# Grafting norms onto the BDI agent model

Mihnea Tufis, Jean-Gabriel Ganascia

# Grafting norms onto the BDI agent model

**Mihnea Tufiş** and **Jean-Gabriel Ganascia** [1]

**Abstract.** This paper proposes an approach on how to accommodate norms to an already existing architecture of rational agents. Starting from the famous BDI model, an extension of the BDI execution loop will be presented; it will address such issues as norm instantiation and norm internalization, with a particular emphasis on the problem of norm consistency. A proposal for the resolution of conflicts between newly occurring norms, on one side, and already existing norms or mental states, on the other side, will be described. While it is fairly difficult to imagine an evaluation for the proposed architecture, a challenging scenario inspired form the science-fiction literature will be used to give the reader an intuition of how the proposed approach will deal with situations of normative conflicts.

## 1 INTRODUCTION

*"Mistress, your baby is doing poorly. He needs your attention."*
*"Stop bothering me, you f* robot."*
*"Mistress, the baby won't eat. If he doesn't get some human love, the Internet pediatrics book says he will die"*
*"Love the f*ing baby, yourself."*

The excerpt is from Prof. John McCarthy's short story "The Robot and the Baby" [9], which besides being a challenging and insightful look into how a future society where humans and robots might function together, also provides with a handful of conflicting situations that the household robot R781 has to resolve in order to achieve one of its goals: keeping baby Travis alive.

The scenario itself made us think about how such a robot could be implemented as a rational agent and how would a normative system would graft onto it. Granted, McCarthy's story is offering a few clues about the way the robot is reasoning and is reaching decisions, but he also lets us wonder about the architecture of a rational agent, such like R781, and how it would function in a normative context. In the following, we will be trying to look exactly into that: how can the well known Beliefs-Desires-Intentions (BDI) rational agent architecture be combined with a normative system to give what we call a normative BDI agent?

The paper is structured as follows: in the next section we will review the state of the art in the field of normative agent systems and present several approaches which we found of great value to our work. In the third section we describe our proposal for normative BDI agents, which will be supported by the case study scenario in the fourth section. In the fifth section we will give details on the future work, before summing up the conclusions of our work so far.

[1] Laboratoire d'Informatique de Paris 6 (LIP6), Université Pierre et Marie Curie – Sorbonne Universités, France, email: mihnea.tufis@lip6.fr

## 2 STATE OF THE ART

### 2.1 Agents, norms, normative agent systems

In the following we will be using what we consider a satisfying definition of (intelligent) agent as given by Michael Wooldridge [13]. Please refer to it, for your convenience.

One of the first key points is defining the notion of norm. This turns out to be a bit more difficult than expected in the context of intelligent agents. Norms are interesting for many domains: law, economics, sports, philosophy, psychology etc. However, we would be interested in such definitions specific to the field of multiagent systems (MAS). Since this domain itself is very much interdisciplinary, defining a norm remains a challenge. For example, we would be interested in a definition applicable to social groups, since MAS, can be seen as models of societies. Thus, in [2] the definition of a norm is given as "a principle of right action binding upon the members of a group and serving to guide, control, or regulate proper or acceptable behaviour". On a slightly more technical approach, in distributed systems norms have been defined as regulations or patterns of behaviour meant to prevent the excess in the autonomy of agents [6].

We can now refer to the normchange definition of a normative multiagent system as it has been proposed in [1]. We find this definition to be both intuitive and to underline very well the idea of coupling a normative system to a system of agents:

**Definition 1** *A **normative multiagent system** is a multiagent system together with normative systems in which agents on the one hand can decide whether to follow the explicitly represented norms, and on the other the normative systems specify how and in which extent the agents can modify the norms.*

An alternative definition of a normative multiagent system, as it was formulated in [3] is given:

**Definition 2** *A **normative multiagent system** is a multiagent system organized by means of mechanisms to represent, communicate, distribute, detect, create, modify and enforce norms and detect norm violations and fulfilment.*

### 2.2 NoA agents

An interesting approach to the problem of norm adoption by a multiagent system has been provided by Kollingbaum and Norman in [8].

