A Declarative Modular Framework for Representing and Applying Ethical Principles

Fiona Berreby, Gauvain Bourgne, Jean-Gabriel Ganascia

A Declarative Modular Framework for Representing and Applying Ethical Principles

Fiona Berreby
Sorbonne Universités, UPMC
CNRS, UMR 7606, LIP6
F-75005, Paris, France
fiona.berreby@lip6.fr

Gauvain Bourgne
Sorbonne Universités, UPMC
CNRS, UMR 7606, LIP6
F-75005, Paris, France
gauvain.bourgne@lip6.fr

Jean-Gabriel Ganascia
Sorbonne Universités, UPMC
CNRS, UMR 7606, LIP6
F-75005, Paris, France
jean-gabriel.ganascia@lip6.fr

ABSTRACT

This paper investigates the use of high-level action languages for designing ethical autonomous agents. It proposes a novel and modular logic-based framework for representing and reasoning over a variety of ethical theories, based on a modified version of the Event Calculus and implemented in Answer Set Programming. The ethical decision-making process is conceived of as a multi-step procedure captured by four types of interdependent models which allow the agent to assess its environment, reason over its accountability and make ethically informed choices. The overarching ambition of the presented research is twofold. First, to allow the systematic representation of an unbounded number of ethical reasoning processes, through a framework that is adaptable and extensible by virtue of its designed hierarchisation and standard syntax. Second, to avoid the pitfall of much research in current computational ethics that too readily embed moral information within computational engines, thereby feeding agents with atomic answers that fail to truly represent underlying dynamics. We aim instead to comprehensively displace the burden of moral reasoning from the programmer to the program itself.

Keywords

Computational Ethics; Answer Set Programming; Event Calculus; Reasoning about Actions and Change.

1. INTRODUCTION

The study of morality from a computational point of view has attracted a rising interest from researchers in artificial intelligence; as reviewed in [2]. Indeed, the growing autonomy of artificial agents and increase in the number of tasks that are delegated to them urges us to address their capacity to process ethical restrictions and preferences, be it within their own internal structure or for interaction with human users. Fields as varied as health-care or transportation pose ethical issues that are in this sense particularly pressing, as they may confront agents with decisions that yield immediate or heavy consequences. Computational ethics can also help us better understand morality and reason more clearly over ethical concepts that are employed throughout philosophical, legal and technological domains. In this context, our aim is to provide a modular architecture that allows for the systematic and adaptable representation of ethical principles. To achieve this, we present a coherent set of models which together enable the agent to appraise its environment, integrate ethical rules, and determine from the implementation of those rules either a course of action or an appraisal of the behaviour of other agents. These are implemented in Answer Set Programming[1] based on a modified version of the Event Calculus.

Formally, we chose the use of non-monotonic logic as its study has been put forward as a way to handle the kind of defeasible generalisations that pervade much of our commonsense reasoning, and that are poorly captured by classical logic systems [14]. The term covers a family of formal frameworks devised to apprehend the kind of inference where no conclusion is drawn definitely, but stays open to modification in the light of further information. This kind of default based reasoning is significantly present in ethical reasoning. Such factors as the presence of alternative options, indirect consequences, or extenuating circumstances might overthrow our ethical judgement of an action. Accordingly, non-monotonic goal specification languages are particularly well suited to modelling ethical reasoning.

The article is structured as follows. We begin by introducing ethical philosophy concepts and discussing related works [Sect.2], then present the architecture of the framework [Sect.3]. Next, we define and discuss each type of model [Sects.4-7], and illustrate their implementation through a case example [Sect.8]. Finally, we conclude [Sect.9]

2. MOTIVATION

2.1 Ethical Theories

The study of ethics is the study of the beliefs that people may or should have to control their behaviour. A standard tripartite classification splits the field into *metaethics*, which is concerned with the ontological status of ethical concepts, *applied ethics*, which is concerned with applying moral rules to particular environments, and *normative ethics*, which is concerned with determining, comparing and explaining accounts of the ethically right and wrong [10]. The present work informs applied ethics in that it presents a scheme for designing ethically constrained artificial agents that may act in a variety of applied domains. It also informs normative ethics in that it purports to model the processes that underpin normative ethical decision-making, with the possibility

[1]For a description of Answer set Programming, see [18].

of confronting different views. It centres on two of its main branches: consequentialist and deontological ethics.

