Engineering Multiagent Organizations through Accountability

Stefano Tedeschi[0000-0002-9861-390X]

Università degli Studi di Torino - Dipartimento di Informatica, Torino, Italy
stefano.tedeschi@unito.it

Abstract. In my PhD I have been investigating the notions of *accountability* and *responsibility* in multiagent organizations. The main objective of the work is to develop both a conceptual model and a programming framework encompassing the two concepts as engineering tools, which, in our view, are fundamental for the realization of robust systems.

Keywords: Accountability · Responsibility · Multiagent Organizations.

1 Introduction

Multiagent systems (MAS) proved to be effective in the development of complex systems composed of heterogeneous actors operating in distributed environments. *Normative agent organizations* offer abstractions defining strategies for decomposing complex goals into simpler ones and for allocating them to agents. Key features of most organizational models, e.g. [6], are a functional decomposition of the goal and a normative system. Norms establish what agents should perform to achieve the organizational goal. However, none of the approaches addresses *accountability*, i.e. who should report to whom for the fulfillment of its duties and how, resulting in the following limits: (i) difficulty to identify who should give restitution to whom for a certain state of the organization; (ii) difficulty to take appropriate countermeasures in case of abnormal situations.

On this foundation, the aim of my PhD has been to investigate the notions of *accountability* and *responsibility* as engineering tools to systematize the development of MAS organizations. We claim that the realization of distributed systems would benefit from an explicit representation of accountability and responsibility, two fundamental concepts at the basis of human organizations. The overall objective is to develop both a formal conceptual model and a programming framework to guide the design and engineering of robust MAS organizations.

2 Related work

Agent organizations [7] embody patterns of interaction imposed to agents to ensure a coherent global behavior. Normative organizations provide the means to realize the *correct* behavior, capturing what agents should do and which sanction is applied if they do not comply. One drawback is that, when the system faces

an *abnormal situation* and some agent *fails*, sanctions are of little utility. What is missing is some support for agents to provide accounts on what happened, propagating such feedback through appropriate channels, and reaching those agents equipped for coping with them, making the system *robust*. We claim that accountability and responsibility serve the purpose intuitively, yet effectively.

In [8], accountability is "a primary characteristic of governance where there is a sense of agreement and certainty about the legitimacy of expectations." Along this line, accountability implies that some actors have the right to hold other actors to a set of standards [10]. Notably, [11] points out that the lack of an adequate representation of relationships between actors obfuscates accountability, possibly compromising the system functioning. Concerning responsibility, in the context of information systems, [9] represents it as a unique charge assigned to an agent. It is worth noting that accountability and responsibility are not primitive concepts. Rather, they are properties that emerge in carefully designed systems.

3 Proposal

We propose to explicitly introduce the notions of accountability and responsibility as software engineering tools for use in MAS organizations. By this, I mean the realization via software of the abilities to trace, evaluate, and communicate accountability and responsibility. The problem will be addressed (i) by supplying a formal model and definition of computational accountability, clarifying its relation with the sibling notion of responsibility; (ii) by providing modeling and programming tools to simplify the realization of accountability supporting organizations. The purpose is to come up with a formalization of the concepts as first-class entities to be used both by the designer to describe the expected behavior of the system and by the agents to direct their conduct.

The first part of my project has been focused on the development of a methodology and framework for the design of such organizations. The construction of a comprehensive system requires many elements: a formal model of accountability and responsibility, an engine to distribute responsibilities, an automated forum to discern the accountability of the involved agents, and a mechanism to keep track of who could be accountable for what in which situation. The second part will be devoted to the development of an actual programming platform implementing the previously mentioned framework in some of the main platforms for MAS organizations. One platform that seems particularly promising is JaCaMo [6]. Such a choice is due to the fact that it provides a very good integration of the concepts characterizing agents, environments and organizations.

4 Preliminary results

The support to accountability and responsibility has found a first realization in the ADOPT protocol for creating and manipulating accountability relationships [3]. The main intuition is that, when an agent participates in an organization,

