Extensions to the BDI model: Obligations and Motivations

Michael Sewell

No Institute Given

1 Obligations

To be able to support collaborative behaviour in multi-agent systems, it is essential to provide individual agents with various forms of social awareness. Standard BDI agents operate on an isolated, non-social level, caring about their own desires, beliefs and intentions. Ideally, we want agents to be able to reason about the beliefs, desires and intentions of other agents, as well. This is the idea behind *obligations* [Dig+00].

In 2001, Broersen et al. presented the *BOID* (Beliefs-Obligations-Intentions-Desires) architecture, an framework that uses Deontic Logic to extend the standard BDI model by adding obligations (and later, *norms*) to the agent reasoning process [Bro+01]. Deontic Logic is a form of modal logic where the modal operators concern aspects such as permissions and obligations [Fis+07].

What follows is a brief overview of the different mental attitudes in the BOID architecture.

Beliefs
Beliefs represent an agent's knowledge and knowledge defaults, meaning they are a representation of the information an agent has about the current state of the world. Observations (percepts) are processed and turned into beliefs as they arrive [Bro+02; DKS02].

Beliefs also act as a filter for *realistic* agents, explained further below, by allowing only desires and obligations that are consistent with the agent's beliefs to be considered during the goal-finding process [Dig+00].

Obligations
Obligations are the new mental attitude that has been added by BOID to the standard BDI architecture. Obligations represent prohibitions and permissions, and as such they reflect the social nature of individual agents or groups of agents. They arise from interactions and relationships between agents, for example an agent buying an item from another agent would receive the obligation to pay for the item [Bro+02].

To remain autonomy, agents can violate obligations if they choose to. This can happen when an agent values its desires over its obligations, for example. In the shop example, an agent may be obligated to pay for an item, but he desires to keep his money, so he leaves the shop without paying [DKS02].

It is possible for obligations of different sources (different agents, usually) to conflict. For example, an agent may receive the obligation to perform action p from one agent and the obligation to perform $\neg p$ from another agent [Dig+00].

Intentions
Intentions in BOID remain as they are in the standard BDI framework. They represent the commitments and plans of an agent.

Desires
Desires represent the wishes and wants of an agent. As they model the internal motivations and long-term preferences of agents, desires can be considered to resemble an emotional process. Generally, an agent can select its goals, but not its desires, and desires can even conflict with each other (as opposed to goals) [DKS02].

1.1 More on obligations

Obligations are often associated with penalties (or *sanctions*) that apply when they are not fulfilled. In the shop example, if the agent leaves without having paid for the item, he might get punished. There may be some mechanism, organization or other body which is responsible for enforcing the penalty, for example the police may deal with thieving agents.

Usually, obligations on actions to perform carry with them a time aspect, which indicates that if the action has not been performed before a certain deadline, the agent will receive his punishment. In the shop example, the merchant agent expects his buyers to pay for bought items now and not later[DKS02].

1.2 Authorization

Authorization is the counterpart of obligation. It descibres the same dependency between agents as an obligation, but from the point of view of the other agent. Generally, if an agent has the authorization to perform some action it has some basis on which to justify it. Example: if the shopkeeper agent is *authorized* to demand payment from a shopper agent, then the shopper agent is *obligated* to to do so. This is not the case if the shopkeeper agent is *not* authorized [Dig99].

1.3 BOID agents

Different agent types are distinguished in the BOID architecture, where the type of the agent depends on the *priority* of the mental attitudes (Beliefs, Obligations, Intentions, Desires) in the agent's goal generation routine. There are *realistic*, *social*, *selfish* and *stable* agents, for example.

The priority relation on the mental attitudes is used as a means of conflict resolution. A mental attitude conflicts with another mental attitude if both

cannot be used to generate a consistent set of goals. To resolve such a conflict, one attribute has to be chosen to override the other. If there is a conflict between e.g. a desire and an obligation ("I *desire* to quit my job, but I am *obligated* to pay my bills"), the prioritized attitude would "win out" and decide the next action to perform. Strictly speaking, the preference relation is defined across rules rather than attitudes, and whenever a new rule is derived in the goal generation process, the new set of rules is returned to the beginning of the process in a feedback loop [Bro+02].