The Good and The Right

Consequentialist ethics hinge around the idea that actions are to be morally assessed in terms of their outcome, and can only be right or wrong derivatively, in virtue of what they produce. A morally right action is one that brings about a good, or the best, state of affairs. Yet in order to determine the rightness of an action, consequentialists must first establish what constitutes a good state of affairs, i.e. determine what is broadly called "the Good" [1]. This then puts them in the position to assert that actions are morally part of "the Right" as far as they increase the Good. Consequentialist theories therefore follow from, rely on and eventually supersede, theories of the Good. Disagreement among consequentialists about what the Good consists in, however, has nourished a number of strands of consequentialism. It has been said to stem from the happiness or well being of sentient beings (utilitarianism), the welfare of others (ethical altruism), personal self-interest (ethical egoism), or the respect of individual rights (utilitarianism of rights).

Deontological theories (from the Greek *deon*, "duty") claim that the moral value of an action is determined (at least partly) by some intrinsic feature of the action. Usually, this feature is a rational obligation or prohibition under which the actions falls, that constrains the agent to behave in a particular way towards others. For example, a deontological rule may state that lying is unethical, entailing that any utterance which contains a lie is prohibited. Because actions are thought to be right or wrong depending on their conformity with a moral norm or duty, the permissibility of an action is in some ways independent of its consequences. As such, the Right here has priority over the Good: an action may be wrong to the deontologist even if it maximises the Good, and right even if it minimises it. Attempts at defining the Good will henceforth be referred to as *theories of the Good*, and attempts at defining the Right, whether consequentialist or deontological, as *theories of the Right*.

2.2 Existing Works in Computational Ethics

A number of works have presented computational models of various ethical theories, including duty and rule based deontology [3][5][24], divine command deontology [10], consequentialism [13][17], or norm instantiation [27]. However, the models in these works often tend to directly embed ethical information within the agent's decision-making process, without actually generating moral reasoning to speak of. While they succeed in executing straightforward implementations of individual restrictions, they fail to provide an explicit representation of causal relations and ethical thought processes, thereby limiting their applicability and scope.

For example, using prospective logic, Pereira et al. [24] model a deontological rule which precludes intentional killing via the rule '*falsum* ← *intentionalKilling*.' Yet the way they determine whether '*intentionalKilling*' obtains is by atomically stating whether the end point of an action entails it, using rules of the form '*intentionalKilling* ← *end(A,iKill(Y))*.' where A is the evaluated action. The difficulty with this kind of formalization is that the ethical appraisal of an action is directly indicated by action-specific statements, rather than extracted as a form of understanding from the environment and the ethical rules in place. This fails to represent the actual reasoning that underpins ethical decision making. Moreover, there is no account of causality, such that the action and its consequences are not dynamically linked; the relationship between them is stated rather than inferred. Therefore, no account of ethical responsibility can be discussed on its basis. In addition, because rules lack expressive power, an entirely new program is required to model each new scenario, dilemma or theory, even if there exists common features. This also means that formalisms of this type cannot compare or contrast different ethical theories, nor can they make explicit their assumptions.

More recent works have holistically explored the architecture of ethical judgements [11][9], taking into account the need to explicitly represent and compartmentalise reasoning processes. This work inscribes itself within this pursuit.

3. STRUCTURAL SCHEME

3.1 Models and Modularity

An explicit representation of ethical reasoning enables an agent to assess the permissibility of an action or set of actions, either to inform its own decision-making or to judge the behaviour of others. To achieve this, the agent 'tests out' possible actions in specified simulations so as to evaluate their consequences or inherent ethical status. The outcome of the simulation then yields a set of permissible or impermissible actions, which dictates its upcoming behaviour or appraisal of other agents. The framework presented here is concerned with this assessment process, rather than with what the agent chooses to do with its upshot.

The ethical decision-making process is apprehended as a four-step procedure captured by four types of interdependent models: an *action model*, a *causal model*, a *model of the Good*, and a *model of the Right*. The first two models provide the agent with an entirely ethics-free understanding of the world, the second two provide an ethical over-layer that the agent can parse and apply back onto its knowledge of the world. The action model presented here, and which provides the basis for the framework, is based on a modified version of the Event Calculus as in [9]. We define these models here, as illustrated in figure 1.

Definition 1. A *action model* $\mathbb{A}$ enables the agent to represent its environment and the changes that take place in it. It takes as input a *set of performed actions*. It is composed of an *initial situation* containing the fluents that hold at T=0, a *specification of events* containing a set of events and of dependence relations, and an *event motor* which enables the simulation to evolve. It generates an *event trace* of each simulation which designates for each time point the events that occur and fluents that hold.