Kollingbaum and Norman study what happens when a new norm is adopted by an agent: what is the effect of a new norm on the normative state of the agent? Is a newly adopted norm consistent with the previously adopted norms?

To this extent they propose a normative agent architecture, called NoA, which is built as a reactive agent architecture. The **NoA architecture** is fairly simple and it comprises of a set of beliefs, a set of plans and a set of norms.

The second reason for which we gave a great deal of attention to NoA is the formalization of the way an agent will adopt a norm following the consistency check between a newly adopted norm and its current normative state. Due to lack of space, we allow the reader to refer to [8] for the exact details.

Using some of the ideas of NoA, we will try to work on what we consider to be its limits. We recall that NoA is based on a reactive architecture; considering our BDI approach we will have to extend the consistency check such as it applies not only to the normative state of the agent but also on its mental states (i.e. check whether a newly adopted norm is consistent with the BDI agent's current mental states). The second point we will study is the consistency check during the norm acquisition stage.

## 2.3 A BDI architecture for norm compliance - reasoning with norms

The second study which we found relevant in our endeavour to adapt the BDI agent architecture to normative needs is the work of Criado, Argente, Noriega and Botti [6]. Their work is particularly interesting since it tackles the problem of norm coherence for BDI agents. They propose a slight adaptation of the BDI architecture in the form of the n-BDI agent for graded mental states. Since our work won't use graded mental states, we will omit details regarding to these in the description of the n-BDI architecture:

- Mental states. Represent the mental states of the agent, same as for the BDI agent. We distinguish the Beliefs Context (belief base), Desires Context (desires/goal base) and the Intentions Context (intentions base/plan base).
- Functional contexts. Address the practical issues related to an agent through the Planning Context and the Communication Context.
- Normative contexts. Handle issues related to norms through the Recognition Context and the norm application context.

Another important point of the cited work is the distinction between an abstract norm and instance of a norm.

**Definition 3** *An **abstract norm** is defined by the tuple:* $n_a = \langle M, A, E, C, S, R \rangle$*, where:*

- $M \in \{F, P, O\}$ *is the modality of the norm: prohibition, permission or obligation*
- $A$ *is the activation condition*
- $E$ *is the expiry condition*
- $C$ *is the logical formula to which the modality is applied*
- $S$ *is the sanction in the case the norm is broken*
- $R$ *is the reward in case the norm is satisfied*

**Definition 4** *Given a belief theory* $\Gamma_{BC}$ *and an abstract norm* $n_a$ *as defined above, we define a **norm instance** as the tuple:* $n_i = \langle M, C' \rangle$*, where:*

- $\Gamma_{BC} \vdash \sigma(A)$
- $C' = \sigma(C)$*, where* $\sigma$ *is a substitution of variables in* $A$*, such that* $\sigma(A)$*,* $\sigma(S)$*,* $\sigma(R)$ *and* $\sigma(E)$ *are grounded*

The specific architectural details regarding the normative contexts and the bridge rules used during a norm's life cycle will be awarded more attention in section 3.2.

In [6], a base is set for the study of the dynamics between norms and the mental states of a BDI agent. Additionally, it provides with a good idea for checking coherence between the adopted norms and the agent's mental states. The main drawback of the approach is the lack of coverage concerning the topic of norm acquisition. Therefore, a big challenge will be to integrate this approach, with the consistency check presented in section 2.2, as well as finding a good way to integrate everything with the classic BDI agent loop, as presented in [13].

## 2.4 Worst consequence

An important part of our work will focus on solving conflicts between newly acquired norms and the previously existing norms or the mental contexts of the agent. Beforehand we draw from some of the definitions given by Ganascia in [7]. Those will later help us define what a conflict set is and how we can solve it.

