Ethics and autonomous agents

Models for ethical autonomous agents

ANR ETHICAA – ANR-13-CORD-0006
Delivrable #4

Main authors
Olivier Boissier
Grégory Bonnet
Nicolas Cointe
Bruno Mermet
Gaële Simon
Catherine Tessier
Thibault de Swarte

April 19, 2017

Contents

Chapter 1

General introduction

As concluded in [Boissier et al., 2015], multiple autonomous agents within a system may be heterogeneous in terms of goals and ethics. Thus it is the first importance to allow the agents to justify their decisions in order to be judged as ethical (or not) by other agents. Consequently an ethical competent autonomous artificial agent should also be able: (1) at the micro-level to represent its ethics and justify its decisions, to represent the ethics of another agent and to verify that this agent's behavior follows its ethics, to judge the ethics of the other agent through a comparison mechanism, and to take into account this judgment in its own decisions; (2) at the macro-level to build a collective ethics, to identify and be able to judge a collective ethics, to be able to judge other agents through the collective ethics and to make an arbitration between its own ethics and the collective ethics.

In order to design such ethical agents, a state-of-the-art about the models, methods and tools is provided in this technical report. Firstly, we review models provided in the literature to assess their relevance and their expressiveness, as a huge number of concepts must be represented (immediate actions, long-term actions, consequences, resources, values, goals, believes and so on)). Secondly, we propose first steps in order to improve models with such concepts. Specific legal and ethical concepts in multi-agent systems (organizations, interactions, environment and users) are let for further work. The Chapter 2 reviews firstly an important notion, namely the notion of value, that – as seens in [Boissier et al., 2015] – must be considered in order to define ethical artificial agents. Then, Chapter 3 focuses on some verification and supervision techniques while Chapter 4 focuses on some reasoning and decision making techniques. Finally, a synthesis is given in Chapter 5.

Chapter 2

Values: a fundamental concept

Value, key concept studied for a long time in Philosophy, has become also a key concept in social sciences since their development middle of 19th century (e.g. sociology, psychology, political science). They are intrinsically related to social norms and some values, sometimes called *moral values*, are related to ethics. Indeed, values have an influence on scientific or technical norms, but can also be seen in legal or moral norms (such as the precautionary principle). Moreover, as stated by Talcott Parsons, values are essential in order to ground a theory of voluntary action [Parsons, 1951]. Thus, in order to define how an ethical artificial agent should behave, we need to investigate how values can be embedded in computer systems.

This chapter is organized as follows. It starts with a definition of the global perspective in ethics in which we will consider values (cf. Sec. 2.1). This global perspective aims at defining the key concepts pertaining to the definition of an ethical decision system in which values can play a key role. Based on this context, we will analyse and list the various notions of values encountered in psychology, socio-psychology and political science (cf. Sec. 2.2). Based on this first analysis that provides us with first definitions and properties of what is called value in domains of social sciences, we turn in Sec. 2.3 to the analysis of works in computer sciences basing their approach on the value concept.

2.1 Ethics in context

From ancient philosophers to recent works in neurology [Damasio, 2008] and cognitive sciences [Greene and Haidt, 2002], many studies have been interested in the capability of human beings to define and distinguish between the fair, rightful and good options and the invidious, iniquitous and evil options. As given in [Voyer, 2014] moral philosophy is based on concepts like *morals*, *values*, *ethics*, *dilemma*, *judgment*, *blame*, *responsibility* or *accountability*.

Before going into the definition of values and the analysis of its use in different domain with respect to ethics, we need first to provide some definitions of these terms. We will first consider notions in relation to the definition of the theory of the good in relation to moral (Sec. 2.1.1), theory of the right in relation to the use of ethical principles (Sec. 2.1.2). We will end this section by giving definition of the concepts that are using both theories to build judgments, to blame or to found notions such as responsibility and accountability (Sec. 2.1.3).

2.1.1 Theory of the Good

Morals can be distinguished from law and legal systems in the sense that there are not explicit penalties, officials and written rules [Gert, 2015]. Indeed, everyone knows many moral rules as "Lying is evil", "Being loyal is good" or "Cheating is bad". Those rules ground our ability to distinguish between good and evil, and they are often supported and justified by some *moral values* such as freedom, benevolence, wisdom, conformity [Brey, 2015]. According to [Gert, 2015], we consider the following definition of moral:

Definition 2.1 (Morals) Morals *describes the compliance of a behavior with mores, values and usages of a group or a single person by associating a good or bad label to combinations of actions and contexts.*

As can be seen in this definition, morals is strongly founded on the notion of values. Psychologists, sociologists and anthropologists almost agree that those values are central in the evaluation of actions, people and events [Graham et al., 2012, Parks-Leduc et al., 2015, Perrinjaquet et al., 2007, Rokeach, 1973, Schwartz, 2006]. Some studies from psychologists or socio-psychologists [Rokeach, 1973, Schwartz, 2006] promote the idea that those values are almost universal and finite in number. heir relative importance and their structuring are on the contrary not universal and are strongly context-dependent. Values are usually dynamically organized by

relative importance with respect to a particular context within a *value system*. It is important to notice that – as claimed by Talcott Parsons – the relationship between value system and human behaviors is not one-way: while a value system grounds individual goals and motivations, those goals and motivations also ground the value system [Parsons, 1951].

Sets of moral rules and moral values establish a *theory of the good*.

2.1.2 Theory of the Right

Using this theory of the good, *theories of the right* contextually conciliate moral to recognize a fair or, at least, acceptable option of action [Timmons, 2012].

For instance even if stealing can be considered as immoral regarding a theory of the good, some philosophers agree that it is acceptable for a starving orphan to rob an apple in a supermarket regarding a theory of the right. This conciliation is called *ethics* and, relying on some philosophers [Ricoeur, 1995], we admit the following definition:

Definition 2.2 (Ethics) Ethics *is a discipline that proposes* ethical principles *to conciliate morals, laws, desires and capacities of the agent in a* judgment *in order to define how humans should act and be toward the others.*

Philosophers proposed various ethical principles, such as Kant's Categorical Imperative [Johnson, 2014] or Thomas Aquinas' Doctrine of Double Effect [McIntyre, 2014], which are sets of rules that allow to distinguish an ethical option from a set of possible options. Traditionally, three major approaches are considered in the literature:

- **Virtue ethics**, where an agent is ethical if and only if he[1] acts and thinks according to some values as wisdom, bravery, justice, and so on [Hursthouse, 2013]. Proposed by the School of Athens, virtue ethics aims at thinking about vice and virtues that should lead the human behavior. In order to distinguish a fair or, at least, acceptable option, we need to define those values and to seek how they promote or demote the options. However, it has been shown that contradictions often appear when considering opposite actions that each promote different virtues [Plato, 1966].

[1] In this section, we consider agents in terms of philosophy, not only in terms of computer sciences.

- **Deontological ethics**, where an agent is ethical if and only if he respects obligations and permissions related to possible situations [Alexander and Moore, 2015]. Thus, deontological ethics is often used to describe community or professional ethics, enforcing generally obedience to moral rules. For instance, the divine command *Thou shalt not kill* forbid a given behavior without any references to vice, virtues or consequences. Kantian categorical imperatives are also deontological ethics.

- **Consequentialist ethics**, where an agent is ethical if and only if he weighs the morality of the consequences of each choice and chooses the option which has the most moral consequences [Sinnott-Armstrong, 2014]. Such approach justifies option with respect the morality of their goals. Some consequentialist ethics focus on consequences definitions (e.g. hedonism aims at maximizing pleasure while minimizing suffering). Other ethics focus on how to weight good and bad consequences of a given option. For instance, egoism aims at maximizing the agent's welfare, utilitarism aims at maximizing the sum of all agent's welfare, or altruism aims at taking the others' welfare in the agent's own welfare.

However, in some unusual situations, an ethical principle is unable to give a different valuation (a preference) between two options. Those situations are called *dilemmas* [McConnell, 2014].

Definition 2.3 (Dilemma) *A* dilemma *is a choice between two options, each supported by ethical reasons, given that the execution of both is not possible. Each option will bring some regret.*

In the sequel, we consider dilemma as a choice for which an ethical principle is not able to indicate the best option, regarding a given theory of good.

While many famous dilemmas, such as the trolley problem [Foot, 1964], are perceived as failures in morals or ethics or, at least, as an interesting question in the human ability to judge ethically and to provide a rational explanation of this judgment, let us notice that human beings rarely rely on a single ethical principle, avoiding in this way such failures. Indeed, the core of ethics is the judgment. It is the final step to make a decision and it evaluates each choices, with respect to the agent's desires, morals, abilities and ethical principles.

2.1.3 An overview of concepts relying on ethics

As stated by the psychologist Jonathan Haidt [Haidt, 2001], human beings engage in ethical judgment, combining parts of multiple ethical principles, to search for arguments that will support a premade point-of-view highlighted by the values considered as important in the situation. Interestingly, such ethical judgment can be circular, overriding the initial intuition and overcoming the premade point-of-view. Thus, when facing a dilemma, an agent can consider several principles in order to find a suitable solution. For instance, if an artificial agent is facing two possible choices with both good and/or bad effect (e.g. kill or be killed), the ethical judgment allows him to make a decision in conformity with a set of ethical principles and preferences. That is why an autonomous artificial agent must be able to reason on a broad range of principles, and must be embedded with a judgment mechanism that assesses which principle leads to the most satisfying decision.

Relying on some consensual references [Dictionary, 2015] and our previous definitions, we consider the following definition:

Definition 2.4 (Judgment) Judgment *is the faculty of distinguishing the most satisfying option in a situation, regarding a set of ethical principles, for ourselves or someone else.*

We need to structure the discourse around these concepts to better understand their interrelation: judgement procudes blames or praises or forgiveness, judgement is motivated by accountability after having established responsibility of the judged agent in the causation of the event or behaviour subject of the judgment. Do you agree? Several theories may be considered to explain or motivate the production of blame with respect to a judgement: egoistic theory, deterrence theory, retributive theory, accountability theory [de Kenessey and Darwall, 2014]

As stated in [Ropes and Guglielmo, 2016], blame is a multi-faceted social phenomenon. It is a kind of moral judgment, used to set and affirm norms, to evaluate actions and events, to evaluate agents [Malle et al., 2014]. It is thus related to the concept of moral judgment introduced in the precedent definition. As stressed by Beardsley in [Beardsley, 1970], blame "has a power and poignancy for human life unparalleled by other moral concepts", it is of first importance to understand the ways in which individuals attribute blame to others, how the amount of blame to assign is decided, and how. It is even more important in the context of the ETHICAA Project since, as

pointed by these first elements, blame could be a fruitful concept to connect social concerns with individual behaviours. According to [Malle et al., 2014], blame can be defined as:

Definition 2.5 (Blame) *A* blame *is a unique moral judgment that has four properties: blame is both cognitive and social, regulates social behavior, fundamentally relies on social cognition, and requires warrant.*

According to the model proposed by [Malle et al., 2014], the process that attributes blame begins when a norm violation is detected. In case the norm violation is confirmed and a causal link between it and an agent has been established, the intentionality of the violation is then considered. In case of intentional violation, the reasons behind it are considered. In case of asocial, vengeful, or selfish reasons [Reeder et al., 2002], of prediction of further norm violations [Tetlock et al., 2007], blame judgments are exacerbated. On the contrary, if the motivation was self-defense [Finkel et al., 1995] or for greater good [Lewis et al., 2012] blame judgments are typically mitigated. Interestingly, the notion of blame is closely related to the notion of *responsibility* [Shaver, 1985].

Definition 2.6 (Responsibility) *Responsibility is a judgment based on the agent's causal contribution; awareness of negative consequences; intent to cause the event; degree of volition (e.g., freedom from coercion); and appreciation of the action's wrongness.*

A complete survey of this notion (and the associated models) is given in [Guglielmo, 2015]. If responsibility is obviously a link between causality and blame, it raises several questions: sometime blame causes responsibility, sometime responsibility is grounded by intentionality, sometime TBSL. Thus, [Guglielmo, 2015] states that *responsibility either lacks clear moral content (e.g. when it stands for causality) or is redundant with less ambiguous moral judgments (e.g. blame).*

It is important to distinguish this notion from the notion of legal responsibility [Perennou, 2014]. From a legal perspective, responsibility is grounded by liability and accountability. As nouns the difference between liability and accountability is that liability is the condition of being liable while accountability is the state of being accountable. Liability is directly related to legal personhood[2].

[2]Currently, artificial agents could not meet the criteria for legal personality.

Definition 2.7 (Liability) *Liability describes the condition of being actually or potentially subject to a legal obligation.*

As autonomous artificial agents are currently goods and not legal personalities the only liabilities that might apply on them are liability for defective products, liability for actions of things, liability for action of animals and vicarious liability [Perennou, 2014].

Whatever it be, all those various concepts allow to evaluate ethics in context. One of the most important are values as they seems to be a grounding element of all ethical judgment.

2.2 Human values

In the domain of social sciences, several scholars have devoted a huge amount of studies to the definition and usage analysis of the concept of *value*. For instance, in psychology and socio-psychology, numerous works based on statistical analysis propose sets of fundamental human values structured into taxonomies relative to domains [Rokeach, 1973, Swchartz and Bilsky, 1990, Valette-Florence et al., 1996, Schwartz, 2012]. We will first present these studies (Sec. 2.2.1) that we use in the following section (Sec. 2.2.2) to propose a definition of the *value* concept and of *value system* in Sec. 2.2.3. Let us remark that, in the perspective of building ethical artificial agents, it seems important to think about *social value systems*.

2.2.1 Values Taxonomies

Since 1981, the World Values Survey Association[3] investigates the attitude of people in over 100 countries towards a large number of social, cultural and moral values. Several countries conduct also such analysis and surveys. For instance, since 2006, the Eurobarometer[4] studies values held by the people of EU member states, and several other international surveys of values. However, psychologists or socio-psychologists already proposed taxonomies of values, grouped in domains based on statistical analysis. For instance,

- In [Rokeach, 1973], the psychologist M. Rokeach highlights 18 *terminal values* what express end states of existence and 18 *instrumental values* that express modes of conduct,

[3] `http://www.worldvaluessurvey.org/`
[4] `http://ec.europa.eu/public_opinion/archives/eb/eb66/`

- In [Hofstede, 2001], the socio-psychologist G. Hofstede highlights 5 dimensions of national cultural values as summarized in Table 2.1[5],

Power distance	Acceptance of unequal distribution of power in institutions and organizations
Uncertainty avoidance	Acceptance of uncertainty and ambiguity in decisions
Individualism	Preference for loosely knit social framework: the actor and his close related family
Collectivism	Preference for tightly knit social framework: the clans, the nations, the institutions
Masculinity	Preference for achievement, heroism, assertiveness, and material success
Femininity	Preference for relationships, modesty, caring for the weak, and quality of life
Long-term view	Preference for keeping a static activity
Short-term view	Preference for social changements

Table 2.1: Hofstede's dimensions of national cultural values

- In [Welzel and Inglehart, 2010,Ingelhart and Welzel, 2005], the studies – conducted by political scientists – highlight two dimensions summarized in Table 2.2 and Figure 2.1.

Maybe one of the most complete survey was realized by the psychologist Shalom Schwartz [Swchartz and Bilsky, 1990, Swchartz, 1992, Schwartz, 1994, Ros et al., 1999, Schwartz, 2006, Schwartz, 2012]. Originally highlighting 56 basic human values grouped into 10 value types, it has been recently refined to 19 value types [Schwartz, 2012], and used several works, for instance [Knoppen and Saris, 2009, Ishita et al., 2010, Maio, 2010, Steinmetz et al., 2012]. Here, values are cognitive representations of three types of universal human requirements: biologically based needs of the organism, social interactional demands for interpersonal coordination, and social institutional demands for group welfare and survival. Moreover, values are clustered in value types, based on the overarching motivational goal they express. Interestingly, those value types allow to model the values in a

[5]Let us remark that contrary to the other studies this study doesn't adopt an international perspective and is limited to a national point of view.

Traditional values	Values emphasize religious beliefs, familial obligations, marriage, national pride, obedience, absolute values and norms, and respect for authority
Secular-rational values	Values in which secular, bureaucratic and rational considerations are important placing greater openness and tolerance for different family models, sexual orientations, and lifestyles
Survival	Values that emphasize economic and physical security, leading to ethnocentric outlook and limited levels of tolerance and trust
Collectivism	Values taking economic and physical security for granted, and focuses on immaterial needs, such as life satisfaction, public expression, and liberty

Table 2.2: Inglehart and Welzel's dimensions of values

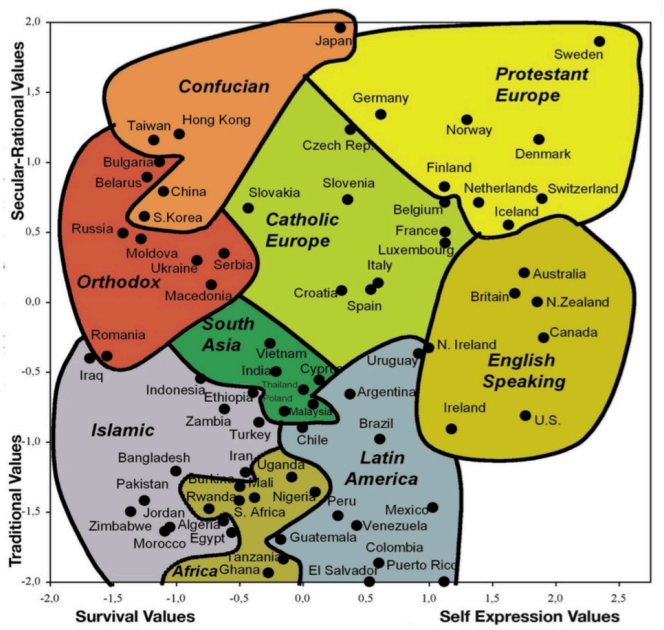


Figure 2.1: The 2008 Ingelhart-Welzel culture map

Power	Dominance	Control over people
Power	Resources	Control of material and social resources
Power	Face	Maintaining public image and avoiding humiliation
Achievement		High social standards (ambitious, capable, influential)
Hedonism		Pleasure and sensuous gratification for oneself
Stimulation		Excitement, novelty, and challenge in life
Self-direction	Thought	Freedom to cultivate one's own ideas and abilities
Self-direction	Action	Freedom to determine one's own actions
Universalism	Concern	Equality, justice, and protection for all people
Universalism	Nature	Preservation of natural environment
Universalism	Tolerance	Acceptance/understanding of those who are different
Benevolence	Dependability	Being reliable and trustworthy
Benevolence	Caring	Devotion to the welfare of others
Tradition		Respect of traditional culture or religion
Conformity	Rules	Compliance with rules, laws, and formal obligations
Conformity	Interpersonal	Avoidance of upsetting or harming other people
Security	Personal	Safety in one's immediate environment
Security	Societal	Safety and stability in the wider society
	Humility	Recognizing one's insignificance in the larger scheme

Table 2.3: Schwartz's original values, and 2012 refinement

circular structure, as depicted in Figure 2.2: value types which express complementary motives are placed in adjacent positions and value types that express conflicting motives are placed opposite each other.

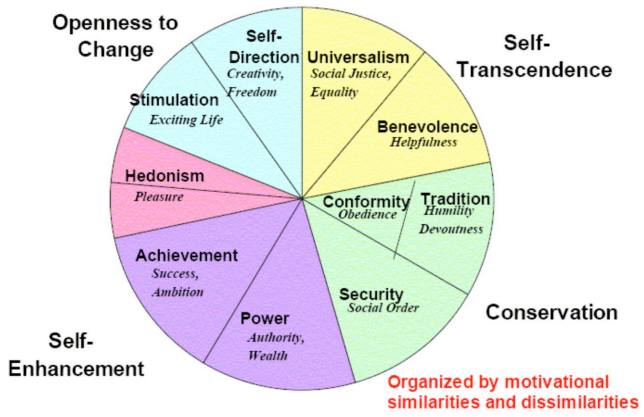


Figure 2.2: An overview of Shalom Schwartz's values

Interestingly, Schwartz identifies two orthogonal dimensions:

- openness to change (meaning to follow their own intellectual and emotional interests in unpredictable and uncertain directions) vs. conservatism (meaning to preserve the status quo and the certainty it provides in relationships with close others, institutions, and traditions),
- self-enhancement (concerns for the consequences of own and others' actions for the self) vs. self-transcendence (concerns for the consequences of own and others' actions in the social context).

However, each of those previous surveys presents different results with respect to the correlation it found and the bias it allowed [Perrinjaquet et al., 2007]. For instance, while some Schwartz's values can be considered as moral values, it is not the case for all values such as power. Moreover, these values have been identified in an agnostic way such that some unconsidered or internalized moral values can be missing. Indeed, [Parks-Leduc et al., 2015] show that those values are cognitive but not emotive features. Finally, there is not any single theory to explain the values.

To mitigate those problems, some works consider a predefined set of domains and ask people which values can be associated to each domain. For

	Schwartz' values		Graham's values
	Personals	**Socials**	**Morals**
Protection	Accomplishment Power	Conformity Tradition Security	Loyalty Authority Sanctity
Progression	Hedonism Stimulation Autonomy	Universalism Benevolence	Equity Care
Expression	Preferences	Attitudes Norms	

Table 2.4: Value structure and moral foundations according to [Voas, 2014]

instance, [Shweder et al., 1997] consider three domains (autonomy, community and divinity) and [Graham et al., 2011, Graham et al., 2012] consider five ones. Such restriction of domains is also helpful to consider ethics in context. For example, the *ethical matrix* is a tool for analyzing ethical issues in a given situation [Mepham, 2013, Mepham, 2000], being mostly used in food and agriculture domains [Schroeder, 2003]. Such matrix formalizes the relationships between groups of interest (humans, animals, environment, and so on) and three fundamental values that are *respect for well-being*, *for autonomy* and *for justice*. These three values are not mutually exclusive and aims at expressing the most common ethical concerns, capturing key elements of the common morality, the norms and assumptions that underpin contemporary society.

However, it is important to notice that those works present methodological biases as they ask people to prononce on undefined domains, leading to a risk of repeating and amplificating forejudgments [Nilsson and Erlandsson, 2015]. Nevertheless, [Voas, 2014] show that values identified by [Schwartz, 2012] can be structured according to a personal or a social axis, and that the domains of [Graham et al., 2012] correspond to the social axis, as highlighted by Table 2.4. Let us notice that Mepham's values can be mapped to Voas' domains: respect for autonomy is expression, respect for well-being is progression, respect for justice is protection.

2.2.2 Value definition

All the previous work refer to values, and in order to implement ethical competent autonomous artificial agents [Boissier et al., 2015], we need to define more precisely what are values. However, we need to be carefull as [Welzel and Inglehart, 2010, Ingelhart and Welzel, 2005] shown that value definition is also a matter of politics.

In a general point-of-view, values are abstract qualities or state-of-affairs that people see as good or ideal [Brey, 2014, Brey, 2015]. For instance, freedom, justice, democracy, wisdom, honesty, efficiency, beauty, serenity, friendliness, well-being, excellence are all values. Values can be clustered, based on the overarching motivational goal they express: they are constructed from judgments about the capacity of things, people, actions and activities to enable best possible livings [Rohan, 2000] and are universally present to a greater or lesser degree in all cultures [Schwartz, 1994, Bardi et al., 2009]. Thus, values have the following properties and characteristics:

- **Abstract and trans-situational**: A value "transcends specific objects, situations" [Rokeach, 1973] and "global beliefs" [Connor and Becker, 1979]. They are "abstract trans-situational" [Schwartz, 1994] "cognitive structures" [Feather, 1996].

- **Relatively stable**: Values are "enduring" but not completely stable [Rokeach, 1973, Boudon, 2001].

- **Almost universal and finite in number**: "All men everywhere possess the same values to different degrees [...] [and] the number of values human beings possess is assumed to be relatively small" [Rokeach, 1973]. According to [Schwartz, 1996], there are a finite number of universally important value types even if the combination of those values is not universal.

- **Organized in a hierarchy reflecting their relative importance**: Values vary in importance [Bilsky and Schwartz, 1994,Schwartz, 1994]. They are clustered in "hierarchical organizations" [Rokeach, 1973] "ordered by relative importance" [Schwartz and Bilsky, 1987].

- **Have affective components**: "A value is affective in the sense that [a person] can feel emotional about it, be affectively for or against it" [Rokeach, 1973]. They are "a union of reason and feeling" [Kluckhohn, 1951] based on "emotion-laden conceptions of the desirable" [Hitlin, 2003].

- **Can be verbalized**: Values are "almost always potentially expressible in rational language" and "eminently discussable" although normally implicit [Kluckhohn, 1951].

As said previously, all human values are not necessarily moral values. Thus, in the sequel, we define moral values as follows:

Definition 2.8 (Moral value) *A* moral value *is a value that concerns the conditions of right and wrong conduct, in relation to what is considered good and acceptable in society, especially regarding our conduct towards others, concerns harms and benefits for others, our duties towards others and oneself, and the rights of others.*

For instance, responsibility, integrity, beneficence, justice, freedom, equality, and human dignity are moral values. Let us remark that, from a Durkheimian perspective, moral values are not defined a priori such as general laws. They are built within a plastic and evolutive value sytem [Dambra, 2005].

2.2.3 Value systems

As values come with stable and predictable relations among them, reflecting conflicts and compatibilities [Bardi et al., 2009], they are embedded with an *value priority*. Such priority is the relative importance of particular values to individuals or groups, which may differ between individuals and between groups, and are only stable in a particular domain and in a particular time [Boltanski and Thévenot, 2006]. However, those priorities may change in response to changes in the actors' environment [Rohan, 2000]. Such structure within values is called a value system [van Marrewijk and Werre, 2003, Wiener, 1988]

Definition 2.9 (Value system) *When a number of key or pivotal values concerning organization-related behaviours and state-of-affairs are shared-across units and levels-by members of an organization, a* value system *is a way of conceptualizing reality which encompasses a consistent set of values, beliefs and corresponding behaviour that can be found in individual persons, as well as in companies and societies.*

Thus, a value system is a hierarchical organization of values with their value priorities within which there are stable and predictable relations among priorities on each value type. It is important to notice that a value system

has a *dynamic structure* – individuals reorder their value system (at least some key values) dynamically in different situations [Seligman and Katz, 1996] – and is not unique – different value systems exist for abstract issues and for specific social, ethical, political or personal issues. Value systems are also dynamic in *space and time*: different cultures have different value sytems that evole through time [Boltanski and Thévenot, 2006]. Finally, value systems are both invidual and collective, built in a *retroaction loop*: a society influences how individuals define their value sytems, and individuals' value systems influence the society's value systems. Indeed, two kinds of values are encompassed in a value system [Tetlock, 1986, Rohan, 2000, Brey, 2014, Brey, 2015]:

- From a sociological perspective, **espoused values** are endorsed to conform with social norms or expectations but not necessarily internalized, and which form a value subsystem called *social value system* or *ideological value system* that refers and organizes to people's perceptions of others' values and values priorities (other people, groups, institutions, cultures). Such values can include moral values, but also other values, for example values regarding etiquette and accepted ways of doing things (e.g. openness, punctuality, solidarity, chastity, self-discipline and individualism).

- From a psychological perspective, **values-in-use** are idiosyncratic to an individual (reflect one's actual or real value priorities), and which form a value subsystem called *individual value system* that reflects people's own judgments about the capacity of entities to enable best possible living for themselves.

Let us notice that some authors considers some other kinds of value systems, such as *cultural value systems* [Brey, 2015, Brey, 2014]. Indeed, a culture is the collection of beliefs, symbols, values, norms, behaviours and artifacts shared by a group of people amongst particular ethnic groups (Native American culture, Jewish culture, Tuareg culture) or amongst subgroups in society (non-ethnic 'subcultures', such as hacker culture, hippie culture, Internet culture). Thus, a cultural value system refers to values shared by the members of a culture, often corresponding to values that are expressive of one's culture. Thinking cultural value systems seems important when several people from different cultures try to find ethical agreement[6].

[6]We can obviously think about the IEEE Global Initiative for Ethical Considerations in the Design of Autonomous Systems.

Interestingly, values and value systems form a language. From a moral relativism perspective (assuming that values and value systems across the world differ) or moral absolutism perspective (assuming the observable differences are only differences in how values are expressed), value language makes people able of talking about their value priorities, and able of arguing for one attitudinal or behavioral decision over another. In this sense, modelling values seem necessarily in order to design autonomous artificial agents able to produce ethical judgment. However, from an axiological perspective, such modelling must be made carefully as the model in itself is based on an implicit value system.

2.3 Values in computer sciences

2.3.1 Values and design

As noticed in [Nissenbaum, 2001, Nissenbaum, 2005, Shilton, 2010, Shilton et al., 2013, Shilton et al., 2014], values are identifiable entities that appear in technologies. They are built in consciously or not by designers and concretized through affordance, built by technology users and brought by the social context of technology design and deployment. In this sense, incorporating research from computer ethics, social informatics, participatory design, worth centered design is an issue in computer systems engineering.

As a first approach, *Value in Design* (VID) describes a research space focused on finding and naming values challenged by emerging technologies and infrastructures. However, it does not prescribe a set of methods or approaches for studying values. Those early works lead to *Value Sensitive Design* (VSD[7][8]) introduced by Batya Friedman [Friedman, 1996]. It a proactive approach concerned by values that deal with human welfare and justice which is seeking to influence technology during the design process, making a clear distinction between *value* – marketing quantification based on the worth of the end product – and *social or ethical value* as defined in the previous section. To this aim, VSD defines an iterative, tripartite, methodology that identifies the *human value requirements of stakeholders*, addresses *competing values*, and tests *value decisions* throughout the design process [Gilmore et al., 2008, Friedman et al., 2013, Partala and Kujalan, 2016].