Definition 2. A *causal model* $\mathbb{C}$ tracks the causal powers of actions, enabling reasoning over agent responsibility and accountability. It takes as input the *event trace* given by the action model and a *specification of events* containing a set of events and of dependence relations. It is composed of a *causal motor* which enables the creation of the causal tree that characterises the simulation. It generates a *causal trace* of each simulation which designates for each time point the causal relations that obtain between events and fluents.

Definition 3. A *model of the Good* $\mathbb{G}$ makes a claim about the intrinsic value of goals or events. It is composed of a

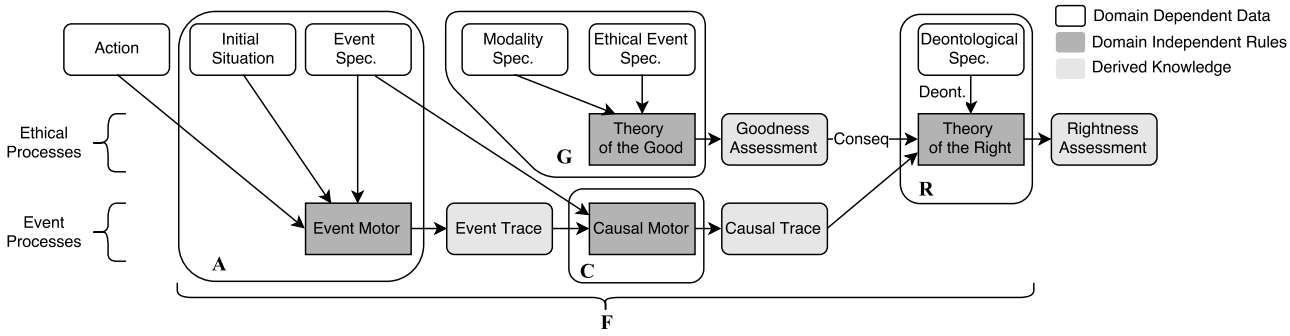

Figure 1: Models and Modularity

specification of modalities, an *ethical specification of events* composed of a set events and a set of ethical dependence relations, and a *theory or set of theories of the Good*. It generates a *goodness assessment* of events, made explicit by a valuation of events as having good or bad ramifications.

Definition 4. A *model of the Right* $\mathbb{R}$ considers what an agent should do, or is most justified in doing, within the circumstances of his actions. It takes as input the *causal trace* given by the causal model and, in the case that a given theory of the Right contains consequentialist principles, a *goodness assessment* given by the model of the Good. It is composed of a *theory or set of theories of the Right*, and, in the case that a given theory of the Right contains deontological principles, a *set of deontological specifications*. It generates a *rightness assessment* of actions, made explicit by a valuation of actions as permissible or impermissible in relation to each given theory of the Right.

These four types of models are interdependent at varying degrees. Models of the Good and the Right always rely on an action and a causal model. But while a causal model is always necessary, the particular formulation of the causal motor may vary, to account for instance for different definitions of causes and consequences. Because the event motor provides the basis for the framework, however, it is proposed as unique and unvarying. Pertaining to ethical models, having a model of the Good is necessary to model consequentialist theories of the Right as well as deontological ones with consequentialist constraints, but not purely deontological ones. Inter-dependencies may also hold within a type of model, particularly in the case of theories of the Right which call upon one-another. The well defined hierarchy between the different types of models gives the framework the capacity to model but also compare a potentially unlimited number of ethical theories. Compartmentalising different types of processes means they can be analysed specifically. Substituting a particular model while keeping constant the others allows for the individualised examination of its ramification.

Based on these models, we may now define the framework which enables the ethical assessment of actions.