The order of conflict resolution determines the agent type. In *realistic* agents, the belief component overrules any other component ($B > OID$). Agents where intentions override desires and obligations ($I > DO$) are called *stable* or *simple−minded*, as they do not remove intentions they have committed to, even if their desires or obligations change. If desires override obligations ($D > O$), the agent is called *selfish*; if obligations override desires ($O > D$), the agent is called *social*. Agent types can be combined: a $I > O > D$ agent would be called stable and social. An agent type is called *primitive* if it contains only one constraint, *complete* if it induces a total strict ordering on the components [Bro+02; Bro+01]. A list of primitive agents is given in Figure 1.

Constraints		Agent type
B ≻ O	(O ≻ B)	Realistic relative to obligations (dogmatic)
B ≻ I	(I ≻ B)	Realistic relative to intentions (over-committed)
B ≻ D	(D ≻ B)	Realistic relative to desires (wishful thinker)
O ≻ I	(I ≻ O)	(Un)Stable relative to obligations
O ≻ D	(D ≻ O)	Social (selfish)
I ≻ D	(D ≻ I)	(Un)Stable relative to desires

Fig. 1. Twelve primitive agent types. Table taken from Broersen et al. [Bro+02].

1.4 Realistic agents

Realistic agents are the most common agent type, and intuitively it is easy to understand that is probably a good idea for an agent to not decide on goals that it knows it can't achieve. In realistic agents, a conflict between a belief and a prior intent means that an intended action should no longer be executed due to the changing environmend, and that it should be removed from the set of intentions. Similarly, a conflict between a belief and an obligation or a desire means that a violation has occured. A list of six complete, realistic agent types is given in Figure 2.

If beliefs overrule desires, we say that the agent is prone to *wishful thinking* [Bro+01].

1.5 Working with rules in BOID

Rules are propositional formulas of the type $a \underset{X}{\rightarrow} b$, with b a conjunction of literals and $X \in \{B, O, I, D\}$. They define the inference steps to make in the

Constraints	Agent type
$B \succ O ; O \succ I ; I \succ D$	Realistic, unstable-O, stable-D, social
$B \succ O ; O \succ D ; D \succ I$	Realistic, unstable-O, unstable-D, social
$B \succ I ; I \succ O ; O \succ D$	Realistic, stable-O, stable-D, social
$B \succ I ; I \succ D ; D \succ O$	Realistic, stable-O, stable-D, selfish
$B \succ D ; D \succ O ; O \succ I$	Realistic, unstable-O, unstable-D, selfish
$B \succ D ; D \succ I ; I \succ O$	Realistic, stable-O, unstable-D, selfish

Fig. 2. Six complete realistig agent types. Table taken from Broersen et al. [Bro+02].

goal resolution process. For $a \underset{B}{\rightarrow} b$, we say that the agent *believes* ($\underset{O}{\rightarrow}$: is *obligated*, $\underset{I}{\rightarrow}$: *intends*, $\underset{D}{\rightarrow}$: *desires*) that as a consequence, b is a goal.

Each component in the BOID goal generation process receives a set of rules (called an *extension* as the input and outputs another extension [Bro+01; Bro+02].

There might be other rules blocking the inference, such as $\top \underset{X}{\rightarrow} \neg b$. BOID uses Default Logic, where rules are only applied if they do not lead to an inconsistency, although goal sets may conflict with each other [Bro+02].

1.6 BOID goal generation

Let B, O, I, D be sets of rules and ρ a priority function from $B \cup O \cup I \cup D$ to the integers. The idea is that in case of multiple applicable rules, the one with the highest ρ value is applied. To generate unique goal sets, ρ must be such that $\rho(x) = \rho(y) \iff x = y$ (complete). This is not always enforced, leading to ambiguities that have to be resoled in other ways.

