Agents dealing with Norms and Regulations

Christian Kammler[0000−0002−8552−713X], René Mellema[0000−0002−4138−937X], and Frank Dignum[0000−0002−5103−8127]

Umeå Universitet, Umeå, Sweden,
{christian.kammler, rene.mellema, frank.dignum}@umu.se

Abstract. Norms influence behaviour in many ways. In situations such as the COVID-19 pandemic where the effect of policies on the spread of the virus is evaluated, this leads to disputes about their effectiveness. In order to build agent-based social simulations that give proper support for this evaluation process we need agents that properly deal with norms. In this paper we present a new agent deliberation architecture that takes more aspects of norms into account than traditional architectures have done. Dealing properly with norms means that agents can reason through the consequences of the norms, that they are used to motivate and not just constrain behaviour, and that the agents can violate the norm as well. For the former we use the ideas of perspectives on norms, while the latter is enabled through the use of values. Within our architecture we can also represent habitual behaviour, context sensitive planning, and through the use of landmarks, reactive planning. We use the example of a restaurant-size based restriction to show how our architecture works.

Keywords: Social Simulation · Normative Reasoning · Values · Needs

1 Introduction

Norms influence behaviour in many ways, and on many different levels [7]. This makes it challenging for policy makers and other decision makers to create policies (which we see as special types of norms), as has been shown by the current COVID-19 pandemic. Here, heavy disputes arose on the effectiveness of the policies that had been introduced to combat the spread of the virus with regards to the effects that those policies had on the people. To tackle this, and to support policy and other decision makers, agent-based social simulation can be a powerful tool [13,15,17,24].

To build agent-based social simulations that give this support, the agents in the simulation need to show realistic human-like behaviour. Part of this is that they need to be able to properly reason with norms, i. e. seeing them as more than just restrictions on behavior. This requires not only that they can see if the consequences of following or breaking the norm are desirable for them, but also how they interact with other parts of their reasoning process. In particular, this means that norms cannot just be seen as simple restrictions, but we also need to take their motivational aspects into account [5,21]. Furthermore, a new policy often interacts with existing social structures (e.g. habits, social practices, other norms, goals) as well, which might cause them to change. Since norm breaking is an important part of norm dynamics [2, Chapter 5], the architecture will also need a flexible way to deal with norm violations that the other agents can react to. What this means for an architecture is that the norms cannot just be added as a module, but it has to be interwoven into the whole decision making process.

Therefore, we are presenting a deliberation architecture with social reasoning as its focus which is motivated by [10]. To enable agents to reason through the consequences of the norms, we use the concept of perspectives on norms that we introduced in [17]. We thereby also take the motivational aspects of norms into account, and enable agents to reason about norm violations. Within our architecture we use context sensitive planning, so agents can adjust their goals and plans reactively with the help of landmarks, based on their current context. Having context sensitive planning also allows us to model habitual behaviour by enabling the agents to recognise whether they are in a familiar context.

While there has been work on norms in the agent community, including work on norm violation behaviour, such as [3,6,7,10,21,26], they do not address all the requirements for properly dealing with norms. Problems with architectures [3,21] for example are that norms are only seen as obligations, and not motivations. Furthermore, they are not allowing an agent to reason "consciously" about a specific norm to determine by themselves if they want to violate the norm or not.

2 Elements of a normative agent architecture

To have an agent architecture which allows agents to take the motivational aspects of norms into account, as well as being able to reason "consciously" about breaking a norm, we need a variety of elements which we are going to describe in more detail in this section.

With these elements we can then also take norms into account which are active later in the day in a different context. For example: Going to a bar in the evening after work with colleagues (norm) means that we might leave the car at home in the morning, so we do not drink and drive (norm, i. e. not violating the no alcohol when driving norm).

Such ahead planning is very complex for an agent compared to humans. For us it is obvious and thought of as one thing, but for agents it is comprised of multiple things where each step of the day requires new deliberation of the agent. To enable agents to have more of this long-term planning, we are using the concepts of context and plan patterns in our architecture.

We first talk about how we use norms in more detail, and then talk about the different elements concerning the agent itself.

2.1 Norms

Norms describe 'normal' behavior and aim at assuring the interests and values of groups or the society as a whole [19]. Furthermore, they are not only constraints on behavior. They can also motivate [5,21] and trigger new behavior [19]. Also, norms promote and demote values. This makes it possible for agents to deliberate if the norm is important for them or if they want to violate it. We take the idea that norms promote values from the real world, as norms are created with a purpose. For example: The norm to wear a face mask is created to reduce the spread of the coronavirus and thus, promotes safety.