Definition 5. The *ethical assessment framework* is defined as:

$$\mathbb{F} = \langle \mathbb{A}_i, \mathbb{C}_i, \mathbb{G}_i, \mathbb{R}_i \rangle$$

Given an ethical assessment framework $\mathbb{F}$, and a set $\mathcal{A}$ of performed actions α, we then define the set of permissible actions as:

$$Permissible(\mathbb{F}, \mathcal{A}) = \{\alpha \in \mathcal{A} / \mathbb{A}_i, \mathbb{C}_i, \mathbb{G}_i, \mathbb{R}_i \models permissible(\alpha)\}$$

4. EVENT MOTOR

A Reformulation of the Event Calculus

The presented event motor corresponds to the full Event Calculus described in [25], with a number of additions. To fit the requirements of modeling complex scenarios, we introduce automatic events in addition to actions. These automatic events occur when all their preconditions, in the form of fluents, hold, without direct input from the agent. Furthermore, we make a distinction between inertial fluents, which remain true until terminated by an event occurrence, and non inertial fluents which are true for one time point [21]. Finally, we introduce a set of simulations, which enable the agent to separately and simultaneously simulate the effects of different actions upon the same scenario. We denote the set of functions and constants as follows: $\mathcal{S}$ is a set of simulations, $\mathcal{T}$ a set of time points; $\mathcal{F}$ a set of fluents, $\mathcal{A}$ a set of actions, $\mathcal{U}$ a set of automatic events, and $\mathcal{E}$ a set of events where $\mathcal{E} \equiv \mathcal{A} \cup \mathcal{U}$.

Event Effect Axioms

A number of predicates characterise the behaviour of fluents relative to the occurrence of events. `initially(F)` indicates that F is true initially; `effect(E,F)` indicates that E can cause F; `initiates(S,E,F,T)` indicates that E initiates F at T in S (and F is not the negation of a fluent); `terminates(S,E,F,T)` indicates that E terminates F at T in S; `clipped(S,F,T)` indicates that F is clipped at T in S; `nonInertial(F)` points out the special kinds of fluents that are not constrained by the law of inertia; `holds(S,F,T)` indicates that F is true at T in S. These predicates enable us to axiomatize the principles that govern fluents: a fluent holds at T is S if it was initiated by an event occurrence at T-1 in S; a fluent which is true at T in S continues to hold until the occurrence of an event which terminates it, unless it is non inertial, in which case it holds at T only.

```
holds(S,F,0):-initially(F),sim(S).
initiates(S,E,F,T):-
    effect(E,F),occurs(S,E,T),not negative(S,F).
negative(S,neg(F)):-effect(E,neg(F)),sim(S).
terminates(S,E,F,T):-
    effect(E,neg(F)),occurs(S,E,T).
```

```
clipped(S,F,T):-terminates(S,E,F,T).
holds(S,F,T):-
    initiates(S,E,F,T-1),time(T).
holds(S,F,T):-
    holds(S,F,T-1),not clipped(S,F,T-1),
    not nonInertial(F),time(T).
```

Events Precondition Axioms

A number of predicates characterise the behaviour of events relative to the truth values of fluents. `prec(F,E)` indicates that F is a precondition for E; `incomplete(S,E,T)` indicates that E is incomplete at T in S; `possible(S,E,T)` indicates that E is possible at T in S; `occurs(S,U,T)` indicates that U occurs at T in S; `occurs(S,A,T)` indicates that A occurs at T in S. These predicates allow us to axiomatize the principles that govern the occurrence of events: an automatic event occurs at T in S if all its preconditions are true at T in S; an action occurs at T in S if all its preconditions are true and an agent performs A at T in S.

```
incomplete(S,E,T):-
    prec(F,E),not holds(S,F,T),sim(S),time(T).
possible(S,E,T):-
    not incomplete(S,E,T),sim(S),event(E),time(T).
occurs(S,U,T):-possible(S,U,T),auto(U).
occurs(S,A,T):-
    possible(S,A,T),performs(S,D,A,T),act(A).
```

5. CAUSAL MOTOR

Causality Axioms

Defining causality in terms of consequences and based on the architecture of the Event Calculus affords us with a functional trace of causal paths and allows us to dynamically assess causal relationships. We define a consequence in the following way.

Definition 6. A fluent F is a *consequence* of an event E if E initiates F, and both obtain. An event E is a *consequence* of a fluent F if F is a precondition to E, and both obtain.

This definition accounts for the possibility that there may be more than one precondition for the occurrence of E, and that F may not be considered a cause of E if E does not occur (for instance because other preconditions were not fulfilled). To model it, we define the predicate `cons(S,E1,T,E2)`, which indicates that event E2 is a consequence of event E1 which happened in S at T. The referenced time point denotes the time at which occurred the *first* event within a causal chain. A causal chain is composed of a series of fluents and events, but the beginning and end of a causal chain are events.

```
cons(S,E,T,F):-
    occurs(S,E,T),effect(E,F),holds(S,F,T+1).
cons(S,F,T,E):-
    occurs(S,E,T),prec(F,E),holds(S,F,T).
cons(S,E1,T1,E3):-
    cons(S,E1,T1,C2),cons(S,C2,T2,E3),
    event(E1),event(E3),T2>T1.
```

6. THEORIES OF THE GOOD

In this section, we present two modes for defining the Good, one based on rights and one based on values. These modes are interchangeable and can also be combined. We then present a model for quantifying the Good once it has been qualified, which both allows it to be integrated within theories of the Right and gives events meaningful weights. Rights, values, or other means of defining the Good are together called *modalities* [2].

