Institutional Situatedness in Multi-Agent Systems

Maiquel de Brito[1], **Jomi Fred Hübner**[1]

[1]PPGEAS – Universidade Federal de Santa Catarina (UFSC)
Florianópolis – SC – Brazil

maiquel.b@posgrad.ufsc.br, jomi.hubner@ufsc.br

Abstract. *Institutions regulating multi-agent systems are not autonomous. Their state changes as the result of facts occurring in the environment where agents act. The modelling of institutional dynamics allowing the institutional regulation based on environmental facts is not a trivial matter. Different institutional abstractions represent different social aspects, having different natures, semantics, life cycles, etc. This work reviews the institutional situatedness analysing how it is currently addressed. We propose a classification for the existing approaches and point some open issues to be tackled in such subject.*

1. Introduction

Institutional abstractions such as norms, roles, goals, missions, scenes, etc, have been proposed to conciliate the goals of the system and the issues raised from system openness and agents autonomy [Artikis et al. 2002, Boissier et al. 2007, Piunti 2009, Esteva et al. 2004]. A regulated MAS may be viewed not just as a set of agents acting in the environment. Besides, a set of mechanisms implementing institutional abstractions composes the institutional dimension (or simply the *institution*) of the multi-agent system (MAS), as shown in the Figure 1.

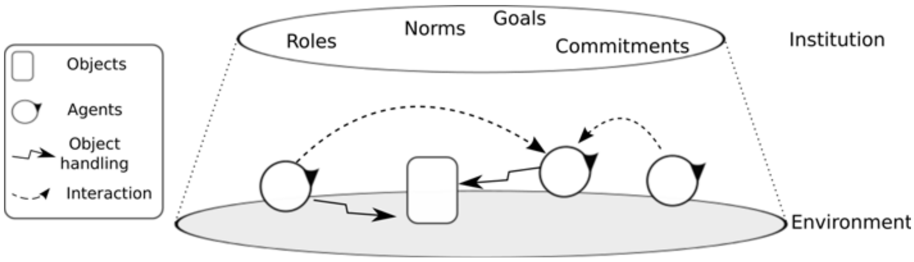

Figure 1. The institution regulates the agents acting in the environment.

Different from the agents, the institution is not autonomous and its state does not change spontaneously. Rather, its state changes as result of facts occurring in the environment, composed of agents and other objects [Searle 1995, Searle 2009]. We consider an institution as *situated* in the environment when its regulation tasks are performed based on environmental facts so that it does not depend on agents informing norm violations, goal achievements, role adoptions, etc. We refer as *situatedness* to the property of an institution being situated in the environment. While specifying the institution requires models and languages to represent institutional abstractions, specifying the institutional situatedness requires models and languages to represent how environmental facts affect the different components of the institution.

Some works have proposed models and languages to specify institutional situatedness. But, as far as we know, there is not any work explicitly reviewing such proposals, comparing them and pointing open issues in the field. To fill this gap, this work analyses how current works deal with situatedness. We are not interested in understanding how specific institutional abstractions are situated. Rather, we aim to have a wide view of the subject, understanding how it is addressed regardless institutional abstractions that are situated.

The main contributions of this analysis are (i) a classification for the current approaches in which other approaches (current or further) can fit (ii) a better understanding about different approaches, according to the proposed classification and (iii) some directions about open issues in this subject. The rest of this paper is organized as follows: Section 2 describes some relevant points to understand the situatedness problem, Section 3 describes how the problem is currently addressed, Section 4 discusses the proposed solutions and, finally, Section 5 points some final remarks about the work.

2. Background

In MAS, several abstractions are proposed to represent social aspects needed to regulate and direct the autonomous acting of the agents towards the achievement of system goals. Some examples of these abstractions are norms, goals, roles, missions, interaction scenes, etc. Each abstraction has a specific nature, representing a specific social aspect of MAS. For example, norms represent what the agents must achieve or avoid, roles represent coherent behaviour expected from an agent, missions represent sets of coherent goals to be achieved by an agent, etc [1]. These abstractions compose the institution that regulates the system while agents act in the environment.

Institutional abstractions, in general, do not represent how the environment, where agents effectively act, influences the regulating tasks. As consequence, implementations of the abstractions do not consider the institution evolving as result of facts external to it. Some of them leave to some external element (including the agents) the responsibility of informing norms violations, role adoptions, goal achievements, etc through institutional interfaces [Gutknecht and Ferber 2001, Campos et al. 2009, Hübner et al. 2006, Hübner et al. 2009]. This is illustrated in the Figure 2: agents act in the environment and use an interface to handle the institutional platform, adopting roles and committing to missions[2].