To formulate norms, we use our ADICDlRO framework [17]. The elements in the ADICDlRO framework are as follows.

`A` defines the agent group that the norm is applicable for.

`D` is the deontic part of the norm, and together with the aim `(I)`, they form the {`fulfilment, violation`} condition of the norm.

`C` defines the contexts in which the norm is active and not active, therefore representing the {`activation, deactivation`} condition of the norm.

The deadline element `(Dl)` states when the norm is supposed to be fulfilled.

The repair part `(R)` of the framework defines the action(s) to 'undo' the potential breaking of the norm, and the 'Or else' `(O)` specifies the punishment of the norm violation.

2.2 Needs

Needs are motivators that drive us constantly to perform behaviour that satisfies them. We make use of this in our architecture in the form of *long-term goals*. Long-term goals are abstract in nature and can be seen as the root element of a goal-plan tree or a goal-goal tree. They have a variety of sub-goals which we call *short-term goals*, which present steps towards fulfilling the long-term goal. We assume that the long-term goals are never achieved. They are ideals that can be seen as points on a horizon, but have no concrete state that can be true or false. A way to implement this is using a homeostatic model, see e.g. [10,14]. The needs are represented as containers in the architecture that deplete over time, and can be filled by achieving subgoals related to the specific long-term goal. The priority in values (see next section) determines thereby the urgency to satisfy the need for taking steps towards a certain long-term goal.

Furthermore, they are a fixed, pre-determined set in our architecture. To determine them, we can, for example, ask the stakeholders that are represented in the specific model.

2.3 Goals & Values

Values are used to evaluate behaviour and events [17,19,21], and function as standards for the evaluation criteria [23,25]. We use values in the architecture to determine which goals are important to achieve, which actions are most desirable to take to achieve the goals, and which needs are more urgent to satisfy. Each perspective has its own priority of values [17] which are constant over time during the whole simulation.

Goals are states that the agent wants to achieve. Goals in our architecture are, called *short-term goals*. These are representing steps towards the desired long-term goals of the agent, and are generated in the goal generation step in our architecture, and can be seen as child nodes in a goal-plan tree or a goal-goal tree.

The formation of these short-term goals is influenced by the agent's priorities in values. The norms influence the short-term goals in the way that some goals might be forbidden due to a prohibition, or an obligation is in conflict with the goal, and therefore achieving the goal will be in violation of the obligation, over which the agent can deliberate.

To achieve a goal, specific actions are taken by the agent. This can be done by one action or a sequence of actions might be necessary to achieve the goal. When the goal is reached, the associated needs will be satisfied and connected values will be promoted and demoted. Note here, that a short-term goal (`StG`) can be associated with more that one need, i.e. it is contributing to more than one long-term goal. Previously generated short-term goals can be used in the same context again.

2.4 Perspectives

People use their own motivations and have different goals, plans and capabilities. Therefore, we are using the concept of *perspectives* [17] to connect norms to individual behavior. Thus agents only focus on the parts of the norm which are relevant for them, and only those parts are affecting their behaviour, utilising the ADICDlRO framework of norms [17]. To address this, perspectives have the following elements [17, p.142]: "A perspective is specified by goals (G), available actions (A), effects of those actions (EoA), social affordances (SocAffs), and priorities in values (PrioV)."

The *PrioV* determines which kind of behaviour is important to us and which incentives motivate us the most. In terms of goal selection and formation, they also help us to select the goals which are most desirable for us whereby goals are specific to a perspective, as everyone has specific goals in their minds which fits their needs. Furthermore, we are distinguishing two kinds of actions, the (classical) physical actions, and social actions. Social actions are the social effects of the physical action performed which are the *SocAffs* in our definition. Note that $SocAffs \neq EoA$ as EoA are the physical effects of the action A, and *SocAffs* the social effects of the physical action, i.e. the social action.

2.5 Actions

To achieve the goals, and satisfy the needs, the agent has a set of available actions. We differentiate hereby between physical actions and social actions [17].