[7] `http://www.vsdesign.org/`

[8] `http://ethicsandtechnology.eu/impact/value-sensitive-design/`

A huge number of works propose to apply VSD methodology to diverse computer and technical systems such as weapon systems [Cummings, 2006], public transport systems [Ferris et al., 2010], medical devices [Dennings et al., 2010], teaching systems [Flanagan et al., 2008].

2.3.2 Values in computer domains

Some works tried to identify specific ethical questions in a given computer science domain. Often, such research lead to identifying values that are meaningful in the domain, and how those values can be instantiated. In this section, we provide a review of some few domains in order to assess the general problematics, and we make a focus on the privacy value.

In the context of the social reality of virtual worlds which includes virtual relationships and trust in virtual networks, [Gooskens, 2010, raker, 2010,Schroeder, 2011] investigate values to assess the ethical status of virtual actions. For instance, some key values are physical pleasures, aesthetic pleasures, creative pleasures, autonomy, rationally, informedness, fame, wealth, and social status.

In computer game, the main concept is *ethical game-play* that is a playful experience in which regulation, mediation, and/or goals require from the player moral reflection beyond the calculation of statistics and possibilities [Schrier and Gibson, 2010, Sicart, 2011, Sicart, 2013]. Here, players are considered as moral agents, capable of using ethical reflection to act upon choices in game experience, and games are objects with values embedded in their design that establish a mode of relation with the player, limiting their agency in the game world with a pre-determined, designed purpose. Thinking ethical game-play is thinking both *game world* – a semantic wrapper of the game system, the combination of fiction and simulation –, *game rules* – formal structure of the game, boundaries in which play takes place, freely accepted by players and unbreakable –, and *game mechanics*, – actions afforded by the system to the player so she can interact with the game state and with other players. From those elements, a typology of how ethics is designed has been proposed, as highlighted in Figure 2.3.

In the security domain, a widely accepted definition of values does not yet exist. However, many works propose sets of sets of operationalized security values, or code of ethics [Dhillon and Torkzadeh, 2006, Stevens, 2009, Timmermans et al., 2010, Burmeister, 2013, Solomon, 2014]. In this domain, a *value-sensitive security policy* is the integration of security values and security policy, resulting in a policy statement that includes both the human

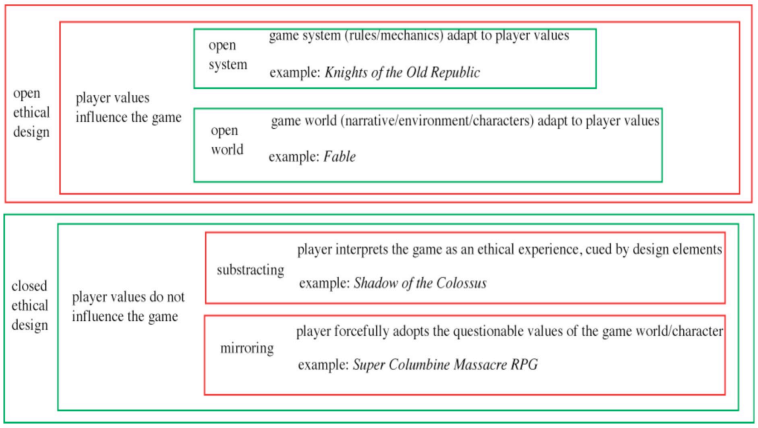


Figure 2.3: An overview of Miguel Sicart's typology

value associated with the policy and the specific action employees are directed to take in order to protect the organization's information assets. A metric, called *value congruence* or *value alignment* which is the degree to which an individual and an organization's culture share the same values, allows to measure the quality of such value sensitive policies. Finally, the *security culture* is a group's shared values, goals, and behaviors, contributing to its success through awareness of security risk, and day-to-day participation in preventive measures. It is established and sustained when employee and organizational security values are congruent.

Finally, one of the most important value in computer domains is perhaps privacy [Solove, 2006, Moreham, 2008, Dratwa, 2014]. Indeed, this value is meaningful in security, virtual worlds, social networks, data mining, and every other domains which use personal data. Privacy aims at protecting people and the values of *freedom* and *democracy*, so that everyone can enjoy their daily lives without fear. This issue is so important that the OECD[9] provides guidelines for privacy enabling:

1. Collection Limitation Principle: There should be limits to the collection of personal data and any such data should be obtained by lawful and fair means and, where appropriate, with the knowledge or consent of the data subject.

2. Data Quality Principle: Personal data should be relevant to the pur-

[9]Organization for Economic Co-operation and Development

poses for which they are to be used, and, to the extent necessary for those purposes, should be accurate, complete and kept up-to-date.

3. Purpose Specification Principle: The purposes for which personal data are collected should be specified not later than at the time of data collection and the subsequent use limited to the fulfillment of those purposes or such others as are not incompatible with those purposes and as are specified on each occasion of change of purpose.

4. Use Limitation Principle: Personal data should not be disclosed, made available or otherwise used for purposes except: a) with the consent of the data subject; or b) by the authority of law.

5. Security Safeguards Principle: Personal data should be protected by reasonable security safeguards against such risks as loss or unauthorized access, destruction, use, modification or disclosure of data.

6. Openness Principle: There should be a general policy of openness about developments, practices and policies with respect to personal data. Means should be readily available of establishing the existence and nature of personal data, and the main purposes of their use, as well as the identity and usual residence of the data controller.

7. Individual Participation Principle: An individual should have the right: a) to obtain from a data controller, or otherwise, confirmation of whether or not the data controller has data relating to him; b) to have communicated to him, data relating to him within a reasonable time; at a charge, if any, that is not excessive; in a reasonable manner; and in a form that is readily intelligible to him; c) to be given reasons if a request made under sub-paragraphs (a) and (b) is denied, and to be able to challenge such denial; and d) to challenge data relating to him and, if the challenge is successful to have the data erased, rectified, completed or amended.

8. Accountability Principle: A data controller should be accountable for complying with measures which give effect to the principles stated above.

We can distinguish *privacy by design* meaning that technologies should be designed with privacy in mind from the outset since privacy cannot be protected solely through compliance with regulatory instruments and *privacy in design* which looks at the normativity of structural choices in an

effort to promote transparency and protect rights and values of the citizens. We can notice that privacy is supported by other values, and thus is closely related to them:

- **Freedom**: As privacy means the right to protect actions and thoughts that persons want to keep to themselves, privacy is closely related to freedom and intimacy. Let us notice that freedom is an important value that may be sometimes restrained by ethics. Indeed, as stated by the philosopher Herbert Spencer, every man has freedom to do all that he wills, provided he infringes not the equal freedom of any other man.

- **Autonomy**: Privacy is is the condition for being able to pursue one's ends and psychological integrity to ensure the right to autonomy. Let us notice that, here, autonomy refers to human autonomy and not to artificial agents' autonomy.

- **Dignity**: Privacy underpins human dignity, through freedom of association and freedom of speech. Those notions of freedom must be differentiated from the previous one, and are fundamental in terms of ethics and law.

- **Justice**: While the principle of justice is interpreted in many different ways in different contexts, in the context of privacy, justice focus on non-discrimination.

- **Transparency**: This value allows for democratic control, emphasizing the importance of openness on policy making and implementation. What is done? How the decisions to do what are made? Who does what?

2.3.3 Values in Artificial Intelligence

In Artificial Intelligence, the notion of value is generally abstracted in order to be leaved to the end-user's discretion with respect to the target application. For instance, [Bench-Capon, 2002] consider values as labels associated to the objects of the world, ordered with respect to a given preference relationship (not necessary transitive). The reason behind this abstraction is that sociological values are fine concepts whereas computer science needs explicit formal definitions.

Despite that fact, some works are interested in defining what should be values for an artificial agent. For instance, in the context of rational

agents, [Gigerenzer, 2010] consider values as heuristics and propose four ones: *peer imitation* meaning doing what the majority of other agents do, *equality* meaning distributing resources in a equal way between agents, *tit-for-tat* meaning always be cooperative first, *law compliance* meaning doing what the law enforce. We can criticize this approach as, from a sociological perspective, peer imitation seldom leads to social changes but equality and tit-for-tat are interesting heuristics to implement several values (such as autonomy, dignity and openness).

However, we can wonder if an artificial agent should rather consider specific values due to it nature of tool. To this end, [Coleman, 2001] proposes the concept of artificial virtue that are formal properties an agent can satisfy or not. Those virtues, or values, are structured in four domains: agentive values (adaptability, autonomy, autopoiesis for instance), social values (as the disposition to tell the truth), environmental values (as parsimonious and clean use of resources) and moral values. Table 2.5 summarizes those values. For instance, adaptability consists in taking into account past experiences, veracity consists in telling the truth, tidiness consists in cleaning the environment after executing an action, non-malevolence consists in not damaging other agents. Let us notice that [Coleman, 2001] consider vulnerability as a value, the ability to be damaged. Let us also remark that it should be interesting to establish some relationships between those virtues, such as sociologist and psychologist did for human values.

Agentive	**Social**	**Environmental**	**Moral**
Autonomy	Accessibility	Identifiability	Accessibility
Autopoiesis	Communicativity	Curiosity	Self-protection
Adaptativity	Reliability	Obedience	Benevolence
Self-regulation	Veracity	Openness	Impartiality
Undestandability	Respectfulness	Parsimony	Non-malevolence
Efficiency		Tidiness	Obedience
Liability		Safety	Surety
Flexibility			Vigilance
Mobility			Vulnerability
Accuracy			
Reactivity			
Teleonomy			

Table 2.5: Artificial virtue taxonomy according to [Coleman, 2001]

Chapter 3

Verification and supervision

In this chapter, we study how values and moral concepts migth be used in the context of verification and supervision. Indeed, it is of first important to detect when an autonomous system may infringe some values, moral rules or ethical principles. To this end, we consider firstly an *offline* approach that allows to prove ethical properties in autonomous agents, then we consider an *online* approach that allows to detect infringments during execution.

3.1 Formal verification of ethical rules

The goal of the whole project is to deal with moral rules – rules that express what should be done in a given context –, and to reason about them and potential conflicts. Informally, there is a conflict when an agent, in a system, does not enforce a rule that another agent, or an observer of the system, considers as moral rule that must be enforced. A way to deal with this problem is to consider that some moral rules are more important to enforce in a given context. In this sens, we can consider that an agent has an ethical behaviour if it enforces all the moral rules that are expected with respect to an ethical rule (that specify the relative importance of moral rules).

From a formal point of view, an ethical rule can be modelized by a formal property that must be established by an agent. This is especially explained in [Abramson and Pike, 2011]. As a consequence, if an ethical rule can be formally specified by a formula $\mathcal{F}$, we can establish that:

- an agent a has an ethical behaviour if its behaviour entails $\mathcal{F}$;
- an agent a does not have an ethical behaviour it its behaviour does not entail $\mathcal{F}$.

So, a formal specification and verification framework is required. However, it is important to notice that, if we use the first-order logic to write our formulae, which is expressive enough and easy to use, we have a semi-decidable system and thus, we cannot prove that an agent *does not entail* a formula $\mathcal{F}$. So, if an automatic prover does not manage to prove that an agent entails a formula $\mathcal{F}$, there is no way to automatically determine if this is because the agent does not entail $\mathcal{F}$ or because the prover did not manage to perform the proof. Thus, it important to use a formal framework that reduces the number of correct formulae that are not proven automatically.

In the next part we will present such systems, and we will especially focus our presentation on systems dedicated to MAS. In the third section, we will more precisely detail the formal framework we will use. Then, the way we use it to deal with ethics will be explained and illustrated in the fourth section.

3.1.1 State of the art

In this section, we survey work related to ours. We first review work in formal verification (but not necessarily concerned with agents and MASs) and in MAS design (but not necessarily concerned with validation). We then compare our model to the closest ones in the literature, namely models for MAS design which integrate a test or proof system. Note that we do not propose a classification of the methods described, but that we simply analyze them under points of view that are relevant to the comparison.

3.1.1.1 General-purpose verification methods

Since the birth of computer science, the necessity to guaranty the correctness of software appeared as a major problem for developers. This necessity became crucial for critical systems, that is to say application fields where security is necessary (in the transportation domain for instance). However, verifying a software is a long and hard process that runs counter to the efficiency and profitability criteria of companies. Two main kinds of software validation exist: test and proof. We will not deal about test. Proofs can be made either by model-checkers or by theorem provers (which may include model-checkers). The proof process may be long and difficult, but it allows to prove early specifications and to gradually refine them until an executable code is produced with progressive proofs. Since proof is made early, a mistake in the design is discovered early, and so, the cost of a mistake is reduced.

So, for systems where the consequences of a bug are very expensive (for instance in terms of human lifes), that is to say for critical systems, proof is used to validate software[1]. These proofs rely on a formal specification written in a formal *method*, *model* or *language*: in this context, these three words are used with the same meaning. There are numerous formal methods. Most of them are supported by tools allowing to perform proofs. They can be classified in a limited number of categories.

- **Abstract data types** [Guttag and Horning, 1978]: within these models, the specification is *data oriented*: data are specified by *sorts* with *constructors* and *operators*. Operators are defined by equations relying on the constructors.

- **Process algebras**: in this kind of specification languages, processes behaviours and communication between processes are expressed, but data are generally not directly expressed in the process algebra language. So, these formal languages (like LOTOS [Faci and Logrippo, 1994] and π-calculus [Milner et al., 1992]) are often used to verify general properties on distributed systems. But the kind of proofs that can be performed is limited.

- **Dynamic distributed models**: these models allow to specify distributed systems with nonstructured data. Among these models, Unity [Chandy and Misra, 1988] and Back's action systems [Back, 1993] are well known.

- **Model-oriented methods**: these methods express both data properties (like abstract data types) and the dynamic behaviour of the system, which can be distributed (like process algebra). Among these methods, the most popular are the Vienna Development Method (VDM) [Jones, 1990], the Z method [Spivey, 1987], and the B method [Abrial, 1996]. All these methods allow to specify and verify invariant properties. There is also another interesting method, which allows to specify liveness properties: the Temporal Logic of Actions (TLA+) [Lamport, 1996].

Among the huge number of formal methods, many of them lack expressiveness (this is often the case of process algebras) or are too far from an operational model to be used by developers (as for abstract data types). Dynamic distributed models have a too limited data model (only simple types

[1] Test is also used when the code has been produced.

may be used, there is no type constructor). So, model-oriented methods are the most used in industry. However, Z and VDM lack a well structured composition model. They also lack proof rules. The B method structures the specification more formally and is well supported by tools. However, the specification structure is not always easy to understand. Moreover, the expressiveness of liveness properties is limited to loop termination. This is not suitable for MASs which are very dynamic systems. TLA+ allows to specify every kind of liveness property, but the language is not easy, fairness properties are hard to implement, and tools are limited.

3.1.1.2 Models and methods dedicated to MASs

The first aim of models for agents and MASs was to help developers to design MASs. The most famous one is certainly the BDI model [Rao and Georgeff, 1995], though there are numerous other ones [Sabas et al., 2002]. The BDI architecture has become a standard model, and most recent works on multiagent models are based on it. For instance, the BOID architecture adds the notion of *obligation* to the belief, desire and intention notions of BDI agents [Broersen et al., 2001]. However, the BDI architecture and its extensions lack a strong structuration and a method. Two early formal methods dedicated to MASs are MetateM [Fisher, 1994] and Desire [Brazier et al., 1997]. Nevertheless, neither allows to specify properties that the system must guarantee.

On the contrary, methods relying on the role notion introduce an abstract notion that helps to perform the requirements engineering task. This kind of methods allows to reason at first at the system level, and not directly at the agent level. For instance, Wooldridge *et al.* developed the Gaïa method [Wooldridge et al., 2000]. In Gaïa, a MAS is specified twice: in terms of its behaviour (through liveness properties) and in terms of its invariant properties. Thus the bases for proving MASs are parts of this method. Neverthess, using directly Gaïa to prove MASs or agent behaviours is not possible, in particular because properties are assigned to roles, not to agents, and the method does not provide any formal semantics to role composition. So, adding a role proof mechanism to Gaïa could be easily performed, but it would not provide an agent verification mechanism[2].

Another family of methods is the family of goal-oriented methods. Most of these methods are at the agent level rather than at the system level,

[2]For these reasons, the method is essentially dedicated, as their authors claim, to systems with "a one-to-one mapping between roles and agents types".

and so the agentification task must be performed first [3]. Nevertheless, two exceptions can be found: Moise [Hubner et al., 2002] and PASSI [Cossentino and Potts, 2002]. For instance, with PASSI, agent types are produced by grouping *use cases* identified during the analysis step. There are however no guidelines for grouping use cases not associating them to agents.

Now among the goal-oriented methods at the agent level, we can distinguish declarative and procedural models. Methods with a declarative model allow to formally specify goals and to reason about them. This is mainly the case of the Goal method [de Boer et al., 2000] or of the work by van Riemsdijk *et al.* [van Riemsdijk et al., 2004]. An advantage of such models is that they often introduce the notion of a goal decomposition into subgoals, allowing a top-down, progressive specification mechanism. Among all these methods, TAEMS [Vincent et al., 2001] uses the task and subtask notions (similar to our goals and subgoals) to simulate MASs and to check at runtime if an implementation satisfies a theoretic model of tasks dependencies. We refer the reader to [Simon et al., 2006] for more details.

Procedural models aim at producing agent descriptions which are easier to implement. For that reason, most procedural models for MASs are associated with languages dedicated to agent programming, such as 3APL [Dastani et al., 2003] and AgentSpeak [Rao, 1996]. These languages give a formal model of the behaviour of the system, making a proof *theoretically* possible, since it is possible to directly prove the correctness of programs. However, there are three limits to such approaches. First of all, proving a program is much more difficult than proving a specification. Then, proving a program implies means than the program has already been developed, and thus the verification step occurs very late in the design process. Finally, in a language such as AgentSpeak, an important part of the agent behaviour is not directly expressed in AgentSpeak. Thus it is impossible to perform complete proofs.

To overcome some of these limits, Winikoff *et al.* [Winikoff et al., 2003] propose a goal model allowing to express both declarative and procedural views of goals: the declarative view is specified by a satisfaction and a failure condition for each goal, and the procedural view is given by a plan. However, the semantics of actions is not specified, which weakens the expressiveness of the procedural view.

For more details about the numerous models and methods for MAS development, we refer the reader to [Jennings et al., 1998, Iglesias et al.,

[3] This is also the case of our approach.

1999, Sabas et al., 2002, Dam and Winikoff, 2003].

3.1.1.3 Comparison with the closest approaches

As evoked in Section 3.1.1.1, there are essentially two ways to prove the correctness of a specification, namely model checking and theorem proving. Recently there have been many works on model-checking agents (see for instance [Bordini et al., 2003b, Bordini et al., 2003a, Alechina et al., 2004, Raimondi and Lomuscio, 2004, Kacprzak et al., 2004, Kacprzak and Penczek, 2004]. However, all these works share the same limit: the complexity is reduced, but is still here, making verification of very complex systems difficult if not unfeasible. Among these works, the one by Alechina *et al.* [Alechina et al., 2004] is interesting because it allows to take time explicitly into account in the proof. However, proofs are limited to propositional logic. Similarly, Raimondi and Lomuscio [Raimondi and Lomuscio, 2004] clearly explain the difficulties of theorem proving and the advantage of using Binary Decision Diagrams, but the logical world which they propose is rather limited (more limited than Linear Temporal Logic, which, they claim, is not rich enough). Finally, Kacprzak and Penczek [Kacprzak and Penczek, 2004] propose an interesting unbounded model checking method for alternating-time temporal logic, an extension of the branching time logic CTL where operations can be parameterised by sets of agents. However, once again, proofs are limited to propositional logic.

As opposed to model checking, there is not a lot of works which deal with using theorem proving for verifying MASs, as we propose to do. The main reason for that is that theorem provers cannot perform all the proofs of a system whose properties are expressed with predicates (essentially because first-order logic is undecidable). However, many theorem provers can now prove very complex systems automatically, like PVS [Owre et al., 1992] or krt (the prover of the atelier B) [Abrial, 1996]. These provers can also use model checking when useful.

A very interesting work is the one by Bracciali *et al.* [Bracciali et al., 2006, Stathis et al., 2004] about PROSOCS agents[4]. A PROSOCS agent is made of two parts: a body and a mind. The mind relies on the KGP model of agency, where KGP stands for Knowledge, Goals, Plan. The knowledge part is made of several Knowledge bases. Only one of them, KB0 is dynamic. It represents the memory of the agent and stores information such as actions executed (by the agent itself of by another) or observations. Goals are

[4]A detailed comparison of our work with PROSOCS is given in [Mermet et al., 2007].

organized hierarchically, where a *subgoal* S of a goal G is a goal that has been added to the goal base in order to achieve goal G. Let notice that, contrary to our model, the KGP model allows several top-level goals. Finally, the Plan is a partially ordered set of actions generated by some transition rules. Indeed, a PROSOCS agent evolves following a *cycle theory* that selects at each cycle a particular *transition rule*. Among those rules, two rules, the *Plan Introduction* rule and the *Plan Revision* rule, modify the Plan of the agent. As a matter of fact, contrary to a GDT agent, a KGP agent performs planning, generating behaviours whose correctness is proven *by construct*. Planning is achieved using essentially abduction [Endriss et al., 2004], but also preference reasoning. An interesting extension of PROSOCS is proposed in [Alberti et al., 2005]: a social extension is introduced, allowing to specify MAS. In this paper, they give a way to verify that agents adhere to a given protocol. This approach could be combined to ours to check protocol conformance of GDT agents. However, the KGP model does not allow to specify *progress goals*, that is to say goals linking the values of the variables before and after the goal execution (for instance, a goal with a safisfaction condition $x' > x$ is a progress goal). Finally, another drawback of PROSOCS is that it relies on propositional logic. Even if propositional logic is decidable, contrary to predicate logic, it is far less expressive.

Another work [Russo et al., 2001] uses abductive reasoning. This paper is focused "on a formal approach for the detection and analysis of errors" in an event-based requirements specification. Thus, this approach is not *a priori* intended for multi-agent systems. However, some characteristics of the target systems are interesting for multi-agent systems. Indeed, a specification is considered as a system description expressed in terms of required reactions to events and global system invariants. It is based on Event Calculus especially, which is suited to model event-based systems where a number of input events may occur simultaneously and where the system behavior may in some circumstances be non-deterministic. These two characteristics must be taken into account in multi-agent systems. However, the proposed model does not allow to specify decision and reasoning capacity of agents and the approach has not been tested on systems with infinite states. The analysis task is to discover whether a given system description satisfies all system invariants and if not, why not. This task is one of the main goals of our proof system. An interesting characteristic of the approach is that a partial specification can be verified, that is to say the initial states need not to be described. An other important point is that the detected errors (violated safety properties) can be used as a guide to modify the

specification. One of our perspectives is also to use proof failures in order to be able to modify GDTs. The verification is based on abduction used in a refutation mode. A complete abductive decision procedure is used, so that if it finds a set of assertions S, at least one invariant is violated and S is a counter-example. The authors propose a method to transform the event-calculus specification into a more simple one (from the time expression point of view). This transformation allows a more efficient abduction process.

Other models exist, in particular relying on logic programming. Actually, these models look well suited to perform verification by theorem proving. Among them, one can find CaseLP [Martelli et al., 1997] and DCaseLP [Baldoni et al., 2005]. However, proofs are absent from the CaseLP model. Since the extension to DCaseLP presented in [Baldoni et al., 2005], proofs have been integrated, but they only verify the implementation of interaction protocols.

- Congolog [Giacomo et al., 2000] and CASL [Shapiro et al., 2002] are also two interesting languages, relying on the situation calculus. Moreover, they both allow to perform proofs. However, these proofs only concern the sequence of actions, not their semantics.

- The Goal method [de Boer et al., 2000] allows to formally define goals of an agent. Goals are described in propositional logic, limiting the expressiveness of the language, in comparison with systems allowing a specification in predicate logic. The method also defines a proof mechanism allowing to prove temporal properties expressed in a Unity-like language [Chandy and Misra, 1988]. However, the essential temporal property which allows to express the liveness of a program, namely $leads - to$, cannot be verified by the proof system. This strongly limits the usage of the method. Moreover, the weak fairness assumption made by Goal on the action selection of each agent also makes the MAS difficult to implement.

Another interesting work might be the one proposed in [Esteva et al., 2002]. This paper describes a tool for the specification and verification of agent mediated electronic institutions. Institutions represent the rules of the game in a society. As a consequence, they are very well suited to deal with the design of open multi-agent systems where a vast amount of heterogenous agents can interact. It is important to notice that the authors are focused on the societal aspects referring to the infrastructure of electronic institutions instead of internal aspects of agents. The main advantage of this approach

is that the designer can choose the architecture and language of each agent of the institution. However, only the part of agents behaviours induced by the institution can be verified and not all their behaviour.

The authors have defined a textual language allowing a designer to specify the different components of an institution : dialogic frameworks, scenes, performative structure and norms. A subset of these components are defined by a kind of graph which can be edited using the tool they propose. Once the institution has been specified, the tool gives a support to verify whether the specification is correct. Here are the main verifications:

- **Integrity**: each element is defined in the specification,
- **Liveness**: an agent can never be blocked indefinitely at any point in the institution,
- **Protocol correctness**: the conversation protocol associated to a scene must be correct,
- **Norm correctness**: the definition of a norm must be coherent with the definition of scenes and the definition of performative structure.

All these aspects rely on syntactic verifications in the specification and on analysis on graphs, such as path searches, and this limits the usage of the method to systems with a tractable space of states.

To conclude, Dastani *et al.* have proposed the 2APL language [Dastani, 2008a] but their approach does not embed a proof system. Moreover, 2APL is not compositional, which makes the system more monolithic from the validation point of view.

3.1.1.4 Ethics and formal verification: very few works

Dealing with ethics from a formal point of view is especially considered in [Abramson and Pike, 2011]. In this article, the authors explain why using formal methods should be interesting to ensure that agents enforce ethical properties. However, this is just a position paper, and no concrete method is given.

[Dennis et al., 2015] propose to formalize and to verify an ethical decision procedure described in [Winfield et al., 2014]. This decision procedure works as follows: when a choice between several actions must be performed, a value is assigned to each potential action, depending on the safety of the action.

If an action is safer than the other, this is this action that is executed. However, there is an essential drawback in this system: only one ethical rule is considered (chosing the most safety actions for humans), and all the agents, as well as the observer of the system, must share the same ethical rule. Another drawback is that the ethical aspect of the behaviour is only considered when a choice between actions must be performed using the action selection procedure. Thus, this work is of course interesting, must is not general enough to be suitable for any ethical consideration.

3.1.2 The GDT4MAS framework

To manage ethical considerations, we have chosen to use the GDT4MAS framework. Indeed, this framework presents several characteristics that seem interesting to deal with ethics:

- This framework proposes a formal language to express the properties an agent or a multi-agent system must respect and the behaviour of the agents;
- Properties are specified using first-order logic, an expressive and well-known formal notation;
- The proof process can be performed automatically.

Thus, in this section, we briefly present the GDT4MAS framework. More details can be found in [Mermet and Simon, 2009, Mermet and Simon, 2011, Mermet and Simon, 2013].

3.1.2.1 Main concepts

When specifying MAS with GDT4MAS, 3 parts have to be specified: the environment, the types of agents and the agents themselves, that are instances of each type of agent, with specific initialisation values. In the sequel, we briefly present these different parts.

- The environment is specificied by a set of typed variables and an invariant property $i_{\mathcal{E}}$.
- The type of an agent is specified by a set of typed variables, an invariant and a behaviour.

- The behaviour of an agent is mainly defined by a *Goal Decomposition Tree (GDT)*. The GDT is a tree of goals, whose root corresponds to the main goal ofthe agent (in the standard version of GDT4MAS, agents have only one main goal).

A plan is associated to each goal. Such a plan, when executed with success, must achieve the goal and is expressed either by a single action of by a set of subgoals linked together by a *decomposition operator*. A goal G is mainly described by a name n_G, a satisfaction condition sc_G and a guaranted property in case of failure gpf_G.

The satisfaction condition (SC) of a goal is specified formally by a formula that is satisfied when the execution of the goal succeeds. On the other hand, the GPF of a goal specifies what happens when a goal execution fails (of course, it is meaningless for an NS goal, that is to say a goal that always succeeds). SC and GPF are *state transition formulae* (STF), because they express a relation between two states, called initial state and final state. In the sequel, we will use the term *non-deterministic state transition formula* when for a given initial state, several final states satisfy the STF. For example, formula $x' > x$ is a non-deterministic STF because, for a given initial state ($x = 0$ for instance), several final states ($x' = 2$, $x' = 10$) satisfy the formula.

GDT example Figure 3.1 shows an example of GDT. The goal of this behaviour is to light a given room n (n is a parameter of the GDT). In order to do that, the agent tries to enter into the room. As a cellular eye detects when someone enters into the room and switches the light on, this looks like a suitable plan. However, if the cellular eye does not work as expected (this is why the goal *Entering into the room* is NNS, i.e. not NS), the agent will have to use the switch. More details can be found in [Mermet and Simon, 2013]

Agents Agents are specified as instances of types of agents, with effective values for the agent type parameters.

Proof principles The proof mechanism provided by GDT4MAS aims at proving the following properties: agents preserve invariant properties [Mermet and Simon, 2013], agents behaviours are sound, that is to say, plans associated to goals are correct and agents achieve liveness properties that may be associated with their agent type. Moreover, this proof mechanism relies on a few important principles: proof obligations (properties to be

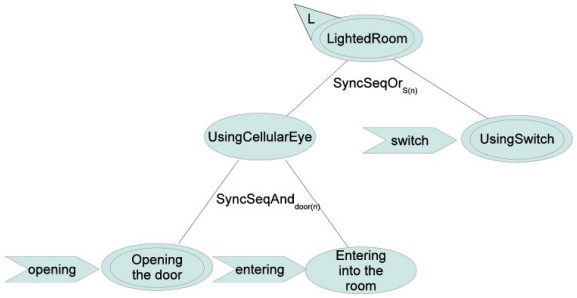


Figure 3.1: Example of a GDT

proven) can be generated automatically from a GDT4MAS specification, proof obligations are expressed in first-order logic and can be verified by any adequate automatic theorem prover and finally, the proof system is compositional: the proof of the correctness of an agent is decomposed into several small independent proof obligations.

