intended to be a call to arms, to invest in robot-based research. The drop in robot prices and consequent rise of the machines make robot-based artificial intelligence research in general, and AAMAS research in particular, both compelling and practical. One no longer needs to have an in-house mechanical and electronics shop to successfully conduct research involving robots. Stable platforms are now cheaply available, and their commercial maintenance makes maintaining a robot lab a feasible effort. As for venues for meeting other like-minded researchers, the appropriately named, ARMS (Autonomous Robots and Multirobot Systems) workshop works hand-in-hand with the AAMAS conference to promote robotics research within the AAMAS community. This is a good place to start, even if your paper is yet not quite up to AAMAS rigorous standards. Similar workshops take place on occasion with other AI conferences, including AAAI.

There is a great opportunity for AI and Agent researchers to begin exploring essentially philosophical ideas in real-world robots. The potential impact that we can have on the scientific community, and on the exploding commercial world of robotics, is huge. The alternative—letting roboticists re-discover and re-invent decades of our work—is unforgivable.

References

Agmon, N.; Sadov, V.; Kaminka, G. A.; and Kraus, S. 2008. The impact of adversarial knowledge on adversarial planning in perimeter patrol. In *Proceedings of the Seventh International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-08)*, volume 1, 55–62.

Agmon, N.; Hazon, N.; and Kaminka, G. A. 2008. The giving tree: Constructing trees for efficient offline and online multi-robot coverage. *Annals of Math and Artificial Intelligence* 52(2–4):143–168.

Agmon, N.; Urieli, D.; and Stone, P. 2011. Multiagent patrol generalized to complex environmental conditions. In *Proceedings of the Twenty-Fifth Conference on Artificial Intelligence*.

Balch, T., and Arkin, R. 1998. Behavior-based formation control for multi-robot teams. *IEEE Transactions on Robotics and Automation* 14(6):926–939.

Balch, T., and Hybinette, M. 2000. Social potentials for scalable multirobot formations. In *Proceedings of IEEE International Conference on robotics and automation (ICRA-00)*.

Basilico, N.; Gatti, N.; and Amigoni, F. 2009. Leader-follower strategies for robotic patrolling in environments with arbitrary topologies. In *Proceedings of the Eighth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-09)*, 57–64.

Batalin, M., and Sukhatme, G. 2002. Spreading out: A local approach to multi-robot coverage. In *Proc. of the 6th Internat. Symposium on Distributed Autonomous Robotic Systems*, 373382.

Beetz, M.; Arbuckle, T.; Bennewitz, M.; Burgard, W.; Cremers, A.; Fox, D.; Grosskreutz, H.; Hähnel, D.; and Schulz, D. 2001. Integrated plan-based control of autonomous service robots in human environments. *IEEE Intelligent Systems* 16(5):56–65.

Beetz, M. 2001. Structured Reactive Controllers. *Journal of Autonomous Agents and Multi-Agent Systems. Special Issue: Best Papers of the International Conference on Autonomous Agents '99* 4:25–55.

Bererton, C. A.; Gordon, G. J.; and Thrun, S. 2003. Auction mechanism design for multi-robot coordination. In *Advances in Neural Information Processing Systems 16*.

Boehm, B. W. 1981. *Software Engineering Economics*. Upper Saddle River, NJ, USA: Prentice Hall PTR, 1st edition.

Butler, Z.; Rizzi, A.; and Hollis, R. L. 2000. Complete distributed coverage of rectilinear environments. In *Workshop on the Algorithmic Foundations of Robotics*.

Carpin, S., and Parker, L. 2002. Cooperative leader following in a distributed multi-robot system. In *Proceedings of the IEEE International Conference on Robotics and Automation*.

CogniTeam, Ltd. 2009. CogniTAO (Think As One) software development kit and run-time kernel.

Cohen, P. R., and Levesque, H. J. 1991. Teamwork. *Nous* 35.

Delle Fave, F.; Canu, S.; Iocchi, L.; Nardi, D.; and Ziparo, V. 2009. Multi-objective multi-robot surveillance. In *Proc. of 4th International Conference on Autonomous Robots and Agents*, 68–73.

Desai, J. P.; Ostrowski, J. P.; and Kumar, V. 2001. Modeling and control of formations of nonholonomic mobile robots. *IEEE Transactions on Robotics and Automation* 17(6):905–908.

Desai, J. P. 2002. A graph theoretic approach for modeling mobile robot team formations. *Journal of Robotic Systems* 19(11):511–525.

Dias, M. B., and Stentz, A. T. 2000. A free market architecture for distributed control of a multirobot system. In *6th International Conference on Intelligent Autonomous Systems (IAS-6)*, 115–122.

Dias, M. B.; Zlot, R.; Kalra, N.; and Stentz, A. 2006. Market-based multirobot coordination: A survey and analysis. *Proceedings of the IEEE* 94(7):1257–1270.

Earl, C., and Firby, R. J. 1997. Combined execution and monitoring for control of autonomous agents. In *Proceedings of the First International Conference on Autonomous Agents (Agents-97)*, 88–95.