There are several issues related to the approach of agents handling the institutional platform [Brito et al. 2013]. First, agents must be aware of institutional abstractions and about its implementation. For example, to adopt a role, an agent must be aware of the semantics of the *role* concept and must also know how a particular platform implements the role adoption. Besides, agents can avoid institutional consequences of their actions. For example, it is reasonable that an agent running through a red traffic light does not inform the institution about the norm violation. Thus, it is worth that institutions are situated in the environment, performing the institutional tasks based on environmental

[1] A comprehensive and well organized analysis of institutional abstractions can be found in [Coutinho et al. 2009]

[2] *adoptRole()* and *commitMission()* are primitives provided by interfaces to handle institutions according to the $\mathcal{M}$oise model. Descriptions of these interfaces are found in [Hübner et al. 2006, Hübner et al. 2009].

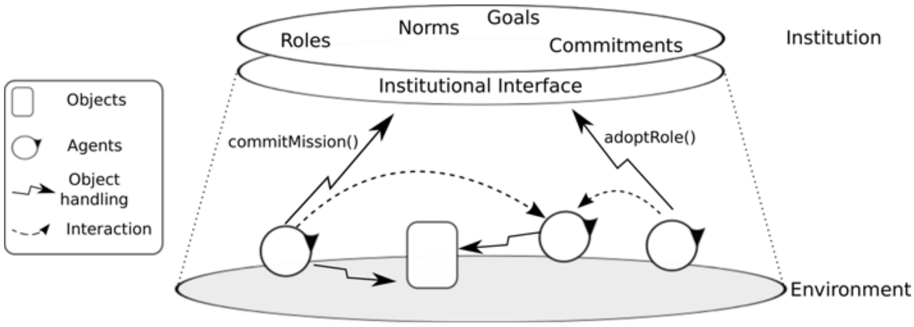

Figure 2. Agents act in the environment and handle the institutional platform.

	Ontological	Functional	
		Concept	Interface
[Dastani et al. 2013]		✓	
[Dastani et al. 2008]		✓	
[Campos et al. 2009]			✓
[Piunti et al. 2010]			✓
[Brito et al. 2013]			✓
[Aldewereld et al.]	✓		

Table 1. Approaches for situatedness

facts rather than on agents handling [Campos et al. 2009].

In human societies, the relation between concrete facts and the institutional state raises naturally and is analysed by the social sciences [Searle 1995]. But in MAS, which can be considered artificial societies [Castelfranchi 2000], such relation is not natural and must be modelled, defining how environmental facts affect the different components of the institution. This modelling, i.e. the conception of institutional situatedness, is not a trivial matter: different institutional abstractions represent different social aspects, having different natures, semantics, life cycles, etc. These differences must be taken into account in situatedness models. The next section describes some works that face this issue.

3. Approaches for situatedness

Some works have addressed situatedness in MAS. They are usually inspired in the *count as* theory of John Searle [Searle 1995, Searle 2009]. According to that theory, human institutions are based on constitutive rules stating that concrete elements *count as* institutional ones. For example, a piece of paper *counts as* a five dollar bill, the eldest son of the deceased king *counts as* the new king, etc.

A contribution of this paper is a classification of the approaches for situatedness. This classification is proposed based on the analysis of some related works. It divides the works into two groups: ontological, described in the Section 3.1 and functional, described in the Section 3.2. The Table 1 summarises the related works according to their approaches.

3.1. Ontological approach for situatedness

Situatedness can be viewed as an ontological matter. Once institutions are specified through abstract concepts, ontological situatedness relates such concepts to concrete elements from the environment. For instance, when a norm states that "authors are forbidden to submit papers having over than 15 pages", ontological situatedness defines what is an author, what is a submission, how to detect the number of pages of the paper, etc.

In the literature, the work of [Aldewereld et al.], in line with [Grossi et al. 2006a, Grossi et al. 2006b, Aldewereld et al. 2009] proposes situatedness according to the ontological approach. The work deals with situatedness of norms and proposes a way to define the meaning of abstract concepts used in norm specifications. This is done through rules in a production system. If a norm states that a is obliged to do b, the rules define that j counts as a and k counts as b, where j and k are environmental elements.