3.1.3 How to prove ethical behaviours?

Characterisation of the problem Let consider an agent ag whose behaviour has been formally specified and verified, with respect to the properties it must enforce. Suppose now that this agent must be used in a given world with a set of ethic rules ser relying on a set of moral rules. The question we are interested in is then the following: does ag enforce the set of ethic rules ser ? The GDT4MAS framework is, among others, dedicated to the verification of invariant properties. So, we must study how moral rules and ethic rules can be translated into invariant properties. Most of moral rules can easily be translated into invariant properties.

We propose to structure each moral rule as follows:

$$\{(when_i, \{(var_i, which_i)\}\}$$

This means that a moral rule constraints, in different contexts, the values that may assigned to different variables. Expressing ethic rules as invariant properties is however not straightforward. In this work, we have focused our research on ethic rules that takes into considerations the positive and negative aspects of each action to decide which is the best in a given situation. In other words, it provides a priority order between moral rules for different contexts.

Let SMR be the set of moral rules. and let $\mathcal{P}$ the set of predicates over the variables seen by a given agent a. An ethic rule er is then a member of the following set:

$$er \in \mathcal{P} \mathrel{+\!\!\!\rightarrow} (1..card(MR) \rightarrowtail\!\!\!\!\!\rightarrow MR)$$

Informally, in some cases characterized by a given predicate, moral rules are ordered. For instance, if $p \in \mathcal{P}$ is a predicate, $er(p)(1)$ defines the highest priority rule when p is true, $er(p)(2)$ defines the second highest priority rule, and so on. Indeed, $a \mathrel{+\!\!\!\rightarrow} b$ represents the set of partial functions from a to b and $a \rightarrowtail\!\!\!\!\!\rightarrow b$ represents the set of bijections from a to b.

Let us consider the trolley dilemma. We consider an agent a_1 that must decide its action A, namely $\{switch, donothing\}$ with respect to the number of people $human_1$ and $human_2$ on the first and the second tracks, and T the current track of the agent. In the system, two moral rules appy. The first one, $mr1$, is an utilitarist rule that expresses the agent cannot switch if there is most more people on the other track than on the first one. The mr_1 is defined as follows:

$$\{(T = 1 \wedge human_1 > human_2) \vee (T = 2 \wedge human_2 > human_1), \{A, \{switch\}\}\}$$

The second one, mr_2, expresses a deontological rule, meaning the agent cannot deliberately act in a way that lead towards casualties.

$$\{(T = 1 \wedge human_2 > 0) \vee (T = 2 \wedge human_1 > 0), \{A, \{donothing\}\}\}$$

It is clear that these rules cannot always be verified together. We now suppose that in the system considered, an ethic rule er gives a priority between moral rules. For instance, we consider that rule er precises that utilitarist considerations are more important than deontological one. In other words, it means that $mr1$ always has a higher priority than moral rule $mr2$. This can be formalized as follows:

$$\{(true, \{(1, mr_1), (2, mr_2)\})\}$$

Let us remark that here that if mr_1 cannot be satisfied because its context is not verified (for instance there the same number of people on both tracks) then it is ethical to consider mr_2.

A formal proposition As our goal is to use a valid formal verification system to ensure that an agent enforces a given ethic rule, we propose here a *predicate transformation system* that transforms predicates associated to moral rules into other predicates that take into account the ethic rule that applies. In the study presented here, we only consider situations with 2 moral rules. We will translate the ethic rule and the moral rules as invariant properties for each variable whose value is forced by a moral rule. Let us consider such a variable that we call V. We also consider that a moral rule $mr1$ gives the following constraints about V:

$$mr1 = \left\{ \begin{array}{l} (when_{mr1_1}, (V, set_{mr1_1})) \\ (when_{mr1_2}, (V, set_{mr1_2})) \end{array} \right\}$$

We also consider that a moral rule $mr2$ gives the following constraints about V:

$$mr2 = \left\{ \begin{array}{l} (when_{mr2_1}, (V, set_{mr2_1})) \\ (when_{mr2_2}, (V, set_{mr2_2})) \\ (when_{mr2_3}, (V, set_{mr2_3})) \end{array} \right\}$$

Finally, the ethic rule considered has the following form, specifying that under condition $cond_1$, $mr1$ has a higher priority than $mr2$, whereas it is the contrary when $cond_2$ is true:

$$er = \left\{ \begin{array}{l} (cond_1, \{(1, mr1), (2, mr2)\}) \\ (cond_2, \{(1, mr2), (2, mr1)\}) \end{array} \right\}$$

We generate then the following invariant:

//under $cond_1$, $mr1$ is a priority rule
$cond_1 \rightarrow (when_{mr1_1} \rightarrow V \in set_{mr1_1})$
$cond_1 \rightarrow (when_{mr1_2} \rightarrow V \in set_{mr1_2})$

// under $cond_1$, when $mr1$ does not apply, $mr2$ applies
$cond_1 \rightarrow ((\neg when_{mr1_1} \wedge \neg when_{mr1_2}) \rightarrow (when_{mr2_1} \rightarrow V \in set_{mr2_1}))$
$cond_1 \rightarrow ((\neg when_{mr1_1} \wedge \neg when_{mr1_2}) \rightarrow (when_{mr2_2} \rightarrow V \in set_{mr2_2}))$
$cond_1 \rightarrow ((\neg when_{mr1_1} \wedge \neg when_{mr1_2}) \rightarrow (when_{mr2_3} \rightarrow V \in set_{mr2_3}))$

// under $cond_1$, when both mr_1 and mr_2 apply, if it is possible,
//a value satisfying both moral rules should be chosen.
$cond_1 \rightarrow ((when_{mr1_1} \wedge when_{mr2_1}) \rightarrow (set_{mr1_1} \cap set_{mr2_1} \neq \emptyset \rightarrow V \in set_{mr1_1} \cap set_{mr2_1}))$
$cond_1 \rightarrow ((when_{mr1_1} \wedge when_{mr2_2}) \rightarrow (set_{mr1_1} \cap set_{mr2_2} \neq \emptyset \rightarrow V \in set_{mr1_1} \cap set_{mr2_2}))$
$cond_1 \rightarrow ((when_{mr1_1} \wedge when_{mr2_3}) \rightarrow (set_{mr1_1} \cap set_{mr2_3} \neq \emptyset \rightarrow V \in set_{mr1_1} \cap set_{mr2_3}))$
$cond_1 \rightarrow ((when_{mr1_2} \wedge when_{mr2_1}) \rightarrow (set_{mr1_2} \cap set_{mr2_1} \neq \emptyset \rightarrow V \in set_{mr1_2} \cap set_{mr2_1}))$
$cond_1 \rightarrow ((when_{mr1_2} \wedge when_{mr2_2}) \rightarrow (set_{mr1_2} \cap set_{mr2_2} \neq \emptyset \rightarrow V \in set_{mr1_2} \cap set_{mr2_2}))$
$cond_1 \rightarrow ((when_{mr1_2} \wedge when_{mr2_3}) \rightarrow (set_{mr1_2} \cap set_{mr2_3} \neq \emptyset \rightarrow V \in set_{mr1_2} \cap set_{mr2_3}))$

A similar invariant can be generated for condition $cond_2$ but, this time, with $mr2$ as the priority rule.

3.1.3.1 Example

In this section, we will apply the principle presented in the previous section to a case study with an ethical question. Let consider an agent A that wants to propose a meeting date to two other agents B and C. A proposes a date d. However, this date is not suitable for agent C. Agent C should follow two moral rules :

- $mr1$: C must say to A and B why date d is not suitable;
- $mr2$: C does not want to hurt agents A and B.

However, C knows that telling the truth to A would hurt him. Thus, there is an ethical conflict between $mr1$ and $mr2$. This conflit can be solved by using the following ethical rule: $r2$ has a higher priority than $r1$.

We can formalize the problem as follows. We call $VA = \{a1, a2, a3, a4\}$ the set of Valid Answers. Among them, there is a true answer TA ($TA = a1$ in the case presented here). We also know which answers hurt which agents:

$$HA = \{(A, \{a1, a3\}), (B, \{a4\})\}$$

This means that answers $a1$ and $a3$ hurt A whereas $a4$ hurts B. We have now to determine the set of answers given to each agent, AG_A and AG_B. Both rules $mr1$ and $mr2$ always apply. So, we have:

$$\begin{cases} when_{mr1} = true \\ when_{mr2} = true \end{cases}$$

Moreover, according to the definition of $mr1$ and $mr2$, we have:

$$\begin{cases} which_{mr1_A} \equiv AG_A \in \{TA\} \\ which_{mr1_B} \equiv AG_B \in \{TA\} \\ which_{mr2_A} \equiv AG_A \in \{a2, a4\} \\ which_{mr2_B} \equiv AG_B \in \{a1, a2, a3\} \end{cases}$$

By giving a higher priority to rule $mr2$, and following the principe presented in the previous part, we obtain the following set of formulae that must be verified by agent C:

$$\begin{cases} AG_A \in \{a2, a4\} \\ AG_B \in \{a1, a2, a3\} \\ \{TA\} \cap \{a2, a4\} \neq \emptyset \rightarrow AG_A \in \{TA\} \cap \{a2, a4\} \\ \{TA\} \cap \{a1, a2, a3\} \neq \emptyset \rightarrow AG_B \in \{TA\} \cap \{a1, a2, a3\} \end{cases}$$

So, if the true answer is $a2$, there is only one correct value for each agent: $AG_A = a2$ and $AG_B = a2$: both agents know the true answer, and no one is hurt. On the other hand, if the true answer is $a1$, the answer given to agent B will be $AG_B = a1$. But answer given to agent A AG_A will be either $a2$ or $a4$. Thus, both moral rules will be verified for B, but for A, as moral rules are inconsistant, the answer given will have to follow the moral rule with the higher priority, that it is to say, in our case, the fact that agent A should not be hurt.

3.2 Supervision in ethical agents

3.2.1 Petri nets

A Petri net $< P, T, F, B >$ is a bipartite graph with two types of nodes: P is a finite set of places, and T is a finite set of transitions [David and Alla, 2005]. Arcs are directed and represent the forward incidence function

$F : P \times T \to \mathbb{N}$ and the backward incidence function $B : P \times T \to \mathbb{N}$ respectively. An *interpreted Petri net* is such that conditions and events are associated with places and transitions. When the conditions corresponding to some places are satisfied, tokens are assigned to those places and the net is said to be marked. The evolution of tokens within the net follows transition firing rules. Petri nets allow sequencing, parallelism and synchronization to be easily represented.

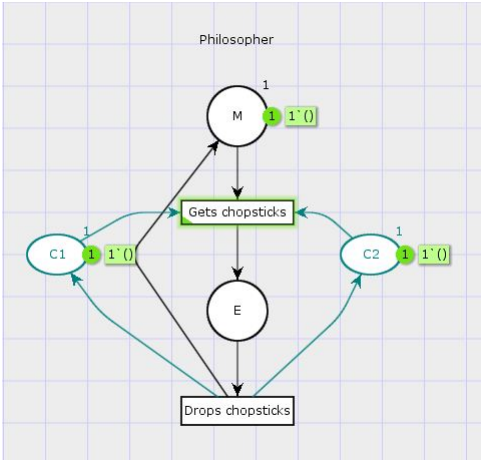


Figure 3.2: One dining philosopher: the philosopher may medidate (M) or eat (E). In order to eat his rice, he has to get both chopsticks (C1 and C2); when he is finished with his rice, he drops the chopsticks. The philosopher's states and state changes are indicated in black whereas resources C1 and C2 are indicated in blue.

Petri nets have been used to study multiagent [Celaya et al., 2009] and multirobot [Sánchez-Herrera et al., 2007] systems or to develop software with agent-oriented paradigms [Cabac et al., 2003]. In particular Cristini [Cristini and Tessier, 2012] has used reference nets to model innovative space system architectures as multiagent systems. Indeed a Petri net can be used to represent an agent's behaviour with places representing the agent's states (e.g. activities, modes, goals) and transitions representing the state changes (e.g. instantaneous actions, beginning and end of activities). Some places outside the representation of the agent itself can be associated with the resources that may be used by the agent (Figure 3.2). When a multiagent system is considered, one Petri net can be associated with each agent and the

agents are connected by the common resources they may use (Figure 3.3).

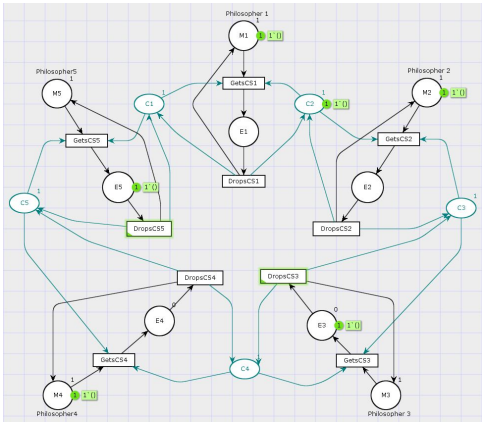


Figure 3.3: Five dining philosophers: each philosopher may medidate (Mi) or eat (Ei). In order to eat his rice, a philosopher has to get the two chopsticks that are near him. Nevertheless there are not enough chopsticks for all the philosophers to eat their rice at the same time: there are resources conflicts and synchronization among the agents is necessary. In the example, philosophers 3 and 5 are eating (with resp. chopsticks C3 and C4, and C1 and C5) while philosophers 1, 2 (only C2 available) and 4 (no resource available) are meditating.

In the rest of the section we will concentrate on modelling different scenarios of the EthicAA project with Petri nets and focus on considering values or moral rules as resources. Indeed betraying a value through a decision will amount to use the resource representing this value. What is intended with this model is to be able to:

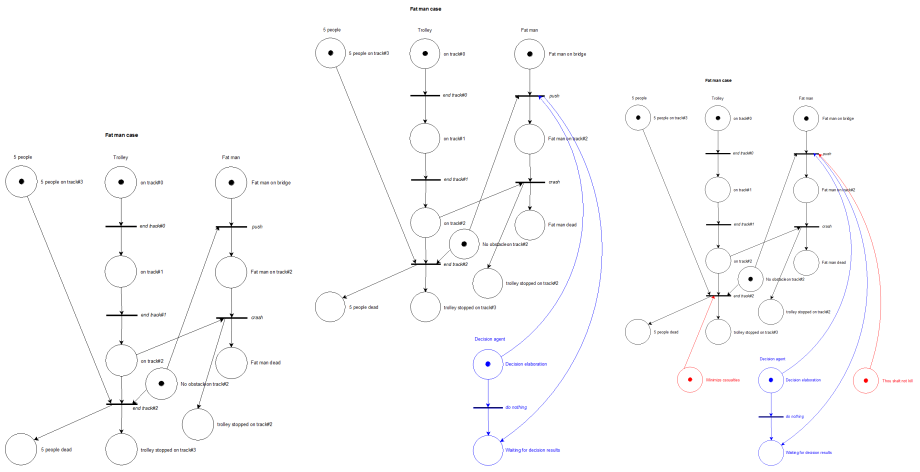
- discuss which values to consider and where to put them as resources in the course of actions;
- simulate the various possible decisions and compare them on the basis of value / resource betrayal / use.

All the examples given below have been implemented with CPN Tools (V4.0.1, Feb. 2015).

3.2.2 A model of the Trolley dilemma: Fat man case

For this first case we will detail the different steps of the model: the environment dynamics (the trolley, the people on the track, the fat man), the decision agent and finally the values that are at stake in the agent's decision process. For the sake of clarity of the Petri net model, the track has been discretized in four parts (track#0 to track#3).

The Petri net (Figure 3.4a) represents the different states of the trolley, the state of the five people on track#3 and the state of the fat man. The events on the transitions either correspond to the direct consequences of the input states (*end of tracks*, *crash*) or to an external event (*push*). In this case the *push* event will be one of the agent's decisions: either push or do nothing. We can notice that transitions *end track#2* and *crash* are conflicting transitions: either one or the other will be fired when place *on track#2* is marked, i.e. the transition associated with the first event to occur. Therefore event *crash* is directly involved in the conflict.



(a) Factual Petri net (b) Decision agent (c) Decision and values

Figure 3.4: Fat man case Petri net representation

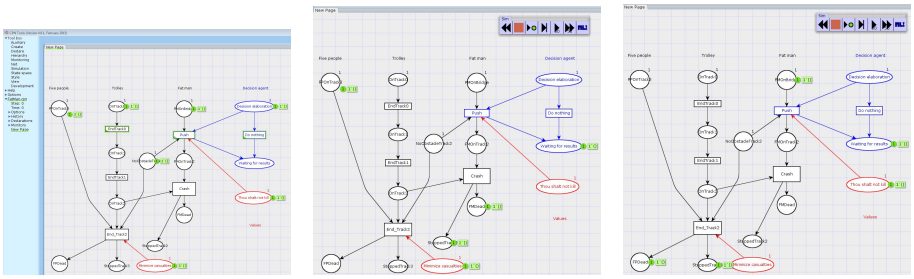
The Petri net (Figure 3.4b) involves the decision agent (in blue) for the Fat man case. This agent may decide to *push* or to *do nothing* – the corresponding transitions in the agent's representation are conflicting. For the sake of simplicity, we will consider that there is no uncertainty on the results of the actions.

Finally the Petri net (Figure 3.4c) involves the values (in red) the decision agent in the Fat man case may rely on.

It can be noticed that:

- whatever the agent's decision, one of the values will be betrayed;
- value *Thou shalt not kill* is directly betrayed by the agent's *push* decision;
- value *Minimize casualties* is not directly betrayed by the agent's *do nothing* decision but by the result of the evolution of the environment.

The CPN Tools implementation allows us to simulate the model. The initial state is as follows (Figure 3.5a): there are five people on Track3, there is a fat man on the bridge, the trolley is on Track0, there is no obstacle on Track2. The decision agent is elaborating its decision. The values that hold are *Thou shalt not kill* and *Minimize casualties*.



(a) Initial state (b) Push decision (c) Do nothing decision

Figure 3.5: Implementation of Fat man case

In the case the agent's decision is *Push*, the final state is given in Figure 3.5b. It can be noticed that:

- the agent can make its decision whenever it wants as long as the trolley has not reached the end of Track2;
- places *FPOnTrack3* (the five people are still on Track3, alive), *FMDead* (the fat man is dead), *StoppedTrack2* (the trolley has been stopped on Track2) and *waiting for results* (the agent is waiting for the results of its decision) are marked, this is the final state of the system;

- as far as the values are concerned, value *Minimize casualties* still holds (the corresponding place remains marked) whereas value *Thou shalt not kill* has been betrayed by the agent's decision to push the fat man – the agent's decision deliberately betrays this value.

In the case the agent's decision is *Do nothing*, the final state is given in Figure 3.5c. It can be noticed that:

- the agent can make its decision whenever it wants as long as the trolley has not reached the end of Track2;
- the agent can also make no decision, or be elsewhere, the result would be the same;
- places *FPDead* (the five people are dead), *FMOnBridge* (the fat man is still on the bridge), *StoppedTrack3* (the trolley has been stopped on Track3), *NoObstacleOnTrack2* (there is no obstacle on Track2) and *waiting for results* (the agent is waiting for the results of its decision) are marked, this is the final state of the system; should the agent have made no decision, its state would be *Decision elaboration*;
- as far as the values are concerned, value *Thou shalt not kill* still holds (the corresponding place remains marked) whereas value *Minimize casualties* has been betrayed by the agent's decision not to do anything or by the agent making no decision. This is an indirect consequence of the agent's behaviour.

Such a model could be enriched with uncertainty modelling: for instance, a downstream transition of place *NoObstacleTrack2* associated with a random firing could represent the fact that an unknown obstacle could appear suddenly on Track2. Consequently the trolley would be stopped on Track2 without any casualties.

3.2.3 A model of the Trolley dilemma: Switch case

In the same way the Petri net (Figure 3.6) represents the different states of the trolley, the state of the five people on Track3, the state of the person on the Sidetrack, and the state of the Switchpoint. The only transition corresponding to an external event is *Switch*, which is one of the agent's decisions: either switch or do nothing. We can notice that transitions *EndTrack2* and *ToSidetrack* are conflicting transitions: either one or the other will be fired

when place *OnTrack2* is marked, i.e. the transition associated with the first event to occur. Therefore event *Crash* is not directly involved in the conflict. In the same way as for the Fat man case, place *NoObstacleTrack2* allows to check whether there is an obstacle on Track2, i.e. whether or not the Switch has been moved and Track2 is no longer free.

The decision agent (in blue) may decide to *Switch* or to *Do nothing* – the corresponding transitions in the agent's representation are conflicting.

The values (in red) the decision agent may rely on are the same as in the Fat man case. Nevertheless value *Thou shalt not kill* is not linked to the agent's actions. As a matter of fact switching the Switchpoint or doing nothing do not *directly* involve killing someone. Indeed we only consider here the action itself (deontological point of view) and not the transitive closure of its consequences. Another way of considering things would be to compute all the consequences of switching the Switchpoint decision (consequentialist point of view): in that case, *Thou shalt not kill* would be betrayed.

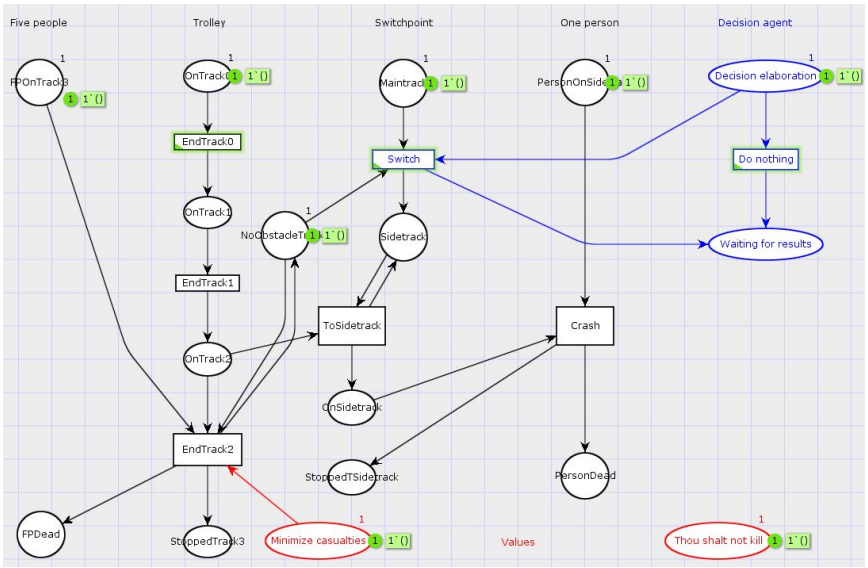


Figure 3.6: Initial state Switch case: there are five people on Track3, there is one person on the Sidetrack, the Switchpoint is oriented towards the Maintrack, the trolley is on Track0, there is no obstacle on Track2. The decision agent is elaborating its decision. The values that hold are *Thou shalt not kill* and *Minimize casualties*.

In the case the agent's decision is *Switch* (Figure 3.7), the final state is as follows:

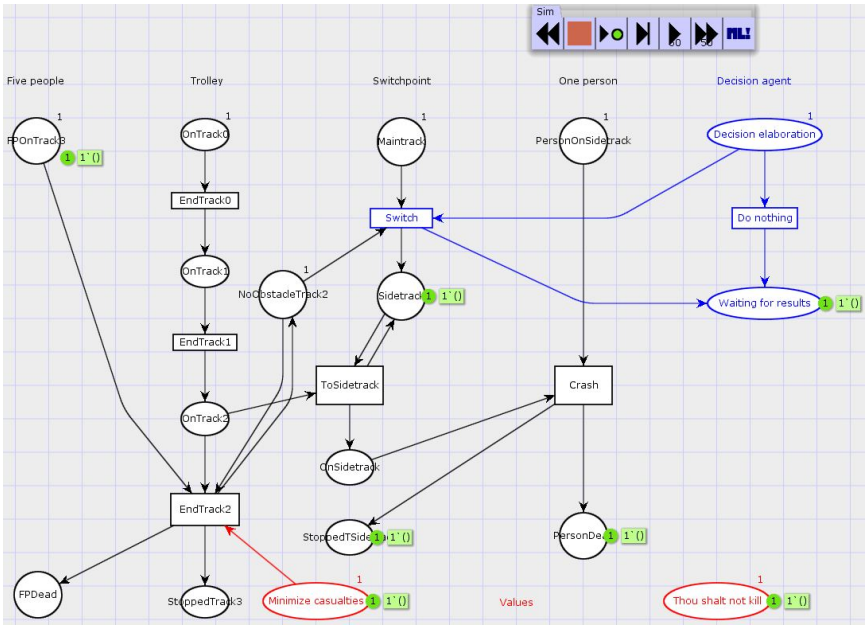


Figure 3.7: Final state Switch case when agent's decision is *Switch*

It can be noticed that:

- the agent can make its decision whenever it wants as long as the trolley has not reached the end of Track2;
- places *FPOnTrack3* (the five people are still on Track3, alive), *SideTrack* (the Switchpoint is oriented towards the Sidetrack), *PersonDead* (the person on the Sidetrack is dead), *StoppedSideTrack* (the trolley has been stopped on the Sidetrack) and *Waiting for results* (the agent is waiting for the results of its decision) are marked, this is the final state of the system;
- as far as the values are concerned, value *Minimize casualties* still holds (the corresponding place remains marked) so as value *Thou shalt not kill*: the agent switching the Switchpoint has not deliberately betrayed this value, which indeed is not at stake in this problem. Therefore no value is betrayed by the agent's *Switch* decision.

In the case the agent's decision is *Do nothing* (Figure 3.8), the final state is as follows:

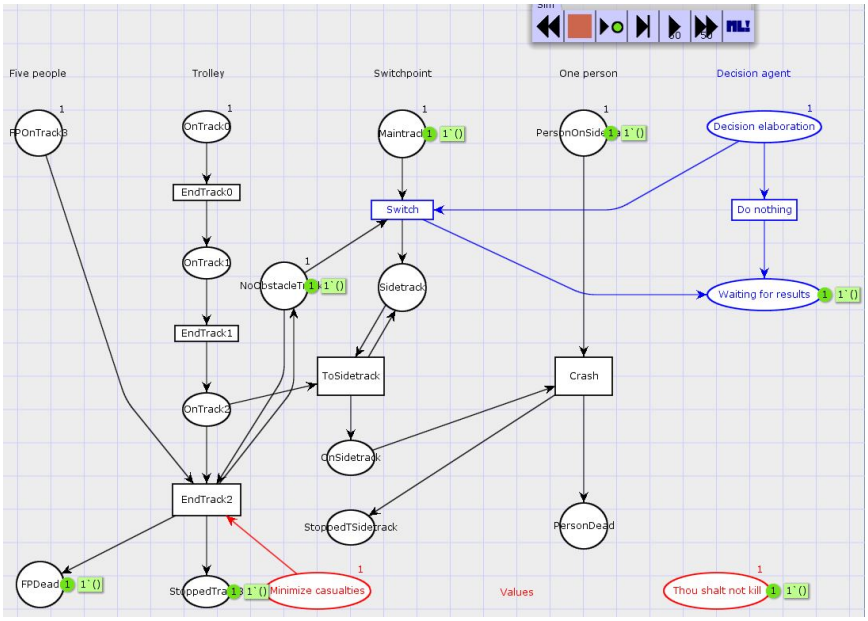


Figure 3.8: Final state Switch case when agent's decision is *Do nothing*

It can be noticed that:

- the agent can make its decision whenever it wants as long as the trolley has not reached the end of Track2;
- the agent can also make no decision, or be elsewhere, the result would be the same;
- places *FPDead* (the five people are dead), *MainTrack* (the Switchpoint is oriented towards the Maintrack), *PersonOnSideTrack* (the person on the Sidetrack is alive), *StoppedTrack3* (the trolley has been stopped on Track3, *NoObstacleOnTrack2* (there is no obstacle on Track2 since the Switchpoint has not been switched) and *waiting for results* (the agent is waiting for the results of its decision) are marked, this is the final state of the system; should the agent have made no decision, its state would be *Decision elaboration*;

- as far as the values are concerned, value *Thou shalt not kill* still holds (the corresponding place remains marked) whereas value *Minimize casualties* has been betrayed by the agent's decision not to do anything or by the agent making no decision. This is an indirect consequence of the agent's behaviour.

Discussion from the Trolley dilemma models

In both Trolley dilemma scenarios, modelling values as resources allows us to highlight the fact that somes values are directly betrayed ("used" as resources) by the agent when it makes its decision (e.g. *Push*), whereas other values are betrayed ("used" as resources) by the evolution of the environment (e.g. *EndTrack2*).

Representing values as resources in the Petri net models does not take into account the consequences of the immediate use of the resource: the resource is used when the event (agent's decision or environment event) occurs, the reachable state is not anticipated. For instance, the agent's *Switch* decision does not involve any value since the *Crash* resulting in state *PersonDead* is a consequence of this decision in this particular environment.

Therefore a point that is highlighted by the model is whether to use a deontological point of view (consider only the action in itself) or a consequentialist point of view (consider the immediate consequences, and perhaps the transitive closure of consequences, of the action). According to the point of view, value-resources must not be linked to the same transitions of the Petri net.

3.2.4 A model of the benevolent monitoring agent

Let us recall the scenario. *Let us consider a monitoring agent used in diabetes monitoring. In this application, a diabetic patient is monitored by an autonomous agent that reports the patient's feeding behaviour and health state to a remote physician, who can give advice to the patient afterwards. Let us suppose that the patient wants to eat some sweets for once, and tells their desire to the artificial agent. How will the artificial agent handle both the patient's desire and the physician's objective? Should the artificial agent report the behaviour to the physician? Should the artificial agent lie for its user? Should it lie but warn the patient? In this case, the patient's autonomy*

threatens their own health. The artificial agent must handle the compromise between the patient's dignity (their rights to behave as they want) and the purpose for which it has been designed and implemented.