**Definition 5** *Given* $(\phi_1, ..., \phi_n, \phi') \in \mathcal{L}_\neg^{n+1}$*,* $\phi'$ ***is a consequence of*** $(\phi_1, ..., \phi_n)$ *according to the belief-set* $B$ *(we write* $\phi' = csq(\phi_1, ..., \phi_n)[B]$ *if and only if:*

- $\phi' \in (\phi_1, ..., \phi_n)$ *or*
- $\exists \Phi \subseteq (\phi_1, ..., \phi_n)\ s.t.\ \Phi \rightarrow \phi' \in B$ *or*
- $\exists \phi'' \in \mathcal{L}_\neg\ s.t.\ \phi'' = csq(\phi_1, ..., \phi_n)[B] \wedge \phi' = csq(\phi_1, ..., \phi_n, \phi'')[B]$

**Definition 6** $\phi$ ***is worse than*** $\phi'$ *given the belief-set* $B$ *(we write* $\phi \succ_c \phi'$*) if and only if one of the consequences of* $\phi$ *is worse than any of the consequences of* $\phi'$*.*

- $\exists \eta \in \mathcal{L}_\neg\ s.t.\ \eta = csq(\phi)[B]$ *and*
- $\exists \phi'' \in \mathcal{L}_\neg\ s.t.\ \phi'' = csq(\phi')[B] \wedge \eta \succ_c \phi''[B]$ *and*
- $\forall \phi'' \in \mathcal{L}_\neg, if\ \phi'' = csq(\phi')[B]\ then\ \eta \succ_c \phi''[B] \vee \eta \parallel \phi''[B]$

*Notation:* $\forall (\phi, \phi') \in \mathcal{L}_\neg,\ \phi \parallel \phi'[B]$ *means that* $\phi$ *and* $\phi'$ *are not comparable under B, i.e. neither* $\phi \succ_c \phi'[B]$ *nor* $\phi' \succ_c \phi[B]$*.*

**Definition 7** $\alpha$ *and* $\alpha'$ *being subsets of* $\mathcal{L}_\neg$*,* $\alpha$ ***is worse than*** $\alpha'$ *given the belief-set* $B$ *(we write* $\alpha \succ_c \alpha'[B]$*) if and only if:*

- $\exists \phi \in \alpha . \exists \eta \in \alpha'\ s.t.\ \phi \succ_c \eta[B]$ *and*
- $\forall \eta \in \alpha' . \phi \succ_c \eta[B] \vee \phi \parallel \eta[B]$

# 3 A NORMATIVE EXTENSION ON THE BDI ARCHITECTURE

## 3.1 The classical BDI architecture

A cornerstone in the design of practical rational agents was the **B**eliefs-**D**esires-**I**ntentions model (BDI), first described by Rao and Georgeff in [10]. This model is famous for being a close model of the way the human mind makes use of the mental states in the reasoning process. It is based on what are considered to be the three main mental states: the beliefs, the desires and the intentions of an agent. In the following we will discuss each element of the BDI architecture.

- Beliefs represent the information held by the agent about the world (environment, itself, other agents). The beliefs are stored in a belief-set.

- Desires represent the state of the world which the agent would like to achieve. By state of the world we mean either an action an agent should perform or a state of affairs it wants to bring upon. In other words, desires can be seen as the objectives of an agent.
- Intentions represent those desires to which an agent is committed. This means that an agent will already start considering a plan in order to bring about the goals to which it is committed.
- Goals. We can view goals as being somehow at the interface between desires and intentions. Simply put, goals are those desires which an agent has selected to pursue.
- Events. These trigger the reactive behavior of a rational agent. They can be changes in the environment, new information about other agents in the environment and are perceived as stimuli or messages by an agent's sensors. Events can update the belief set of an agent, they can update plans, influence the adoption of new goals etc.

For the pseudocode for the execution loop of a BDI agent, please refer to [13].

## 3.2 Normative BDI agents

Starting from the BDI execution loop earlier described we will now introduce and discuss a solution for taking into account the normative context of a BDI agent.

First, the agent's mental states are initialized. The main execution loop starts with the agent observing its environment through the `see()` function and interpreting the information as a new percept $\rho$. This could be an information given by its sensors about properties of the environment or information about other agents, including messages received from other agents. These messages may be in some cases *about* a norm (e.g. the performative of an ACL message specifying an obligation or a prohibition).