3.2. Functional approach for situatedness

The different institutional abstractions have their specific life cycles. By *life cycle* we mean the set of states where an instance of an institutional abstraction can be found and how the abstraction changes its state. A norm, for example, can be *active*, *fulfilled*, *violated* and *disabled*. A possible life cycle for a norm is *active* and then *fulfilled*. The functional approach for *situatedness* specifies how the life cycle of institutional abstractions evolves as result of facts occurring in the environment.

To exemplify the functional approach and state a clear difference from the ontological one we can again use the norm "authors are forbidden to submit papers having over than 15 pages". While the ontological approach allows to specify what is an author, the functional approach specifies what must happen in the environment so that the norm is active, violated, satisfied, etc.

There are two kinds of functional approaches for situatedness, that we refer as *concept* situatedness and *interface* situatedness.

3.2.1. Concept situatedness

As previously described, different institutional abstractions have specific life cycles. For instance, (i) a *role* may be adopted and then leaved; (ii) a norm may become active and then satisfied or violated; (iii) a goal may be pending and then fulfilled. Taking these differences into account, concept situatedness defines how the life cycle of *specific* abstractions, which represent institutional concepts, are affected by facts from the environment[3].

The proposal of [Dastani et al. 2013] is an example of concept situatedness. The situated abstraction is *commitment*. The work deals with the changes in the state of commitments (according to the life cycle shown the Figure 3) as consequence of environmental facts. The code excerpt (cex1) below shows an example of specification according to such approach. The line 1 states that an agent x proposing to an agent y to achieve q before the instant d_2 if y achieve p before d_1 counts as a new commitment whose initial state is *conditional* (C^c). The line 2 states that, from the commitment defined in the line 1, the

[3]An institutional abstraction represents an institutional concept.

agent x informing to y that has done q before d_1 counts as satisfying of the commitment, whose state changes from *conditional* to *satisfied* (C^s).

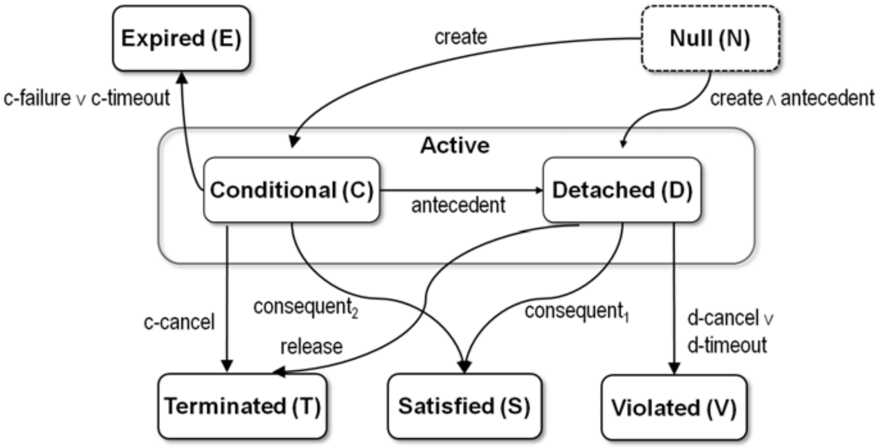

Figure 3. Life cycle of a commitment [Dastani et al. 2013]

$$\begin{aligned} 1: &\quad offer(x,y,p,q,d_1,d_2) \Longrightarrow_{cr} \mathsf{C}^c(x,y,p,q,d_1,d_2) \\ 2: &\quad tell(x,y,q) \wedge \mathsf{C}^c(x,y,p,q,d_1,d_2) \wedge \neg d1 \wedge q \Longrightarrow_{cr} \mathsf{C}^s(x,y,p,q,d_1,d_2) \end{aligned} \quad \text{(cex1)}$$

Concept situatedness is also the approach in the set of works related to the 2OPL model, composed by [Dastani et al. 2009, Dastani et al. 2008, Tinnemeier et al. 2009, Tinnemeier et al. a, Tinnemeier et al. b]. In this case, the abstraction considered is *norm*. The works propose a way to represent how the life cycle of a norm, illustrated in Figure 4, is affected by the environmental state. The code excerpt (cex2) below shows a rule according to this approach. The rule defines that if the environment has properties pointing that a paper having more that 15 pages has been submitted to a conference, then there is a norm violation.