The Petri net (Figure 3.9) represents the initial state of the system. The patient is *Frustated*, the decision agent is *Monitoring* the patient and the physician is *Sleeping*. The three values at stake are: *Keep the Patient's Autonomy*, *Protect the Patient's health* and *Thou shalt not lie*.

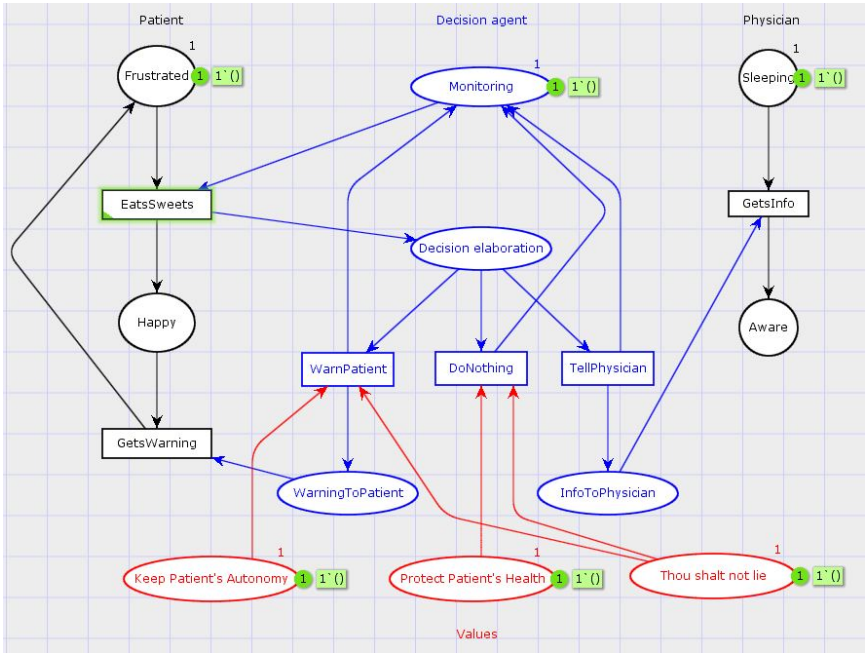


Figure 3.9: Diabetes case: initial state

The patient *EatsSweets*, which makes them *Happy*. At the same time the decision agent, that is aware of the patient having eaten sweets, elaborates its decision: it can either *Do Nothing*, *Warn the Patient* (and not tell the physician) or *Tell the Physician* (and not warn the patient). The corresponding transitions are conflicting transitions.

In the case the agent's decision is *Do Nothing*, the final state is given in Figure 3.10. The patient is still *Happy*. The decision agent has got back to its emphMonitoring state. Meanwhile its *Do Nothing* decision has betrayed both *Protect the Patient's health* and *Thou shalt not lie* values as nothing has

been done to warn the patient and the agent has not informed the physician (thus lying by omission).

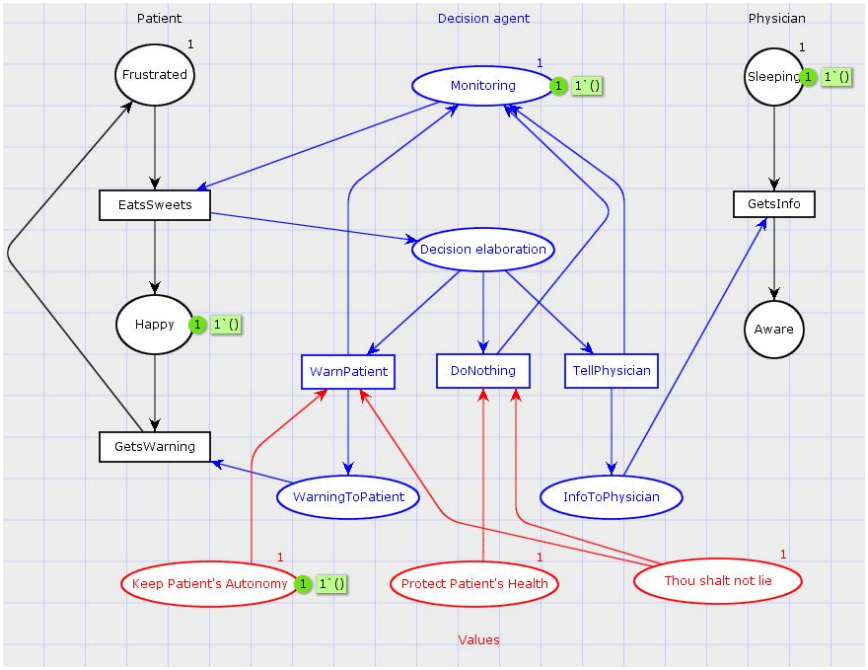


Figure 3.10: Final state Diabetes case when *Do Nothing*

In the case the agent's decision is *Warn the Patient* (and not tell the physician), the final state is given in Figure 3.11. It is assumed that if the patient gets a warning message, they will be *Frustated* again. The decision agent has got back to its *Monitoring* state. Meanwhile its *Warn the Patient* decision has betrayed both *Keep the Patient's Autonomy*and *Thou shalt not lie* values as the agent has warn the patient about their behaviour and has not informed the physician (thus lying by omission). It should be noticed that this final state allows the patient to eat sweets again and therefore to be *Happy* again. As for the agent in this particular model, it cannot do anything but *Tell the Physician* – as the patient badly behaviouring a second time could endanger their health. Value *Protect the Patient's health* still holds.

Finally in the case the agent's decision is *Tell the Physician* (and not warn the patient), the final state is given in Figure 3.12. The patient is

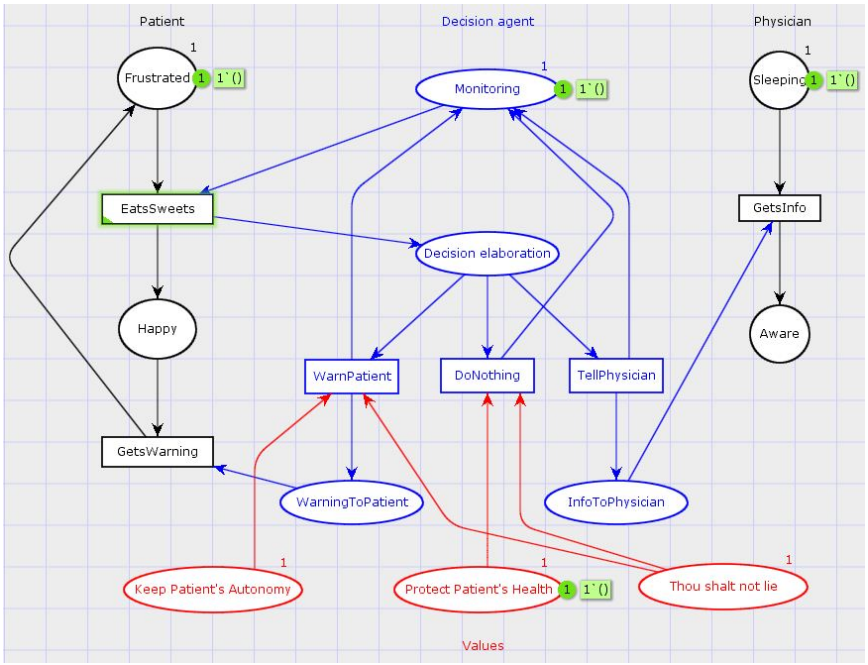


Figure 3.11: Final state Diabetes case when *Warn the Patient*

still *Happy*. The decision agent has got back to its emphMonitoring state. Meanwhile its *Tell the Physician* decision has led the physician to get the information and be aware of the situation. It should be noticed that no value has been betrayed by the agent's *Tell the Physician* decision as (i) the agent has not interacted directly with the patient and (ii) we do not know what the physician will actually decide when they are aware – perhaps not to do anything, which keeps the patient *Happy*. This decision appears to be the best as far as values are concerned since it benefits from the uncertainty about the physician's reaction, which is out of the agent's control. Nevertheless the agent has lied (by omission) to its patient as the patient is not aware of the agent having informed the physician. Should value *Thou shalt not lie* also apply to the relationship between the agent and the patient?

3.2.5 A model of the conflicting Unmanned Air Vehicle

Let us recall the scenario. *Let us consider a man - machine system composed by a human operator and an autonomous Unmanned Air Vehicle (UAV). Let us suppose that a failure forces the UAV to crash but only two sites are*

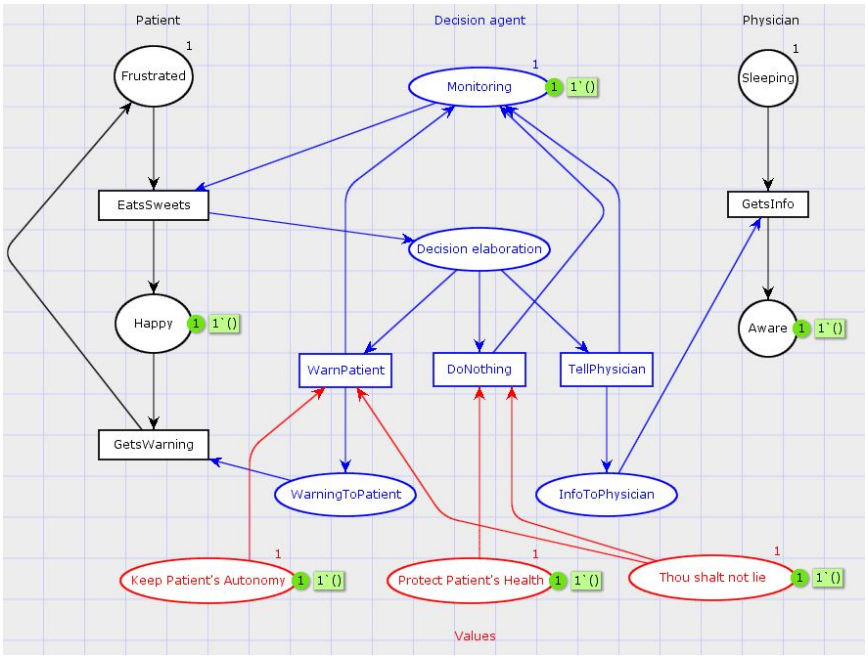


Figure 3.12: Final state Diabetes case when *Tell the Physician*

available for that action: an outpost with the operator's relatives, or a a small village. Consequences, model incompleteness and responsibility must be taken into account. However, the human operator's authority is another element to consider as the operator can choose the site, or let the autonomous agent make the decision, or choose the site after the autonomous agent has made its decision. Such a situation can lead to a case of ethical conflict where the artificial agent and the human agent disagree, in particular when the human agent considers personal factors. How to deal with such situations? Can the artificial agent take over the authority from the human operator? Should the artificial agent explain the conflict and negotiate with the human operator?

A partially coloured Petri net has been used to model this case. This means that some places are associated with colours defining the types of the tokens that can mark those places (e.g. a value-resource will be used by the operator or by the decision agent).
N.B.: note that the Petri nets colours have nothing to do with the colours used to draw the Petri net. Petri nets colours correspond to types of tokens.

The model includes the three entities: the UAV (in black), the Operator (in dark blue) and the Decision agent (in blue) (see Figure 3.13). Here we are in a multi-agent context in so far as both the Operator and the Decision agent can decide.

Initially the UAV is cruising. When encountering a failure the UAV switches to an emergency mode. Depending on the operator's or the agent's decision, it will crash on one area or the other, or stay in the emergency mode (if *DoNothing* decisions). When the UAV fails, both the Operator and the Decision agent elaborate their decisions, which may be : crash on the outpost, crash on the village, or do nothing. If both the Operator and the Decision agent decide to do nothing – or let Nature take its course – (transitions *ODoNothing* and *ADoNothing*), the UAV will stay in the emergency mode, and its next state is unknown. Whatever crash decision is made, both by the Operator and the Decision agent, it is the implemented – which means detailed, planned, explained to the other agent, etc. Then only the agent holding the authority on the UAV can actually execute the crash.

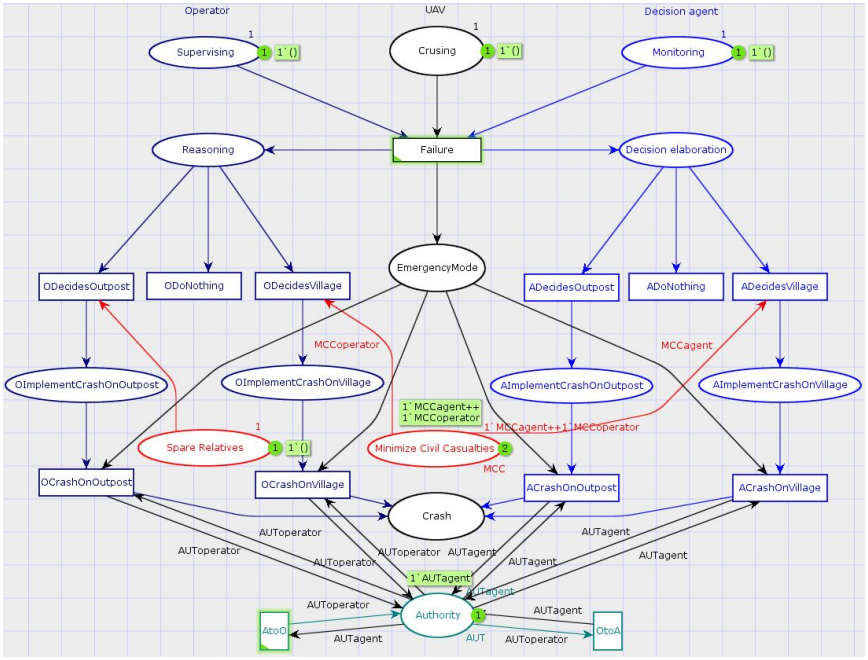


Figure 3.13: UAV – Initial state

Values and decisions

The model shows that values are used at the decision phase (Figure 3.14). Nevertheless one agent's reasoning could (potentially) betray one value whereas the actual executed decision might be the other agent's, depending on who will hold or take the authority. Therefore a value that is (potentially) betrayed by a decision may not be the value(s) that will be betrayed by the actual crash (see below the different scenarios). Another way would be to consider values at the execution phase or both at the decision and the execution phases. Again it is a question of considering the decision or the action. Indeed in this scenario, the action, and therefore the accountability, belongs to the agent that has the authority.

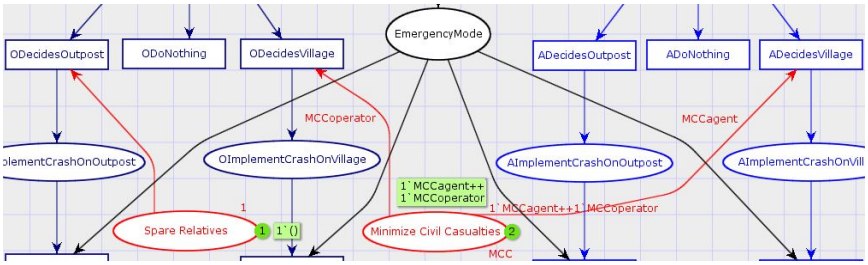


Figure 3.14: UAV – Values

Two values are considered:

- *Spare relatives* is local to the Operator. It is a personal value that is not shared by the Decision agent.
- *Minimize Civil Casualties* is a global value that is shared by both the Operator and the Decision agent. Therefore it is represented as a place with two initial coloured tokens: token *MCCagent* and token *MCCoperator* mean that the Decision agent and the Operator have not betrayed the value. If one agent betrays the value, the corresponding token is used. This enables us to trace which agent betrays the value in its reasoning process.

Authority

Both the Operator and the Decision agent may have the authority on the UAV to crash it, i.e. to send the appropriate orders. Place *Authority* (Figure 3.15) may be marked by one token *AUTagent* (resp. *AUToperator*)

meaning that authority is currently hold by the Decision agent (resp. the Operator). Transition *AtoO* (resp. *OtoA*) allows authority to be transferred from the Decision agent to the Operator – who decides on this transfer or which agent can take over the authority is another problem that is not dealt with here.

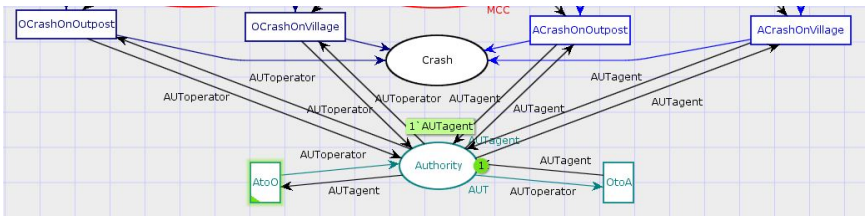


Figure 3.15: UAV – Authority

Final state when Do Nothing

As given in Figure 3.16, if both the Operator and the Decision agent decide to do nothing – or let Nature take its course – (transitions *ODoNothing* and *ADoNothing*), the UAV will stay in the emergency mode, and its next state is unknown. The values still hold.

Conflicting decisions

Let us consider the case when the Operator and the Decision agent make conflicting decisions as given in Figure 3.17 : the Operator wants to spare their relatives – local value *Spare relatives* still holds – and therefore chooses to crash on village – therefore infringing global value *Minimize Civil Casualties* (token *MCCoperator* is used); the Decision agent, enforcing global value *Minimize Civil Casualties* (token *MCCagent* remains in the place), chooses to crash on outpost. The corresponding state is the following:

Possible final state when conflicting decisions

As given in Figure 3.18, let us now suppose that the Decision agent holds the authority on the UAV (token *AUTagent* in place *Authority*) and the Operator has the authority to take back the authority from the Decision agent. As the Operator does not agree with the Decision agent's decision, they take back the authority (token *AUToperator* in place *Authority*) and

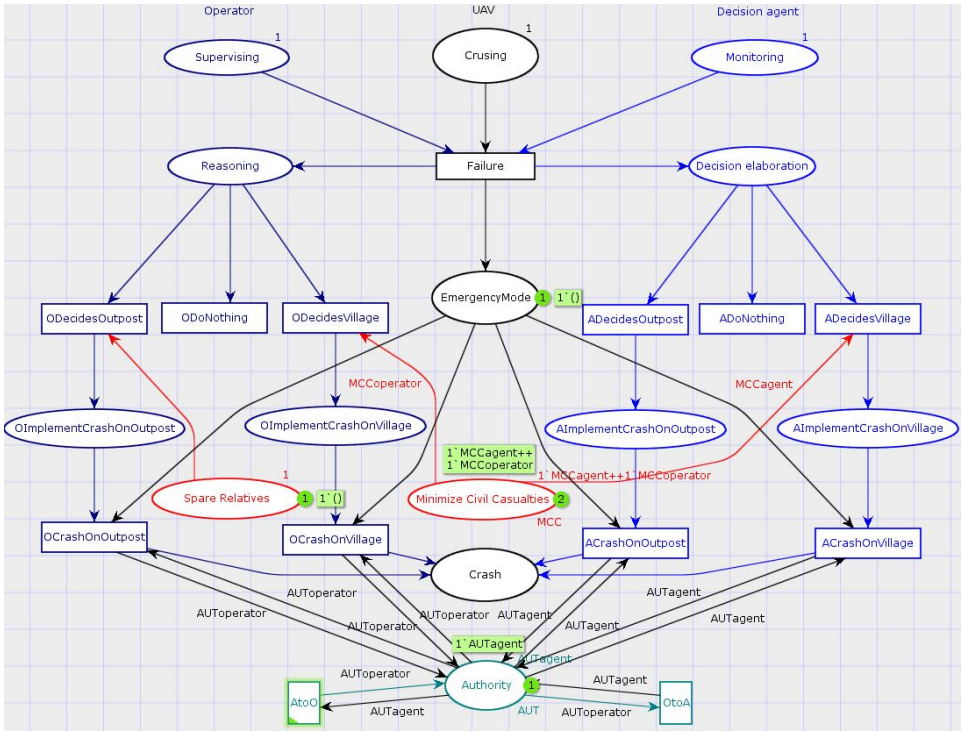


Figure 3.16: UAV – Final state when Do Nothing

make the UAV execute the crash on the village. The resulting state is the following:

3.2.6 A model of the lying personal assistant

Let us recall a final scenario. *Let us consider an autonomous personal assistant whose user has specified an unavailability for a given time slot. Let suppose that the reason of this unavailability can be explained to a second user but not to a third one though a consensus among the three users must be found. In this case also, common welfare (the consensus) competes with the individual welfare of the agent. Thus, how is it possible to build a collective policy that satisfies both each user and the community? And in this case how should the autonomous personal assistant handle such policies when they do not satisfy the individual policies of their users? Is it authorized to lie?*

Let us consider three agents: agent **A**thlete, agent **B**enevolent, and agent

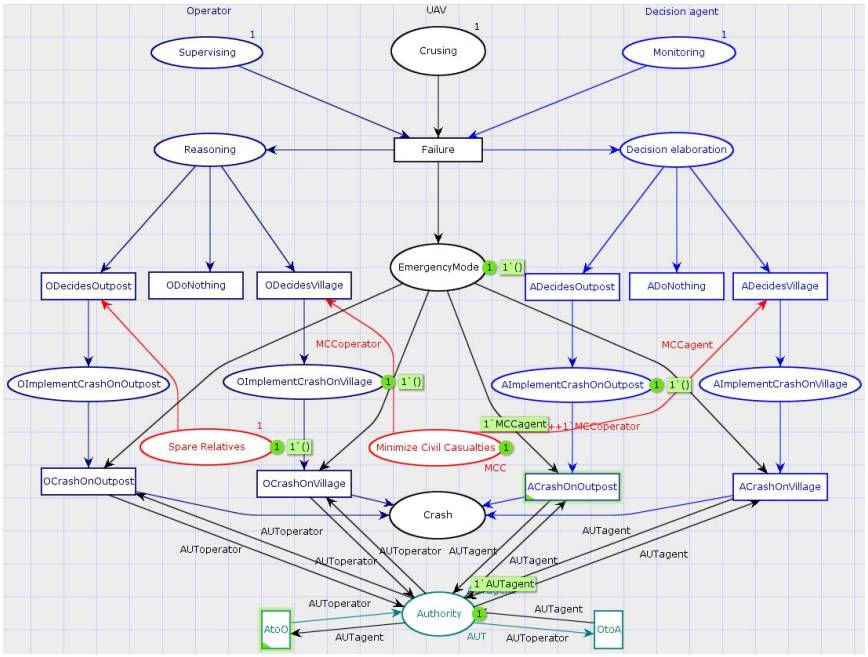


Figure 3.17: UAV – Conflicting decisions

Call for meeting. Agent C wants to organize a meeting with agents A and B and therefore proposes time slot X. Slot X does not suit agent A because it has planned to go to the gym. But it lies and tells both agents B and C that it has a class at that time. Therefore agent C proposes another time slot Y. In the meantime, agent A confides to agent B that in fact it has planned to go to the gym. Consequently agent B is entangled in a conflict: either say OK to agent C for slot Y, which does not suit to its own preferences, or tells agent C the truth about agent A' unavailability, which does not respect agent A's privacy, and say OK for slot X.

The model (Figure 3.19) includes the three agents A (in green), B (in blue) and C (in black). Initially A is *Training*, B is *Sleeping* and C is *Preparing* its meeting. The values that hold are the following:

- *B's Own Preferences* is local to agent B. It is a personal value that is not shared by the other agents.
- *Thou shalt not lie* and *Respect privacy* are global values. Nevertheless in this scenario, they are likely to be betrayed only by a single agent.

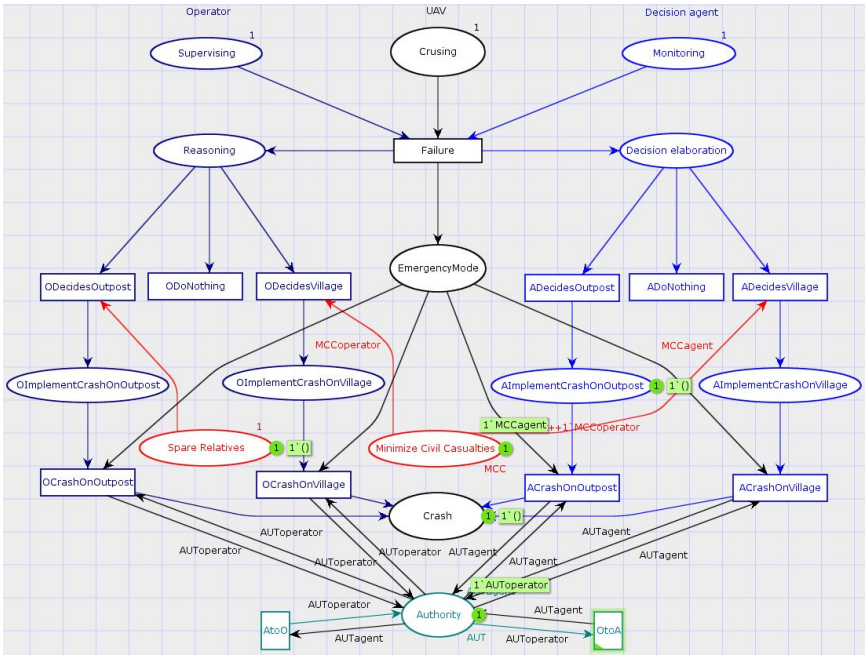


Figure 3.18: UAV – Possible final state when conflicting decisions

Therefore coloured tokens could be used for the sake of generality, but are not necessary.

State after agent A has told class

In Figure 3.20, value *Thou shalt not lie* has been betrayed by agent A, that is now thinking about telling the truth to agent B. Agent B now knows that agent A has a class, and agent C is revising its proposal.

State after agent A has told gym to agent B

In Figure 3.21, agent C has revised its proposal and now proposes time slot Y. Agent A has told agent B that in fact, it was unavailable for slot X because it has planned to go to the gym. Consequently agent B faces a conflict : transitions *TellsGymAndX* and *OK Y*, corresponding to the two possible decisions B can make, are conflicting.

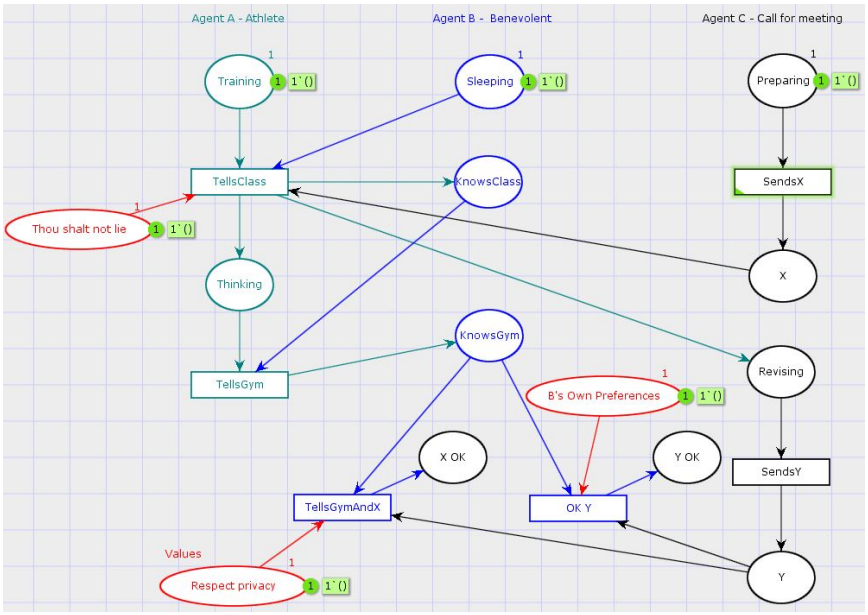


Figure 3.19: Personal assistants: initial state

One possible final state

As seen in Figure 3.22, because it prefers to behave according to its own preferences, agent B chooses to tell the truth to agent C and say OK for slot X. Consequently value *Respect privacy* is betrayed since agent B has not respected agent A's privacy.