Physical actions are the classical actions when we think of actions. They may require one or more objects (o_i, o_j, ...), with $o_i, o_j \in O$, to perform the action, whereby O is the total set of n objects $O = \{o_1, ..., o_n\}$ that we have in our simulation. Actions also have a pre-condition that needs to be met so the action can be executed. The actions also have an effect, the result of the action [17]. Furthermore, they are specific to a perspective, as different groups have different actions available to them.

Social actions are defined by the social effects of the physical actions (including all physical actions that the agent can perform= performed [17]. They call this the *"social affordances (SocAffs) of an object (o_i)"* [17, p.6]. In general, we relate them to the purpose that an object fulfils for a person or a group [8]. People have different purposes for the same object based on their perspective.

Norms influence actions in the way that some actions might become obligatory or forbidden. Identifying the actions that are affected by a norm is done using the object of the norm (I_{Object}), by checking if $o_i == I_{Object}$ holds, where o_i is the object required by the action. We note that we assume here that an action always requires an object.

Finally, actions promote and demote values. We note here however, that a sequence of actions to achieve a goal can promote and demote different values in the end than the individual actions in that chain. This happens because achieving goals can also promote and demote values.

2.6 Context

Keeping all the different goals, actions, and norms in mind, greatly increase the complexity of the decision making process. To constrain this, we use the notion of *context* [28], which contains all the information that is currently *relevant* [4]. For example, this can mean that the currently active norms are part of the context, but the ones that are currently deactivated need not be. Thereby making the deliberation faster for the agent since it needs to take less things into account. Furthermore, keeping track of the context also helps in detecting when the goal might no longer be achievable.

To determine what information is relevant, we base ourselves on the work of [4,27]. Here, relevancy of information is defined in terms of the *goal* that the agent has. This means that the context should contain information that:

1. is about the content (proposition) of the goal;
2. is about the relationships between the goals;
3. is about the conditions for the pertinent actions.

What exactly this information is depends on the exact simulation, but there are some general rules for this. Firstly, the goal itself is included. At least the active norms are incorporated based on their active contexts (`C`), since they can allow or block certain actions. To further determine what information is relevant, we check which information is needed to execute the current plan. This would also be a good place to bring in expert knowledge. If we know which plans and actions are available, and we know which information these need to determine their pre-conditions, we can use this to determine what information is relevant towards the goal.

To determine whether the agent has to switch contexts, we can set up *acceptable* ranges for the parameters. This can be done using a similar process as determining which parameters are relevant, but now keeping in mind not just what parameters are relevant, but also the values that they need to have. Then in order to check if we are still in the same context, we can compare the current value of the relevant parameters to these ranges. Once a parameter has left that range, the context has changed too much, and the agent has to replan or change their goal entirely.

To make it easier to talk about the context, and to link actions and plans to specific contexts, we *label* contexts. Actions/plans are linked to these context labels instead of having to be linked to more detailed context descriptions.

Using the goal as the cornerstone in the context detection, does mean that we are making the assumption that there will always be a goal. This might not be the case when a goal has just been achieved. In this case, we assume that there is a default context that defines the relevant information. This should include at least its current needs, the roles the agent fulfils, and the time/location.

2.7 Plan patterns

When designing agent behaviour, there is often a balance that needs to be made between proactive, goal directed behaviour, and reactive situational behaviour. Our solution to this problem is through the use of plan patterns [8]. Plan patterns are sets of sequences of actions, defined in terms of *landmarks* [11]. Landmarks represent states that need to be achieved along the way, without specifying how they are achieved. In this way, they can be seen as sub-goals in a plan, but achieving them does not have the same effect that achieving a goal has. Using this system, the agents can make a rough plan to follow, and then fill in the details depending on the current context.

This requires some changes to the traditional planning paradigm. Instead of giving full fledged plan, the planner now needs to produce plan patterns instead. As long as there is a mechanism for chaining landmarks together, this is not a big change. The context can also help here in speeding things up, since within a certain context, only certain landmarks might be available. Since we are trying to mimic human behaviour, which is often repetitive (i.e. habitual) but not always optimal, we can also store plan patterns that were used before in the same context. Thereby giving us the possibility to simulate habitual behaviour.

After a plan pattern has been selected, the landmarks need to be filled in. To do this, a lower level plan is selected, where the plan can be just one action. This is done in a similar manner as the selection of an initial plan, but now the landmark is what is being planned towards instead of the goal. Here again the context, which might be slightly different from the context when the initial plan was made, can be used to limit or make available certain options in the planning.

