Agents are Dead. Long live Agents!

Blue Sky Ideas Track

Virginia Dignum, Frank Dignum
UmeåUniversity
UmeSweden
{virginia.dignum,frank.dignum}@umu.se

ABSTRACT

In recent years, the future of agent research has often been discussed. Most prominent is the issue whether agents should be seen as a conceptual framework or as a software development paradigm. At the same time, developments on AI seem to have taken the field into a new direction. In this paper we argue that in order for agents research to create added value for actual, real problems in the world we need to reconsider possible agent architectures and their strengths and weaknesses, their overlaps and commonalities. Finally we present a first sketch of an architecture for such agents.

KEYWORDS

AAMAS; multi-agent systems; software agents; AI; social AI

ACM Reference Format:
Virginia Dignum, Frank Dignum. 2020. Agents are Dead. Long live Agents!. In *Proc. of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020), B. An, N. Yorke-Smith, A. El Fallah Seghrouchni, G. Sukthankar (eds.), Auckland, New Zealand, May 2020,* IFAAMAS, 5 pages.

1 INTRODUCTION

Over the years the emphasis in the research has shifted from a focus on single, intelligent agents towards multi-agent systems. Given the distribution of papers at AAMAS it seems that a theoretical view on agent research is prevalent. This is exemplified by the attention on game theory as a way to guide the interactions between agents. At the same time the research on agent communication languages totally disappeared. It seems that the agent coordination itself is so abstract and takes place in contexts where the actual implementation of the communication no longer warrants a separate agent communication language. Can we conclude that agent research provides us with a more abstract foundation of distributed autonomous systems? If that is the case, it would be good to reconsider what characteristics these autonomous systems should have. Are they BDI (or BDI-like) systems? Do they cooperate or compete? Is the system as a whole supposed to solve one problem or is it meant to model cooperating organizations in a dynamic environment? In the latter case the agents will have different knowledge and capabilities and possibly incomplete knowledge about the other agents.

One might argue that different people can give different answers to the above questions and that is ok, because a general, abstract framework can be instantiated in many different ways. However, it becomes problematic if there seems to be no single conceptual framework that forms the basis for all the different instantiations.

Proc. of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020), B. An, N. Yorke-Smith, A. El Fallah Seghrouchni, G. Sukthankar (eds.), , May 2020, Auckland, New Zealand.

In this paper we propose some possible directions to ensure added value for actual, real problems in the world. Firstly, we need to reconsider agent architectures and their strengths and weaknesses, their overlaps and commonalities. This would give a more solid basis for developing agents for different domains and applications using a uniform background and thus strengthen the usefulness of the paradigm. Next to a reconsideration of agent architectures there is also a need to determine what the theoretical contributions of the agent reseach should be. Are they a bunch of very abstract mechanisms (such as deliberation cycles, interaction strategies, etc.) that can be assembled and implemented as one sees fit? Or should we have more restrictions on what "real" agents are? Third, we need to decide whether techniques can be developed that are usable by others to quickly develop agents and solve problems. This latter part drives e.g. the current success of deep learning. One can easily download neural networks or deep learning software packages and start using the software very quick to solve actual problems. We are not saying that all problems are solved using deep learning, but the availability of easy to use software is of added value.

In the next section we will discuss a bit more in depth why AAMAS has failed on its promises. In section 3, we argue that this failure does certainly not mean we have to give up, but rather provides many opportunities for young researchers to lead the way. In section 4, we give a sketch of some solution paths given that we develop agents as part of the general research of artificial intelligence. We make some concluding remarks in section 5.

2 AGENTS' FAILED PROMISES?

On its home page, the IFAAMAS organisation states that "*The AAMAS conference series was initiated in 2002 as a merger of three highly respected individual conferences: AGENTS (International Conference on Autonomous Agents), ICMAS (International Conference on Multi-Agent Systems), and ATAL (International Workshop on Agent Theories, Architectures, and Languages). The aim of the joint conference is to provide a single, high-profile, internationally renowned forum for research in the theory and practice of autonomous agents and multiagent systems*[1]." For those, like us, that have attended most of the editions since 2002, it is becoming increasingly clear that research across and combining the three original areas has never really taken off. Having worked for more than 25 years in the agent research, we have seen a number of very promising application areas where agents intuitively should provide an added value. However, in each of these areas (e-commerce, web-services, serious games, social simulations,...) agents have not delivered on their promise. Moreover, the last years have shown a decrease of activity in agent technology (programming languages, design

[1] http://www.ifaamas.org/

methodologies, development frameworks) and on the development and application of software agents themselves. Whereas the concept of agent is generally accepted outside the community, very few real software agent applications are actually available, as we have described in the previous section.