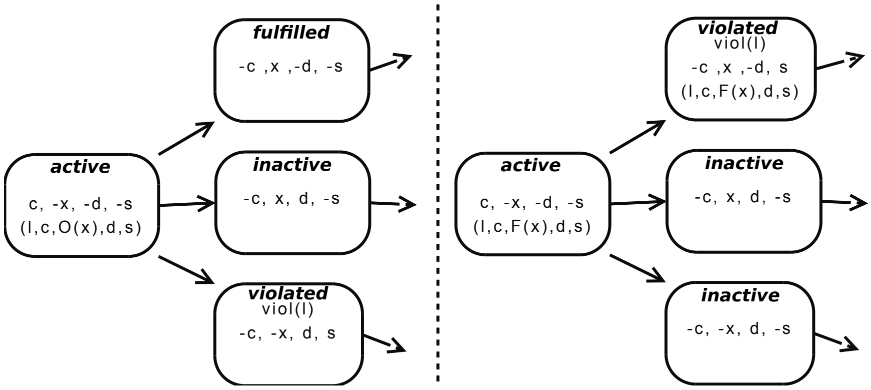

Figure 4. Life cycle of obligations (left) and prohibitions (right) (adapted from [Tinnemeier et al. 2009])

.

$$received(As) \wedge member((A,Id),As) \wedge pages(Id) > 15 \Longrightarrow viol\_size(A) \quad \text{(cex2)}$$

3.2.2. Interface situatedness

Different from concept situatedness, interface situatedness is not concerned about which are the institutional abstractions affected by environmental facts. Institutional concepts are not included in the models. At a glance, in this approach, a situatedness interface observes the environment and, by interpreting the situatedness specification, produces informations about *what should happen* in the institution. These informations do not have, themselves, institutional meaning. It is assumed that the institution takes such informations and changes its own state accordingly (Figure 5).

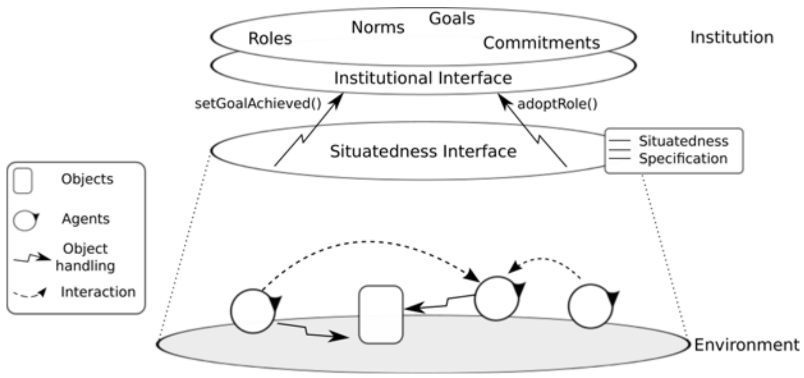

Figure 5. The situatedness interface observes the environment and informs the institution when necessary.

The work of Piunti et al. [Piunti 2009, Piunti et al. 2010] deals with situatedness of the $\mathcal{M}$oise organisational model. A language is proposed to specify that events from the environment trigger operations in ORA4MAS artifacts, that implement the organizational model [Hübner et al. 2009, Ricci et al. 2011]. The code excerpt cex3, extracted from [Piunti et al. 2010], shows a specification according to this approach. It specifies that the event *pay*, produced by the environmental object *BillingMachine*, triggers the operation *setGoalAchieved* in the artifact named *visitSchBoard*. In the application scenario, the rule states that when a billing machine produces the event *toPay*, the organisational goal *pay_visit* is achieved.

$$\begin{aligned}&+\texttt{op\_completed}(\texttt{"BillingMachine"}, \texttt{Ag}, \texttt{pay})\\&\quad - > \texttt{apply}(\texttt{"visitorSchBoard"}, \texttt{setGoalAchieved}(\texttt{Ag}, \texttt{pay\_visit})).\end{aligned} \tag{cex3}$$

Interface situatedness is also the approach in the work of Brito et al. [Brito 2012, Brito et al. 2013], where a programming language is proposed to specify properties that the institution should have as consequence of both events or states of the environment. The code excerpt cex4, extracted from [Brito et al. 2013], is an example of this approach. It states that if the environmental object *Art* has the property *auctionStatus(closed)*, then the institution should have the property *play(Winner,Role,g1)*. In the application scenario, the property *play(Winner,Role,g1)* holds when the agent *Winner* plays the role *Role* in the group *g1*.