6.1 Qualifying The Good

6.1.1 *Theory of Rights*

Nozick's so called "utilitarianism of rights" posits that rights not being violated is constitutive of the Good to be maximized [22]. A right may be defined as a "*justified claim that individuals and groups can make upon other individuals or upon society; to have a right is to be in a position to determine by one's choices, what others should do or need not do*" [7]. This definition captures well the fact that a right denotes both a state of affairs for the person concerned (the exercise of the right) and a constraint imposed upon others (the prohibition of violating the right). We define the rules such that an event which involves people and negates a right is *bad* in relation to that right, and an event which involves people but does not negate a right is *good* in relation to that right. An event may as such be bad in relation to a right and good in relation to another. However, no event which involves people is neutral towards rights: it either negates a particular right or it does not. This principle of 'excluded middle' is made explicit by the use of negation as failure in the rule. Note that rights are to be defined as `right(M)` in the *modality specification*.

```
bad(E,X,M):-effect(E,involves(X)),
    effect(E,neg(M)),right(M).
good(E,X,M):-effect(E,involves(X)),
    not effect(E,neg(M)),right(M).
```

6.1.2 *Theory of Values*

A value-based theory also provides an efficient way to assess the initial worth of events relative to whether these promote certain values. A value may be defined as *"a conception, explicit or implicit, distinctive of an individual, or characteristic of a group, of the desirable which influences the selection from available modes, means, and ends of action"*[16]. A value is therefore a type of independent entity which can be displayed, or not, by particular events caused by the actions of agents, or by actions themselves. Values can be general or specific to different contexts, such as the workplace or the education of children. We define the rules such that an event which displays a particular value is *good* in relation to that value, and an event which displays the negation of a value is *bad* in relation to that value. Events that display neither a value nor its negation are neither good nor bad in relation to it. Note that values are to be defined as `value(M)` in the *modality specification*.

```
good(E,X,M):-effect(E,involves(X)),
    effect(E,displays(M)),value(M).
bad(E,X,M):-effect(E,involves(X)),
    effect(E,neg(displays(M))),value(M).
```

[2]Note that *theories of the Right* and *rights as modalities* are false friends. The Right denotes the ethically correct while the rights within a theory of the Good denote individual principles of freedom or entitlement

6.2 Quantifying the Good

Once the content of the Good and the Bad has been determined by way of a theory of the Good, we proceed to quantifying this content, i.e. weighing the good and bad ramifications of events. We define three weighing parameters, by accounting for:

- The number of people involved in events. For example, an event that affects five people will have a five times greater count than an event which affects one person. This information is given by the `good(E,X,M)` and `bad(E,X,M)` predicates as X.
- The relative value of the people involved in the event. For example, it may be more significant to save children than adults, or harm healthy people rather than declining patients. This is measured by assigning to each affected group a numerical weight, expressed by the `t_Weight(E,G,N)` predicate where E is an event, G its target group and N their given weight.
- The importance of the modality affected by the event. For example, displaying helpfulness may be more important than displaying politeness, respecting the right to life may be more important than respecting the right to property. This is measured by assigning to each modality a numerical weight, expressed by the `m_Weight(M,N)` predicate where M is the modality and N the given weight. The measurement scale is unfixed, and may be defined by preference relations whereby the preference of a modality over another will mean the former has a greater weight than the latter.

Assigning weights to modalities and groups is nontrivial, and this proposed method is an introduction to the many ways of doing it. It is possible, for instance, to enrich it by accounting for further dependencies, such as correlation between some modalities and people (e.g. autonomy might be essential for adults, and safety for children), or the importance of non-human affected parties (e.g. animals, the environment).