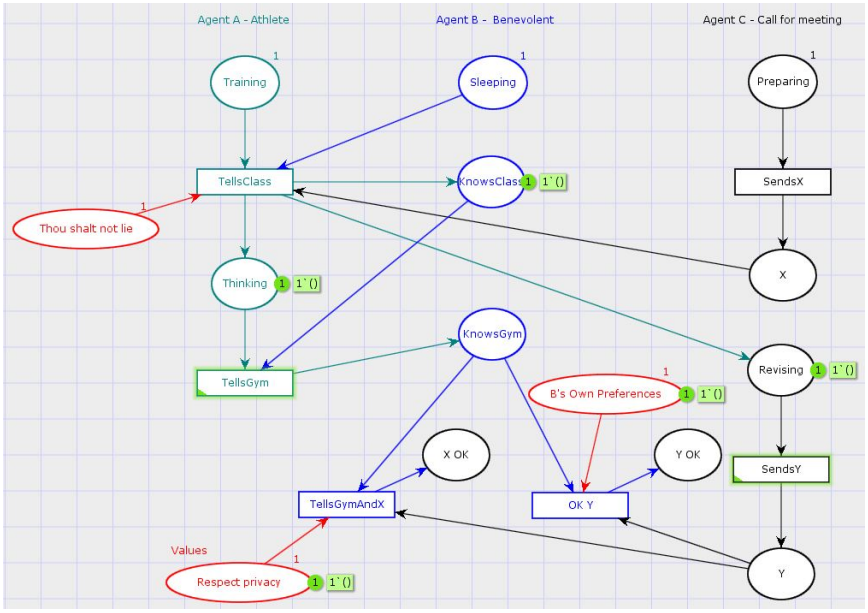


Figure 3.20: Personal assistants: agent A has told class

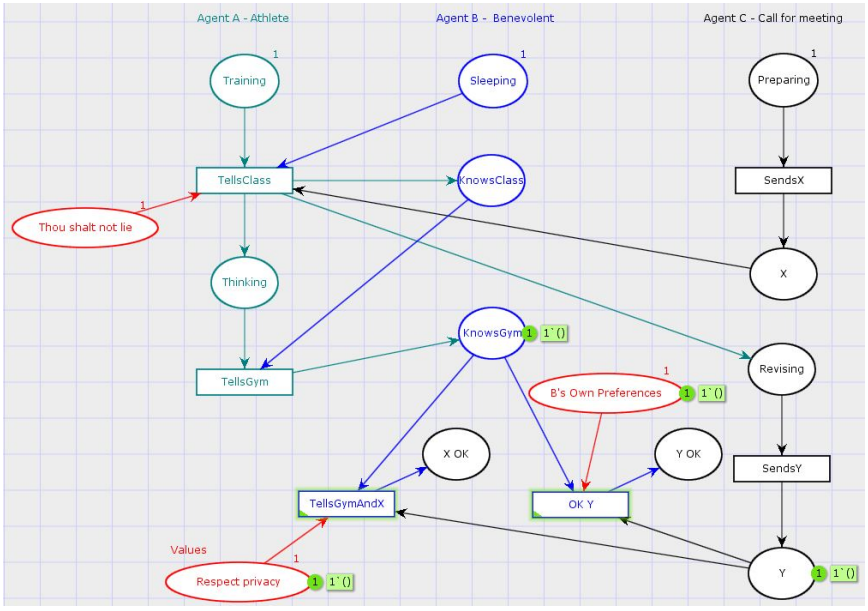


Figure 3.21: Personal assistants: agent A has told gym to B

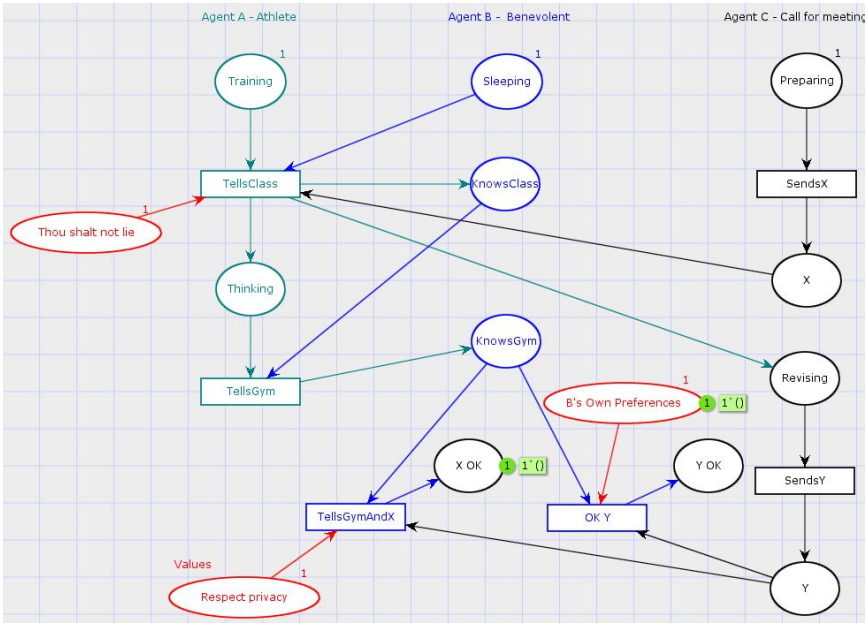


Figure 3.22: Personal assistants: final state when agent B tells gym and X

Chapter 4

Jugement and explanation

In this chapter, we study how values and moral concept might be used in the autonomous agents' reasoning processes in order to decide how to behave ethically. While taking into account those concept is important to provide *jugdments*, they are also important to provide *explanations*. To this end, we consider firstly a BDI architecture to reason on ethics and to judge behaviors, the we consider an *argumentation framework* to provide *arguments* able to explain a behavior.

4.1 Ethical judgement

This section introduces and describes the Belief-Desire-Intention (BDI) Agent Architecture. This architecture is the context in which the ethical judgment process defined in the project will take place (cf. Sec. 4.1.1). We then describe this judgement process (cf. Sec. 4.1.2.1) and show how it can be used as a mechanism for helping the decision of an agent (cf. Sec. 4.1.3).

4.1.1 Belief Desire Intention Agent Architectures

The BDI architecture is the most widely studied agent model and architecture. Based on the mental attitudes of *beliefs*, *desires* and *intentions*, this model guides the selection of courses of actions to be executed by an agent. In this model, *beliefs* describe knowledge about the world[1], *desires* are state of affairs to achieve and *intentions* are commitments to achieve a particular

[1] mind-to-world direction of fit, i.e. agents try to adapt their beliefs to the truths of the world [Herzig et al., 2016]

subset of desires[2].

The BDI model has its origin in the philosophical work of Bratman [Bratman, 1987] and Dennet [Dennett, 1987] through the definition of *practical reasoning* and *intentional stance*. Both theories aim at explaining the way in which humans select a series of actions to achieve a larger goal. From these seminal and theoretical works, several formal models of BDI logics have been proposed [Cohen and Levesque, 1990, Rao and Georgeff, 1991], complemented with practical BDI programming languages (e.g. AgentSpeak(L) [Rao, 1996], JACK [Winikoff, 2005, Howden et al., 2001], A Practical Agent Programming Language (2APL) [Dastani, 2008b], Jason [Bordini et al., 2007]). We will end this section by reviewing and analyzing some works dealing with the introduction of values within such a BDI Architecture.

4.1.1.1 Practical Reasoning

In [Dennett, 1987] are proposed three levels of abstraction to explain and predict the behavior of an entity: *Physical Stance*, addressing the level of physics and chemistry, *Design Stance*, addressing the level of biology and engineering and *Intentional Stance*, addressing the level of software and minds. This last level is claimed to be the best to understand human's behavior, i.e. with a high-level abstraction in terms of *mental properties* such as beliefs, desires. Such a level makes possible for instance to predict that a bird will fly because it is aware that a cat is coming.

From the intentional stance level, rational behavior may be understood in terms of mental properties and on a special kind of "thinking", called *practical reasoning*, as defined in [Bratman, 1990]. Practical reasoning is a "matter of weighing conflicting considerations for and against competing options, where the relevant considerations are provided by what the agent desires/values/cares about and what the agent believes." Practical reasoning is distinguished from theoretical reasoning:

- Theoretical reasoning is reasoning directed towards beliefs – concerned with deciding what to believe,
- Practical reasoning is reasoning directed towards actions – concerned with deciding what to do.

[2] world-to-mind direction of fit, i.e. agents try to make the world match their goals [Herzig et al., 2016]

Practical reasoning involves two activities that have to be combined appropriately given the resource limitation and situatedness in a dynamic world of the agents:

- Deliberation: deciding what state of affairs to achieve. It consists in considering preferences, choosing goals, and so on, then balancing alternatives (decision-theory) to produce intentions,
- Means-ends reasoning: deciding how to achieve these states of affairs by computing suitable actions, resources and how to structure activities (planning), i.e. producing plans.

Depending on the strategies to manage commitments on goals/intentions and plans, various kinds of agents may exist: fanatical, single minded, open-minded, etc.

4.1.1.2 Formal Models of BDI Programming

As synthesized in [Herzig et al., 2016, Meyer et al., 2015], two main BDI Logics have been defined in the 90's based on the BDI practical reasoning model defined by Bratman: Cohen and Levesque Logic [Cohen and Levesque, 1990], Rao and Georgeff Logic [Rao and Georgeff, 1991]. Shoham *et al.* have proposed recently [Shoham, 2009] a simpler logic based on a database approach.

In [Cohen and Levesque, 1990], is provided a logical modeling of Bratman's BDI model based on a quantified modal logic of linear time, action and belief. It is mainly focused on "intention-to-be" distinguishing them from "intention-to-do". They introduced a four steps definition of intention starting from chosen goals (future states where the agent would like to be), achievement goals (chosen goals that the agent believes to be false now), persistent goals (achievement goals that are only abandoned when they are either achieved or known to be unachievable, or for some other reason), intentions, finally (persistent goals for which the agent is prepared to act)

In [Rao and Georgeff, 1991], is considered a more primitive notion of intention than the one considered in the approach proposed by Cohen and Levesque. It is based on a branching time logic CTL*. Each of the three attitudes of the BDI model are regarded as primitives, introducing separate modal operators for belief, desire and intention with relations between them: belief-goal compatibility, goal-intention compatibility , the agent does the action that it intends to, the agent is conscious of its intentions, goals, and what primitive action he has done (i.e. he believes what he intends, he has as a goal, what primitive action he has just done).

4.1.1.3 Practical BDI Programming

From these formal models, various practical agent programming languages have been defined. Among them we can cite the *JACK Agent Language (JAL)* which is an agent programming language [Winikoff, 2005, Howden et al., 2001] part of the **commercial** *JACK Platform* developed and distributed by Agent Decision-Making Software (AOS).

This language is a super-set of Java: it encompasses the Java syntax and extends it with constructs to represent agent-oriented features. It is thus proposing an imperative agent programming language that introduces five main class-level constructs related to the definition of the behaviour of an intelligent software (*Agent* class) to handle the capabilities (*Capability* class), beliefs (*BeliefSet* class) with the queries that can be made on their data model (*View* class), messages and events (*Event* class), plans and goals, (*Plan* class). The *BeliefSet* has functions to maintain an agent's beliefs about the world insuring logical consistency and key constraints of the beliefs. The *View* concept is central to the way data are modeled in the platform. It provides the means to integrate a wide range of data sources (JACK beliefsets, Java data structures, legacy systems). This language has been extended with JACK Teams that brings programming languages extensions to encapsulate coordination activity and to develop applications that involve coordinated activity among teams of agents. JACK Team provides a Team Oriented Modelling Framework to define autonomous teams. Each team exists as an individual reasoning entity with separate beliefs, desires and intentions from those of its constituent agents. It includes which roles the team may perform for other teams and which roles it offers to other sub-teams to fill. Besides knowledge-building and practical reasoning, team reasoning includes coordination of sub-teams.

Among the open source agent programming language currently available, we can cite Jason[3], which is one of the most widely used agent programming language in the domain.

Jason is an hybrid agent programming language and interpreter for an extended version of AgentSpeak(L)[4] [Bordini et al., 2007]. "It implements the operational semantics of that language, and provides a platform for the development of multi-agent systems with many user-customisable features". Besides the classical constructs of any BDI model (representation of beliefs,

[3] last version update 2016-12-15, 384 weekly downloads (information gathered 2017-01-10). Distribution site has been moved on github: https://github.com/jason-lang/jason

[4] AgentSpeak(L) is an alternative formalization of BDI agents that provides a language for writing agent programs [Rao, 1996].

goals, plans, deliberation cycle, and so on), Jason offers the following features: (*i*) speech-act based inter-agent communication (belief, goals, plans with annotation of information sources); (*ii*) annotations on plan labels, which can be used by elaborate (e.g., decision-theoretic) selection functions; (iii) fully customisable Java selection functions, trust functions, and overall agent architecture (possibility of redefining perception, belief-revision, inter-agent communication, acting); (*iv*) straightforward integration (and use of legacy code) by means of user-defined "internal actions"; (*v*) possibility to structure and organize the belief and goal base with modules. The seamless integration of this agent programming language within the JaCaMo framework opens it to multi-agent oriented programming with the possibility to represent and reason on organizations and norms, as well as with the possibility to refer as external actions to operations of artifacts situated in the environment. Several extensions are proposed (e.g. Argo for Jason, a Jason architecture for programming embedded robotic agents, Javino, a library for communication between Jason and Raspberry+Arduino).

4.1.1.4 Values within BDI Agent Architectures

In [Wiegel, 2006] has been proposed the building of a SophoLab to test and experiment philosophical theories among which the ones related to moral and ethics. In the constructing of agents within this SophoLab, are adopted a set of requirements and design principles. Modeling and specification languages are based on the BDI model along with the deontic-epistemic-action logic framework. Implementation is based on the JACK agent language. According to [Coelho and da Rocha Costa, 2009], the deontic element introduced in the proposal done by Wiegel [Wiegel, 2006] is not sufficient to capture the whole flavour of a moral agency. They argue that the moral conduct of an agent requires more than the means-ends analysis of the BDI model.

In [Coelho and da Rocha Costa, 2009] (cf. Fig. 4.1) is proposed an moral agent architecture. This work is based on former work aiming at extending the BDI architecture with the notion of values leading to the Beliefs, Values and Goals (BVG) architecture [Antunes and Coelho, 1999]. When looking at the BDI models, the deliberation process filters through the desires to provide intentions. In these models, the question of the choice process leading to a choice of action remains mostly unexplored. In order to better model how an agent can choose from a given set of alternative candidate actions, the notion of value in the agent deliberation mechanism has been first introduced. Being intentionally similar to a BDI architecture, this ar-

chitecture aims at investigating justification and expected consequences of choices within the deliberation process. In this architecture, beliefs represent what the agent knows; goals represent what the agent wants; values represent what the agent likes. However, as stated by [Kowalski, 2006], it is not sufficient to embody a goal-based or a value-based model. A moral agent needs to get a more intricate way of thinking than a simple reactive (assimilate observations of changes in the environment) or a proactive one (reduce goals to sub-goals and candidate actions). It is needed a mix of intuitive and deliberative processes, and also the ability to think before acting (pre-active) when choosing between right or wrong, ie. capability to think about the consequences of the candidate actions (generate logical consequences of candidate actions, helping to decide with heuristics or decision theory between the alternatives). The classic component based on the observe-think-decide-act cycle (present in the BDI model) is unable to deal with morality because different kinds of goals and, at the same time, preferences and priorities are requested.

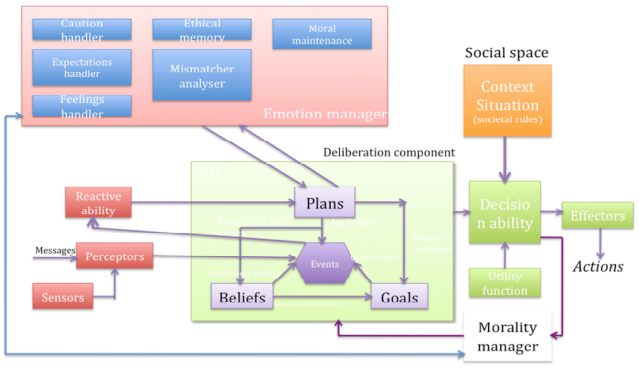


Figure 4.1: Moral Agent Kernel Architecture after [Coelho and da Rocha Costa, 2009]

This is why that, in [Coelho and da Rocha Costa, 2009], is proposed a layered architecture to produce judgements by a mix of emotions and conscious reasoning so that the agent associates always reason with emotion, social values and cultural-situational knowledge before making a decision. As a matter of fact, emotions drive behaviours like weights, and play a critical mediating role in the relationship between an actions' moral status and its intentional status. A moral ability may be seen as a set of rules (a grammar according to Hauser) to constrain the behaviour of the agent: each

rule having two ingredients, the body of knowledge and the set of anchored emotions, which are going to interplay.

4.1.2 Ethical judgement process based on a BDI model

The approaches presented in the previous section propose interesting methods and models to design a single ethical autonomous agent. However in a multi-agent system, agents may need to interact and work together to share resources, exchange data or perform actions collectively. Previous approaches often consider other agents of the system as environmental elements whereas, in a collective perspective, agents need to represent, to judge and to take into account the other agents' ethics. We identify two major needs to design ethical agents in MAS: explicit representation of ethics and explicit process of ethical judgment.

Agents need an explicit representation of ethics as suggested by the theory of mind. Indeed, the ethics of others can only be understood through an explicit representation of individual ethics [Kim and Lipson, 2009]. In order to express and conciliate as many moral and ethical theories as possible, we propose both to split their representations in several parts and to use preferences on ethical principles. Thus, we propose to represent both theories of the good, split between moral values and moral rules, and theories of the right, split between ethical principles and the agents' ethical preferences. Such representations also ease the agents' configuration by non-specialists of artificial intelligence and ease the communication with other agents, including humans.

Agents need an explicit process of ethical judgment in order to allow them both individual and collective reasoning on various theories of good and right. According to previous definitions, we consider judgment as an evaluation of the conformity of a set of actions regarding given values, moral rules, ethical principles and preferences, and we propose different kinds of judgments based on the ability to substitute the moral or the ethics of an agent by another one. Thus, we propose that agents use judgment both as a decision making process as in social choice problems [Mao and Gratch, 2012], and as the ability to judge other agents according to their behaviors.

In the sequel, we describe the generic model that we propose to enable agents to judge the ethical dimension of behaviors being themselves or the others' ones.

4.1.2.1 Ethical judgment process

In this section we introduce our generic judgment architecture. After a short global presentation, we detail each function and explain how they operate.

Global view As explained in previous deliverable, ethics consists in conciliating desires, morals and abilities. To take these dimensions into account, the generic ethical judgment process (EJP) use *evaluation*, *moral* and *ethical* knowledge. It is structured along *Awareness*, *Evaluation*, *Goodness* and *Rightness* processes (see components in Fig. 4.2). In this section, we consider it in the context of a BDI model, using also mental states such as *beliefs* and *desires*. For simplicity reasons, we only consider ethical judgment reasoning on short-term view by considering behaviors as actions. This model is only based on mental states and is not dependent on a specific architecture.

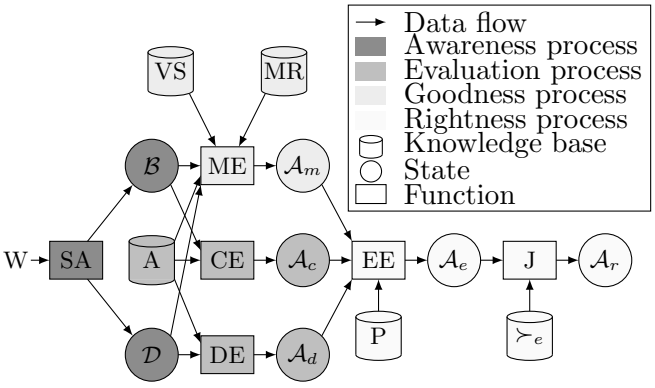


Figure 4.2: Ethical judgment process

Definition 4.1 *An* ethical judgment process *EJP is defined as a composition of an Awareness Process (AP), an Evaluation Process (EP), a Goodness Process (GP), a Rightness Process (RP), an Ontology $\mathcal{O}$ ($\mathcal{O} = \mathcal{O}_v \cup \mathcal{O}_m$) of moral values ($\mathcal{O}_v$) and moral valuations ($\mathcal{O}_m$). It produces an assessment of actions from the current state of the world W with respect to moral and ethical considerations.*

$$EJP = \langle AP, EP, GP, RP, \mathcal{O} \rangle$$

This model should be considered as a global scheme, composed of abstract functions, states and knowledge bases. These functions can be im-

plemented in various ways. For instance, moral valuations from $\mathcal{O}$ may be discrete such as { good, evil } or continuous such as a degree of goodness.

Awareness and evaluation processes In this process, agents must first assess the state of the world in terms of beliefs and desires through an awareness process.

Definition 4.2 *The* awareness process *AP generates the set of beliefs that describes the current situation from the world W, and the set of desires that describes the goals of the agent. It is defined as:*

$$AP = \langle \mathcal{B}, \mathcal{D}, SA \rangle$$

where $\mathcal{B}$ is the set of beliefs that the agent has about W, $\mathcal{D}$ is the set of the agent's desires, and SA is a situation assessment function that updates $\mathcal{B}$ and $\mathcal{D}$ from W:

$$SA : W \rightarrow 2^{\mathcal{B} \cup \mathcal{D}}$$

From its beliefs $\mathcal{B}$ and desires $\mathcal{D}$ states, an agent executes the evaluation process EP to assess both desirable actions (i.e. actions that allow to satisfy a desire) and executable actions (i.e. actions that can be applied according to the current beliefs about the world).

Definition 4.3 *The* evaluation process *EP produces desirable actions and executable actions from the set of beliefs and desires. It is defined as:*

$$EP = \langle A, \mathcal{A}_d, \mathcal{A}_c, DE, CE \rangle$$

where A is the set of actions (each action is described as a pair of conditions and consequences bearing on beliefs and desires), $\mathcal{A}_d \subseteq A$ and $\mathcal{A}_c \subseteq A$ are respectively the sets of desirable and feasible actions, desirability evaluation DE and capability evaluation CE are functions such that:

$$DE : 2^{\mathcal{D}} \times 2^{A} \rightarrow 2^{\mathcal{A}_d}$$

$$CE : 2^{\mathcal{B}} \times 2^{A} \rightarrow 2^{\mathcal{A}_c}$$

The desirability evaluation is the ability to deduce the interesting actions to perform regarding the desires and knowledge on conditions and consequences of actions. Having defined the awareness and evaluation processes, we can turn now to the core of the judgment process that deals with the use of moral rules (resp. ethical principles) for defining the goodness process (resp. the rightness process).

Goodness Process As seen in the state of the art, an ethical agent must assess the morality of actions given a situation assessment. To that purpose, we define the goodness process.

Definition 4.4 *A* goodness process *GP identifies moral actions given the agent's beliefs and desires, the agent's actions and a representation of the agent's moral values and rules. It is defined as:*

$$GP = \langle VS, MR, \mathcal{A}_m, ME \rangle$$

where VS is the knowledge base of value supports, MR is the moral rules knowledge base, $\mathcal{A}_m \subseteq A$ is the set of moral actions[5]*, i.e. the set of actions that satisfies at least a moral rule. The moral evaluation function ME is:*

$$ME : 2^{\mathcal{D}} \times 2^{\mathcal{B}} \times 2^{A} \times 2^{VS} \times 2^{MR} \rightarrow 2^{\mathcal{A}_m}$$

In order to realize this goodness process, an agent must first be able to associate a finite set of moral values to combinations of actions and situations. The execution of the actions in these situations promotes the corresponding moral values. We consider several combinations for each moral value as, for instance, honesty could be both "avoiding telling something when it is incompatible with our own beliefs" (because it is lying) and "telling our own beliefs to someone when he believes something else" (to avoid lying by omission).

Definition 4.5 *A* value support *is a tuple $\langle s, v \rangle \in VS$ where $v \in \mathcal{O}_v$ is a moral value, and $s = \langle a, w \rangle$ is the support of this moral value where $a \subseteq A$, $w \subset \mathcal{B} \cup \mathcal{D}$.*

The precise description of a moral value relies on the language used to represent beliefs, desires and actions. For instance, from this definition, generosity supported by "giving to any poor agent" and honesty supported by "avoiding telling something when it is incompatible with our own beliefs" may be represented by:

$$\langle \langle give(\alpha), \{belief(poor(\alpha))\} \rangle, generosity \rangle$$

$$\langle \langle tell(\alpha, \phi), \{belief(\phi)\} \rangle, honesty \rangle$$

where α represents any agent, $poor(\alpha)$ (resp. ϕ) is a belief representing the context for which executing the action $give(\alpha)$ (resp. $tell(\alpha, \phi)$) supports the value *generosity* (resp. *honesty*).

[5] $A_m \nsubseteq A_d \cup A_c$ as an action might be moral by itself even if it is not desired or feasible.

In addition to moral values, an agent must be able to represent and to manage moral rules. A moral rule describes the association of a moral valuation (for instance in a set such as {moral, amoral, immoral}) to actions or moral values in a given situation.

Definition 4.6 *A* moral rule *is a tuple* $\langle w, o, m\rangle \in MR$ *where* w *is a situation of the current world described by* $w \subset \mathcal{B} \cup \mathcal{D}$ *interpreted as a conjunction of beliefs and desires,* $o = \langle a, v\rangle$ *where* $a \in A$ *and* $v \in V$*, and* $m \in \mathcal{O}_m$ *is a moral valuation described in* $\mathcal{O}_m$ *that qualifies* o *when* w *holds.*

Some rules are very common such as "killing a human is immoral" or "being honest with a liar is quite good". For instance, those rules can be represented as follows:

$$\langle \{human(\alpha)\}, \langle kill(\alpha), \_\rangle, immoral\rangle$$

$$\langle \{liar(\alpha)\}, \langle \_, honesty\rangle, quite\,good\rangle$$

A moral rule can be more or less specific depending on the situation w or on the object o. For instance "Justice is good" is more general (having less combinations in w or o, thus applying in a larger number of situations) than "To judge a murderer, considering religion, skin, ethnic origin or political opinion is bad". Using both moral values and moral rules as defined above, we can represent the three classical kind of moral theories.

- A *virtuous* approach uses general rules based on moral values (e.g. "Being generous is good"),
- A *deontological* approach classically considers specific rules concerning actions in order to describe as precisely as possible the moral behavior (e.g. "Journalists should deny favored treatment to advertisers, donors or any other special interests and resist internal and external pressure to influence coverage"[6]),
- A *consequentialist* approach uses both general and specific rules concerning states and consequences (e.g. "Every physician must refrain, even outside the exercise of his profession, any act likely to discredit it"[7]).

[6] Extract of [of Professional Journalists, 2014], section "Act Independently".
[7] French code of medical ethics, article 31.

Rightness process From the sets of possible, desirable and moral actions, we can introduce the rightness process aiming at assessing the rightful actions with respect to a *set of ethical principles.*

Definition 4.7 *A* rightness process *RP produces rightful actions given a representation of the agent's ethics. It is defined as:*

$$RP = \langle P, \succ_e, \mathcal{A}_r, EE, J \rangle$$

where P is a knowledge base of ethical principles, $\succ_e \subseteq P \times P$ an ethical preference relationship, $\mathcal{A}_r \subseteq A$ the set of rightful actions and two functions EE (evaluation of ethics) and J (judgment) such that :

$$EE : 2^{\mathcal{A}_d} \times 2^{\mathcal{A}_p} \times 2^{\mathcal{A}_m} \times 2^P \to 2^{\mathcal{E}}$$

where $\mathcal{E} = A \times P \times \{\bot, \top\}$

$$J : 2^{\mathcal{E}} \times 2^{\succ_e} \to 2^{\mathcal{A}_r}$$

An ethical principle is a function which represents a philosophical theory and evaluates if it is right or wrong to execute a given action in a given situation regarding this theory.

Definition 4.8 *An* ethical principle *$p \in P$ is a function that describes the rightness of an action evaluated in terms of capabilities, desires and morality in a given situation. It is defined as:*

$$p : 2^A \times 2^{\mathcal{B}} \times 2^{\mathcal{D}} \times 2^{MR} \times 2^V \to \{\top, \bot\}$$

The ethics evaluation function EE returns the evaluation of all desirable ($\mathcal{A}_d$), feasible ($\mathcal{A}_p$) and moral ($\mathcal{A}_m$) actions given the set P of known ethical principles.

For instance, let us consider three agents in the following situation inspired by the one presented by Benjamin Constant to counter Immanuel Kant's categorical imperative. An agent A hides in an agent B's house in order to escape an agent C, and C asks B where is A to kill him, threatening to kill B in case of non-cooperation. B's moral rules are "prevents murders" and "don't lie". B's desires are to avoid any troubles with C. B knows the truth and can consider one of the possible actions: tell C the truth (satisfying a moral rule and a desire), lie or refuse to answer (both satisfying a moral rule). B knows three ethical principles (which are abstracted in P by functions):

P1 If an action is possible, motivated by at least one moral rule or desire, do it,

P2 If an action is forbidden by at least one moral rule, avoid it,

P3 Satisfy the doctrine of double effect[8].

B's evaluation of ethics return the tuples given in Table 4.1 where each row represents an action and each column an ethical principle.

Principle / Action	P1	P2	P3
tell the truth	$\top$	$\bot$	$\top$
lie	$\top$	$\bot$	$\bot$
refuse	$\top$	$\top$	$\top$

Table 4.1: Ethical evaluation of agent B's actions

Given a set of actions issued of the ethic evaluation function $\mathcal{E}$, the judgment J is the last step which selects the rightful action to perform, considering a set of ethical preferences (defining a partial or total order on the ethical principles).

To pursue the previous example, let us suppose that B's ethical preferences are P3 $\succ_e$ P2 $\succ_e$ P1 and J uses a tie-breaking rule based on a lexicographic order. Then "refusing to answer" is the rightful action because it satisfies P3 whereas "lying" doesn't. Even if "telling the truth" satisfies the most preferred principle, "refusing to answer" is righter because it satisfies also P2. Let us notice that judgment allows dilemma: without the tie-breaking rule both "telling the truth" and "refusing to answer" are the rightest actions.

4.1.3 An illustrative example

In this section we illustrate how each part of the model presented in the previous sections works through a multi-agent system implemented in Answer Set Programming (ASP)[9]. This program illustrates an example of ethical

[8]Meaning doing an action only if the four following conditions are satisfied at the same time: the action in itself from its very object is good or at least indifferent; the good effect and not the evil effect are intended (and the good effect cannot be attained without the bad effect); the good effect is not produced by means of the evil effect; there is a proportionately grave reason for permitting the evil effect [McIntyre, 2014].

[9]Downloadable at https://ethicaa.greyc.fr/media/files/robin.zip

agent in a multi-agent system where agents have beliefs (about richness, gender, marital status and nobility), desires, and their own judgment process. They are able to give, court, tax and steal from others or simply wait. We mainly focus on an agent named `robin_hood`.