To account for desired actions becoming not executable anymore during plan execution, we mark actions with a *purpose* for why they have been selected. With this, the agent can find an alternative action or plan pattern that serves the same purpose. To avoid the alternative action/plan becoming too long, we limit its length before forcing the agent to create a new plan to reach the goal.

2.8 Norm Breaking

So far we have discussed norms and described the different mechanisms we need to deal with the different sources of motivations. To decide between these different sources, we use the notion of plan/action *acceptability*.

Acceptability here means that for each plan/action selected by the agent which is in conflict with a norm, the agent needs to find executing the plan/action acceptable with respect to breaking the norm. We use values for this decision. Since each norm pro-/demotes different values, the agent can compare what values it deems important with those. If the norm demotes values it finds important, then it should break the norm. Other indicators can also be taken into account, such as governmental trust (in the case of legal norms) or its social identity.

If the agent does not find the norm acceptable to break, then it tries to find an alternative plan/action. This can also happen if a norm applies that it had not anticipated would be active, in which case it would end up in the "find alternative action" step. If no alternative is available, the agent tries to generate a new plan in the "planner" to achieve the goal. If that also fails, the agent marks the goal as not achievable, and generates a new goal.

However, if it does find the norm acceptable to break, then it can break the norm. In that case, the norm violation would have to be detected by the software, and in the next time step, both the reparation (`R`) and 'or else' (`O`) parts of the norm have to be activated as well.

2.9 Resulting Agent Architecture

Putting the things together from the previous sections, we get the architecture in Figure 1. The starting point of the deliberation is the *context detection*, and the end point is the *execute action*. The solid arrows indicate the next step, and the dotted arrows of needs, values, and norms are indicating the influence of those elements.

To show how our architecture works in practice, we use a restriction on the number of allowed guests based on the size of the restaurant, similar to our last paper [17]. Given the current COVID-19 pandemic, only a certain amount of guests are allowed inside a restaurant to increase the distance between the guests in the restaurant to promote the value safety. Looking at this norm only as a restriction is not enough and only leads to the trivial insight that the restaurant has too little income and is eventually going bankrupt, because not enough guests are allowed in the restaurant. This view is too limited, as different people are in different ways, such as the guests and the restaurant owner.

Making money with their restaurant is the most important need for **restaurant owner**, given their highest prioritised values are power and achievement. At their restaurant they get informed about the new norm (context detection). The new norm hinders the restaurant owner to have as many guests as possible in their restaurant (goal), and therefore negatively impacts the need of making money with the restaurant. Consequently, the context changed too much for the restaurant owner. Given a high priority in the values of power and achievement,

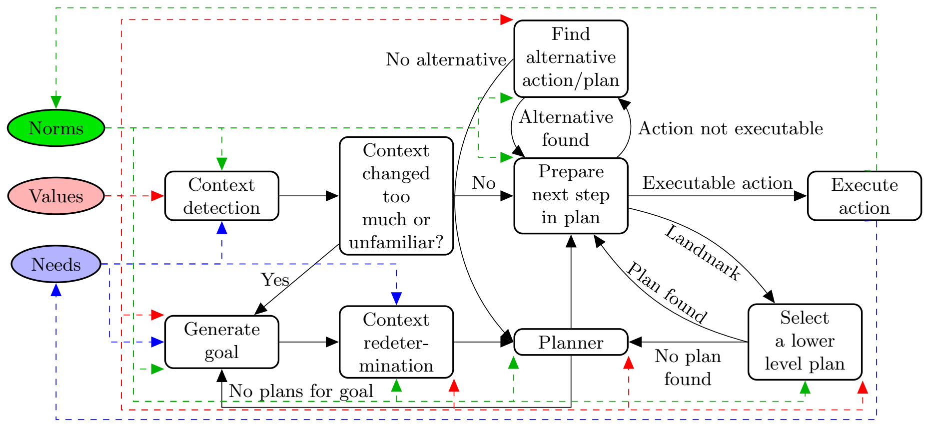

Fig. 1. The resulting agent architecture

it is very important for the restaurant owner to make as much money as possible with their restaurant. Therefore, they are now looking for a way (generate goal) to combat the loss of income which is a consequence of the new norm. The restaurant owner decides to lower their variable costs of the restaurant (new goal), given their priority in power and achievement. To achieve this goal, the restaurant owner determines their new context (which contains information about food prices, contracts with suppliers and staff, norms such as minimum wage or which meat is allowed in food), and forms the plan (in the planner) to use cheaper ingredients in some of their meat dishes. The next step in the plan is a landmark (collecting all information about the meat dishes served in the restaurant), so the restaurant owner selects a lower level plan with all the dishes in menu order. They decide to use cheaper meat in the dishes where a lot of sauce and spices are used (such as goulash), because there guests will not taste the change as much.