$$* \ \texttt{auctionStatus(closed)[source(Art)]} \\ \textbf{count} - \textbf{as} \ \texttt{play(Winner, Role, g1)[source(g1)]} \\ \textbf{in} \ \texttt{currentWinner(Winner)[source(Art)]} \ \& \\ \texttt{not(Winner == no\_winner)} \ \& \ \texttt{auction\_role(Art, Role)}. \quad \text{(cex4)}$$

A third case of interface situatedness is the *Situated Electronic Institutions* model (SEI) [Campos et al. 2009], that provides situatedness to *Electronic Institutions* (EI) [Esteva et al. 2001]. In SEI, special agents named *modeller* and *staff agents* monitor the environment checking facts that may be relevant to the institution. Those facts are informed to special agents named *governors* that are aware about institutional specification and determine the institutional meaning of the facts.

4. Discussion

The Section 3 presented some works related to institutional situatedness proposing a classification to them. While that section describes how different approaches propose to situate an institution, this section discusses two important points observed in the analysed works. The first point, discussed in the Section 4.1 is the comprehensiveness of the approaches regarding to institutional aspects that can be situated. The second point, discussed in the Section 4.2, is the institutional semantic of specifications following the different approaches.

4.1. Institutional coverage

In concept situatedness, each model situates a specific abstraction (Figure 6). This implies a limited coverage to the institutional abstractions that can be situated. The analysed works propose conceptual situatedness for commitments and norms. The proposed models cannot be used to situate additional abstractions. Different concepts require different models and languages. Besides, the same institutional abstraction may have many different conceptions. For example, the life cycle of commitments in [Dastani et al. 2013] is different of the one proposed in [Fornara and Colombetti 2006] e [Chesani et al. 2013]. Thus, different conceptions for the same abstraction require different models of concept situatedness. To provide conceptual situatedness for all institutional abstractions possibly present in an MAS, it would be necessary to provide languages for every abstraction. Admitting that some institutional aspects of MAS may have not been identified yet, new institutional abstractions may be still proposed and to situate them, it is needed additional work to propose models and languages.

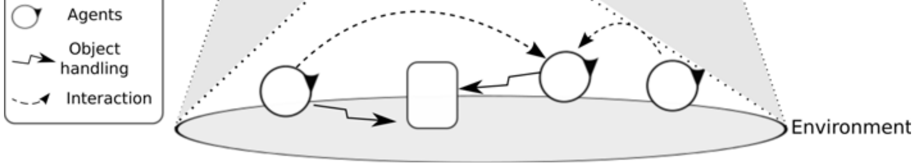

Figure 6. Concept situatedness: an approach situates just one abstraction.

Interface situatedness and ontological situatedness do not have the issue of limited institutional coverage. But it does not mean completeness regarding to the situated institutional abstractions. Rather, such approaches do not take into account specific institutional abstractions. Although the approaches consider that the environmental facts affect the institution, it is not taken into account whether they are affecting norms or roles or goals or anything else. While this feature provides a wider institutional coverage, it raises some issues related to the institutional semantics of the models. These issues are discussed in the next section.

4.2. Institutional semantics

Conceptual situatedness is concerned with the environment affecting the life cycle of specific abstractions. The semantics of the abstractions is taken into account in the semantics of languages for situatedness specification. For example, the code excerpt cex1 shows the specification of institutional effects for messages exchanged by the agents. Such specification uses the syntactic elements C^{c} e C^{s}, that have institutional semantics. The rule shown in the Figure 7 belongs to the operational semantics of the approach and states that from a communication act $com(\alpha)$, the institutional state of the system is updated according to the operator $\oplus$.

$$\frac{com(\alpha) \wedge \sigma_i' = \oplus(\sigma_i \cup \sigma_b \cup \{\alpha\})}{\langle \sigma_b, \sigma_i \rangle \rightarrow \langle \sigma_b, \sigma_i' \rangle}$$

$$where \quad \oplus (\sigma_i \cup \sigma_b \cup offer(x, y, p, q, d_1, d_2)) = \sigma_i \cup \mathsf{C}^{\mathsf{c}}(x, y, p, q, p1, d2)$$

Figure 7. Semantic rule for commitment situatedness [Dastani et al. 2013].