There are two main causes of this failure. The first is that there was (and is) no readily available agent platform that can be used by anyone outside the agent community. The existing agent programming languages (e.g. 2APL [3], JACK [16], AgentSpeak [1], GOAL [12], and more) are loosely based on the concept of BDI agents, but (rightfully so) adapted to make them computational feasible. A number of these languages and platforms are now reasonably mature, and often have a solid theoretical foundation that provide operational semantics for BDI concepts in terms of their implementation [14]. However, in many situations the programs needed to solve the problems in the real world are very convoluted, because many actions and plans of the agents are easier to describe in a procedural way rather than using some type of planning. One can argue that therefore the agent programming languages are inadequate. However, any person that tries to implement the conceptual agent model needs to make some decisions on what to support, what to allow and what to prohibit. It seems, however, that the conceptual model of BDI agents only works well for very specific types of problems. Moreover, very little work has been done to develop methodologies and platforms that enable to specify agents that can join an existing MAS in an open environment, thus enabling extensions and reuse of agent systems [10].

The second issue in trying to apply agent research in practice is the failure to capture or to incorporate essential features of the problem for which the agents are used. For example, it seemed that agents would be perfectly suitable to implement characters in video games (see e.g. [5, 15]). It would give the characters goals and thus consistent behaviour over time, it should provide handles to balance reactive and pro-active behavior and it should support the natural interaction between characters and between characters and the player. However, in practice none of these issues were properly supported by any of the standard agent theories or platforms. Agents appear to be a primarily cognitive concept and thus steering actual bodies in a virtual environment was not supported at all. It is clear that moving a body around should not be done based on existing BDI deliberation cycles. The BDI deliberation should be used for the more tactical or strategic reasoning. But how does that reasoning relate to the operational level? What are the consequences for the BDI deliberation? No answers exist in the literature, neither seems there to be a community that is actively searching for answers.

2.1 Rationality is not enough

The traditional definition of agents as autonomous, proactive and interactive entities where (a) each agent has bounded (incomplete) resources to solve a given problem, (b) there is no global system control, (c) data is decentralized, and (d) computation is asynchronous. That is, rationality is often a central assumption for agent deliberation [9]. Individual agents are typically characterized as bounded rational, acting towards their own perceived interests. The main advantages of a rationality assumption are their parsimony and applicability to a very broad range of situations and environments, and their ability to generate falsifiable, and sometimes empirically confirmed, hypotheses about actions in these environments. This gives conventional rational choice approaches a combination of generality and predictive power not found in other approaches.

Unfortunately, this type of rational behavior fits mostly with strategic choices, where information is all available or can be gathered at will. It does not really suit most human behavior which is based on split second decisions, on habits, on social conventions and power structures. If the aim of MAS is to develop models of societal behaviour or to develop systems that are able to interact with people in social settings, rationality is not enough to model human behaviour. This was exemplified before by all the application areas, where the rational behavior needs to be combined with different types of behavior in order to be effective. In reality, human behaviour is neither simple nor rational, but derives from a complex mix of mental, physical, emotional and social aspects. Realistic applications must moreover consider situations in which not all alternatives, consequences, and event probabilities can be foreseen. Thus it is impossible to "rationally" optimize utility, as the utility function is not completely known, neither are the optimization criteria known. This renders rational choice approaches unable to accurately model and predict a wide range of human behaviours. Already in 2010, [10] shows how different types of variations and models cater for different applications, while no generic model exists that serves as a foundation for all models.

2.2 Open environments, heterogeneous agents

Multi-agent systems are often seen as ideal ways to implement interactions and ensure coordination in open environments, inhabited by heterogenous entities with different aims and capabilities. However, very little results have ever been achieved to ensure that any given software agent is able to determine how, when and why to join such a system [7].

Open systems assume that the heterogeneous agents are designed and run independently from each other and use their own motivations to determine whether to join an existing institution or another interaction space. However, in most cases, all agents are designed from scratch so that their behavior complies with the behavior expected by the multi-agent system. Comprehensive solutions for open environments would require complex agents that are able to reason about their own objectives and desires and thus decide and negotiate their participation in an organization. Work in this area has never developed further than proofs of concept.