If for all $X, Y \in \{B, O, I, D\}$ with $X \neq Y$ we have either $X > Y$ or $X > Y$, ρ induces a strict component order – this is satisfied only by the complete agent types [Bro+02; Bro+01].

For some example calculations, see Broersen et al. [Bro+02].

For the full BOID goal generation process, the input and output are sets of extensions. Another component is needed that selects only one extension from the set of extensions to turn into goals. This extension, the new intentions, is the input for the planning component [Bro+01].

Different strategies for choosing the "best" extension set exist [Bro+02]:

- Selecting the smallest extension set
- Using domain knowledge to make a decision, such as information about costs of actions
- Choose extension based on similarity with previously selected agents (*persistent* agent type).

To go from goals to action, the planning component decides which *actions* should be performed to achieve the *goals* given by the calculated *extension*. Once the planning component decides to $do(b)$, then $\top \underset{I}{\rightarrow} b$ is added the set of intentions for the next goal generation process [Bro+01; Bro+02].

The final control loop for the BOID architecture is shown in Figure 3.

```
select ρ ;
repeat
    Obs := read_environment();
    S := generate_candidate_goal_sets(Obs, B, O, I, D, ρ);
    P := select_goal_set_and_generate_plans(S);
    update(B, O, I, D, ρ, P)
until forever
```

Fig. 3. The control loop for the agent deliberation process in the BOID architecture. Taken from Broersen et al. [Bro+02].

2 Norms

Agents joining a group (or *society*) of agents must undergo a process of socialisation. This means that they are required to accept the normative conventions, called *norms*, that represent the rules of behaviour of the group. Norms can be seen as the desires of society: While obligations are agreements and commitments between individual agents, norms are "obligations" to the society as a whole [**dignum200towards**; Kol05; DKS02].

As to why to bother with norms at all, the knowledge of a society's norms can reduce the amount of computation required for an agent to make a decision, since the behaviour of other agents can be anticipated (with some degree of reliability). Knowledge of norms also allows for easier coordination between agents [Sav+10b].

An agent has various ways of becoming aware of norms. Norms can be hardcoded or given to an agent by an authoritative leader in the system. This is called the *top-down* approach. Preferably, an agent is able to infer the norms of a newly joined society on their own by e.g. observing patterns of interactions and their consequences. This has been done with e.g. association rule mining and is called the *bottom-up* approach.

Norms may change over time and an agent should be able to adapt to a changing environment [Sav+10b].

There are multiple views on how norms originate. The *legalistic* (or *prescriptive* view assues that norms are used to regulate the emergent behaviour of open systems. The idea is that agents are "motivated" by sanctions to stick to norms. The *interactionist* (or *emergent*) view states that autonomous norms are regularities of hebavior that emerge without any enforcement systems. Agents conform to norms because e.ġ their goals happen to coincide with the norms of the system. In this approach authoritative sanctions are not always necessary. Instead, social blame or exclusion from the group is often enough to ensure conformity [BVV08; Sav+10a].

Two types of norms are distinguished: *prohibition* norms, where the occurence of an event is followed by a sanction (littering causes a sanction in a park). *Obligation* norms cause sanctions to occur due to the *absence* of an event (waiter in a restaurant may sanction a customer for not tipping). Obligation norms are harder to detect autonomously than prohibition norms [Sav+10b].

3 Motivation

Motivations (or *motives* can be formalized as preference relations over an agent's desires. In motivated agents, desires are added by a motivational component. Example motivations are greed, altruism, hunger or love, and from that short list one can see that motivations are commonly more abstract (high-level) than desires. Motivations are not desires, but rather the reasons for desires.

An agent may have multiple motivations. In this case, a hierarchy of motivations has to be established, similar to Maslow's Hierarchy of Needs. Motivations further characterize agents [Kru+08; TK09; Kru+11; NL95].

Motivations affect desires by being linked with a *coupling strength* $\in [-1, 1]$ and a condition. Two examples might be $(\text{thirst}, \text{drink\_water}, 0.3, \text{at\_girlfriends\_place})$ and $(\text{thirst}, \text{drink\_beer}, 0.9, \text{at\_home})$.