Interface situatedness, on its turn, does not make any assumption about specific institutional abstractions. Institutional concepts are not part of the models and related languages do not use institutional primitives. As a consequence, languages for interface situatedness cannot express the institutional meaning of environmental facts. The languages define how the facts from the environment are handled to produce some information that is provided to the institution. But this information does not have itself institutional meaning. The institution is in charge of handling that information, updating its institutional abstractions accordingly. That is, the institutional meaning of the produced information is given by the very institution instead of being given by the situatedness model. For example, (i) in the model of [Piunti 2009, Piunti et al. 2010]., is specified that events from the environment trigger operations in artifacts and (ii) in the model of Brito et al., is specified that the institution should have some properties as consequence of events and states from the environment. In both cases, however, the institutional meaning of the environmental facts is placed into the designer's mind rather than be placed in the language semantics. For example, the code excerpt cex3, from [Piunti et al. 2010], specifies that the event *pay*, produced by an environmental object named *BillingMachine*, triggers the operation *setGoalAchieved* in an artifact named *visitorSchBoard*. In its specific application, the rule means that the institutional consequence of the event *pay* produced by an electronic billing machine is the achievement of the institutional goal *pay_visit*. But such institutional meaning is not explicit in the semantics of the language. The semantics just

$$\frac{\langle x,y,c\rangle \in R_s \quad \mathcal{N}\cup\mathcal{I}\cup D \models x \quad \mathcal{N}\cup\mathcal{I}\cup D \models c \quad y \notin \mathcal{I}}{\langle R_e,R_s,D,\mathcal{E},\mathcal{N},\mathcal{I},\mathcal{T}\rangle \longrightarrow \langle R_e,R_s,D,\mathcal{E},\mathcal{N},\mathcal{I},\mathcal{T}\cup y\rangle}$$

Figure 8. Semantic rule for count-as rule evaluation [Brito et al. 2013].

states that the operation *apply*, with two parameters, is triggered in an artifact. The institutional semantics, on its turn, is defined in the artifact operation: *setGoalAchieved(a,g)* means that the goal *g* has been achieved by the agent *a* [Hübner et al. 2009].

The approach of [Brito et al. 2013, Brito 2012] has a similar issue. The code excerpt cex4 states that when the environment has the property *auctionStatus(Closed)*, then institution should have the property *play(Winner,Role,g1)*. But the property *play(Winner,Role,g1)* does not have, itself, any institutional meaning. In its specific application, the property is truth when the agent *Winner* plays the role *Role* in the group *g1*. But the institutional platform is in charge of interpreting the property giving it an institutional meaning. Notice that *role*, that is an institutional concept, does not even belong to the situatedness model. We can clearly see the lack of institutional semantics in this proposal through the operational semantics of the language. The semantic rule shown in the Figure 8 states that when a rule $\langle x, y, c\rangle$ is evaluated, the element y is added to a queue $\mathcal{T}$ to be consumed by the institutional platform. The element y does not have institutional meaning.

Although SEI model does not have a specification language, this approach for situatedness also lacks of institutional semantics. Special agents observe the environment and just inform the *governors* about relevant facts. The *governors* are in charge of give institutional semantics for that information.

Finally, the approach of [Aldewereld et al.], that addresses situatedness as an ontological problem, also lacks of institutional semantics. Resuming a previous example, for a norm stating that "*a* is obliged to *b*", the implementation of ontological situatedness allows to specify that "*j* counts as *a*" and "*k* counts as *b*". But it allows also to specify that "*j* counts as *b*" and "*k* counts as *a*", that is wrong as *a* is an agent while *b* has a different nature. This is possible because the environmental elements are related to the norm elements but are not related to institutional concepts. For example, the rule states that "*j* counts as *a*" but do not takes into account that *j* is an agent. The designer is in charge of to consider this semantics when writes the rules.

The use of institutional primitives to situatedness programming, as allowed by the concept situatedness, is an advantage as the programs have an explicit institutional meaning. This allows the agents (both human and artificial) to reason about the institutional effects of environmental facts and of their own actions. For example, observing the code excerpt cex1, through the semantics of Figure 7, an agent knows that when it utters an *offer*, it produces a commitment. On other hand, when situatedness language lacks of institutional primitives, agents may not be able to use the situatedness program to reason about the institutional effects of environmental facts. By observing the code excerpt cex3, for example, an agent knows that when the object *BillingMachine* produces the event *pay*, the operation *setGoalAchieved(Ag,pay_visit)* in the artifact *visitorSchBoard*. But to know that such operation means the achievement of the goal *pay_visit*, the agent should to know also the ORA4MAS infrastructure. Thus, in this case, it is not sufficient to the agents to reason about the situatedness specification to know how to achieve the goal *pay_visit*.