2.3 MAS or ABMS?

Besides divisions within the AAMAS community, which we have described in section 2.1 and 2.2, there are actually two main approaches in agent research and application: Multi-Agent Systems (MAS) and Agent-Based Modelling and Simulation (ABMS). These approaches differ considerably in methodology, applications and aims. MAS focuses on solving specific complex problems using independent agents, while ABM is used to capture the dynamics of a (social or technical) system for analytical purposes. ABM, on the other hand, is a form of computational modelling whereby a population of individual agents are given simple rules to govern their behaviour such that global properties of the whole can be

analysed [11]. The terminology of ABM tends to be used more often in the social sciences, whereas MAS is more used in engineering and technology.

Although there is considerable overlap between the two approaches, a multiagent system is not always the same as an agent-based model. Historically, the differences between ABM and MAS are often more salient than their similarities. For example, it is often remarked that a main difference between ABM and MAS is that the goal of ABM is to search for explanatory insight into the collective behaviour of agents at the macro level, whereas MAS focuses on solving specific practical or engineering problems, emphasizing complex agent architectures with sophisticated reasoning and decision processes [9]. This has lead to the development of two research communities proceeding on nearly independent tracks. When considering how the two approaches are applied, ABM models are *descriptive* systems, used as analysis tools to support the understanding whereas MAS are often meant to be *operational* systems, acting and affecting its (physical) environment.

3 AI NEEDS AGENTS

Nowadays when people outside computer science talk about AI, they actually mean machine learning and in specific deep learning techniques. Of course this is understandable given the huge successes of deep learning (and it also causes some envy from all of us that have been less successful in practical problems). However, rather than trying to jump on the bandwagon of machine learning it is a good time to reflect where the strengths of agent technology lay and why it is needed more than ever to fullfil the promise of AI. A particular success of current machine learning techniques is classification. I.e. problems where, given a certain amount of input, a decision on an output parameter has to be given. E.g. which object is visible in a picture, which diagnosis is most likely given symptoms of a patient, etc.

However, most problems that we consider to require intelligence are not of this simple input-output type. E.g. should an 87 year old person with heart problems start a chemo therapy fighting bone cancer? The therapy might prolong life with a full year, but also will decrease the quality of life such that the person might not be able to visit family or receive family most of the time. In order to give an answer to this question the system should interact with the person in order to find out which are the preferences and fundamental values that are most important according to which an optimum decision might be reached. It becomes even more difficult if in this process also consideration has to be taken with a partner of the person. Suppose the partner really wants to use every means possible to keep the person alive, while the person itself thinks it is time to leave a wonderful and long life.

In these problems there often is not a lot of data available and thus it will be impossible to use pattern matching to get the right decision given the myriad of possible attitudes of the persons involved in the problem. It is these kinds of problems where agents seem to provide a better framework to model the problem and solution process. The solution inevitably needs interaction between different parties. The interaction is not just an exchange of information, but also information discovery and preference formation based on preferences of other agents.

Moreover, recently many organisations and governments have put forward principles and guidelines to ensure that AI development is human-centred, responsible, trustworthy, and/or ethical. Even though quite different, such guidelines converge into agreement that new approaches are needed to deal with the social interaction between AI applications and humans. Principles such as human oversight, diversity, fairness, accountability or adaptability to cultural, social and contextual differences, seem to point to models and architectures that are fundamentally aligned with classic agent paradigms. The European guidelines for Trustworthy AI, explicitly call for AI architectures extending "sense-plan-act" cycles to integrate: "*(i) at the "sense"-step, the system should be developed such that it recognises all environmental elements necessary to ensure adherence to the requirements; (ii) at the "plan"-step, the system should only consider plans that adhere to the requirements; (iii) at the "act"-step, the system's actions should be restricted to behaviours that realise the requirements.*" All these are aspects core to the agent paradigm, and therefore an important role lays here for the AAMAS community.