it must accept a set of accountability requirements, expressed as social commitments. The protocol specifies the shapes of these commitments and controls their creation. In [4], an information model that describes which data should be available to identify accountabilities in a group of interacting agents has been proposed. The model, provided in Object-Role Modeling, identifies the main concepts in the process of accountability determination: *mutually held expectation* and *control*. Central lies the accountability relationship, constrained so that a principal accountable for an achievement must be in control of it and that there must be a mutually held expectation on that principal for the achievement. In [1], we proposed to improve the specification of an organization with a set of accountability and responsibility specifications. $\mathsf{R}(x, q)$, denoting a responsibility, expresses an expectation on any agent playing role x on pursuing condition q. $\mathsf{A}(x, y, r, u)$ expresses that x, the a-giver, is accountable towards y, the a-taker, for condition u when condition r holds. Accountability relationships are collected in an *accountability specification*, while responsibilities are grouped in a *responsibility distribution*. The designer will specify a set of acceptable accountability specifications and responsibility distributions. Central is the notion of *accountability fitting* that links together accountability and responsibility. We, then, studied how robustness can be achieved via accountability and responsibility. In [2] we presented two programming patterns for developing agents according to the accountability fitting. The proposal allows to map the accountability/responsibility specification into a set of well-defined agent plans. Such plans define the behavior agents should exhibit to produce accounts for the goals they are responsible for, directed to the agents entitled for treating them. The same approach has been applied to the field of Business Processes (BPs), as well. BPs realize a business goal by coordinating the tasks undertaken by multiple parties. When processes are distributed, MAS organizations are promising candidates for realizing them. Again, we claimed that, to effectively engineer distributed processes, a modeler should be equipped with abstractions for capturing relationships between the actors, and not only between the process activities. This line has been developed, e.g., in [5].

5 Evaluation

The proposal is being evaluated in two ways. Regarding the formal model, the proposal has been compared to the main approaches to accountability and responsibility from other areas (e.g., social sciences, public administration, psychology). Accountability is central in many fields that study human interaction and the very same approaches can be effective in the context of intelligent systems. The objective is to propose a characterization for use in MAS capturing as many declinations as possible. Moreover, the proposal has been discussed in the context of two widely accepted models of interaction, i.e., normative ones and the ones based on social commitments. For what concerns the programming model, the plan is to evaluate it mainly inside the JaCaMo framework. We are working to enrich JaCaMo's organizational model and infrastructure with accountability

and responsibility. At the same time, BPs offer interesting real-world scenarios to show the benefits coming from an explicit account of accountability and responsibility both at design and execution time.

6 Conclusions

Accountability offers interesting implications in terms of robustness. The final part of my PhD will be focused on this aspect. Robustness is typically defined as "the ability of a software to keep an 'acceptable' behavior [...] in spite of exceptional or unforeseen execution conditions." Casting such a view into the context of MAS organizations, an organization is robust when its agents can react to abnormal events, possibly encompassing contextual information provided by others. Accountable software results to be robust, that is, capable to keep up working within acceptable standards despite the occurrence of some abnormal situations. By way of accountability, an organization designer can specify how (relevant) contextual information produced during the achievement of a goals flows from an agent to another, so as to provide an adequate context for the a-taker's decision-making, especially in front of invalid or exceptional situations.

References

1. Baldoni, M., Baroglio, C., Boissier, O., May, K.M., Micalizio, R., Tedeschi, S.: Accountability and responsibility in agent organizations. In: Proc. of PRIMA 2018. LNCS, Springer (2018)
2. Baldoni, M., Baroglio, C., Boissier, O., Micalizio, R., Tedeschi, S.: Engineering business processes through accountability and agents. In: Proc. of AAMAS 2019. pp. 1796–1798. IFAAMAS (2019)
3. Baldoni, M., Baroglio, C., May, K.M., Micalizio, R., Tedeschi, S.: Computational Accountability in MAS Organizations with ADOPT. Applied Sciences **8**(4) (2018)
4. Baldoni, M., Baroglio, C., May, K.M., Micalizio, R., Tedeschi, S.: MOCA: An ORM MOdel for Computational Accountability. Intelligenza Artificiale **13**(1), 5–20 (2019)
5. Baldoni, M., Baroglio, C., Micalizio, R., Tedeschi, S.: Implementing business processes in JaCaMo+ by exploiting accountability and responsibility. In: Proc. AAMAS 2019, Demo Track. p. 2330–2332. IFAAMAS (2019)
6. Boissier, O., Bordini, R.H., Hübner, J.F., Ricci, A., Santi, A.: Multi-agent oriented programming with JaCaMo. Sci. Comput. Program. **78**(6), 747–761 (2013)
7. Carabelea, C., Boissier, O.: Coordinating agents in organizations using social commitments. Electronic Notes in Theoretical Computer Science **150**(3), 73–91 (2006)
8. Dubnick, M.J., Justice, J.B.: Accounting for accountability (2004), Annual Meeting of the American Political Science Association
9. Feltus, C.: Aligning Access Rights to Governance Needs with the Responsability MetaModel (ReMMo) in the Frame of Enterprise Architecture. Ph.D. thesis, University of Namur, Belgium (2014)
10. Grant, R.W., Keohane, R.O.: Accountability and Abuses of Power in World Politics. The American Political Science Review **99**(1) (2005)
11. Nissenbaum, H.: Accountability in a computerized society. Science and Engineering Ethics **2**(1), 25–42 (1996)