As another issue, the lack of institutional semantics in situatedness languages, observed in both ontological and interface situatedness, allows specifications that do not produce any institutional effect. For example, although the replacement of the code excerpt cex3 by the code excerpt below(cex5) keeps the program semantic and syntactically correct, it does not specify any institutional consequence for the environmental fact. Concept situatedness, on its turn, induces a consistent programming as the consequences of environmental facts are defined in terms of institutional concepts.

$$\begin{array}{l}+\texttt{op\_completed}(\texttt{"BillingMachine"},\texttt{Ag},\texttt{pay})\\ \quad -> \texttt{apply}(\texttt{"visitorSchBoard"},\texttt{meaninglessParam}(\texttt{Ag},\texttt{pay\_visit})).\end{array} \tag{cex5}$$

5. Final remarks

This work analysed institutional situatedness in MAS. Current approaches can be divided in two approaches: ontological and functional. We observed that (i) some approaches have a limited coverage regarding to situated institutional abstractions, (ii) some approaches lack of institutional semantics and (ii) there is no approach that situates all institutional abstractions and allows situatedness programming using institutional primitives.

Acknowledgements

The authors are grateful for the support given by CAPES and CNPq (grant number 140261/2013-3).

References

Aldewereld, H., Álvarez-Napagao, S., Dignum, F., and Vázquez-Salceda, J. Making norms concrete. In van der Hoek, W., Kaminka, G. A., Lespérance, Y., Luck, M., and Sen, S., editors, *Proc. 9th International Conference on Autonomous Agents and Multiagent Systems*, pages 807–814.

Aldewereld, H., Alvarez-Napagao, S., Dignum, F., and Vázquez-Salceda, J. (2009). Engineering Social Reality with Inheritance Relations. In *Proc. 10th International Workshop on Engineering Societies in the Agents World X*.

Artikis, A., Pitt, J., and Sergot, M. (2002). Animated specifications of computational societies. In *Proc. 1st international joint conference on Autonomous agents and multiagent systems*.

Boissier, O., Hübner, J. F., and Sichman, J. S. (2007). Organization Oriented Programming: From Closed to Open Organizations. In O'Hare, G. M. P., Ricci, A., O'Grady, M. J., and Dikenelli, O., editors, *Engineering Societies in the Agents World VII*, volume 4457 of *LNCS*, pages 86–105. Springer Berlin Heidelberg.

Brito, M. (2012). Uma linguagem para especificação da dinâmica dos fatos institucionais em sistemas multiagentes. Master's thesis, Universidade Federal de Santa Catarina, Centro Tecnológico. Programa de Pós-Graduação em Engenharia de Automação e Sistemas.

Brito, M., Hübner, J. F., and Bordini, R. H. (2013). Programming institutional facts in multi-agent systems. In Aldewereld, H. and Sichman, J. S. a., editors, *Coordination, Organizations, Institutions, and Norms in Agent Systems VIII*, volume 7756 of *LNCS*, pages 158–173. Springer Berlin Heidelberg.

Campos, J., López-Sánchez, M., Rodríguez-Aguilar, J., and Esteva, M. (2009). Formalising Situatedness and Adaptation in Electronic Institutions. In Hübner, J., Matson, E., Boissier, O., and Dignum, V., editors, *Coordination, Organizations, Institutions and Norms in Agent Systems IV*, volume 5428 of *LNCS*, pages 126–139. Springer Berlin / Heidelberg.

Castelfranchi, C. (2000). Engineering social order. 1972:1–18.

Chesani, F., Mello, P., Montali, M., and Torroni, P. (2013). Representing and monitoring social commitments using the event calculus. *Autonomous Agents and Multi-Agent Systems*, 27(1):85–130.

Coutinho, L. R., Sichman, J. S., and Boissier, O. (2009). *Handbook of Research on Multi-Agent Systems: Semantics and Dynamics of Organizational Models*, chapter Modelling Dimensions for Agent Organizations. Information Science Reference.

Dastani, M., Grossi, D., Meyer, J.-J. C., and Tinnemeier, N. (2009). Knowledge Representation for Agents and Multi-Agent Systems. chapter Normative Multi-agent Programs and Their Logics, pages 16–31. Springer-Verlag, Berlin, Heidelberg.