A core challenge for trustworthy AI, concerns the shaping of AI ecosystems comprising autonomous and collaborative, assistive technology in ways that express shared moral values and ethical and legal principles as expressed in e.g. binding codes such as universal human rights and national regulations. This requires the understanding, developing, and evaluating AI applications through the lense of an artificial autonomous system that interacts with others in a given environment. One of the most straightforward approaches to ensure a system behaves "rightly", or in a trustworthy manner, is through regulating its behaviour. This is an area in which the AAMAS community has a proven track record. Work on norms and institutions in multiagent systems can be used to prove or verify that specific rules of behaviour are observed when making decisions, while ensuring that individual agents' rights are taken into consideration [4]. However, while research on norms and normative system is central to AAMAS, the focus has mostly been on the general rules of interaction that coordinates agents' actions. It is important to be able to extend this line of research to understand and model the ethical dilemmas that arise from the need to combine multiple norms, preferences and interpretations, from different agents, cultures, and situations.

Finally, the vision of human-centred AI, requires that AI systems are social. In previous papers ([8, 13]) it is already argued that agents should become more aware of the social context in which they operate. This awareness is not included in the standard BDI model of agents, which is directed on the agents own goals. So, what is needed are agent models and architectures that explicitly embrace the social aspects of AI.

4 DISTRIBUTED SOCIAL AGENTS

In recent years, several researchers in both ABM and MAS, [8, 13, 15], recognise the need for new models of deliberation that show how behaviour derives from both personal drives such as identities, emotions, motives, and personal values as well as from social sources such as social practices, norms, organizations [6].

Socially realistic agent models require to bring together formalization and computational efficiency, and planning techniques, and expertise on empirical validation and on adapting and integrating

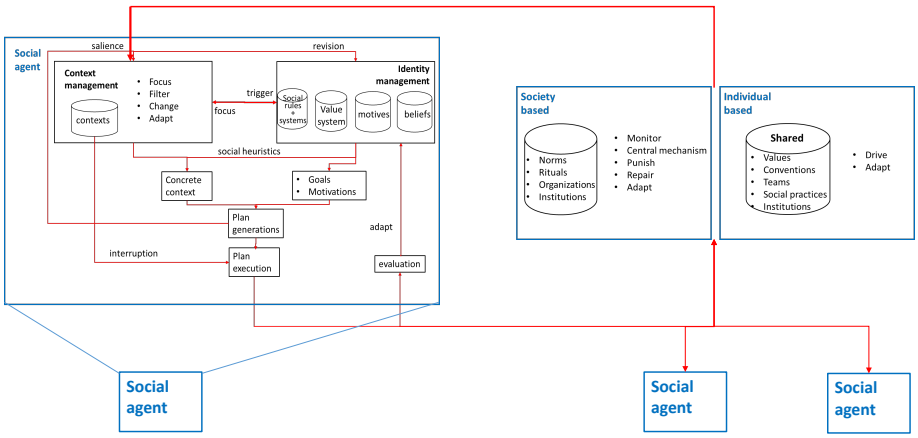

Figure 1: Sketch of a Social System Architecture

social sciences theories into a unified set of assumptions [2]. Main characteristics of sociality-based reasoning are [9]:

- Ability to hold and deal with inconsistent beliefs for the sake of coherence with identity and cultural background.
- Ability to combine innate, designed, preferences with behaviour learned from observation of interactions. In fact, preferences are not only a cause for action but also a result of action, and can change significantly over time.
- Capability to combine reasoning and learning based on perceived situation. Action decisions are not only geared to the optimization of own wealth, but often motivated by altruism, justice, or by an attempt to prevent regret at a later stage.
- Pragmatic, context-based, reasoning capabilities. Often there is no need to further maximize once utility gets beyond some reasonably achievable threshold.
- Ability to pursue seemingly incompatible goals concurrently, e.g. a simultaneous aim for comfort and sustainability.

Our claim is that these types of agents should be based on new architectures that are not primarily goal or utility driven, but are instead situation or (social) context based.