Each motive has a weight associated with it that can change over time (thirst increases steadily over time, for example). The *motivation value* of a desire is generated by combining all applicable couplings and their respective motive weights. If the motivation value then exceeds a certain treshold, it is added to the lits of desires [Kru+11].

The motivation behind motivations is that the standard BDI models assume that the desires of an agent are given, and for simple agents with limited capabilites and tasks this is often sufficient. For more complex agents/environments, motivations are mechanisms that allow an agent to generate and set its own desires. In this case, motivations drive the behaviour of an agent [Kru+11].

3.1 Motivated cooperation

In motivated multi-agent systems, cooperation arises when an agent's goal either cannot be achieved alone or is better achieved through cooperation. In such situtations, an agent uses its knowledge and trust of others to determine who to ask for assistance. On receiving a request for assistance, agents inspect their own motivations and commitments (intentions) to decide how to respons. Motivation determines whether an agent *wants* to cooperate [GL03].

Issues with motivated cooperation arise when motivations change over time. Cooperations that would be valuable in the long-term may be rejected if the motivations are temporarily opposed. A short-term view of cooperations leads to missed opportunities for cooperation; agents' goals may be "out of step" [GL03].

References

[Bro+01] Jan Broersen et al. "The BOID architecture: conflicts between beliefs, obligations, intentions and desires". In: *Proceedings of the fifth international conference on Autonomous agents*. ACM. 2001, pp. 9–16.

[Bro+02] Jan Broersen et al. "Goal generation in the BOID architecture". In: *Cognitive Science Quarterly* 2.3-4 (2002), pp. 428–447.

[BVV08] Guido Boella, Leendert Van Der Torre, and Harko Verhagen. "Introduction to the special issue on normative multiagent systems". In: *Autonomous Agents and Multi-Agent Systems* 17.1 (2008), pp. 1–10.

[Dig+00] Frank Dignum et al. "Towards socially sophisticated BDI agents". In: *MultiAgent Systems, 2000. Proceedings. Fourth International Conference on*. IEEE. 2000, pp. 111–118.

[Dig99] Frank Dignum. "Autonomous agents with norms". In: *Artificial Intelligence and Law* 7.1 (1999), pp. 69–79.

[DKS02] Frank Dignum, David Kinny, and Liz Sonenberg. "From desires, obligations and norms to goals". In: *Cognitive Science Quarterly* 2.3-4 (2002), pp. 407–430.

[Fis+07] Michael Fisher et al. "Computational logics and agents: a road map of current technologies and future trends". In: *Computational Intelligence* 23.1 (2007), pp. 61–91.

[GL03] Nathan Griffiths and Michael Luck. "Coalition formation through motivation and trust". In: *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*. ACM. 2003, pp. 17–24.

[Kol05] Martin Josef Kollingbaum. "Norm-governed practical reasoning agents". PhD thesis. University of Aberdeen, 2005.

[Kru+08] Patrick Krümpelmann et al. "Belief operations for motivated BDI agents". In: *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 1*. International Foundation for Autonomous Agents and Multiagent Systems. 2008, pp. 421–428.

[Kru+11] Patrick Krümpelmann et al. "Motivating agents in unreliable environments: A computational model". In: *Multiagent System Technologies*. Springer, 2011, pp. 65–76.

[NL95] Timothy J Norman and Derek Long. "Goal creation in motivated agents". In: *Intelligent Agents*. Springer, 1995, pp. 277–290.

[Sav+10a] Bastin Tony Roy Savarimuthu et al. "Norm identification in multi-agent societies". In: (2010).

[Sav+10b] Bastin Tony Roy Savarimuthu et al. "Obligation Norm Identification in Agent Societies." In: *Journal of Artificial Societies & Social Simulation* 13.4 (2010).

[TK09] Matthias Thimm and Patrick Krümpelmann. *Know-how for Motivated BDI Agents:(extended Version)*. TU, Department of Computer Science, 2009.