While for many non regular restaurant **guests**, the norm of limiting the amount of guests in a restaurant affects their behaviour strongly and they might decide to reserve a table before hand (similar deliberation as the restaurant owner), for some of the regular guests the norm does not change their context too much. Consider a group of friends who goes to the same restaurant every Friday after work. They have their specific table reserved every week. Therefore, the introduction of the new norm does not change their context too much for them. However, this time their desired beer is not available due to shortages. Therefore, they decide to have a different beer (find alternative action) with a similar taste. In another instance, a new colleague decides to join their regular table. While this changes the context too much for the group (given a new person is introduced), they still decide to go to the restaurant trying to see if there is space. After arriving at the restaurant, the restaurant owner tells them that the restaurant is packed and no free spaces are available. However, the restaurant owner has the highest priority in the values power and achievement, and the value of safety, which the norm promotes, is not important for them. Therefore, the need of making money with the restaurant is more important for the restaurant owner than the strong adherence to the norm. Therefore, they decide to bend the norm, and have a few more guests than allowed.

As a result, the example shows that the same norm has different effects for different perspectives. The restaurant owner is affected differently than the guests. Another implication which we did not talk about in the example above is the reaction of the guests to the restaurant owner's decision. Some guests might learn that the restaurant owner is using cheaper meat in some of their dishes. This might not change the context for some guests, as they do not eat the affected dishes. However, for the people who eat those dishes it changes the context too much. Also here, for some guests it might not be as important as the restaurant place itself, as sustainability is not so important to them compared to the eating place. However, for guests for whom sustainability has a very high or the highest priority, they are not going to that restaurant anymore. Therefore, modelling norms as just a restriction on behaviour is not adequate to model all the consequences of the introduction of a new regulation or norm.

3 Related work

Having shown our architecture, we can now discuss why existing approaches, such as the ones mentioned in the introduction are not suitable for our purposes.

3.1 BDI, its Extensions and Utility Functions

One of the big downsides of using BDI [22] and its extensions, such as [3,20], is that they tend to not take into account the full agent deliberation cycle, in particular norm importance in planning. Besides this, they also tend to not take into account the context that an agent is in and how that interacts with decision making, nor other motivational reasons for norm following/breaking besides the agents desires/intentions.

While there are works that address the first concern, such as [21], these tend to not address the reasons why an agent might want to break or follow a norm. As described in Chapter 11 in [2], agents might have various different reasons for breaking norms, not all of which are depent on their other desires or intentions, but sometimes simply how much they like the norm, or who instituted it. Specifying when and how an agent should find breaking a norm "acceptable" in a BDI framework has to our knowledge not yet happened.

The context problem is easiest seen with BOID [3]. In the BOID architecture, agents are of a certain type to solve conflicts between different types of motivations, such as obligations and desires. This type does not differ over the lifetime of the agent. However, in real life a lot of the decisions that we make are dependent upon the context in which we make them, which includes our reasons for breaking or following a norm [2]. These kinds of factors are hard to include in the BOID architecture, but are accounted for in our proposal.

Approaches based on utilities (including sanctions and rewards), such as [20] or the EMIL-A architecture by [6] with its EMIL-I-A [26] extension, solve some of these issues, but it still has a few drawbacks as pointed out by [12]. Usually, a utility function is conceptualised in such a way that norm breaking behaviour results in utility penalties [12]. Furthermore, these functions only function well in a static environment [9]. Furthermore, given the same set of alternatives, the choice is always the same [9], even though the context might have changed.

3.2 Social Reasoning

[10] in their ASSOCC project shows the benefits of having an architecture that is capable of social reasoning. The results of their COVID-19 simulations seem to be far more close to actual human behavior than most other approaches. At the core of the agent's decision making process is a homeostatic needs model, which is described in detail in [16].