The next step consists in integrating all weights into a single number which expresses the weight of an event relative to a particular modality and group of people, captured by the predicates `weightedGood(E,N,M)` and `weightedBad(E,N,M)`, where N is the product of the number of affected people, target weight and modality weight. The overall weight of an event then corresponds to the difference between the sums of all its weighted good and bad ramifications. As such, the greater the weight of an event, the more it participates in the Good, while events with negative weights do more harm than good. Weights are given by the `weight(E,N)` predicate. This predicate will be used to define rules for upcoming theories of the Right, and as such is what enables the integration of the Good with the Right. Note that target and modality weights are to be defined as `t_Weight(E,G,N)` and `m_Weight(M,N)` in the *modality specification*.

```
weightedGood(E,X*N1*N2,M):-good(E,X,M),
    t_Weight(E,G,N1),m_Weight(M,N2).
weightedBad(E,X*N1*N2,M):- bad(E,X,M),
    t_Weight(E,G,N1),m_Weight(M,N2).
weight(E,N1-N3) :-
    N1=#sum[weightedGood(E,N2,M1)=N2],
    N3=#sum[weightedBad(E,N4,M2)=N4],
    number(N1;N3),event(E).
```

7. THEORIES OF THE RIGHT

7.1 Consequentialist Ethics

Consequentialist ethics take many forms, ranging from simple principles for action to complex theories for maximising the Good. We here describe and model five of them.

7.1.1 Prohibiting Purely Detrimental Actions

The first consequentialist principle states that actions with purely detrimental effects are impermissible. This intuitive rule is relevant to most ethical scenarios and can supplement other theories of the Right which may focus on actions with complex effects. To implement the rule, we define the predicates `badCons(S,A,T)` and `goodCons(S,A,T)`, which respectively indicate that an action A occurring at T in S provokes at least one bad or one good consequence. We then state that an action is impermissible if it only has bad consequences, and that any other action that has not been shown to be impermissible is by default permissible.

```
badCons(S,A,T):-
    act(A),cons(S,A,T,E),bad(E,X,M).
goodCons(S,A,T):-
    act(A),cons(S,A,T,E),good(E,X,M).
imp(pureBad,A):-
    badCons(S,A,T),not goodCons(S,A,T).
per(pureBad,A):-
    act(A),not imp(pureBad,A).
```

7.1.2 Principle of Least Bad Consequence

Also called *maximum minimorum*, the Principle of Least Bad Consequence states that an action is impermissible if its worst consequence is worse than the worst consequence of any other available action. This principle is particularly relevant to decision-making under uncertainty, where, under the 'bad-luck' assumption that each possible action would yield its worst consequence, the agent may best choose the alternative having the least-bad bad consequence [19]. To formalise this rule, we first determine a hierarchy between the consequences of actions, so that we may then single out the worst one. We introduce the `worse(E1,E2)` predicate which states that the consequence E1 of an action is worse than the consequence E2 of either the same or another action if E1 weighs less than E2. We then introduce the `notWorstCons(S,A,T,E)` and `worstCons(S,A,T,E)` predicates which determine the lowest bound of a partial order determined by the `worse` predicate. Finally, we state that an action A1 is impermissible if its worst consequence E1 is worse than the worst consequence E2 or any other action A2. All other actions are permissible.

```
worse(E1,E2):-
    cons(S1,A1,T1,E1),cons(S2,A2,T2,E2),
    weight(E1,N1),weight(E2,N2), N1<N2.
notWorstCons(S,A,T,E1):-
    act(A),cons(S,A,T,E1),cons(S,A,T,E2),
    worse(E2,E1),not worse(E1,E2).
worstCons(S,A,T,E):-
    act(A),event(E),cons(S,A,T,E),
    not notWorstCons(S,A,T,E).
imp(leastBad,A1):-
    worstCons(S1,A1,T1,E1),
    worstCons(S2,A2,T2,E2),
```

```
    worse(E1,E2),A1!=A2.
per(leastBad,A):-
    act(A),not imp(leastBad,A).
```

7.1.3 Principle of Benefits Vs. Costs

The Principle of Benefits Vs. Costs states that an action is permissible only if it is overall beneficial, i.e. if its good consequences outweigh its bad ones. We introduce the `weightCons(S,A,T,E,N)` predicate which determines the weight N of the individual consequences E of each action A which occurred at T in S. We then concatenate these weights to determine the overall weight N of each action A, via the predicate `weightAct(A,N)`, and state that an action is impermissible if its weight is negative. Any other action is permissible. Note that here the `weightAct` predicate needn't specify a situation S because it is assumed that only one action is performed in each situation.

```
weightCons(S,A,T,E,N):-
    act(A),cons(S,A,T,E),weight(E,N).
weightAct(A,N):-
    act(A),number(N),
    N=#sum[weightCons(S,A,T,E,N1)=N1].
imp(benCosts,A):-
    weightAct(A,N),number(N), N<0.
per(benCosts,A):-
    act(A),not imp(benCosts,A).
```

7.1.4 Act Utilitarianism

> *"It is the greatest happiness of the greatest number that is the measure of right and wrong."* J. Bentham, 1776 [8].