In the architecture sketched in Figure 1 a first step into the direction of these social agents is given. The context management of the agent filters the (social) context to lead to standard behaviour appropriate for that context. Whenever the context is uncertain, not recognized or not standard a second process of deliberation is started based on the motives and values of the agent and the current concrete goals. After the performance of each behaviour there is a feedback loop that is used to adapt all the elements of the agent based on the rate of success or failure of the behaviour in that particular context. However, there is also an input to the context management from the internal drives of the agent. I.e. the agent will actively search for a context to satisfy some of its needs if it can. E.g. if one feels lonely then one will actively search for a situation in which one meets with friends and/or family. Thus context management is not just passively filtering the environment, but also directing focus on parts of a context or seeking it to get the right context. Sociality-based agents are fundamental to the new generations of intelligent devices, and interactive characters in smart environments. These agents need to be fundamentally pro-active, reactive and adaptive to their social context, because basically the social context with people is not a static given situation, but is actively created and maintained based on mutual satisfaction of motives, values and needs. Thus the agents not only must build (partial) social models about the humans they interact with, but also need to take social roles in a mixed human/digital reality and start co-creating the social reality in which they operate. More work is needed to test and validate social agent architectures such as the exemplary one suggested in Figure 1.

5 CONCLUSIONS

Although the agent community seems to deliver many results, none of these seems able to provide the highly needed contribution to ensure that AI applications are truly human-centred. One of the main issues is the lack of a common basic model that could underpin the different developments and can be used as a start of pragmatic techniques that can be used by software engineers to build applications for complex domains where agent technology provides real added value. We claim that in order to provide the characteristics that are needed for human centred AI we need to start with a social agent architecture like presented in this paper. Crucial in this architecture is the situatedness of the agent and the bidirectional nature of the pro-active and reactive deliberation of the agent that allows it to co-create the context in which it operates. Of course the architecture is very rich and thus not suited for every application. An important direction of research is to investigate modular implementations that allow to use only relevant parts of this architecture.

Acknowledgements. This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

REFERENCES

[1] Rafael H Bordini, Jomi Fred Hübner, and Michael Wooldridge. 2007. *Programming multi-agent systems in AgentSpeak using Jason*. Vol. 8. John Wiley & Sons.

[2] S. Chai. 2001. *Choosing an Identity: A General Model of Preference and Belief Formation*. University of Michigan Press.

[3] Mehdi Dastani. 2008. 2APL: a practical agent programming language. *Autonomous agents and multi-agent systems* 16, 3 (2008), 214–248.

[4] Louise Dennis, Michael Fisher, Marija Slavkovik, and Matt Webster. 2016. Formal verification of ethical choices in autonomous systems. *Robotics and Autonomous Systems* 77 (2016), 1–14.

[5] F. Dignum, J. Bradshaw, B. Silverman, and W. van Doesburg (eds.). 2010. *Agents for Games and Simulations: Trends in Techniques, Concepts and Design*. LNAI 5920, Springer Verlag.

[6] F. Dignum, V. Dignum, R. Prada, and C.M. Jonker. 2015. A conceptual architecture for social deliberation in multi-agent organizations. *Multiagent and Grid Systems* 11, 3 (2015), 147–166.

[7] Frank Dignum, Virginia Dignum, John Thangarajah, Lin Padgham, and Michael Winikoff. 2007. Open agent systems???. In *International Workshop on Agent-Oriented Software Engineering*. Springer, 73–87.

[8] F. Dignum, R. Prada, and G.J. Hofstede. 2014. From autistic to social agents. In *AAMAS 2014*.

[9] Virginia Dignum. 2017. Social Agents: Bridging Simulation and Engineering. *Commun. ACM* 60, 11 (2017).

[10] Virginia Dignum and Frank Dignum. 2010. Designing agent systems: state of the practice. *IJAOSE* 4, 3 (2010), 224–243.

[11] J.M. Epstein and R. Axtell. 1996. *Growing Artificial Societies: Social Science from the Bottom Up*. The Brookings Institution, Washington, DC, USA.

[12] Koen V Hindriks. 2009. Programming rational agents in GOAL. In *Multi-agent programming*. Springer, 119–157.

[13] G. Kaminka. 2013. Curing Robot Autism: A Challenge. In *AAMAS 2013*. 801–804.

[14] Brian Logan. 2015. A future for agent programming. In *International Workshop on Engineering Multi-Agent Systems*. Springer, 3–17.

[15] B. Silverman, D. Pietrocola, B. Nye, N. Weyer, O. Osin, D. Johnson, and R. Weaver. 2012. Rich socio-cognitive agents for immersive training environments: case of NonKin Village. *Journal of Autonomous Agents and Multi-Agent Systems* 24, 2 (March 2012), 312–343.

[16] Michael Winikoff. 2005. JACK™ intelligent agents: an industrial strength platform. In *Multi-Agent Programming*. Springer, 175–193.