The agent is then updating its beliefs through the `brf()` function. If the agent realizes that percept $\rho$ is about a norm, it should initialize the acquisition phase of a potential norm. There is a multitude of ways in which an agent can detect the emergence of norms in its environments and a good review of those is given in [11]. For simplicity, we will consider that norms are transmitted via messages and our agent will consider the sender of such a message to be a trusted normative authority. Therefore, the function above will treat a "normative" percept:

```
brf(B, ρ)
{
 ...
 if (ρ about abstract norm n_a) then
 {
  acquire(n_a)
  add(n_a, ANB)
 }
 ...
 return B
}
```

The agent will acquire a new abstract norm $n_a$ (see section 2.3) and store it in the Abstract Norms Base(ANB). Drawing from the normative contexts described in [6], we define the ANB as a base of in-force norms. It is responsible with the acquisition of new norms based on the knowledge of the world as well as the deletion of obsolete norms. However, at this point the agent is simply storing an abstract norm which it detected to be in-force in its environment; it has not yet adhered to it!

Next, a BDI agent will try to filter its desires, based on its current beliefs about the world and its current intentions. It does so by calling the `options(B, I)` method. However, a normative BDI agent should at this point take into account the norms which are currently in force and check whether the instantiation of such norms will have any impact on its current normative state as well as on its mental states.

### 3.2.1 Consistency check

It is at this stage that we will perform the consistency check for a given abstract norm $n_a$.

Drawing from the formalization in [8] regarding norm consistency, we give our own interpretation of this notion.

Let us define the notion of consistency between a plan $p$ and the currently in-force norms to which an agent has also adhered and which are stored in the Norm Instance Base (NIB). By contrast to the ANB, the NIB stores the instances of those norms from the ANB which become active according to the norm instantiation bridge rule (see below).

**Definition 8** *A plan instance $p$ is **consistent** with the currently active norms in the NIB, if the effects of applying plan $p$ are not amongst the forbidden effects of the active norms and the effects of current obligations are not amongst the negated effects of applying plan $p$.*

$$consistent(p, NIB) \iff$$
$$(effects(n_i^F) \setminus effects(n_i^P)) \cap effects(p) = \emptyset$$
$$\wedge$$
$$effects(n_i^O) \cap neg\_effects(p) = \emptyset$$

The types of consistency / inconsistency which can occur between a newly adopted norm and the currently active obligations are:

- **strong inconsistency** occurs when all plan instantiations $p$ which satisfy the obligation $o$ are either explicitly prohibited actions by the NIB or the execution of such a plan would make the agent not consistent with its NIB
- **strong consistency** occurs when all the plan instantiations $p$ which satisfy the obligation $o$ are not amongst the explicitly forbidden actions by the NIB and the execution of such a plan would keep the agent consistent with the NIB
- **weak consistency** occurs when there exists at least one plan instantiation $p$ to satisfy obligation $o$ which is not explicitly prohibited by the NIB and the execution of such a plan would keep the agent consistent with its NIB.

It is simple to define the analogous rules for prohibitions and permissions. The second point of consistency check is formalizing the rules about the consistency between a newly adopted abstract obligation and the current mental states of the agent. Prior to this, we define:

**Definition 9** *A plan instance $p$ is **consistent** to the current intentions set $I$ of the agent when the effects of applying the plans specific to the current intentions are not among the negated effects of applying plan $p$.*

$$consistent(p, I) \iff \forall i \in I.(effects(\pi_i) \cap effects(p) = \emptyset$$

Where by $\pi_i$ we denote the plan instantiated to achieve intention $i$.

The types of consistency / inconsistency states between a plan and an intention are almost similar to those between a plan and the norms in the NIB:

- **strong inconsistency** occurs when all plan instantiations $p$ which satisfy the obligation $o$ are not consistent with the current intentions of the agent
- **strong consistency** occurs when all plan instantiations $p$ which satisfy the obligation $o$ are consistent with the current intentions of the agent
- **weak consistency** occurs when there exists at least one plan instantiation $p$ which satisfies the obligation $o$ and is consistent with the current intentions of the agent

### 3.2.2 Norm instantiation

We will now give the norm instantiation bridge rule, adapted from the definition given in [6].

$$\frac{ANB : \langle M, A, E, C, S, R\rangle \quad Bset : \langle B, A\rangle, \langle B, \neg E\rangle}{NIB : \langle M, C\rangle}$$

In other words, if in the ANB there exists an abstract norm with modality M about C and according to the belief-set the activation condition is true, while the expiration condition is not, then we can instantiate the abstract norm and store an instance of it in the NIB. In this way, the agent will consider the instance of the norm to be active.