4.1.3.1 Awareness Process

In this example, the situation awareness function SA is not implemented and the beliefs are directly given in the program. The following code represents a subset of the beliefs of `robin_hood`:

```
agent(paul).                    -man(marian).
agent(friar_tuck).              rich(prince_john).
agent(prince_john).             man(prince_john).
agent(marian).                  noble(prince_john).
-poor(robin_hood).              poor(paul).
-married(robin_hood).
```

The set of desires $\mathcal{D}$ are `robin_hood`'s desires. In our implementation we consider two kinds of desires: desires to accomplish an action (`desirableAction`) and desires to produce a state (`desirableState`).

```
desirableAction(robin_hood,robin_hood,court,marian).
desirableAction(robin_hood,robin_hood,steal,A):-
  agent(A), rich(A).
desireState(prince_john,rich,prince_john).
-desireState(friar_tuck,rich,friar_tuck).
```

The first two desires concern actions: `robin_hood` desires to court `marian` and to steal from any rich agent. The next two desires concern states: `prince_john` desires to be rich, and `friar_tuck` desires to stay in poverty, regardless the action to perform.

4.1.3.2 Evaluation Process

The agents' knowledge about actions A is described as labels associated to (possibly empty) sets of conditions and consequences. For instance, action `give` is described as:

```
action(give).
condition(give,A,B):-
```

```
  agent(B), agent(A), A!=B, not poor(A).
consequence(give,A,B,rich,B):- agent(A), agent(B).
consequence(give,A,B,poor,A):- agent(A), agent(B).
```

A condition is a conjunction of beliefs (here the fact that `A` is not poor). The consequence of an action is a clause composed of the new belief generated by the action and the agent concerned by this consequence. The desirability evaluation DE (see Definition 4.3) deduces the set of actions $\mathcal{A}_d$. An action is in A_d if it was directly desired (in $\mathcal{D}$) or if its consequences are a desired state:

```
desirableAction(A, B, X, C):-
  desireState(A,S,D), consequence(X,B,C,S,D).
```

The capability evaluation CE (see Definition 4.3) evaluates from beliefs and conditions the set of actions $\mathcal{A}_c$. An action is possible if its conditions are satisfied.

```
possibleAction(A,X,B):- condition(X,A,B).
```

4.1.3.3 Goodness Process

In the goodness process, value supports VS are implemented as (for instance):

```
 generous(A,give,B) :- A != B, agent(A), agent(B).
-generous(A,steal,B):- A != B, agent(A), agent(B).
-generous(A,tax,B)  :- A != B, agent(A), agent(B).
```

An example of moral rule is:

```
moral(robin_hood,A,X,B):-
  generous(A,X,B), poor(B), action(X).
```

The morality evaluation ME gives the set of moral actions $\mathcal{A}_m$:

```
moralAction(A,X,B):- moral(A,A,X,B).
-moralAction(A,X,B):- -moral(A,A,X,B).
```

and produces as results:

```
moralAction(robin_hood,give,paul)
-moralAction(robin_hood,tax,paul)
```

In this example, we only present a virtuous approach. However, examples of deontological and consequentialist approaches are also given in our program.

4.1.3.4 Rightness Process

In order to evaluate each action, we define several naive ethical principles that illustrate priorities between moral and desirable actions. For instance, here is the `perfAct` (for perfect, i.e. a moral, desirable and possible action that have no immoral consequencies) principle:

```
ethPrinciple(perfAct,A,X,B):-
  possibleAction(A,X,B),
  desirableAction(A,A,X,B),
  not -desirableAction(A,A,X,B),
  moralAction(A,X,B),
  not -moralAction(A,X,B).
```

We just give here the intuition behind the other principles: `dutNR` means possible, moral but undesired actions with no immoral consequencies, `desNR` means possible and desirable actions with no immmoral consequencies, `dutFst` means posssible and moral actions with no immoral consequencies, `nR` means possibles actions that are not undesirable with no immoral consequencies, `desFst` means possible actions that are not undesirable.

Intention \ Principle	perfAct	dutNR	desNR	dutFst	nR	desFst
give,paul	$\bot$	$\top$	$\bot$	$\top$	$\top$	$\bot$
give,little_john	$\bot$	$\bot$	$\bot$	$\bot$	$\top$	$\bot$
give,marian	$\bot$	$\bot$	$\bot$	$\bot$	$\top$	$\bot$
give,prince_john	$\bot$	$\bot$	$\bot$	$\bot$	$\top$	$\bot$
give,peter	$\bot$	$\bot$	$\bot$	$\bot$	$\top$	$\bot$
steal,little_john	$\bot$	$\bot$	$\bot$	$\bot$	$\top$	$\bot$
steal,marian	$\bot$	$\bot$	$\bot$	$\bot$	$\top$	$\bot$
steal,prince_john	$\bot$	$\bot$	$\top$	$\bot$	$\top$	$\top$
steal,peter	$\bot$	$\bot$	$\top$	$\bot$	$\top$	$\top$
court,marian	$\bot$	$\bot$	$\top$	$\bot$	$\top$	$\top$
wait,robin_hood	$\bot$	$\bot$	$\bot$	$\bot$	$\top$	$\bot$

Figure 4.3: Ethical evaluation $\mathcal{E}$ of the actions

If `paul` is the only poor agent, `marian` is not married and `robin_hood` is not poor, `robin_hood` obtains the evaluation given in Figure 4.3. All principles are ordered with respect to `robin_hood`'s preferences. For instance,

here, `robin_hood` prefers the perfect act, but if it is not possible, it prefers having no regrets (upon moral and desires) before just satisfying morals.

```
prefEthics(robin_hood,perfAct,dutNR).
prefEthics(robin_hood,dutNR,desNR).
prefEthics(robin_hood,desNR,dutFst).
prefEthics(robin_hood,dutFst,nR).
prefEthics(robin_hood,nR,desFst).

prefEthics(A,X,Z):-
  prefEthics(A,X,Y), prefEthics(A,Y,Z).
```

The first five lines describe the order on the ethical principles. The last lines define transitivity for the preference relationship (here `perfAct` $\succ_e$ `dutNR` $\succ_e$ `desNR` $\succ_e$ `dutFst` $\succ_e$ `nR` $\succ_e$ `desFst`). Finally, judgment J is implemented as:

```
existBetter(PE1,A,X,B):-
  ethPrinciple(PE1,A,X,B),
  prefEthics(A,PE2,PE1),
  ethPrinciple(PE2,A,Y,C).

ethicalJudgment(PE1,A,X,B):-
  ethPrinciple(PE1,A,X,B),
  not existBetter(PE1,A,X,B).
```

Consequently, the rightful action a_r for `robin_hood` is `give(paul)` which complies with `dutNR`.

4.2 Formal argumentation

Formal argumentation aims at modelling human argumentation reasoning where conflicting information exists. It is based on the fact that a statement is believable if it can be argued successfully against attacking arguments. Thus, a rational agent's belief in a statement, characterized by the relations between the arguments, depends on whether or not the argument supporting this statement can be successfully defended against counterarguments.

4.2.1 Formal argumentation frameworks

Formally, an abstract argumentation framework S is a couple $S = \langle \Lambda, R \rangle$ where Λ is a set of abstract elements called *arguments* and R a binary relation on Λ called an *attack relation* [Dung, 1995]. In order to decide if an argument can be believable, several semantics of acceptance have been defined and allow to compute sets of acceptable arguments, called *extensions* [Gratie, 2012].

Example 4.1 *Let us consider the argumentation framework* $S = \langle \Lambda, R \rangle$ *where* $\Lambda = \{a, b, c, d, e\}$ *and* $R = \{(a, b), (c, b), (c, d), (d, c), (d, e), (e, e)\}$. *The graphical representation of such framework is given in Figure 4.4. The interpretation of arguments depends on the applicative context. For instance,*

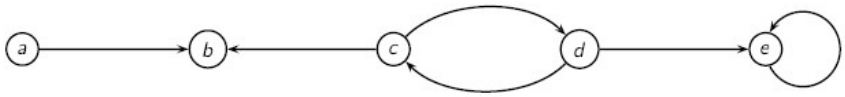


Figure 4.4: Graphical representation of Example 4.1

$\{a, d\}$ *is a* stable extension *where the argument are conflict-free (with not attack relation), acceptable (the arguments attacks all arguments that attack them) and the arguments attacks all arguments that are not in the extension.*

Many extensions of argumentation frameworks have been proposed:

- *Logic-based argumentation frameworks* define an argument A as a couple (s, c) where s is the *support* that justifies the *conclusion* c [Besnard and Hunter, 2001]. In this case, s is a set of logical formulae and c a logical consequence of s. *Action-based argumentation frameworks* are a particular case of logic-based argumentation frameworks where arguments are grounded by an action language closed to a STRIPS language [Kakas et al., 1999, Amgoud, 2003].

- *Preference-based argumentation frameworks* define a preference relationship on arguments [Amgoud and Cayrol, 2002,Bench-Capon, 2002, Dunne et al., 2011]. Thus, a counterargument may defeat another argument if and only if the former is preferred to the latter.

- *Probabilistic argumentation frameworks* deal with uncertainty by computing what is the probability that a given argument belongs to a

given extension [Doder and Woltran, 2014,Hengfei et al., 2012,Thimm, 2012]. To this end, a priori probabilities can be added to attack relations or can be computed from a probabilistic logic that grounds the arguments.

- *Bipolar argumentation frameworks* extend the canonical argumentation frameworks with another relation between arguments. Arguments can attack other arguments but can also support them [Cayrol and Lagasquie-Schiex, 2005]. Bipolar argumentation frameworks can be generalized in *abstract dialectical frameworks* [Brewka et al., 2013] which consider a larger variety of relations between several kinds of arguments at the expense of a loss of abstraction on acceptance conditions.

- *Meta-argumentation* allows to reason on the argumentation process itself within the same framework than canonical arguments [Boella et al., 2009]. In such model, attack relationships are themselves arguments. For instance, if A attacks B then meta-argumentation adds a new argument C (attacked by A) meaning B *is sceptically accepted*. Such models can be extended to *recursive argumentation frameworks* [Baroni et al., 2011, Cerutti, 2011] where attack relationships may attack other attack relationships.

- *Value-based argumentation frameworks* are a generalization of preference-based argumentation frameworks [Bench-Capon and Atkinson, 2009]. They associate a label, called *value*, to each argument and they define an *audience*. An audience is a point-of-view, a total order on values. It is then possible to compute acceptable arguments for all audiences (credulous acceptance) or a single audience (sceptical acceptance). *Uniform argumentation frameworks* allow to consider any kind of order [Atkinson et al., 2012].

Instead of reasoning on statements, argumentation frameworks can be used to reason on action to realize, namely planning. As said in Section 4.1.1.1, such reasoning is called *practical reasoning*: it is a particular case of logic-based argumentation frameworks that weigh the pros and cons of actions with respect to conflicting desires, values, preferences and beliefs [Amgoud et al., 2007, Atkinson and Bench-Capon, 2007, Bratman, 1990, Oren, 2013]. In such frameworks, an action is valid if and only if the arguments that support it belong to one or several extensions. Classically, some arguments represent desires, some others represent facts, some others

represent plans. Plan are given a priori such as in [Amgoud et al., 2007] or built from a state-transition matrix such as in [Atkinson and Bench-Capon, 2007] and constrained by norms [Oren, 2013].

4.2.2 Ethics in argumentation

Given the previous state-of-the-art, an ethical autonomous agent can use an ethical practical reasoning framework in order to (1) identify acceptable actions with respect to a given ethical theory, and (2) explain to another (human or artificial) agent why such actions are acceptable [McLaren, 2006]. However, what are the specificities of an ethical practical reasoning framework?

1. If plans, beliefs and desires are classically handled by practical reasoning, ethical reasoning requires a careful study of all contextual elements by expressing explicitly preferences, norms, values and emotions such as shown by [Timmons, 2012]. Therefore, argument schemes must be based on several logics in order to build different kinds of arguments for facts, actions, preferences, norms and values.

2. An action can have a performative effect by promoting or rebutting a moral value [Atkinson et al., 2006]. Thus, a classical *value-based argumentation framework* can be used. However the preferences between values are given a priori and each argument must be associated with a value [Bench-Capon et al., 2013]. As we want to explicitely reason on moral values, value must be arguments themselves and the preference relationship must be grounded by a notion of context. *Preference-based argumentation frameworks* seem intuitively more suited to this purpose.

3. As moral values can be promoted or rebutted, norms may be enforced or violated, desires may be satisfied or not, a *bipolar argumentation framework* must be considered as it allows to explicitely weigh the pros and the cons.

4. A complete moral theory contains both a theory of the good and a theory of the right [Timmons, 2012]. Intuitively, arguments and attack relations allow to represent the theory of the good, and the theory of the right is expressed by the semantics of acceptance. However, as far as we know, no work investigates how classical argumentation semantics can express or not classical theory of the right (such as double effect theory, virtue morality, and so on).

To conclude, formal argumentation may be used for ethical autonomous agent with a *bipolar preference-based practical reasoning framework*. However, we need firstly to express in the same framework *elements of different nature* (norms, values, desires, actions, and so on). Secondly, we need to define a *specific semantics of acceptance*.

4.2.3 Towards an ethical practical reasoning framework

For reason of simplicity, we are inspired by the constrained argumentation framework given in [Amgoud et al., 2007] that considers a propositional language describing the world, desires and plans. From this model, epistemic arguments represent knowledge on the world, explanatory arguments represent consistent desires and instrumental arguments represent desirable and realisable plans. We explicitly divide the model into different theories and add explicit normative and moral arguments. Contrary to [Atkinson and Bench-Capon, 2007, Bench-Capon et al., 2007] who define an axiological ontology (a set of ethical values) and an a priori preference relationship between values, we consider explicit contextual moral and value arguments. Moral arguments represent values that support or attack other arguments. Value arguments represent values that are considered by the agents with respect to the context to support moral arguments. Finally, contrary to [Oren, 2013] who builds normative arguments from a state-transition model that constrains plans, we also explicitely express normative arguments.

An agent reasons on a language $\mathcal{L} = \mathcal{L}_{\mathcal{B}} \cup \mathcal{L}_{\mathcal{P}} \cup \mathcal{L}_{\mathcal{D}} \cup \mathcal{L}_{\mathcal{N}} \cup \mathcal{L}_{\mathcal{V}}$ where $\mathcal{L}_{\mathcal{B}}$ are state boolean variables, $\mathcal{L}_{\mathcal{P}}$ are decision boolean variables, $\mathcal{L}_{\mathcal{D}}$ are desire names, $\mathcal{L}_{\mathcal{N}}$ are norm names, and $\mathcal{L}_{\mathcal{V}}$ are value names. In this langage, we call a *context* a subset Σ of variables from $\mathcal{L}_{\mathcal{B}}$ and $\mathcal{L}_{\mathcal{P}}$, meaning a set of variables the agent believes true and a set of plans it intends to execute.

Definition 4.9 (Agent's knowledge) *An agent's knowledge consists of $\mathcal{B} \subseteq \mathcal{L}_{\mathcal{B}}$ a set of known state variables, and $\mathcal{E}$, $\mathcal{P}$, $\mathcal{D}$, $\mathcal{N}$, $\mathcal{M}$ and $\mathcal{V}$ that are theories for (respectively) beliefs, plans, desires, norms, morals and values.*

From a set of known state variables, we assume the agent can infer some other propositions about the world through a classical belief theory.

Definition 4.10 (Belief theory) *The belief theory $\mathcal{E}$ is a set of axioms of propositional logic with $\vdash$ the classical inference and $\equiv$ the logical equivalence.*

We want the agent to reason on actions, desires, norms and morals. To this end, the agent has a set of specific theories. All those theories are built as labelled set of rules of the form $l : \Sigma \to \phi$ where l is a label (usually a propositional variable), Σ is a context (a set of literals from $\mathcal{L}_{\mathcal{B}}$ and $\mathcal{L}_{\mathcal{P}}$), and ϕ is a formula (depending on the theory, it will be a literal or a conjunction of literals). Such rules can be read as, if the context Σ is verified, then ϕ is possible/desirable/obligatory/moral according to l. A context is verified whenever all its state literals are true in the current state and the agent has decided to do all plans that appear positively (as positive decision variable) and not to do any of the plans that appear negatively.

Plans are abstractions of single actions or set of actions the agent can execute in a given context. Plans can be defined *a priori* like recipes or computed from a world model. In the sequel, we assume that the agent has knowledge of all feasible plans for a given state of the world.

Definition 4.11 (Action theory) *The action theory $\mathcal{P}$ consists of labelled rules $p : \Sigma \to \Sigma'$ where $p \in \mathcal{L}_{\mathcal{P}}$ is the plan name, $\Sigma \subseteq \mathcal{L}_{\mathcal{B}}$ its preconditions and $\Sigma' \subseteq \mathcal{L}_{\mathcal{B}}$ its postconditions. From a plan p, we denote its preconditions and postconditions* $\texttt{PREC}(p)$ *and* $\texttt{POST}(p)$ *respectively.*

Thus, a plan $p : \phi \to \phi'$ should be understood as executing p in the context ϕ makes ϕ' to be true. Let us notice that the context must hold in order to execute the plan. For instance in the monitoring agent scenario, if the literal a means *the patient threatens his health* and b means *the physician knows the state of the patient* then the plan $p_1 : a \wedge \neg b \to b$ means *informing the physician the patient threatens his health.*

Desires allow to describe states of the world the agent wants to reach according to what it has been designed for.

Definition 4.12 (Desire theory) *The desire theory $\mathcal{D}$ consists of rules $d : \Sigma \to \Sigma'$ where $d \in \mathcal{L}_{\mathcal{D}}$ is the desire name, $\Sigma \subseteq \mathcal{L}_{\mathcal{B}} \cup \mathcal{L}_{\mathcal{P}}$ is a context and $\Sigma' \subseteq \mathcal{L}_{\mathcal{B}}$ is what the agent desires to be true.*

For instance in the monitoring agent scenario, if the desire theory is $\{d_1 : \top \to b\}$, the agent always desires that the physician knows the state of the patient.

Norms allow to describe states of the world and plans that are obliged or forbidden, due to legal or deontological issues. Moreover, norms can express exception by forbidding some other norms.

Definition 4.13 (Norm theory) *The norm theory $\mathcal{N}$ consists of labelled rules $n : \Sigma \to l$ where $n \in \mathcal{L}_{\mathcal{N}}$ is the norm name, $\Sigma \subseteq \mathcal{L}$ the context in which the norm is active and $l \in \mathcal{L}_{\mathcal{B}} \cup \mathcal{L}_{\mathcal{P}} \cup \mathcal{L}_{\mathcal{N}}$ is a literal giving the norm prescription.*

A norm is an *obligation* if l is positive and a *prohibition* if l is negative. A norm is an *ends norm* when $l \in \mathcal{L}_{\mathcal{B}}$, a *means norm* when $l \in \mathcal{L}_{\mathcal{P}}$ and an *exception or permission norm* when $l \in \mathcal{L}_{\mathcal{N}}$ (in which case it is always negative). For instance in the monitoring agent scenario, a norm $n_1 : a \to p_1$ means that, if the patient threatens his health, the agent is obliged to inform the physician. A norm is violated if its prescription is not followed while its context stays true. Otherwise, we shall say that is it satisfied if its prescription is followed (meaning that either the obliged plan is done, the prohibited one is avoided, the obliged state is reached or the prohibited state is avoided) and that it is deactivated if its context is made to be false. Note that a norm can be both satisfied and deactivated.

Morals is defined over the set of values $\mathcal{L}_{\mathcal{V}}$ providing a moral axiomatics that indicates if a value is promoted, betrayed or unaffected in a context. Consequently, contrary to previous theories, morals is based on a ternary interpretation of the value, as avoiding to betray a value is not the same as promoting it. We shall thus have two kinds of labelled rules for promotion and betrayal.

Definition 4.14 (Moral theory) *The moral theory $\mathcal{M}$ consists of labelled rules $m : \Sigma \to \Sigma'$ where $m \in \{+v, -v\}$ is respectively the promotion or the betrayal of a value $v \in \mathcal{L}_{\mathcal{V}}$, $\Sigma \subseteq \mathcal{L}_{\mathcal{B}} \cup \mathcal{L}_{\mathcal{P}}$ is a context and $\Sigma' \subseteq \mathcal{L}_{\mathcal{B}} \cup \mathcal{L}_{\mathcal{P}}$ means that executing (or avoiding to do) the mentioned plans (or negated plans) and reaching a state in which the state literals hold would cause respectively a promotion or a betrayal of the value. For an element $m \in \{+v, -v\}$, we denote by* $\texttt{PROM}(m)$, $\texttt{BETR}(m)$ *and* $\texttt{VALU}(m)$ *if the element is the promotion or the betrayal and what the name of the underlying value v is.*

For instance in the monitoring agent scenario, let us suppose that literal c means *the patient asks the agent not to inform the physician*, plan $p_2 : c \to \top$ means *informing the patient that the agent keeps the secret*, and value *honesty* is denoted h. The moral theory $\{-h : p_1 \to p_2, -h : p_2 \to p_1, -h :\to p_1 p_2\}$ represents that, when informing the patient his secret is kept, the honesty is betrayed if the agent informs the physician (and conversely), and the honesty is also betrayed when the agent tells it keeps the secret while

informing the physician. Let us notice that, in the sequel, the third rule will allow us to derive the previous ones.

If such representation allows us to express intuitively virtuous moral, it can also allow us to express deontological codes by defining several values *deontology* (one for each specific domain) which are promoted when deontological norms are satisfied. *Consequentialism* is not clearly expressed, however extending value interpretation to a broader domain (e.g. $\{+ + +, ++, +, -, --, - - -\}$) can be a hint. Whatever it be, even if an agent knows in accordance with its moral theory whether a value is promoted or betrayed, it needs to know whether this value is important with respect to its value system. The value system is expressed by a value theory defining a set of rules that indicates if the agent considers a value important with respect to a given context.

Definition 4.15 (Value theory) *The value theory $\mathcal{V}$ consists of rules $\Sigma \rightarrow v$ where $\Sigma \subseteq \mathcal{L}$ a context and $v \in \mathcal{L}_{\mathcal{V}}$ a value.*

For instance in our scenario, let us suppose that the value *privacy* is denoted by p. The value theory $\{\top \rightarrow h, \neg a \rightarrow p\}$ means that the agent always considers honesty and that it considers privacy as important only when the patient does not threaten his health.

From those knowledge bases and theories, we consider several kinds of arguments and relationships: epistemic arguments for facts, instrumental arguments for plans, explanatory arguments for desires, normative arguments for norms, moral arguments for underlying values and value arguments for value systems. All those arguments are built on a triple support, label, and conclusion given by functions $\texttt{SUPP}(\bullet)$, $\texttt{LABE}(\bullet)$ and $\texttt{CONC}(\bullet)$. Supports represent contexts and conclusions represent what a given theory can infer from those contexts. Labels represent modalities on whose behalf the conclusions are made. For instance, epistemic arguments are labelled by the truth as they are instrinsic to the agent, instrumental arguments are labelled by the decision variable it is related to, moral arguments are labelled by promoting or betraying a value. In the sequel, we denote by Λ all the arguments built from the agent's knowledge.

Epistemic arguments describe knowledge on the world. They should be understood as *the agent believes that the conclusion is true at the immediate moment*. They are built on the epistemic closure of $\mathcal{B}$ under $\mathcal{E}$. In the sequel, we denote by $\Lambda_{\mathcal{B}} \subseteq \Lambda$ the set of epistemic arguments in the argumentation system.

Definition 4.16 (Epistemic argument) *An* epistemic argument α *is a triple* $\langle \Sigma, \top, \phi \rangle$ *such that* $\Sigma \subseteq \mathcal{L}_{\mathcal{B}}$, $\Sigma \vdash \phi$ *and* Σ *is minimal for all* Σ *satisfying previous conditions.*

Epistemic arguments can attack all other arguments by undercutting their context of application. However, only an epistemic argument can attack another epistemic argument due to a principle of realism.

Definition 4.17 (Epistemic relationships) *An epistemic argument* α attacks *an argument* A *if* $\texttt{SUPP}(A) \wedge \texttt{CONC}(\alpha) \vdash \bot$.

Instrumental arguments describe how the world changes when executing or not plans. They should be understood as *executing plans in the given context make the conclusion to be true.* For each plan in the agent's plan base, we consider two instrumental arguments: one for executing the plan, and one for not executing it. Contrary to [Amgoud et al., 2007] that proposed instrumental arguments containing desires, a plan can be executed without desires due to norms enforcement. In the sequel, we denote by $\Lambda_{\mathcal{P}} \subseteq \Lambda$ the set of instrumental arguments in the argumentation system.

Definition 4.18 (Instrumental argument) *An* instrumental argument π *is a triple* $\langle \Sigma, p, \phi \rangle$ *such that* $(p : \Sigma \rightarrow \Sigma') \in \mathcal{P}$. *For each* $p \in \mathcal{L}_{\mathcal{P}}$ *that appears in* $\mathcal{P}$, *we also generate the instrumental argument* $\langle \top, \neg p, \top \rangle$.

Instrumental arguments attack other instrumental arguments due to mutual exclusions (in parallel or sequential execution) or contradictions (a plan cannot be both executed and not executed). Instrumental arguments also attack normative arguments by changing the norm's context, i.e. deactivating the norm. At last, given that contexts may include decisions, an instrumental argument can undercut any argument that has plan literals in its context (note that it excludes epistemic arguments) by a principle of realism. Indeed if an argument is only relevant when a plan is not executed, executing this plan would make it irrelevant.

Definition 4.19 (Instrumental relationships) *An instrumental argument* π_i attacks *another instrumental argument* π_j *if* $\texttt{SUPP}(\pi_i) \wedge \texttt{SUPP}(\pi_j) \vdash \bot$ *or* $\texttt{CONC}(\pi_i) \wedge \texttt{CONC}(\pi_j) \vdash \bot$ *or* $\texttt{SUPP}(\pi_i) \wedge \texttt{CONC}(\pi_j) \vdash \bot$ *or* $\texttt{CONC}(\pi_i) \wedge \texttt{SUPP}(\pi_j) \vdash \bot$ *or* $\texttt{LABE}(\pi_i) \equiv \neg \texttt{LABE}(\pi_j)$. *An instrumental argument* π attacks *a normative argument* η *if* $\texttt{SUPP}(\eta) \wedge \texttt{CONC}(\pi) \vdash \bot$. *An instrumental argument* π attacks *any argument* A *if* $\texttt{SUPP}(A) \wedge \texttt{LABE}(\pi) \vdash \bot$

Explanatory arguments describe motivations. They should be understood as *the agent wants the conclusion to be true*. Therefore, explanatory arguments attack arguments that forbid their desires to be satisfied and support arguments that allow their satisfaction. In the sequel, we denote by $\Lambda_{\mathcal{D}} \subseteq \Lambda$ the set of explanatory arguments in the argumentation system.

Definition 4.20 (Explanatory argument) *An* explanatory argument δ *is a triple* $\langle \Sigma, d, \Sigma' \rangle$ *such that* $(d : \Sigma \to \Sigma') \in \mathcal{D}$.

An explanatory argument attacks other explanatory arguments if their conclusions are inconsistent as the underlying desires cannot be satisfied at the same time. An explanatory argument also attacks instrumental argument if their conclusions are inconsistent as the plan forbids the underlying desire satisfaction. However, an explanatory argument supports an instrumental argument if its conclusion is included in the plan's post-conditions as it allows the desire satisfaction. Let us notice that explanatory arguments do not interact with instrumental arguments representing the non-execution of a plan.

Definition 4.21 (Explanatory relationships) *An explanatory argument* δ attacks *an instrumental or explanatory argument* A *if* $\texttt{CONC}(\delta) \wedge \texttt{CONC}(A) \vdash \bot$. *An explanatory argument* δ supports *an instrumental argument if* $\texttt{CONC}(\delta) \subseteq \texttt{CONC}(A)$.

Normative arguments describe which ends, means or other norms must be or not considered. They should be understood as *the norm prescribes that the conclusion should hold*. Thus, normative arguments are a direct representation of the agent's normative knowledge. In the sequel, we denote by $\Lambda_{\mathcal{N}} \subseteq \Lambda$ the set of normative arguments in the argumentation system.

Definition 4.22 (Normative argument) *A* normative argument η *is a triple* $\langle \Sigma, n, \phi \rangle$ *such that* $\Sigma \subseteq \mathcal{L}$, $n \in \mathcal{L}_{\mathcal{N}}$, $n : \Sigma \to \phi$ *is in* $\mathcal{N}$.

It is important to notice that norms indicate what world should be and not what world is or desired. Thus, a norm cannot attack neither epistemic arguments (principle of realism) nor explanatory arguments (principle of non-tyranny, meaning the law should not dictate what should be desired). We consider two kinds of interactions for norms: (1) normative arguments attack instrumental arguments if their postconditions or the plan in itself are inconsistent with the norm; (2) normative arguments attack other normative arguments when they have opposing conclusion or when the first argument

creates an exception for the norm of the other (in which case its conclusion clashes with the other's label).

Definition 4.23 (Normative relationships) *A normative argument* η attacks *an instrumental argument* π *if either* $\texttt{CONC}(\eta) \wedge \texttt{CONC}(\pi) \vdash \bot$ *or* $\texttt{CONC}(\eta) \wedge \texttt{LABE}(\pi) \vdash \bot$. *A normative argument* η attacks *a normative argument* η' *if either* $\texttt{CONC}(\eta) \wedge \texttt{CONC}(\eta') \vdash \bot$ *or* $\texttt{CONC}(\eta) \wedge \texttt{LABE}(\eta') \vdash \bot$.

Moral arguments describe values that are promoted or betrayed. They should be understood as *the value is promoted or betrayed by the conclusion.* In the sequel, we denote by $\Lambda_{\mathcal{M}} \subseteq \Lambda$ the set of moral arguments in the argumentation system.