Each of the agents has several needs, which are modelled as containers (based on [14]). These containers deplete over time if no action towards satisfying that need is taken [16]. The actions can also cause the draining of the container by removing some satisfaction. An action can thus influence multiple needs at the same time. For example, going to a park with friends satisfies the need of belonging, as one is with their friends. However, in the COVID crisis it also has a negative impact on risk-avoidance, as one is outside around people [16]. The decision which action to take next is made based on the combined need satisfaction over all needs that the action would provide, if executed.

A major issue which makes the ASSOCC architecture not usable for our purposes is that norms are only modelled implicitly, in their effect on the need satisfaction of an action. This makes changing the norms very difficult, as each of the parts of the model has to be inspected to see if this part of the code is affected by the desired change. Furthermore, this also means that the agents cannot reason about the norms themselves when determining whether they should violate them. These aspects make it is not suitable for our more general architecture.

4 Discussion

Our architecture provides a deliberation that can be very fast and simple when plan (patterns) are available for the current context, i. e. mimiking habitual behavior (the middle line from context detection to action execution in Figure 1. The contextual planning involving values, goals and norms is only used if no ready plan is available. This makes the agent very efficient in all standard situations, while taking all the social aspects into consideration when it is needed.

It is clear that agents with different perspectives react on different aspects of norms. Thus, modelling norms as just a restriction on behaviour would not be adequate to model all the consequences of the introduction of a new norm. We showed this in our example discussion of the architecture even if different groups share the same values (in our case sustainability).

Since our framework allows for norm violation, there are also multiple directions for future work in this area. The first of these is that our notion of "acceptable plan" can be expanded. For example, most norms are put in place by some form of institution, either a formal institute such as a state, or an informal one such as a culture. One thing that could be taken into account is how much an agent might trust such an institution, since this also has a large effect in human societies [18]. Another aspect which we did not discuss in the example is the reaction of the guests to the norm violation of the restaurant owner (having a few more guests than allowed). While for some guests this does not matter, others might leave, because they feel that their safety is endangered.

Norm internalisation [1], which is an important aspect of normative reasoning, also needs to be explored. A norm that is internalised is harder to violate than a norm that is not internalised. In general a norm can be said to be internalised if the norm is in line with the agents values. Thus an agent would choose the behaviour that follows the norm even if that norm would not be there. Having the norm has more effects as we have seen, and thus, the norm is an important driver of behaviour.

For our immediate future work, we are going to implement our proposed architecture, based on the formalisation which we omitted in this paper to not distract from the main goal of the paper to present an agent architecture capable of incorporating different perspectives on norms, the motivation components of norms, and enabling agents to explicitly reason about norm violations.

References

1. Axelrod, R.: An evolutionary approach to norms. American Political Science Review **80**(04), 1095–1111 (Dec 1986). https://doi.org/10.2307/1960858
2. Brennan, G., Eriksson, L., Goodin, R.E., Southwood, N.: Explaining Norms. Explaining Norms, Oxford University Press, Oxford (2013)
3. Broersen, J., Dastani, M., Hulstijn, J., Huang, Z., van der Torre, L.: The boid architecture: conflicts between beliefs, obligations, intentions and desires. In: Proceedings of the fifth international conference on Autonomous agents. pp. 9–16. Association for Computing Machinery, New York, NY, USA (2001)
4. Castelfranchi, C.: Guarantees for autonomy in cognitive agent architecture. In: Wooldridge, M.J., Jennings, N.R. (eds.) Intelligent Agents, LNCS, vol. 890, p. 56–70. Springer, Berlin, Heidelberg (1995)
5. Castelfranchi, C., Dignum, F., Jonker, C.M., Treur, J.: Deliberative normative agents: Principles and architecture. In: Jennings, N.R., Lespérance, Y. (eds.) Intelligent Agents VI LNCS 1757. pp. 364–378. Springer Berlin Heidelberg (2000)
6. Conte, R., Andrighetto, G., Campennl, M.: Minding norms: Mechanisms and dynamics of social order in agent societies. Oxford University Press (2014)
7. Dignum, F.: Autonomous agents with norms. AI and law **7**, 69–79 (1999)
8. Dignum, F.: Interactions as social practices: towards a formalization (Sep 2018), `https://arxiv.org/abs/1809.08751v1`
9. Dignum, F.: Foundations of social simulations for crisis situations. In: Dignum, F. (ed.) Social Simulation for a Crisis: Results and Lessons from Simulating the COVID-19 Crisis, pp. 15–37. Springer International Publishing, Cham (2021)