In our pseudo-code description of the BDI execution loop, we will take care of the instantiation after the belief-set update and just before the desire-set update. The instantiation method should look like this:

```
instantiate(ANB, B)
{
 for all n_a = ⟨ M, A, E, C, S, R ⟩ in ANB do
 {
  if (exists(A in B) and
  not exists(E in B)) then
  {
   create norm instance n_i = ⟨ D, C ⟩ from n_a
   add(n_i, NIB)
  }
 }
}
```

This method will return the updated Norm Instance Base (NIB) containing the base of all in-force and active norms, which will further be used for the internalization process.

### 3.2.3 Solving the conflicts

When following its intentions an agent will instantiate from its set of possible plans (capabilities) $\mathcal{P} \subseteq \mathcal{L}_{\neg}$, a set of plans $\Pi(B, D)$. We call $\Pi(B, D)$ the conflict set, according to the agent's beliefs and desires. Sometimes, the actions in $\Pi(B, D)$ can lead to inconsistent states. We solve such inconsistency by choosing the maximal non-conflicting subset from $\Pi(B, D)$.

**Definition 10** *Let $\alpha \subseteq \Pi(B, D)$. $\alpha$ is a **maximal non-conflicting subset** of $\Pi(B, D)$ with respect to the definition of consequences given the belief-set B if and only if the consequences of following $\alpha$ will not lead the agent in a state of inconsistency and for all $\alpha' \subseteq \Pi(B, D)$, if $\alpha \subseteq \alpha'$ then the consequences of following $\alpha'$ will lead the agent in an inconsistent state.*

The maximal non-conflicting set may correspond to the actions required by the newly acquired norm or, on the contrary, to the actions required by the other intentions of the agent. Thus, an agent may decide either:

- to internalize a certain norm, if the consequences of following it are the better choice or
- to break a certain norm, if by 'looking ahead' it finds out that the consequences of following it are worse than following another course of actions or respecting another (internalized) norm

A more comprehensive example of how this works is presented in section 4.

### 3.2.4 Norm internalization

With the instantiation process being finished and the consistency check having been performed, the agent should now take into account the updated normative state, which will become part of its cognitions. Several previous works treat the topic of norm internalization [5] arguing which of the mental states should be directly impacted by the adoption of a norm. For this initial state of our work and taking into account the functioning of the BDI execution loop, we propose that an agent updates only its desire-set; subsequently, this will impact the update of the other mental states in the next iterations of the execution loop. We first give the norm internalization bridge rule and then provide with the adaptation of the BDI execution loop for handling this process.

$$\frac{NIB : \langle O, C1\rangle}{Dset : \langle D, C1\rangle} \qquad \frac{NIB : \langle F, C2\rangle}{Dset : \langle D, \neg C2\rangle}$$

In other words, if there is a **consistent** obligation for an agent with respect to $C1$, the agent will update its desire-set with the desire to achieve $C1$; whereas if there is a prohibition for the agent with respect to $C2$, it will update its desire-set with the desire not to achieve $C2$.

```
options(B, I)
{
 ...
 for all new norm instances n_i in NIB do
 {
  if (consistent(n_i, NIB)
  and consistent(n_i, I)) then
  { internalize(n_i, D) }
  else
  { solve_conflicts(NIB, I) }
 }
 ...
}
```

In accordance with the formalization provided, the `options()` method will look through all new norm instances and will perform consistency check on each of them. If a norm instance is consistent with both the currently active norm instances as well as with the current intentions, as defined in section 3.2.1, the norm can be internalized in the agent's desires. Otherwise we attempt to solve the conflicts as described by Ganascia in [7]. In this case, if following the

norm brings about the better consequences for our agent, the respective norm will be internalized; otherwise the agent will simply break it.