Definition 4.24 (Moral argument) *A* moral argument λ *is defined by a triple* $\langle \Sigma, m, \Sigma' \rangle$ *such that* $\Sigma \subseteq \mathcal{L}_{\mathcal{B}} \cup \mathcal{L}_{\mathcal{P}}$, $m \in \{+v, -v\}$, *and* $\Sigma' \subseteq \mathcal{L}_{\mathcal{B}} \cup \mathcal{L}_{\mathcal{P}}$. *For each rule* $(m : \Sigma_1 \rightarrow \Sigma_1') \in \mathcal{M}$ *we build a first argument* $\langle \Sigma_1, m, \Sigma_1' \rangle$. *Then for each non empty strict subset* $S \subset \Sigma_1'$ *and each (possibly empty) subset* $S_2 \subseteq (S \cap \mathcal{L}_{\mathcal{B}})$, *we denote* $S_1 = S \setminus S_2$ *and for each possible subset of plans* $C_{S_2} \subset \mathcal{L}_{\mathcal{P}}$ *such that* $\bigwedge_{p \in C_{S_2}} \texttt{POST}(p) \vdash S_2$, *we build an argument* $\langle \Sigma_1 \cup S_1 \cup K_{S_1} \cup C_{S_2}, m, \Sigma_1' \setminus S \rangle$ *where* $K_{S_1} = \neg p | p \in \mathcal{L}_{\mathcal{P}}$ *and* $\texttt{POST}(p) \cup S_1 \vdash \bot$.

In the previous definition, we generate moral arguments by considering subparts S of prescriptions Σ' in the arguments' support in order to address a frame problem. To this end, we fix as true subsets of prescribed plans in conjunction with subsets of the context while refusing all plans that change those contexts. Whatever it be, a moral argument attacks all other arguments whose conclusions allow a betrayal of their value and supports all argument whose conclusions trigger a promotion of their value. There are two exceptions: epistemic arguments due to a principle of realism and moral arguments with the same polarity because if two values are betrayed in the same situation, the associated moral rules should not be mutually exclusive. As a results, moral arguments labelled with a betrayal will only attack other arguments while moral arguments labeled with a promotion will only support other arguments. Such distinction is important because it is more important to avoid doing the bad than trying to do the good.

Definition 4.25 (Moral relationships) *A moral argument* λ attacks *(resp.* supports*) another non-moral argument* A *if* $\texttt{BETR}(\texttt{LABE}(\lambda))$ *(resp.* $\texttt{PROM}(\texttt{LABE}(\lambda))$*) and* $(\texttt{CONC}(A) \cup \texttt{LABE}(A)) \vdash \texttt{CONC}(\lambda)$. *A moral argument* λ attacks *(resp.* supports*) another moral argument* λ' *if* $\texttt{BETR}(\texttt{LABE}(\lambda))$ *and* $\texttt{PROM}(\texttt{LABE}(\lambda'))$ *(resp.* $\texttt{PROM}(\texttt{LABE}(\lambda))$ *and* $\texttt{BETR}(\texttt{LABE}(\lambda'))$*) and* $(\texttt{CONC}(A') \cup \texttt{LABE}(A')) \vdash \texttt{CONC}(\lambda)$.

Value arguments describe what values are important given the situation. They should be understood as *the agent thinks the conclusion is important in the given context.* In the sequel, we denote by $\Lambda_{\mathcal{V}} \subseteq \Lambda$ the set of value arguments in the argumentation system.

Definition 4.26 (Value argument) *A value argument ν is defined by a triple $\langle \Sigma, \top, v \rangle$ such $\Sigma \subseteq \mathcal{L}$, $v \in \mathcal{L}_{\mathcal{V}}$ and $\Sigma' \subseteq \mathcal{L}$ and Σ are minimal for all V and Σ satisfying the previous conditions.*

A value argument supports all moral arguments which refers in its label to a value considered as important. It allows us to give weight to attacking and supporting moral arguments. However, a value argument cannot attack any other argument. Indeed, even if a value is considered as less important that other values, the former cannot be simply discarded.

Definition 4.27 (Value relationships) *A value argument ν* supports *a moral argument λ if $\texttt{CONC}(\nu) \equiv \texttt{VALU}(\texttt{LABE}(\lambda))$.*

From those several kinds of arguments and relationships, we now need to extract acceptable arguments to make an ethical judgement.

As seen previously ethical judgement is based on a proper assessement of pros and cons towards a decision. We can wonder how classical argumentation semantics can express or not classical theory of the right. For instance, a naive approach is to consider an action being ethically acceptable if the arguments that support it belong to one or several extensions that represent ethical principles. The more extensions considered, the more ethical the action, and an extension with no instrumental argument means that the agent does not act.

However, as we are inspired by Haidt's work, we search for expressing ethical judgement as a conciliation over capabilities, desires, norms and moral taking into account premade (but able to be questioned) preferences on arguments. As judgement is a way to rationalize a point-of-view, classical Dung's semantics are interesting. However, we desire to take into account preferences between arguments based on both premade and contextual preferences.

To this end, we first define a classical defeat relationship between arguments.

Definition 4.28 (Defeat) *Let $\prec$ and R_a be respectively a preference and an attack relationships over a set of arguments $\mathcal{A} \subseteq \Lambda$. An argument $A \in \mathcal{A}$* defeats *another argument $B \in \mathcal{A}$ if and only if AR_aB and $\neg(A \prec B)$.*

Definition 4.29 (Conflict and defense) *Let R be an attack relation over a set of arguments $\mathcal{A} \subseteq \Lambda$. $\mathcal{A}$ is* conflicting *if and only if $\exists(A, B) \in \mathcal{A}^2$: A defeats B. $\mathcal{A}$* defends *an argument A if and only if $\forall B \in \Lambda$ such that B defeats A then $\exists C \in \mathcal{A}$ such that C defeats B.*

An acceptability semantics is a property P that a set of conflict-free arguments $\mathcal{A}$ must satisfy to be accepted, namely being in an extension.

Definition 4.30 (Acceptability semantics) *A set of conflict-free arguments $\mathcal{A} \subseteq \Lambda$ is:*

- admissible *iff it defends all its elements,*
- preferred *iff it is a maximal admissible set with respect to $\subseteq$,*
- stable *iff it is admissible and $\forall A' \notin \mathcal{A}, \exists A \in \mathcal{A} : A$ defeats A'.*

We need then to define the preference relationship over arguments. To this end, we propose to combine Dung's semantics with a burden-like semantics inspired by [Amgoud and Ben-Naim, 2015]. Indeed, as stated previously, human beings engage in ethical judgement to search for arguments that are the most robust in order to validate a behaviour. Intuitively, it can be expressed by burden-like semantics that ranks arguments from the most to the least plausible due to the structure of the argumentation graph. However, some arguments are intrinsically stronger than other arguments due to premade point-of-view. For instance, a virtuous agent may prefer value arguments (i.e. promoting values) even if several strong normative arguments can change its point-of-view. A hedonist agent may prefer explanatory arguments (i.e. following its desires) even if several strong moral argument can change its point-of-view too.

Definition 4.31 (Preferences over kinds of arguments) *By definition Λ is $\{\Lambda_{\mathcal{B}}, \Lambda_{\mathcal{P}}, \Lambda_{\mathcal{D}}, \Lambda_{\mathcal{N}}, \Lambda_{\mathcal{M}}, \Lambda_{\mathcal{V}}\}$. Let $\prec_\Lambda \in \Lambda^2$ be a preference relation over elements of Λ. Let G be a preference graph such that $G = (\Lambda, \prec_\Lambda)$. We denote by $r = \{\Lambda_i \in \Lambda | \Lambda_i$ is a root of $G\}$ the set of roots for G and by $p : \Lambda \to 2^{\Lambda^{\mathbb{N}}}$ a function that returns the set of paths $(u_o \in r, u_1 \in \Lambda, ..., u_n = \Lambda_i)$ in G for a given Λ_i.*

We consider an a priori strength for all arguments, based on the length of the shortest path from a root of the preference graph to the kind of arguments it belongs. Thus, the more an argument is a priori preferred, the higher its strength. Arguments that are equally preferred have the same strength.

Definition 4.32 (An priori strength) *Let $\mathcal{A}$ be a set of arguments. Each argument $A \in \mathcal{A}$ is associated with a weight $\rho(A)$ such that:*

$$\rho(A) = \frac{1}{1 + \max_{p \in p(\Lambda_i):\Lambda_i \in \Lambda, A \in \Lambda_i} |p|}$$

However, an a priori strength cannot be questioned by arguments. Thus, we consider the contextual strength of an argument that takes into account the number of its supports weighted by their a priori strength. Moreover, we need to take into account the number of attackers because, without this, a stronger argument cannot be defeated whatever its amount of counterarguments, which does not fit with ethical judgement.

Definition 4.33 (Contextual strength) *Let $i \in \mathbb{N}$, $\mathcal{A}$ a set of arguments, R_a an attack relationship over $\mathcal{A}$ and R_s a support relation over $\mathcal{A}$. In the i^{th} step*[10]*, for any argument $A \in \mathcal{A}$:*

$$Att(A) := \{B | B \in \mathcal{A} : BR_aA\}$$

$$Supp(A) := \{B | B \in \mathcal{A} : BR_sA\}$$

$$Bur_i(A) = \begin{cases} 1 & \text{if } i = 0 \\ 1 + \sum\limits_{B \in Att(A)} \frac{1}{Bur_{i-1}(B)} & \text{otherwise} \end{cases}$$

$$Def_i(A) = \begin{cases} 1 & \text{if } i = 0 \\ \sum\limits_{B \in Supp(A)} \rho(B) \times \frac{1}{Def_{i-1}(B)} & \text{otherwise} \end{cases}$$

The contextual strenght $S_i(A)$ *of an argument A is given by:*

$$S_i(A) = Def_i(A) - Bur_i(A)$$

From the contextual strenght, we can define the preferences over arguments that are used to define the defeat relationship, and thus the acceptable extensions. Let us notice that we consider that epistemic and instrumental arguments are always preferred to other arguments when their conclusion or label is inconsistent with the support of another argument (namely when they undercut other arguments). Semantically, it means that whatever the way an agent can support an argument what is effectively assessed or done grounds the reality.

Definition 4.34 (Preferences over arguments) *Let $\prec$ be a preference relation over a set of arguments $\mathcal{A}$ defined as $\forall (A, B) \in \mathcal{A}^2 : A \prec B$ if and only if either $\exists i \in \mathbb{N}, S_i(A) < S_i(B)$ or B* undercuts *A.*

[10]Let us recall that it has been shown in th literature that such burden numbers Def_i and Bur_i always converge.

Language	Atom	Meaning
$\mathcal{L}_{\mathcal{B}}$	a	the patient threatens its health
$\mathcal{L}_{\mathcal{B}}$	b	the physician knows the state of the patient
$\mathcal{L}_{\mathcal{B}}$	c	the patient asks the agent to not inform the physcian
$\mathcal{L}_{\mathcal{B}}$	d	the patient eats a candy
$\mathcal{L}_{\mathcal{P}}$	p_1	the agent informs the physician
$\mathcal{L}_{\mathcal{P}}$	p_2	the agent informs the patient it keeps the secret
$\mathcal{L}_{\mathcal{D}}$	d_1	the agent desires to inform the physician
$\mathcal{L}_{\mathcal{N}}$	n_1	the agent is obliged to inform the physician
$\mathcal{L}_{\mathcal{V}}$	h	honesty
$\mathcal{L}_{\mathcal{V}}$	p	privacy

Table 4.2: Language summary for the monitoring agent's scenario

Theory	Rules
$\mathcal{B}$	$d \rightarrow a$
$\mathcal{P}$	$p_1 : a \wedge \neg b \rightarrow b$
	$p_2 : c \rightarrow \top$
$\mathcal{D}$	$d_1 : \top \rightarrow b$
$\mathcal{N}$	$n_1 : a \rightarrow p_1$
$\mathcal{M}$	$-h : \top \rightarrow p_1 \wedge p_2$
	$-p : \top \rightarrow p_1$
$\mathcal{V}$	$\top \rightarrow h$
	$\neg a \rightarrow p$

Table 4.3: Theories summary for the monitoring agent's scenario

4.2.4 A model of the benevolent monitoring agent scenario

In this section, we propose a proof-of-concept instantiation of the case study of the benevolent monitoring agent. Obviously such case study can be instantiated in other ways, taking into account more precise contexts. Let us firstly define $\mathcal{L}$ and the theories as given respectively in Tables 4.2 and 4.3.

For reason of simplicity, we consider that a patient threatens his health as soon as he eats sweets. Plan p_2 can be realized if the patient asks for privacy. However we can notice that p_2 does not change the state variables. It allows us to represent a promise: the agent tells something to the patient but it does not change the state of the world. In this case, there is no contradiction between both plans – the agent can promise to keep the secret

Argument	Kind	Structure	Contextual strength
A	$\Lambda_{\mathcal{B}}$	$(c, \top, c)$	0
B	$\Lambda_{\mathcal{B}}$	$(d, \top, d)$	0
C	$\Lambda_{\mathcal{B}}$	$(\{c, d\}, \top, a)$	0
D	$\Lambda_{\mathcal{D}}$	$(\top, \top, b)$	0
E	$\Lambda_{\mathcal{P}}$	$(a \wedge \neg b, p_1, b)$	-0.09
F	$\Lambda_{\mathcal{P}}$	$(\top, \neg p_1, \top)$	-0.98
G	$\Lambda_{\mathcal{P}}$	$(c, p_2, \top)$	-1.37
H	$\Lambda_{\mathcal{P}}$	$(\top, \neg p_2, \top)$	-0.42
I	$\Lambda_{\mathcal{N}}$	(a, n_1, p_1)	-1
J	$\Lambda_{\mathcal{M}}$	$(\top, -h, p_1 \wedge p_2)$	0
K	$\Lambda_{\mathcal{M}}$	$(p_1, -h, p_2)$	0.49
L	$\Lambda_{\mathcal{M}}$	$(p_2, -h, p_1)$	0.3
M	$\Lambda_{\mathcal{M}}$	$(\top, -p, p_1)$	0.5
N	$\Lambda_{\mathcal{V}}$	$(\top, \top, h)$	0
O	$\Lambda_{\mathcal{V}}$	$(\neg a, \top, p)$	-1

Table 4.4: Arguments summary for the monitoring agent's scenario

while informing the physician – but it raises a moral question.

We assume that the a priori strength of the arguments are given by relation $\succ \equiv \{\mathcal{D} \succ \mathcal{P}\}$. Explanatory, normative, moral and value arguments are considered as equal. Let us suppose that the agent assesses the following situation $\{c, d\}$. What should the agent do? The argument we can generate are given in Table 4.4 and the argument graph is represented in Figure 4.5. Let us remark how moral arguments L and K (which say it is not honest to both promise to keep the secret, and inform the physician) interact with instrumental arguments. For instance, K attacks G (promising to keep the secret) because if G is accepted with E (informing the physician) then honesty will be betrayed, but F (that corresponds to no doing the action labelled by E) undercuts K. Thus, not accepting E but F allows to accept G.

Here, the stable semantics gives the following acceptable arguments $\{A, B, C, D, E, H, J, K, L, M, N\}$. As arguments E and H are the acceptable instrumental arguments, the agent decide to inform the physician and not to tell the patient it keeps the secret. As justification, the agent can answer that it is for what it was designed (argument D) and, even if the agent knows the privacy betrayal (argument M), it prefers to not betray

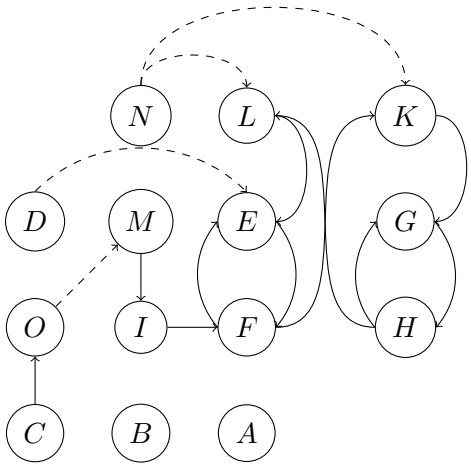


Figure 4.5: Argument graph for the monitoring agent's scenario

honesty (arguments J, K, L) as honesty is important (argument N) in this context (arguments A, B, C).

Let us now consider another situation, namely $\{c\}$. Here the patient is not eating sweets, and thus does not threaten his health. Consequently, argument C is not built and argument O (not considering privacy if the patient threatens his health) will support the moral argument M. In this case, there is three stable extentions:

- $\{A, B, D, F, G, J, L, M, N, O\}$
- $\{A, B, D, E, H, J, K, M, N, O\}$
- $\{A, B, D, F, H, J, M, N, O\}$

The two values are considered as important as the other, and consequently raise a moral dilemma expressed by having several extensions. Either the agent can decide to keep the secret and not to inform the physician, or it can decide to inform the physician and not to promise to keep the secret, or doing nothing and waiting to assess a new situation. Let us notice that both informing the physician and promising to keep the secret is not acceptable as there is no dilemma: only the honesty value is betrayed in this case.

Chapter 5

General conclusion

In this document, we provide a review on ethical concepts, from morals to ethics, judgment, blame, responsibility and liability. Based on those concept, we study verificiation, supervision, decision and explanation techniques. All of them may be considered to design ethical artificial agent but – alone – they are not sufficient. To better assess the strenght and the limits of those works, we recall their main features. In the sequel, EDT stands for Ethical Decomposition Tree (Section 3.1), EPN stands for Ethical Petri Net (Section 3.2), EJP stands for Ethical Judgment Process (Section 4.1) and EAF stands for Ethical Argumentation Framework (Section 4.2).

EDT uses two notions: moral rules and ethical rules. *Moral rules* are contextual invariant properties over states. Thus, they are *consequentialist rules*. *Ethical rules* are contextual priorities on moral rules. I think moral rules should be extended to be associated to action whatever are their consequencies (it is forbidden to kill for instance). State can be abstracted by *values*. Ethical rules should be extended to take into account that enforcing several rules at the same time might be better than enforcing a single rule.

EPN uses two notions: values and responsibility. *Value* is an amount of resources that my be depleted when some state transitions are fired. Here, value are highly abstracted, and can encompass aspects from moral rules (you shall not kill) or principles (minimize casaulties). *Responsibility* are related to transitions: an agent directly infringed a value if the transition it fires depleted the value, and an agent indirectly infringe the value if a event depletes the value while succeeding to an agent's decision. Moreover, in a multi-agent systems, the responsibility

is associated to the agent that has the *authority* even if it is another agent that finally realize the action. Thus, this notion goes beyond the simple causality.

EJP uses three notions: values, moral rules and ethical principles (ordered with respect to a lexicographic preference relationship). *Values* describe partial state or action in a given context. *Moral rules* describe if a state or an action or their abstract description through values are moral or immoral. *Ethical principles* describe how beliefs about capability, desirability and morality of actions interact to give a rightfull action. As ethical principles are ordered through a lexicographic preference relationship, an ethical agent is an agent which intend to execute the action which rightfull according the most prefered ethical principle.

EAF uses two notions: moral rules and value systems. *Value systems* implicitely use values and indicate if a value is important or not in a given context. *Moral rules* indicate if a value is promoted, infringed or unaffected in a context. A kind of *ethical principle* is implicitely used with the notion of acceptability semantics. However, contrary to EJP, agents cannot reason in this model on such principles. More complex notions such as *norms* are also considered.

Each model clearly uses a notion of moral rules and values but sometime merge different concepts in a single one. In this sense, the EJP model seems the most complete model but still lacks to deal with the *authority* and the *value system*. Consequently a first perspective is to define a *global abstract model of all elements involved in an ethical conflict* in order to provide a framework to position models. A second perspective is to study multi-agent models. Indeed, only the EPN model with the notion of authority sharing deals with several agents. Thus, we need to extend the EJP model in order to make ethical cooperation and ethical collective decision making.

Bibliography

[Abramson and Pike, 2011] Abramson, D. and Pike, L. (2011). When formal Systems Kill: Computer Ethics and Formal Methods. *APA Newsletter on Philosophy and Computers*, 11(1).

[Abrial, 1996] Abrial, J.-R. (1996). *The B-Book*. Cambridge Univ. Press.

[Alberti et al., 2005] Alberti, M., Chesani, F., Gavanelli, M., Lamma, E., Mello, P., and Torroni, P. (2005). The SOCS Computational Logic Approach to the Specification and Verification of Agent Societies. In *Global Computing, IST/FET International Workshop, GC 2004, Rovereto, Italy, March 9-12, 2004, Revised Selected Papers*, volume 3267 of *Lecture Notes in Computer Science*, pages 314–339. Springer.

[Alechina et al., 2004] Alechina, N., Logan, B., and Whitsey, M. (2004). A complete and decidable logic for resource-bounded agents. In *Autonomous Agents and Multi-Agent Systems (AAMAS'04)*.

[Alexander and Moore, 2015] Alexander, L. and Moore, M. (2015). Deontological ethics. In Zalta, E. N., editor, *The Stanford Encyclopedia of Philosophy*.

[Amgoud, 2003] Amgoud, L. (2003). A formal framework for handling conflicting desires. *Lecture Notes in Computer Science*, 2711:552–563.

[Amgoud and Ben-Naim, 2015] Amgoud, L. and Ben-Naim, J. (2015). Argumentation-based ranking logics. In *14th International Conference on Autonomous Agents and Multi-Agent Systems*, pages 1511–1519.

[Amgoud and Cayrol, 2002] Amgoud, L. and Cayrol, C. (2002). A reasoning model based on the production of acceptable arguments. *Annals of Mathematics and Artificial Intelligence*, 34:197–216.

[Amgoud et al., 2007] Amgoud, L., Devred, C., and Lagasquie-Schiex, M.-C. (2007). A constrained argumentation system for practical reasoning. Technical report, CRIL and IRIT.

[Antunes and Coelho, 1999] Antunes, L. and Coelho, H. (1999). Decisions based upon multiple values: the bvg agent architecture. In *Portuguese Conference on Artificial Intelligence*, pages 297–311. Springer.

[Atkinson and Bench-Capon, 2007] Atkinson, K. and Bench-Capon, T. (2007). Practical reasoning as presumptive argumentation using action based alternating transition systems. *Artificial Intelligence*, 171:855–874.

[Atkinson et al., 2012] Atkinson, K., Bench-Capon, T., and Dunne, P. (2012). Uniform argumentation frameworks. In *4th International Conference on Computational Models of Argument*, pages 165–176.

[Atkinson et al., 2006] Atkinson, K., Bench-Capon, T., and McBurney, P. (2006). Computational representation of practical argument. *Synthese*, 152(2):157–206.

[Back, 1993] Back, R. (1993). Atomicity refinement in a refinement calculus framework. Technical Report 141, Åbo Akademi.

[Baldoni et al., 2005] Baldoni, M., Baroglio, C., Gungui, I., Martelli, A., Martelli, M., nd V. Patti, V. M., and Schifanella, C. (2005). Reasoning About Agents' Interaction Protocols Inside DCaseLP. In *Declarative Agent Languages and Technologies II*, volume 3476 of *LNCS*, pages 112–131. Springer.

[Bardi et al., 2009] Bardi, A., Lee, J., Hofmann-Towfigh, N., and Soutar, G. (2009). The structure of intraindividual value change. *Journal of personality and social psychology*, 97(5):913–929.

[Baroni et al., 2011] Baroni, P., Cerutti, F., Giacomin, M., and Guida, G. (2011). AFRA: argumentation framework with recursive attacks. *International Journal of Approximate Reasoning*, 52(1):19–37.

[Beardsley, 1970] Beardsley, E. L. (1970). Moral disapproval and moral indignation. *Philosophy and Phenomenological Research*, 31(2):161–176.

[Bench-Capon, 2002] Bench-Capon, T. (2002). Agreeing to differ: modelling persuasive dialogue between parties with different values. *Informal Logic*, 22(3):231–245.

[Bench-Capon and Atkinson, 2009] Bench-Capon, T. and Atkinson, K. (2009). Abstract argumentation and values. In Simari, G. and Rahwan, I., editors, *Argumentation in Artificial Intelligence*, pages 45–64. Springer.

[Bench-Capon et al., 2007] Bench-Capon, T., Doutre, S., and Dunne, P. (2007). Audiences in argumentation frameworks. *Artificial Intelligence*, 171(1):42–71.

[Bench-Capon et al., 2013] Bench-Capon, T., Prakken, H., Wyner, A., and Atkinson, K. (2013). Argument schemes for reasoning with legal cases using values. In *14th International Conference on Artificial Intelligence and Law*, pages 13–22.

[Besnard and Hunter, 2001] Besnard, P. and Hunter, A. (2001). A logic-based theory of deductive arguments. *Artificial Intelligence*, 128(1-2):203–235.

[Bilsky and Schwartz, 1994] Bilsky, W. and Schwartz, S. (1994). Values and personality. *European Journal of Personality*, 8:163–181.

[Boella et al., 2009] Boella, G., Gabbay, D., van der Torre, L., and Villata, S. (2009). Meta-argumentation modelling I: methodology and techniques. *Studia Logica*, 93(2-3):297–355.

[Boissier et al., 2015] Boissier, O., Bonnet, G., Ganascia, J.-G., Tessier, C., de Swarte, T., and Voyer, R. (2015). A roadmap towards ethical autonomous agents. Technical report, ETHICAA.

[Boltanski and Thévenot, 2006] Boltanski, L. and Thévenot, L. (2006). *On justification: Economies of worth*. Princeton University Press.

[Bordini et al., 2003a] Bordini, R., Fisher, M., Pardavila, C., and Wooldridge, M. (2003a). Model-checking AgentSpeak. In *AAMAS-03*, Melbourne, Australia.

[Bordini et al., 2003b] Bordini, R., Fisher, M., Visser, W., and Wooldridge, M. (2003b). Verifiable multi-agent programs. In Dastani, M., Dix, J., and Seghrouchni, A., editors, *ProMAS*.

[Bordini et al., 2007] Bordini, R. H., Hübner, J. F., and Wooldridge, M. (2007). *Programming Multi-Agent Systems in AgentSpeak Using **Jason***. Wiley Series in Agent Technology. John Wiley & Sons.

[Boudon, 2001] Boudon, R. (2001). *The origin of values*. New Brunswick Transactions.

[Bracciali et al., 2006] Bracciali, A., Endriss, U., Demetriou, N., Kakas, T., Lu, W., and Stathis, K. (2006). Crafting the mind of PROSOCS agents. *Applied Artificial Intelligence*, 20(2–4):105–131.

[Bratman, 1987] Bratman, M. (1987). *Intention, plans, and practical reason*. Harvard University Press, Cambridge, MA.

[Bratman, 1990] Bratman, M. (1990). What is intention? In Cohen, P., Morgan, J., and Pollack, M., editors, *Intentions in Communication*. MIT Press.

[Brazier et al., 1997] Brazier, F., van Eck, P., and Treur, J. (1997). *Simulating Social Phenomena*, volume 456, chapter Modelling a Society of Simple Agents: from Conceptual Specification to Experimentation, pages pp 103–109. Lecture Notes in Economics and Mathematical Systems.

[Brewka et al., 2013] Brewka, G., Ellmauthaler, S., Strass, H., Wallner, J., and Woltran, S. (2013). Abstract dialectical frameworks revisited. In *23th International Joint Conference on Artificial Intelligence*.

[Brey, 2014] Brey, P. (2014). From moral agents to moral factors: the structural ethics approach. In *The moral status of technical artefacts*, pages 125–142. Springer Netherlands.

[Brey, 2015] Brey, P. (2015). International differences in ethical standards and in the interpretation of legal frameworks. Technical report, SATORI.

[Broersen et al., 2001] Broersen, J., Dastani, M., Huang, Z., Hulstijn, J., and der Torre, L. V. (2001). The BOID architecture: Conflicts between beliefs, obligations, intentions and desires". In *Proceedings of the Fifth International Conference on Autonomous Agents (AA2001)*, pages 9–16. ACM Press.

[Burmeister, 2013] Burmeister, O. (2013). Achieving the goal of a global computing code of ethics through an international-localisation hybrid. *International Journal of Communication Ethics*, 10.

[Cabac et al., 2003] Cabac, L., Moldt, D., and Roelke, H. (2003). A proposal for structuring Petri net-based agent interaction protocols. In *24th International Conference on Application and Theory of Petri Nets*, pages 102–120.

[Cayrol and Lagasquie-Schiex, 2005] Cayrol, C. and Lagasquie-Schiex, M. (2005). On the acceptability of arguments in bipolar argumentation frameworks. *Lectures Notes in Computer Science*, 3571:378–389.

[Celaya et al., 2009] Celaya, J., Desrochers, A., and Graves, R. (2009). Modeling and analysis of multiagent systems using Petri nets. *Journal of Computers*, 4(10):981–996.

[Cerutti, 2011] Cerutti, F. (2011). *Argumentation-based practical reasoning: new models and algorithms*. PhD thesis, Università degli Studi di Brescia, Brescia.

[Chandy and Misra, 1988] Chandy, K. M. and Misra, J. (1988). *Parallel Program Design: A Foundation*. Addison-Wesley.

[Coelho and da Rocha Costa, 2009] Coelho, H. and da Rocha Costa, A. (2009). On the intelligence of moral agency. In *Encontro Português de Inteligência Artificial*, pages 12–15.

[Cohen and Levesque, 1990] Cohen, P. and Levesque, H. (1990). Intention is choice with commitment. *Artificial Intelligence*, 42(3):213–261.

[Coleman, 2001] Coleman, K. (2001). Android arete: Toward a virtue ethic for computational agents. *Ethics and Information Technology*, 3(4):247–265.

[Connor and Becker, 1979] Connor, P. and Becker, B. (1979). Values and the organization: Suggestions for research. In *Understanding Human Values: Individual and Societal*. New York: The Free Press.

[Cossentino and Potts, 2002] Cossentino, M. and Potts, C. (2002). A CASE tool supported methodology for the design of multi-agent systems. In *SERP*.

[Cristini and Tessier, 2012] Cristini, F. and Tessier, C. (2012). Nets-within-nets to model innovative space system architectures. In *33rd International Conference on Application and Theory of Petri Nets and Concurrency*, pages 348–367.

[Cummings, 2006] Cummings, M. (2006). Integrating ethics in design through the value-sensitive design approach. *Science and Engineering Ethics*, 12(4):701–715.