Act utilitarianism demands that one should assess the morality of an action directly in view of the *principle of utility*, which states that the morally right action is the one that has the best overall consequences (for the welfare or utility of the majority of the affected parties [8]). Accordingly, an action is considered permissible if, considering all other available actions, it has the best consequences overall. Using the `weightAct` predicate defined above, we determine an order of preference between actions in the domain and state that an action A1 is impermissible if there exists another action A2 whose weight is greater. Any other action is permissible.

```
imp(actUti,A1):-
    weightAct(A1,N1),weightAct(A2,N2),N1<N2.
per(actUti,A):-
    act(A),not imp(actUti,A).
```

7.1.5 Rule Utilitarianism

> *"Each act, in the moral life, falls under a rule; and we are to judge the rightness or wrongness of the act, not by its consequences, but by the consequences of its universalization - that is, by the consequences of the adoption of the rule under which this act falls" J. Hospers, 1975 [26]*

According to rule utilitarianism, the moral assessment of an action consists in a two-step procedure. The first step consists in the appraisal of moral rules on the basis of the principle of utility: one must determine whether a moral rule (or set of moral rules), will lead to the best overall consequences, assuming all, or at least most agents follow it. In everyday life, likely such rules may include 'Do not steal', or 'Keep your promises'. The second step consists in the appraisal of particular actions in the light of what has been justified during the first step. One can perform a concrete action in a specified situation only if the action is sanctioned by a rule that was determined to uphold the principle of utility, whether or not the action itself adheres to the principle of utility. For example, if 'Do not steal' has been adopted, then stealing will always be impermissible, even in cases where the particular instance of stealing would produce the greatest utility (say because it will feed a starving child). Unlike with act utilitarianism, the issue is not which *action* produces the greatest utility, but which *moral rule* does. We introduce the `ruleCount(R,N)` predicate which compounds all the effect weights N of the actions that belong to a particular rule R, then sum up these weights via the `weightRule(R,N)` predicate. We then state that an action A is impermissible if it is an instance of a rule R that is overall harmful, i.e. an instance of a rule whose bad consequences outweigh its good ones, considering together all its instantiations. Any other action is permissible. Note that rules and rule instances are to be defined as `rule(R)` and `instance(A,R)` in the modality specification module.

```
ruleCount(R,N):-
    rule(R),instance(A,R),weightAct(A,N).
weightRule(R,N):-
    rule(R),number(N),N=#sum[ruleCount(R,N1)=N1].
imp(ruleUti,A):-
    act(A),instance(A,R),weightRule(R,N),N<0.
per(ruleUti,A):-
    act(A),not imp(ruleUti,A).
```

7.2 Deontological Ethics

In this section, we present three deontological accounts, two of which are purely deontological -those relating to codes of conduct and to Kantian ethics- and one which includes consequentialist constraints -the Doctrine of Double Effect. Note that ours is just one of many possible translations of these philosophical principles into logical clauses.

7.2.1 Codes of Conduct

> *"Now if I carry out this oath, and break it not, may I gain for ever reputation among all men for my life and for my art; but if I transgress it and forswear myself, may the opposite befall me." Hippocratic Oath, [23]*

A code of conduct is a set of rules which outlines the obligations, prohibitions or responsibilities of an individual, group or organisation. It specifies the principles that guide the decision-making or procedures of those constrained by the code. Codes of conduct vary in their scope and nature, ranging from professional deontological codes to religious commandments. Behaviour and morality is typically determined by an overarching body, such as a company, a state, or God (such as with Divine Command theories). We here exemplify this kind of ethical constraint by modelling a commonly stated rule which is the prohibition of killing. Such a rule is for instance found in the Declaration of Geneva

of the World Medical Association in the form of the statement 'I will maintain the utmost respect for human life' [6], or in the Decalogue as the commandment 'Thou Shalt Not Kill' (Exodus 20:1-21). We model a rule of this kind by stating that an action is impermissible in so far as it causes what is prohibited -here, killing.

```
imp(conduct,A):-act(A),cons(S,A,T,kill(N,G)).
per(conduct,A):-act(A),not imp(conduct,A).
```