Elmaliach, Y., and Kaminka, G. A. 2008. Robust multi-robot formations under human supervision and control. *Journal of Physical Agents* 2(1):31–52.

Elmaliach, Y.; Shiloni, A.; and Kaminka, G. A. 2008. A realistic model of frequency-based multi-robot fence patrolling. In *Proceedings of the Seventh International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-08)*, volume 1, 63–70.

Fredslund, J., and Mataric, M. J. 2002. A general algorithm for robot formations using local sensing and minimal communications. *IEEE Transactions on Robotics and Automation* 18(5):837–846.

Gat, E. 1992. Integrating planning and reacting in a heterogeneous asynchronous architecture for controlling real-world mobile robots. In *Proceedings of the Tenth National*

Conference on Artificial Intelligence (AAAI-92). Menlo Park, Calif.: AAAI press.

Gerkey, B. P., and Mataric, M. J. 2002. Sold!: Auction methods for multi-robot coordination. *IEEE Transactions on Robotics and Automation* 18(5):758–768.

Goldberg, D., and Matarić, M. 2001. Design and evaluation of robust behavior-based controllers for distributed multi-robot collection tasks. In *Robot Teams: From Diversity to Polymorphism*, 315–344.

Grosz, B. J., and Kraus, S. 1996. Collaborative plans for complex group actions. *Artificial Intelligence* 86:269–358.

Haigh, K. Z., and Veloso, M. M. 1997. High-level planning and low-level execution: Towards a complete robotic agent. In *Proceedings of the First International Conference on Autonomous Agents (Agents-97)*, 363–370.

Inalhan, G.; Busse, F.; and How, J. 2000. Precise formation flying control of multiple spacecraft using carrier-phase differential GPS. In *Proceedings of AAS/AIAA Space Flight Mechanics*.

Jager, M., and Nebel, B. 2002. Dynamic decentralized area partitioning for cooperating cleaning robots. In *ICRA 2002*, 3577–3582.

Jensen, E.; Franklin, M.; Lahr, S.; and Gini, M. 2011. Sustainable multi-robot patrol of an open polyline. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA-11)*, 4792–4797.

Kaminka, G. A., and Frenkel, I. 2005. Flexible teamwork in behavior-based robots. In *Proceedings of the Twentieth National Conference on Artificial Intelligence (AAAI-05)*.

Kaminka, G. A., and Frenkel, I. 2007. Integration of coordination mechanisms in the BITE multi-robot architecture. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA-07)*.

Kaminka, G. A.; Erusalimchik, D.; and Kraus, S. 2010. Adaptive multi-robot coordination: A game-theoretic perspective. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA-10)*.

Kaminka, G. A.; Schechter-Glick, R.; and Sadov, V. 2008. Using sensor morphology for multi-robot formations. *IEEE Transactions on Robotics* 271–282.

Kaminka, G. A. 2012. I have a robot, and I'm not afraid to use it! *AI Magazine*. In press.

Köse, H.; Meriçli, c.; Kaplan, K.; and Akin, H. 2003. All bids for one and one does for all: Market-driven multi-agent collaboration in robot soccer domain. In Yazici, A., and Sener, C., eds., *Computer and Information Sciences (ISCIS 2003)*, volume 2869 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg. 529–536.

Lagoudakis, M. G.; Berhault, M.; Koenig, S.; Keskinocak, P.; and Kleywegt, A. J. 2004. Simple auctions with performance guarantees for multi-robot task allocation. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS-04)*, volume 1, 698–705.

Lagoudakis, M. G.; Markakis, E.; Kempe, D.; Keskinocak, P.; Kleywegt, A. J.; Koenig, S.; Tovey, C. A.; Meyerson, A.; and Jain, S. 2005. Auction-based multi-robot routing. In *Robotics: Science and Systems*, 343–350. The MIT Press.

Lee, J.; Huber, M. J.; Kenny, P. G.; and Durfee, E. H. 1994. UM-PRS: An implementation of the procedural reasoning system for multirobot applications. In *Proceedings of the Conference on Intelligent Robotics in Field, Factory, Service, and Space (CIRFFSS-94)*, 842–849.

Lin, L., and Zheng, Z. 2005. Combinatorial bids based multi-robot task allocation method. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA-05)*, 1145–1150.

Lötzsch, M.; Risler, M.; and Jüngel, M. 2006. XABSL - A pragmatic approach to behavior engineering. In *Proceedings of IEEE/RSJ International Conference of Intelligent Robots and Systems (IROS)*, 5124–5129.

Marino, A.; Parker, L. E.; Antonelli, G.; and Caccavale, F. 2009. Behavioral control for multi-robot perimeter patrol: A finite state automata approach. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA-09)*.

Michael, N.; Zavlanos, M. M.; Kumar, V.; and Pappas, G. J. 2008. Distributed multi-robot task assignment and formation control. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA-08)*, 128–133.

Murphy, R. R.; Tadokoro, S.; Nardi, D.; Jacoff, A.; Fiorini, P.; Choset, H.; and Erkmen, A. M. 2008. Search and rescue robotics. In Siciliano, B., and Oussama, K., eds., *Springer Handbook of Robotics*. Springer Verlag. 1151–1173.