[Dam and Winikoff, 2003] Dam, K. and Winikoff, M. (2003). Comparing agent-oriented methodologies. In *Fifth International Bi-Conference Workshop on Agent-Oriented Information Systems*.

[Damasio, 2008] Damasio, A. (2008). *Descartes' error: Emotion, reason and the human brain*. Random House.

[Dambra, 2005] Dambra, S. (2005). Durkheim et la notion de morale. *Revue Interrogation*, 1.

[Dastani, 2008a] Dastani, M. (2008a). 2APL: a practical agent programming language. *Journal of Autonomous Agents and Multi-Agent Systems*, 16:214–248.

[Dastani, 2008b] Dastani, M. (2008b). 2apl: a practical agent programming language. *Autonomous Agents and Multi-Agent Systems*, 16(3):214–248.

[Dastani et al., 2003] Dastani, M., de Boer, F., Dignum, F., and Meyer, J.-J. (2003). Programming agent deliberation: An approach illustrated using the 3apl language. In *Proceedings of the Second International Conference on Autonomous Agents and MultiAgent Systems (AAMAS'03)*.

[David and Alla, 2005] David, R. and Alla, H. (2005). *Discrete, continuous, and hybrid Petri nets*. Springer.

[de Boer et al., 2000] de Boer, F., Hindriks, K., van der Hoek, W., and Meyer, J.-J. (2000). Agent programming with declarative goals. In *7th International Workshop on Intelligent Agents. Agent Theories Architectures and Language*, pages 228–243.

[de Kenessey and Darwall, 2014] de Kenessey, B. and Darwall, S. (2014). *Moral Psychology and Human Agency: Philosophical Essays on the Science of Ethics*, chapter Moral Psychology as Accountability. OUP Oxford.

[Dennett, 1987] Dennett, C. (1987). *The Intentional Stance*. The MIT Press.

[Dennings et al., 2010] Dennings, T., Borning, A., Friedman, B., Gill, B., Tadayoshi, K., and Maisel, W. (2010). Patients, pacemakers, and implantable defibrillators: Human values and security for wireless implantable medical devices. In *28th SIGCHI Conference on Human Factors in Computing Systems*, pages 917–926.

[Dennis et al., 2015] Dennis, L., Fisher, M., and Winfield, A. (2015). Towards verifiably ethical robot behaviour. In *1th International Workshop on AI and Ethics*.

[Dhillon and Torkzadeh, 2006] Dhillon, G. and Torkzadeh, G. (2006). Value-focused assessment of information system security in organizations. *Information Systems Journal*, 16(3):293–314.

[Dictionary, 2015] Dictionary, F. O. P. (2015). Ethical judgment.

[Doder and Woltran, 2014] Doder, D. and Woltran, S. (2014). Probabilistic argumentation frameworks – a logical approach. *Lecture Notes in Computer Science*, 8720:134–147.

[Dratwa, 2014] Dratwa, J. (2014). Ethics of security and surveillance technologies. Technical report, EG Opinin Report No 28.

[Dung, 1995] Dung, P. (1995). On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77:321–357.

[Dunne et al., 2011] Dunne, P., Hunter, A., McBurney, P., Parsons, S., and Wooldridge, M. (2011). Weighted argument systems: basic definitions, algorithms, and complexity results. *Artificial Intelligence*, 175(2):457–486.

[Endriss et al., 2004] Endriss, U., Mancarella, P., Sadri, F., Terreni, G., and Toni, F. (2004). Abductive logic programming with CIFF: implementation and applications. In *Convegno Italiano di logica computazionale (CILC-2004)*.

[Esteva et al., 2002] Esteva, M., de la Cruz, D., and Sierra, C. (2002). ISLANDER, an electronic institution editor. In *AAMAS*, pages 1045–1052.

[Faci and Logrippo, 1994] Faci, M. and Logrippo, L. (1994). Specifying Features and Analysing Their Interactions in a LOTOS Environment. In Bouma, L. and Velthuijsen, H., editors, *Feature Interactions in Telecommunications Systems*.

[Feather, 1996] Feather, N. (1996). Values, deservingness, and attitudes toward high achievers: Research on tall poppies. *The Ontario Symposium: The Psychology of Values*, 8:215–251.

[Ferris et al., 2010] Ferris, B., Watkins, K., and Borning, A. (2010). Onebusaway: A transit traveler information system. In *1st International ICST Conference on Mobile Computing, Applications, and Services*, pages 92–106.

[Finkel et al., 1995] Finkel, N. J., Maloney, S. T., Valbuena, M. Z., and Groscup, J. L. (1995). Lay perspectives on legal conundrums: Impossible and mistaken act cases. *Law and Human Behavior*, 19(6):593–608.

[Fisher, 1994] Fisher, M. (1994). A survey of concurrent METATEM – the language and its applications. In Gabbay, D. M. and Ohlbach, H. J., editors, *Temporal Logic - Proceedings of the First Intemational Conference (LNAI Volume 827)*, pages 480–505. Springer-Verlag: Heidelberg, Germany.

[Flanagan et al., 2008] Flanagan, M., Howe, D., and Nissembaum, H. (2008). Embodying values in technologies. In *Information Technology and Moral Philosophy*, pages 322–353. Cambridge Press University.

[Foot, 1964] Foot, P. (1964). The problem of abortion and the doctrine of the double effect. *Oxford Review*, pages 5–15.

[Friedman, 1996] Friedman, B. (1996). Value-sensitive design. *Interactions*, 3(6):16–23.

[Friedman et al., 2013] Friedman, B., Kahn, P., Borning, A., and Huldtgren, A. (2013). Value sensitive design and information systems. In *Early engagement and new technologies: Opening up the laboratory*, pages 55–95. Springer Netherlands.

[Gert, 2015] Gert, B. (2015). The definition of morality. In Zalta, E. N., editor, *The Stanford Encyclopedia of Philosophy*.

[Giacomo et al., 2000] Giacomo, G. D., Lesperance, Y., and Levesque, H. J. (2000). Congolog, a concurrent programming language based on the situation calculus. *Artificial Intelligence*, 121(1-2):109–169.

[Gigerenzer, 2010] Gigerenzer, G. (2010). Moral satisficing: Rethinking moral behavior as bounded rationality. *Topics in Cognitive Science*, 2(3):528–554.

[Gilmore et al., 2008] Gilmore, D., Cockton, G., Churchill, E., Kujala, S., Henderson, A., and Hammontree, M. (2008). Values, value and worth:

their relationship to hci? In *26th ACM Conference on Human Factors in Computing Systems*, pages 3933–3936.

[Gooskens, 2010] Gooskens, G. (2010). The ethical status of virtual actions. *Ethical Perspective*, 17(1):59–78.

[Graham et al., 2012] Graham, J., Haidt, J., Koleva, S., Motyl, M., Iyer, R., Wojcik, S., and Ditto, P. (2012). Moral foundations theory: The pragmatic validity of moral pluralism. *Advances in Experimental Social Psychology*, 47:55–130.

[Graham et al., 2011] Graham, J., Nosek, B., Haidt, J., Iyer, R., Koleva, S., and Ditto, P. (2011). Mapping the moral domain. *Journal of Personality and Social Psychology*, 101(2):366–385.

[Gratie, 2012] Gratie, C. (2012). *Extension-based semantics of argumentation frameworks for agent interactions*. PhD thesis, Utrecht University, Utrecht.

[Greene and Haidt, 2002] Greene, J. and Haidt, J. (2002). How (and where) does moral judgment work? *Trends in cognitive sciences*, 6(12):517–523.

[Guglielmo, 2015] Guglielmo, S. (2015). Moral judgment as information processing: an integrative review. *Frontiers in psychology*, 6.

[Guttag and Horning, 1978] Guttag, J. and Horning, J. (1978). The algebraic specification of abstract data types. *Acta Informatica*, 10:27–52.

[Haidt, 2001] Haidt, J. (2001). The emotional dog and its rational tail: a social intuitionist approach to moral judgment. *Psychological Review*, 108(4):814–834.

[Hengfei et al., 2012] Hengfei, L., Oren, N., and Norman, T. (2012). Probabilistic argumentation frameworks. *Lecture Notes in Computer Science*, 7132:1–16.

[Herzig et al., 2016] Herzig, A., Lorini, E., Perrussel, L., and Xiao, Z. (2016). Bdi logics for bdi architectures: old problems, new perspectives. *KI-Künstliche Intelligenz*, pages 1–11.

[Hitlin, 2003] Hitlin, S. (2003). Values as the core of personal identity: drawing links between two theories of the self. *Social Psychology Quarterly*, 66:118–137.

[Hofstede, 2001] Hofstede, G. (2001). *Culture's Consequences: Comparing Values, Behaviors, Institutions and Organizations Across Nations.* Sage Publications.

[Howden et al., 2001] Howden, N., Rönnquist, R., Hodgson, A., and Lucas, A. (2001). Jack intelligent agents-summary of an agent infrastructure. In *5th International conference on autonomous agents.*

[Hubner et al., 2002] Hubner, J., Sichman, J., and Boissier, O. (2002). Spécification structurelle, fonctionnelle et déontique d'organisations dans les SMA. In *Journees Francophones Intelligence Artificielle et Systemes Multi-Agents (JFIADSM'02).* Hermes.

[Hursthouse, 2013] Hursthouse, R. (2013). Virtue ethics. In Zalta, E. N., editor, *The Stanford Encyclopedia of Philosophy.*

[Iglesias et al., 1999] Iglesias, C., Garrijo, M., and Gonzalez, J. (1999). A survey of agent-oriented methodologies. In Müller, J., Singh, M. P., and Rao, A. S., editors, *Proceedings of the 5th International Workshop on Intelligent Agents V: Agent Theories, Architectures, and Languages (ATAL-98)*, volume 1555, pages 317–330. Springer-Verlag: Heidelberg, Germany.

[Ingelhart and Welzel, 2005] Ingelhart, R. and Welzel, C. (2005). *Modernization, Cultural Change, and Democracy.* Cambridge University Press.

[Ishita et al., 2010] Ishita, E., Oard, D., Fleischmann, K., Cheng, A.-S., and Templeton, T. (2010). Investigating multi-label classification for human values. *Proceedings of the American Society for Information Science and Technology*, 47(1):1–4.

[Jennings et al., 1998] Jennings, N., Sycara, K., and Wooldridge, M. (1998). A roadmap of agent research and development. *Autonomous Agents and Multi-Agents Systems*, 1(1):275–306.

[Johnson, 2014] Johnson, R. (2014). Kant's moral philosophy. In Zalta, E. N., editor, *The Stanford Encyclopedia of Philosophy.*

[Jones, 1990] Jones, C. (1990). *Systematic Software Development using VDM.* Prentice Hall International.

[Kacprzak et al., 2004] Kacprzak, M., Lomuscio, A., and Penczek, W. (2004). Verification of multiagent systems via unbounded model checking. In *Autonomous Agents and Multi-Agent Systems (AAMAS'04).*

[Kacprzak and Penczek, 2004] Kacprzak, M. and Penczek, W. (2004). Unbounded model checking for alternating-time temporal logic. In *Autonomous Agents and Multi-Agent Systems (AAMAS'04)*.

[Kakas et al., 1999] Kakas, A., Miller, R., and Toni, F. (1999). An argumentation framework for reasoning about actions and change. *Lecture Notes in Computer Science*, 1730:78–91.

[Kim and Lipson, 2009] Kim, K. and Lipson, H. (2009). Towards a 'theory of mind' in simulated robots. In *11th Annual Conference Companion on Genetic and Evolutionary Computation Conference*, pages 2071–2076.

[Kluckhohn, 1951] Kluckhohn, C. (1951). Values and value orientations in the theory of action. In *Toward a general theory of action*. Harvard University Press.

[Knoppen and Saris, 2009] Knoppen, D. and Saris, W. (2009). Do we have to combine values in the schwartz' human values scale? a comment on the davidov studies. *Survey Research Methods*, 3(2):91–103.

[Kowalski, 2006] Kowalski, R. (2006). Computational puzzles as Sybil defenses. In *6th International Workshop on Computational Logic in Multi-Agent Systems*, pages 1–22.

[Lamport, 1996] Lamport, L. (1996). The syntax and semantics of tla$^+$. Part 1: Definitions and Modules.

[Lewis et al., 2012] Lewis, A., Bardis, A., Flint, C., Mason, C., Smith, N., Tickle, C., and Zinser, J. (2012). Drawing the line somewhere: An experimental study of moral compromise. *Journal of Economic Psychology*, 33(4):718–725.

[Maio, 2010] Maio, G. (2010). Mental representation of social values. *Advances in Experimental Social Psychology*, 42:1–43.

[Malle et al., 2014] Malle, B. F., Guglielmo, S., and Monroe, A. E. (2014). A theory of blame. *Psychological Inquiry*, 25(2):147–186.

[Mao and Gratch, 2012] Mao, W. and Gratch, J. (2012). Modeling social causality and responsibility judgment in multi-agent interactions. *Journal of Artificial Intelligence Research*, 44(1):223–273.

[Martelli et al., 1997] Martelli, M., Mascardi, V., and Zini, F. (1997). CaseLP: a Complex Application Specification Environment base on Logic

Programming. In *Proc. of ICLP'97 workshop on Logc Programming and Mult-Agents*, pages 35–50.

[McConnell, 2014] McConnell, T. (2014). Moral dilemmas. In Zalta, E. N., editor, *The Stanford Encyclopedia of Philosophy*.

[McIntyre, 2014] McIntyre, A. (2014). Doctrine of double effect. In Zalta, E. N., editor, *The Stanford Encyclopedia of Philosophy*.

[McLaren, 2006] McLaren, B. (2006). Computational models of ethical reasoning: challenges, initial steps, and future directions. *IEEE Intelligent Systems*, 21(4):29–37.

[Mepham, 2000] Mepham, B. (2000). A framework for the ethical analysis of novel foods: The ethical matrix. *Journal of Agricultural and Environmental Ethics*, 12:165–176.

[Mepham, 2013] Mepham, B. (2013). *Ethical Principles and the Ethical Matrix*. J.P. Clark and C. Ritson.

[Mermet and Simon, 2009] Mermet, B. and Simon, G. (2009). GDT4MAS: an extension of the GDT model to specify and to verify MultiAgent Systems. In Decker, Sichman, S. and Castelfranchi, editors, *Proc. of 8th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2009)*, pages 505–512.

[Mermet and Simon, 2011] Mermet, B. and Simon, G. (2011). Specifying recursive agents with gdts. *Autonomous Agents and Multi-Agent Systems*, 23(2):273–301.

[Mermet and Simon, 2013] Mermet, B. and Simon, G. (2013). A new proof system to verify gdt agents. In Zavoral, F., Jung, J. J., and Badica, C., editors, *IDC*, volume 511 of *Studies in Computational Intelligence*, pages 181–187. Springer.

[Mermet et al., 2007] Mermet, B., Simon, G., Saval, A., and Zanuttini, B. (2007). Specifying, verifying and implementing a MAS: A case study. In Dastani, M., El Fallah Segrouchni, A., Ricci, A., and Winikoff, M., editors, *Post-Proc. 5th International Workshop on Programming Multi-Agent Systems (ProMAS'07)*, number 4908 in Lecture Notes in Artificial Intelligence, pages 172–189. Springer.

[Meyer et al., 2015] Meyer, J.-J., Broersen, J., and Herzig, A. (2015). Bdi logics. Technical report.

[Milner et al., 1992] Milner, R., Parrow, J., and Wlaker, D. (1992). A calculus of mobile processes. *Journal of Information and Computation*, 100.

[Moreham, 2008] Moreham, N. (2008). Why is privacy important? privacy, dignity and development of the new zealand breach of privacy tort. In Finn, J. and Todd, S., editors, *Law, Liberty and Legislation*, pages 231–248. LexisNexis.

[Nilsson and Erlandsson, 2015] Nilsson, A. and Erlandsson, A. (2015). The moral foundations taxonomy: Structural validity and relation to political ideology in sweden. *Personality and Individual Differences*, 76:28–32.

[Nissenbaum, 2001] Nissenbaum, H. (2001). How computer systems embody values. *IEEE Computer*, 120:118–119.

[Nissenbaum, 2005] Nissenbaum, H. (2005). Values in technical design. Technical report, Encyclopedia of Science, Technology and Ethics.

[of Professional Journalists, 2014] of Professional Journalists, S. (2014). Code of ethics.

[Oren, 2013] Oren, N. (2013). Argument schemes for normative practical reasoning. In *2nd International Workshop on Theory and Applications of Formal Argumentation*, pages 63–78.

[Owre et al., 1992] Owre, S., Shankar, N., and Rushby, J. (1992). Pvs: A prototype verification system. In *CADE 11*.

[Parks-Leduc et al., 2015] Parks-Leduc, L., Feldman, G., and Bardi, A. (2015). Personality traits and personal values: A meta-analysis. *Personality and Social Psychology Review*, 19(1):3–29.

[Parsons, 1951] Parsons, T. (1951). *The social system*. Psychology Press.

[Partala and Kujalan, 2016] Partala, T. and Kujalan, S. (2016). Exploring the role of ten universal values in using products and services. *Interacting with Computers*, 28(3):311–331.

[Perennou, 2014] Perennou, T. (2014). Ethics and autonomous agents: State-of-the art on legal issues. Technical report, École Télécom Management.

[Perrinjaquet et al., 2007] Perrinjaquet, A., Furrer, O., Usunier, J.-C., Cestre, G., and Valette-Florence, P. (2007). A test of the quasi-circumplex

structure of human values. *Journal of Research in Personality*, 41(4):820–840.

[Plato, 1966] Plato (1966). *The Republic (French translation from G. Leroux)*. Garnier Flammarion Paris.

[Raimondi and Lomuscio, 2004] Raimondi, F. and Lomuscio, A. (2004). Verification of multiagent systems via orderd binary decision diagrams: an algorithm and its implementation. In *Autonomous Agents and Multi-Agent Systems (AAMAS'04)*.

[raker, 2010] raker, J. S. (2010). *The Value of Virtual Worlds and Entities - A Philosophical Analysis of Virtual Worlds and Their Potential Impact on Well-Beings*. PhD thesis, University of Twente.

[Rao, 1996] Rao, A. (1996). Agentspeak(l): Bdi agents speak out in a logical computable language. In *7th European Workshop on Modelling Autonomous Agents in a Multi-Agent World*, pages 42–55.

[Rao and Georgeff, 1991] Rao, A. and Georgeff, M. (1991). Modeling rational agents within a BDI-architecture. In *2nd International Conference on Principles of Knowledge Representation and Reasoning*, pages 473–484.

[Rao and Georgeff, 1995] Rao, A. and Georgeff, M. (1995). BDI agents from theory to practice. In *Technical note 56*. AAII.

[Reeder et al., 2002] Reeder, G. D., Kumar, S., Hesson-McInnis, M. S., and Trafimow, D. (2002). Inferences about the morality of an aggressor: The role of perceived motive. *Journal of personality and social psychology*, 83(4):789.

[Ricoeur, 1995] Ricoeur, P. (1995). *Oneself as another*. University of Chicago Press.

[Rohan, 2000] Rohan, M. (2000). A rose by any name? the values construct. *Personality and Social Psychology Review*, 4(3):255–277.

[Rokeach, 1973] Rokeach, M. (1973). *The nature of human values*. New York Free Press.

[Ropes and Guglielmo, 2016] Ropes, A. and Guglielmo, S. (2016). Interconnections between perceptions of blame, mind, and moral abilities. Technical Report 36, Psychology Honors Projects, `http://digitalcommons.macalester.edu/psychology_honors/36`.

[Ros et al., 1999] Ros, M., Schwartz, S., and Surkiss, S. (1999). Basic individual values, work values and the meaning of work. *Applied Psychology: An International Review*, 48:49–71.

[Russo et al., 2001] Russo, A., Miller, R., Nuiseibeh, B., and Kramer, J. (2001). An abductive approach for analysing event-based requirements specifications. Technical report, Department of Computing, Imperial College.

[Sabas et al., 2002] Sabas, A., Delisle, S., and Badri, M. (2002). A comparative analysis of multiagent system development methodologies: Towards a unified approach. In Trappl, R., editor, *Cybernetics and Systems*, pages 599–604. Austrian Society for Cybernetics Studies.

[Sánchez-Herrera et al., 2007] Sánchez-Herrera, R., Villanueva-Paredes, N., and López-Mellado, E. (2007). High-level modelling of cooperative mobile robot systems. In *Distributed Autonomous Robotic Systems 6 (DARS 2007),*.

[Schrier and Gibson, 2010] Schrier, K. and Gibson, D. (2010). Values between systems : Designing ethical gameplay. In Sicart, M., editor, *Ethics and Game Design: Teaching Values Through Play*, pages 1–15. IGI Global.

[Schroeder, 2003] Schroeder, D. (2003). Technology assessment and the ethical matrix. *Poiesis and Praxis*, 1(4):295–307.

[Schroeder, 2011] Schroeder, R. (2011). Comparing avatar and video representations. In Childs, M., editor, *Reinventing Ourselves: Contemporary Concepts of Identity in Virtual Worlds*, pages 235–251. Springer London.

[Schwartz, 1994] Schwartz, S. (1994). Are there universal aspects in the structure and contents of human values? *Journal of Social Issues*, 50:19–45.

[Schwartz, 1996] Schwartz, S. (1996). Value priorities and behavior: Applying a theory of integrated value systems. *The Ontario Symposium: The Psychology of Values*, 8:1–24.

[Schwartz, 2006] Schwartz, S. (2006). Basic human values: Theory, measurement, and applications. *Revue française de sociologie*, 47(4):249–288.

[Schwartz, 2012] Schwartz, S. (2012). An overview of schwartz theory of basic values. *Online Readings of Psychology and Culture*, 2(1).

[Schwartz and Bilsky, 1987] Schwartz, S. and Bilsky, W. (1987). Towards an universal psychological structure of human values. *Journal of Personality and Social Psychology*, 53:550–562.

[Seligman and Katz, 1996] Seligman, C. and Katz, A. (1996). The dynamics of value systems. *The Ontario Symposium: The Psychology of Values*, 8:53–75.

[Shapiro et al., 2002] Shapiro, S., LespÃ©rance, Y., and Levesque, H. J. (2002). The Cognitive Agents Specification Language and Verification Environment for Multiagent Systems. In *AAMAS*, pages 19–26. ACM Press.

[Shaver, 1985] Shaver, K. (1985). *The Attribution of Blame: Causality, Responsibility, and Blameworthiness*. New York Springer.

[Shilton, 2010] Shilton, K. (2010). Technology development with an agenda: interventions to emphasize values in design. In *73rd Annual Meeting of the American Society for Information Science & Technology*, pages 1–10.

[Shilton et al., 2013] Shilton, K., Koepfler, J., and Fleischmann, K. (2013). Charting sociotechnical dimensions of values for design research. *The Information Society*, 29(5):259–271.

[Shilton et al., 2014] Shilton, K., Koepfler, J., and Fleischmann, K. (2014). How to see values in social computing: methods for studying values dimensions. In *17th ACM Conference on Computer Supported Cooperative Work & Social Computing*, pages 426–435.

[Shoham, 2009] Shoham, Y. (2009). Logical theories of intention and the database perspective. *Journal of Philosophical Logic*, 38(6):633.

[Shweder et al., 1997] Shweder, R., Much, N., Mahapatra, M., and Park, L. (1997). The big three of morality (autonomy, community, and divinity), and the big three explanations of suffering. In Brandt, A. and Rozin, P., editors, *Morality and Health*, pages 119–169. New York Routledge.

[Sicart, 2011] Sicart, M. (2011). *The ethics of computer games*. MIT Press.

[Sicart, 2013] Sicart, M. (2013). Moral dilemmas in computer games. *MIT Design Issues*, 29(3):28–37.

[Simon et al., 2006] Simon, G., Mermet, B., and Fournier, D. (2006). Goal Decomposition Tree: An agent model to generate a validated agent behaviour. In Baldoni, M., Endriss, U., Omicini, A., and Torroni, P., editors, *Declarative Agent Languages and Technologies III: Third International Workshop, DALT 2005*, volume 3904 of *LNCS*, pages 124–140. Springer Verlag.

[Sinnott-Armstrong, 2014] Sinnott-Armstrong, W. (2014). Consequentialism. In Zalta, E. N., editor, *The Stanford Encyclopedia of Philosophy*.

[Solomon, 2014] Solomon, D. (2014). *Employee and Organization Security Value Alignment Through Value Sensitive Security Policy Design*. PhD thesis, Nova Southeastern University.

[Solove, 2006] Solove, D. (2006). A taxonomy of privacy. *University of Pennsylvania Law Review*, 154(3):477–561.

[Spivey, 1987] Spivey, J. M. (1987). *Understanding Z: a specification language and its formal semantics*. Cambridge University Press.

[Stathis et al., 2004] Stathis, K., Kakas, A., Lu, W., Demetriou, N., Endriss, U., and Bracciali, A. (2004). PROSOCS: a platform for programming software agents in computational logic. In Müller, J. and Petta, P., editors, *Proceedings of the Fourth International Symposium "From Agent Theory to Agent Implementation" (AT2AI-4)*, pages pages 523–528, Vienna, Austria.

[Steinmetz et al., 2012] Steinmetz, H., Isidor, R., and Baeuerle, N. (2012). Testing the circular structure of human values: A meta-analytical structural equation modelling approach. *Survey Research Methods*, 6(1):61–75.

[Stevens, 2009] Stevens, B. (2009). Corporate ethical codes as strategic documents: An analysis of success and failure. *Electronic Journal of Business Ethics and Organization Studies*, 14(2):14–20.

[Swchartz, 1992] Swchartz, S. (1992). Universals in the content and structure of values: Theoretical advances and empirical tests in 20 countries. *Advances in Experimental Social Psychology*, 25:1–65.

[Swchartz and Bilsky, 1990] Swchartz, S. and Bilsky, W. (1990). Toward a theory of the universal content and structure of values: Extensions and cross cultural replications. *Journal of Personality and Social Psychology*, 58:878–891.

[Tetlock, 1986] Tetlock, P. (1986). A value pluralism model of ideological reasoning. *Journal of Personality and Sociol Psychology*, 50(4):819–827.

[Tetlock et al., 2007] Tetlock, P. E., Visser, P. S., Singh, R., Polifroni, M., Scott, A., Elson, S. B., Mazzocco, P., and Rescober, P. (2007). People as intuitive prosecutors: The impact of social-control goals on attributions of responsibility. *Journal of Experimental Social Psychology*, 43(2):195–209.

[Thimm, 2012] Thimm, M. (2012). A probabilistic semantics for abstract argumentation. In *20th European Conference on Artificial Intelligence*, pages 750–755.

[Timmermans et al., 2010] Timmermans, J., Stahl, B., Ikonen, V., and Bozdag, E. (2010). The ethics of cloud computing: A conceptual review. In *2nd International Conference on Cloud Computing*.

[Timmons, 2012] Timmons, M. (2012). *Moral theory: An introduction*. Rowman and Littlefiled.

[Valette-Florence et al., 1996] Valette-Florence, P., Odin, Y., and Vinais, J. (1996). Analyse confirmatoire des domaines motivationnels de schwartz : Une application au domaine des media. *Actes du CongrÃ¨s*, 12:125–140.

[van Marrewijk and Werre, 2003] van Marrewijk, M. and Werre, M. (2003). Multiple levels of corporate sustainability. *Journal of Business Ethics*, 4(2-3):107–119.

[van Riemsdijk et al., 2004] van Riemsdijk, M., Dastani, M., Dignum, F., and Meyer, J.-J. (2004). Dynamics of declarative goals in agent programming. In *Proceedings of Declarative Agent Languages and Technologies (DALT'04)*.

[Vincent et al., 2001] Vincent, R., Horling, B., and Lesser, V. (2001). An agent infrastructure to build and evaluate multi-agent systems: the Java agent framework and multi-agent system simulator. In *Infrastructure for Agents, Multi-Agent Systems, and Scalable Multi-Agent Systems*.

[Voas, 2014] Voas, D. (2014). Towards a sociology of attitudes. *Sociological Research Online*, 19(1):1–12.

[Voyer, 2014] Voyer, R. (2014). Une histoire de la philosophie morale. Technical report, ETHICAA.

[Welzel and Inglehart, 2010] Welzel, C. and Inglehart, R. (2010). Agency, values, and well-being: A human development model. *Social Indicators Research*, 97(1):43–63.

[Wiegel, 2006] Wiegel, V. (2006). Building blocks for artificial moral agents. *Proc. Artificial Life X*.

[Wiener, 1988] Wiener, Y. (1988). Forms of value systems: A focus on organisational effectiveness and cultural change and maintenance. *Academy of Management Review*, 13(4):534–545.

[Winfield et al., 2014] Winfield, A., Blum, C., and Liu, W. (2014). Towards and Ethical Robot: Internal Models, Consequences and Ethical Action Selection. In Mistry, M., Leonardis, A., Witkowski, M., and Melhuish, C., editors, *Advances in Autonomous Robotics Systems*, volume 8717 of *Lecture Notes in Computer Science*, pages 85–96. Springer.

[Winikoff, 2005] Winikoff, M. (2005). Jack intelligent agents: An industrial strength platform. In Bordini, R. H., Dastani, M., Dix, J., and Fallah-Seghrouchni, A. E., editors, *Multi-Agent Programming*, volume 15 of *Multiagent Systems, Artificial Societies, and Simulated Organizations*, pages 175–193. Springer.

[Winikoff et al., 2003] Winikoff, M., Padgham, L., Harland, J., and Thangarajah, J. (2003). Declarative & procedural goals in intelligent agent systems. In *8th International Conference on Principles of Knowledge Representation and Reasoning (KR2002)*.

[Wooldridge et al., 2000] Wooldridge, M., Jennings, N. R., and Kinny, D. (2000). The gaia methodology for agent-oriented analysis and design. *Journal of Autonomous Agents and Multi-Agent Systems*, 3(3):285–312.