## 4 WHAT ABOUT BABY TRAVIS?

Now that we have seen how a BDI agent becomes a normative BDI, adapting to to norm occurrence, consistency check and internalization of norms, let's get back to Prof. John McCarthy's story [9]. And let's focus on the short episode with which we started this article, considering that R781 functions according to the normative BDI loop which we have just described.

R781's initial state is the following:

$ANB:$ $\emptyset$
$NIB:$ $\langle F, love(R781, Travis)\rangle$

$Bset:$ $\langle B, \neg healthy(Travis)\rangle$,
$\langle B, isHungry(Travis)\rangle$,
$\langle B, csq(\neg love(R781, x)) \succ_c csq(heal(R781, x))\rangle$
$Dset:$ $\langle D, \neg love(R781, Travis)\rangle, \langle D, isHealthy(Travis)\rangle$
$Iset:$ $\emptyset$

When R781 receives the order from his mistress he will interpret it as a normative percept and the `brf(...)` method will add a corresponding abstract obligation norm to the ANB structure. Since the mistress doesn't specify an activation condition nor an expiration condition (the two "none" values), R781 will consider that the obligation should start as soon as possible and last for an indefinite period of time. Its normative context is updated:

$ANB:$ $\langle O, none, none, love(R781, Travis)\rangle$
$NIB:$ $\langle F, love(R781, Travis)\rangle$,
$\langle O, love(R781, Travis)\rangle$

At this point, R781 will update the desire-set and will detect an inconsistency between the obligation to love baby Travis and the design rule which forbids R781 to do the same thing. Therefore, it will try to solve the normative conflict looking at the consequences of following each of the paths, given its current belief-set. In order to do so, let us take a look at the plan base of R781:

```
PLAN heal(x, y)
{
 pre: ¬ isHealthy(y)
 post: isHealthy(y)
 Ac: feed(x, y)
}

PLAN feed(x, y)
{
 pre: ∃ x.(love(x, y) ∧ hungry(y)
 post: ¬ hungry(x)
}
```

As we know from the story, R781 uses the Internet Paediatrics book to find out that if a baby is provided with love while hungry, it is more likely to accept being fed and therefore not be hungry any more. This is described by the `feed(x, y)`. Moreover, R781 also knows how to make someone healthy through the `heal(x, y)` plan, given that a-priori, that someone is not healthy. In our simplified scenario we consider that R781 knows how to do so only by feeding someone.

Instantiating its plans on both of the paths, R781 will come up with the following maximal non-conflicting sets:

$\{love(R781, Travis), feed(R781, Travis), heal(R781, Travis)\}$
$and$
$\{\neg love(R781, Travis)\}$

And since the current belief set has a rule defining that not loving someone has worse consequences than healing that person, R781 will opt for the first maximal non-conflicting subset. This means R781 will be breaking the prohibition of not loving baby Travis and will follow the action path given by the first maximal non-conflicting subset {`loves(R781, Travis), feed(R781, Travis), heal(R781, Travis)`}, while dropping the contrary. Further on, it will create an intention to achieve this state and will begin the execution of such a plan (simulating love towards baby Travis turns out to involve such plans as the robot disguising himself as human, displaying a picture of a doll as his avatar and learning what it considers to be the "motherese" dialect, mimicking the tone and the language of a mother towards her son).

## 5 CONCLUSION

In this paper we have presented an adaptation of the BDI execution loop to cope with potential normative states of such an agent. We have given a motivation for choosing the mental states model of Bratman which we have enriched with capabilities of reasoning about norms. We have investigated several previous relevant work in the domain in order to come up with a formalization of such issues as norm instantiation, norm consistency, solving consistency conflicts and norm internalization. Finally, we have provided with an intriguing study scenario, inspired from Professor McCarthy's science fiction short story "The Robot and The Baby".