10. Dignum, F. (ed.): Social Simulation for a Crisis: Results and Lessons from Simulating the COVID-19 Crisis. Springer International Publishing, Cham (2021)
11. Dignum, V., Dignum, F.: Coordinating Tasks in Agent Organizations, vol. 4386, p. 32–47. Springer, Berlin, Heidelberg (Aug 2006)
12. Fagundes, M.S., Ossowski, S., Cerquides, J., Noriega, P.: Design and evaluation of norm-aware agents based on normative markov decision processes. International Journal of Approximate Reasoning **78**, 33–61 (2016)
13. Gilbert, N., Ahrweiler, P., Barbrook-Johnson, P., Narasimhan, K.P., Wilkinson, H.: Computational modelling of public policy: Reflections on practice. JASSS **21**(1), 14 (2018). https://doi.org/10.18564/jasss.3669
14. Heidari, S., Jensen, M., Dignum, F.: Simulations with values. In: Advances in Social Simulation: Looking in the Mirror, pp. 201–215. Springer, Cham (2020)
15. Jager, W., van der Vegt, G.: Management of complex systems: Toward agent-based gaming for policy. In: Policy Practice and Digital Science, vol. 10, p. 291–303. Springer International Publishing, Cham (Jun 2015)
16. Jensen, M., Vanhée, L., Kammler, C.: Social simulations for crises: From theories to implementation. In: Dignum, F. (ed.) Social Simulation for a Crisis, pp. 39–84. Springer, Cham (2021)
17. Kammler, C., Dignum, F., Wijermans, N., Lindgren, H.: Changing perspectives: Adaptable interpretations of norms for agents. In: Van Dam, K.H., Verstaevel, N. (eds.) Multi-Agent-Based Simulation XXII. pp. 139–152. Springer, Cham (2022)
18. Marien, S., Hooghe, M.: Does political trust matter? an empirical investigation into the relation between political trust and support for law compliance: does political trust matter? European Journal of Political Research **50**(2), 267–291 (Mar 2011)
19. Mellema, R., Jensen, M., Dignum, F.: Social rules for agent systems. In: Aler Tubella, A., Cranefield, S., Frantz, C., Meneguzzi, F., Vasconcelos, W. (eds.) Coordination, Organizations, Institutions, Norms, and Ethics for Governance of Multi-Agent Systems XIII. pp. 175–180. Springer, Cham (2021)
20. Meneguzzi, F., Rodrigues, O., Oren, N., Vasconcelos, W.W., Luck, M.: Bdi reasoning with normative considerations. Eng. Appl. Artif. Intell. **43**, 127–146 (2015)
21. Panagiotidi, S., Alvarez-Napagao, S., Vázquez-Salceda, J.: Towards the Norm-Aware Agent: Bridging the Gap Between Deontic Specifications and Practical Mechanisms for Norm Monitoring and Norm-Aware Planning. In: Balke, T., et al. (eds.) COIN, LNCS, vol. 8386, pp. 346–363. Springer, Cham (2014)
22. Rao, A.S., Georgeff, M.P., et al.: Bdi agents: From theory to practice. In: Lesser, V., Gasser, L. (eds.) Proceedings of the First International Conference on Multiagent Systems. vol. 95, pp. 312–319. MIT Press, Cambridge, MA, USA (1995)
23. Rohan, M.J.: A rose by any name? the values construct. Personality and social psychology review **4**(3), 255–277 (2000)
24. Rosewell, B.: Complexity science and the art of policy making. In: Johnson, J., Nowak, A., Ormerod, P., Rosewell, B., Zhang, Y.C. (eds.) Non-Equilibrium Social Science and Policy, p. 159–178. Understanding Compl. Sys., Springer, Cham (2017)
25. Schwartz, S.H., Bilsky, W.: Toward a universal psychological structure of human values. Journal of personality and social psychology **53**(3), 550 (1987)
26. Segura, D.V.: Social Norms for Self-policing Multiagent Systems and Virtual Societies. Universitat Autonoma de Barcelona (2012)
27. Vecht, B.v.d.: Adjustable autonomy: controling influences on decision making. Universiteit Utrecht, Utrecht (2009)
28. Zimmermann, A., Lorenz, A., Oppermann, R.: An operational definition of context. In: Kokinov, B., Richardson, D.C., Roth-Berghofer, T.R., Vieu, L. (eds.) Modelling and Using Context. LNCS, vol. 4635. Springer Berlin Heidelberg (2007)