Ostergaard, E.; Sukhatme, G.; and Matarić, M. 2001. Emergent bucket brigading. In *Proceedings of the Fifth International Conference on Autonomous Agents (Agents-01)*, 29–30.

Pollack, M. E. 1990. Plans as complex mental attitudes. In Cohen, P. R.; Morgan, J.; and Pollack, M. E., eds., *Intentions in Communication*. MIT Press.

Py, F.; Rajan, K.; and McGann, C. 2010. A systematic agent framework for situated autonomous systems. In *Proceedings of the Ninth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-10)*.

Rao, A. S., and Georgeff, M. P. 1995. Bdi agents: From theory to practice. In *Proceedings of the First International Conference on Multiagent Systems (ICMAS-95)*, 312–319.

Rekleitis, I.; Lee-Shue, V.; New, A. P.; and Choset, H. 2004. Limited communication, multi-robot team based coverage. In *IEEE International Conference on Robotics and Automation*, 3462–3468.

Rekleitis, I.; Dudek, G.; and Milios, E. 2001. Multi-robot collaboration for robust exploration. *Annals of Mathematics and Artificial Intelligence* 31:7–40.

Rosenfeld, A.; Kaminka, G. A.; Kraus, S.; and Shehory, O. 2008. A study of mechanisms for improving robotic group performance. *Artificial Intelligence* 172(6–7):633–655.

Rybski, P.; Larson, A.; Lindahl, M.; and Gini, M. 1998. Performance evaluation of multiple robots in a search and retrieval task. In *Proc. of the Workshop on Artificial Intelligence and Manufacturing*, 153–160.

Sardiña, S., and Padgham, L. 2011. A bdi agent programming language with failure handling, declarative goals, and planning. *Journal of Autonomous Agents and Multi-Agent Systems* 23(1):18–70.

Schneider-Fontan, M., and Matarić, M. 1996. A study of territoriality: The role of critical mass in adaptive task division. In Maes, P.; Matarić, M.; Meyer, J.-A.; Pollack, J.; and Wilson, S., eds., *From Animals to Animats IV*, 553–561. MIT Press.

Simmons, R. G.; Goodwin, R.; Haigh, K. Z.; Koenig, S.; and O'Sullivan, J. 1997. A layered architecture for office delivery robots. In *Proceedings of the First International Conference on Autonomous Agents (Agents-97)*, 245–252.

Smith, S. L.; Schwager, M.; and Rus, D. 2011. Persistent monitoring of changing environments using a robot with limited range sensing. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA-11)*, 5448–5455.

Tabuada, P.; Pappas, G. J.; and Lima, P. U. 2005. Motion feasibility of multi-agent formations. *IEEE Transactions on Robotics and Automation* 21(3):387–392.

Tambe, M. 1997. Towards flexible teamwork. *Journal of Artificial Intelligence Research* 7:83–124.

Tang, F., and Parker, L. E. 2007. A complete methodology for generating multi-robot task solutions using asymtred and market-based task allocation. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA-07)*, 3351–3358.

Thrun, S.; Beetz, M.; Bennewitz, M.; Burgard, W.; Cremers, A. B.; Dellaert, F.; Fox, D.; Hähnel, D.; Rosenberg, C. R.; Roy, N.; Schulte, J.; and Schulz, D. 2000. Probabilistic algorithms and the interactive museum tour-guide robot minerva. *International Journal of Robotics Research* 19(11):972–999.

Thrun, S.; Burgard, W.; and Fox, D. 2005. *Probabilistic Robotics*. MIT Press.

Tousignant, S.; Wyk, E. V.; and Gini, M. 2011. An overview of xrobots: A hierarchical state machine based language. In *Proceedings of the ICRA-2011 Workshop on Software development and Integration in Robotics*.

Traub, M. 2011. Topics in multi-robot teamwork. Master's thesis, Bar Ilan University. Available at http://www.cs.biu.ac.il/˜galk/Publications/b2hd-meytal-msc.html.

Vig, L., and Adams, J. A. 2006. Market-based multi-robot coalition formation. In Gini, M., and Voyles, R., eds., *Distributed Autonomous Robotic Systems 7*. Springer Japan. 227–236.

Xu, Y.; Scerri, P.; Sycara, K. P.; and Lewis, M. 2006. Comparing market and token-based coordination. In *Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-06)*, 1113–1115.

Zheng, X.; Jain, S.; Koenig, S.; and Kempe, D. 2005. Multi-robot forest coverage. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.

Ziparo, V. A.; Iocchi, L.; Lima, P. U.; Nardi, D.; and Palamara, P. F. 2010. Petri net plans. *Autonomous Agents and Multi-Agent Systems* 23(3):344–383.

Zlot, R., and Stentz, A. 2006. Market-based multirobot coordination for complex tasks. *The International Journal of Robotics Research* 25(1):73–101.

Zuluaga, M., and Vaughan, R. 2005. Reducing spatial interference in robot teams by local-investment aggression. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*.