Dastani, M., Tinnemeier, N., and Meyer, J.-J. (2008). *A programming language for normative multi-agent systems*, chapter XVI, pages 397–417. Information Science Reference, Hershey, PA, USA.

Dastani, M., Torre, L., and Yorke-Smith, N. (2013). Monitoring interaction in organisations. In Aldewereld, H. and Sichman, J. S., editors, *Coordination, Organizations, Institutions, and Norms in Agent Systems VIII*, volume 7756 of *LNCS*, pages 17–34. Springer Berlin Heidelberg.

Esteva, M., Rodríguez-Aguilar, J. A., Sierra, C., Garcia, P., and Arcos, J. L. (2001). On the Formal Specifications of Electronic Institutions. In Dignum, F. and Sierra, C., editors, *AgentLink*, volume 1991 of *LNCS*, pages 126–147. Springer.

Esteva, M., Rosell, B., Rodríguez-Aguilar, J. A., and Arcos, J. L. (2004). AMELI: An Agent-Based Middleware for Electronic Institutions. pages 236–243.

Fornara, N. and Colombetti, M. (2006). Specifying and Enforcing Norms in Artificial Institutions. In Dunin-Keplicz, B., Omicini, A., and Padget, J. A., editors, *EUMAS*, volume 223 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Grossi, D., Aldewereld, H., Vázquez-Salceda, J., and Dignum, F. (2006a). Ontological aspects of the implementation of norms in agent-based electronic institutions. *Computational & Mathematical Organization Theory*, 12(2-3):251–275.

Grossi, D., Meyer, J.-J. C., and Dignum, F. (2006b). Counts-as: Classification or Constitution? An Answer Using Modal Logic. In Goble, L. and Meyer, J.-J. C., editors, *DEON*, volume 4048 of *LNCS*, pages 115–130. Springer.

Gutknecht, O. and Ferber, J. (2001). The madkit agent platform architecture. In Wagner, T. and Rana, O. F., editors, *Infrastructure for Agents, Multi-Agent Systems, and Scalable Multi-Agent Systems*, volume 1887 of *Lecture Notes in Computer Science*, pages 48–55. Springer Berlin Heidelberg.

Hübner, J. F., Boissier, O., Kitio, R., and Ricci, A. (2009). Instrumenting multi-agent organisations with organisational artifacts and agents. *Autonomous Agents and Multi-Agent Systems*, 20(3):369–400.

Hübner, J. F., Sichman, J. S., and Boissier, O. (2006). $\mathcal{S}-\mathcal{M}$oise^{+}: A middleware for developing organised multi-agent systems. In Boissier, O., Padget, J., Dignum, V., Lindemann, G., Matson, E., Ossowski, S., Sichman, J., and VÃ¡zquez-Salceda, J., editors, *Coordination, Organizations, Institutions, and Norms in Multi-Agent Systems*, volume 3913 of *Lecture Notes in Computer Science*, pages 64–77. Springer Berlin Heidelberg.

Piunti, M. (2009). *Designing and Programming Organizational Infrastructures for Agents situated in Artifact-based Environments*. PhD thesis, Universit' di Bologna.

Piunti, M., Boissier, O., Hübner, J. F., and Ricci, A. (2010). Embodied organizations: a unifying perspective in programming agents, organizations and environments. In *MALLOW*.

Ricci, A., Piunti, M., and Viroli, M. (2011). Environment programming in multi-agent systems: an artifact-based perspective. *Autonomous Agents and Multi-Agent Systems*, 23(2):158–192.

Searle, J. (1995). *The Construction of Social Reality*. Free Press.

Searle, J. (2009). *Making the Social World:The Structure of Human Civilization*. Oxford University Press.

Tinnemeier, N., Dastani, M., and Meyer, J.-J. In *Proc. of 8th International Conference on Autonomous Agents and Multiagent Systems - Volume 1*, pages 121–128, Richland, SC.

Tinnemeier, N., Dastani, M., and Meyer, J.-J. In *Proc. of 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1 - Volume 1*, pages 957–964, Richland, SC.

Tinnemeier, N. A. M., Dastani, M. M., Meyer, J.-J. C., and van der Torre, L. (2009). Programming Normative Artifacts with Declarative Obligations and Prohibitions. *Web Intelligence and Intelligent Agent Technology, IEEE/WIC/ACM International Conference on*, 2:145–152.