Finally, it is worth noting that our research effort has been doubled by an implementation part. We have developed a first version of the normative BDI agent, using the Jade platform for agents and its extension for rational agents, Jadex [4]. The normative states (norm representation, ANB, NIB) were described by means of a small XML structured vocabulary. Thus, an agent is fully described using 3 entities: an ADF file (Agent Description File - as required by Jadex), a Java implementation of its plan base (capabilities) and an additional XML file (describing the normative states).

## 6 FUTURE WORK

Some of the limitations of our work which we would like to address in the future are related to the norm acquisition issue as well as the coherence check.

Whereas our work is providing with a very simple case of **norm recognition**, several interesting ideas have been explored based on different techniques. A good review of those as well as a description of a norm's life cycle is given in [11]. Out of those specific approaches, we will probably focus on learning based mechanisms, namely machine learning techniques and imitation mechanisms for norm recognition.

An important part of our future work will be focused on the adaptation to the **coherence theory**. At this point, it is difficult to determine incoherent states based on our architecture. As stated in [6] taking into account the coherence of norm instances will enable us to determine norm deactivation and active norms in incoherent states. As in the previously mentioned paper, we will try to base our approach on Thagard's coherence theory [12].

## REFERENCES


[1] G. Boella, L. van der Torre, and H. Verhagen, 'Introduction to normative multiagent systems', *Computation and Mathematical Organizational Theory, Special issue on Normative Multiagent Systems*, **12**(2-3), 71–79, (2006).

[2] Guido Boella, Gabriella Pigozzi, and Leendert van der Torre, 'Normative systems in computer science - ten guidelines for normative multiagent systems', in *Normative Multi-Agent Systems*, eds., Guido Boella, Pablo Noriega, Gabriella Pigozzi, and Harko Verhagen, number 09121 in Dagstuhl Seminar Proceedings, Dagstuhl, Germany, (2009). Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany.

[3] Guido Boella, Leendert van der Torre, and Harko Verhagen, 'Introduction to normative multiagent systems', in *Normative Multi-agent Systems*, eds., Guido Boella, Leon van der Torre, and Harko Verhagen, number 07122 in Dagstuhl Seminar Proceedings, (2007).

[4] Lars Braubach and Alexander Pokahr. Jadex - bdi agent systems. wiki. features, 2009.

[5] R. Conte, G. Andrighetto, and M. Campeni, 'On norm internalization: a position paper', EUMAS, (2009).

[6] Natalia Criado, Estefania Argente, Pablo Noriega, and Vicente J. Botti, 'Towards a normative bdi architecture for norm compliance.', in *MALLOW*, eds., Olivier Boissier, Amal El Fallah-Seghrouchni, Salima Hassas, and Nicolas Maudet, volume 627 of *CEUR Workshop Proceedings*. CEUR-WS.org, (2010).

[7] Jean-Gabriel Ganascia, 'An agent-based formalization for resolving ethical conflicts', Belief change, Non-monotonic reasoning and Conflict resolution Workshop - ECAI, Montpellier, France, (August 2012).

[8] Martin J. Kollingbaum and Timothy J. Norman, 'Norm adoption and consistency in the noa agent architecture.', in *PROMAS*, eds., Mehdi Dastani, Jrgen Dix, and Amal El Fallah-Seghrouchni, volume 3067 of *Lecture Notes in Computer Science*, pp. 169–186. Springer, (2003).

[9] John McCarthy, 'The robot and the baby', (2001).

[10] Anand S. Rao and Michael P. Georgeff, 'Bdi agents: From theory to practice', in *In Proceedings of the First International Conference on Multi-Agent Systems (ICMAS-95*, pp. 312–319, (1995).

[11] Bastin Tony Roy Savarimuthu and Stephen Cranefield, 'A categorization of simulation works on norms', in *Normative Multi-Agent Systems*, eds., Guido Boella, Pablo Noriega, Gabriella Pigozzi, and Harko Verhagen, number 09121 in Dagstuhl Seminar Proceedings, Dagstuhl, Germany, (2009). Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany.

[12] Paul Thagard, *Coherence in Thought and Action*, MIT Press, 2000.

[13] Michael Wooldridge, *An Introduction to MultiAgent Systems*, Wiley Publishing, 2nd edn., 2009.