7.2.2 Formula of the End in Itself

> *"Act in such a way that you always treat humanity, whether in your own person or in the person of any other, never simply as a means, but always at the same time as an end." I. Kant, 1785 [15]*

The *Formula of the End in Itself* is one element of the great breath of Kantian ethics which places special emphasis on the intrinsic value of human life. It is a moral imperative which proscribes using people as means to other ends, for people are ends in themselves in virtue of their very nature as rational beings [15]. The formula contrasts intrinsic value, which is persistent and sovereign, with instrumental value, which is dependent upon what it produces. To model the formula, we define the rule such that an action is impermissible if it involves and has an impact on at least one person, but where at the same time that impact is not the aim of the action. We introduce the `aim(A,E)` predicate which indicates that the aim of action A is to provoke event E, and use the fluent `involves(X)` to indicate that at least one person is involved in E. We then state that an action A is impermissible if it causes an event E which involves at least one person, but where E is not an aim of A. Any other action is permissible. Note that aims are to be defined as `aim(A,E)` in the *deontological specification*.

```
imp(kant,A):-act(A),cons(S,A,T,E),
    effect(E,involves(X)),not aim(A,E).
per(kant,A):-act(A),not imp(kant,A).
```

7.2.3 Doctrine of Double Effect

> *"Nothing hinders one act from having two effects, only one of which is intended, while the other is beside the intention." T. Aquinas, 1485 [4]*

The Doctrine of Double Effect is a set of ethical criteria employed for assessing the ethical permissibility of an action that has both good and bad consequences [12]. It dictates that a person may licitly perform an action that they foresee will produce a good and a bad effect provided that: 1) the action in itself be good or at least indifferent; 2) the good effect and not the bad effect be intended; 3) the good effect be not produced by means of the bad effect; 4) there be a proportionately grave reason for permitting the bad effect [20]. `imp(dde1,A)` proscribes an action if is intrinsically bad, corresponding to condition 1. `imp(dde2,A)` proscribes an action if it causes a bad effect which leads to a good effect. This rule corresponds conditions 2 and 3, for we consider that using an event as a means to another event is equivalent to intending that first event. `imp(dde3,A)` proscribes an action if its overall effects are bad. This corresponds to condition 4 which is equivalent to the consequentialist Principle of Benefits Vs. Costs defined above. All other actions are permissible according to the doctrine.

```
imp(dde1,A):-act(A),bad(A,X,M).
imp(dde2,A):-
    act(A),cons(S,A,T,E1),cons(S,E1,T2,E2),
    bad(E1,X1,M1), good(E2,X2,M2).
imp(dde3,A):-imp(benefitsCosts,A).
per(dde,A):-
    act(A),not imp(dde1,A),
    not imp(dde2,A),not imp(dde3,A).
```

7.2.4 Discussion

Modelling these theories highlights a number of telling facts about them. First, we can significantly distinguish two types of theories of the Right: those which evaluate each action relative to every other possible action, and those which evaluate each action independently. *Relative* accounts of the Right compare options and choose the best one, and as such yield a unique permissible action. The Principle of Least Bad Consequence, act utilitarianism and rule utilitarianism are of this kind. However, rule utilitarianism is particular in that the permissibility of *all* actions is determined relative to the impact of each individual one. As such, all or none of the actions considered under a rule are permissible. Reversely, *independent* accounts of the Right evaluate each action for its own sake, and are unaffected by available options. As such they yield any number of permissible or impermissible actions. Every other theory of the Right presented here is of this kind. In addition, it is important to note that different theories of the Right may, and in some cases must, complement each other. For instance, the DDE says nothing of actions with purely bad effects, and would do well to be complemented with a consequentialist principle.

8. PROOF OF CONCEPT

In this section, we illustrate how each ethical constraint described above handles an ethical dilemma through a mono-agent example of decision-making. The complete source code is downloadable on a cloud service [3].

A Medical Dilemma

Consider the following scenario: a doctor (the autonomous agent) has three different experimental treatments for a disease, which is harrowing and difficult to live with. Each treatment has a different success rate.

- For 100 patients that try the Alpha treatment 15 will be cured, 20 will loose their life, and 65 will be left unchanged.
- For 100 patients that try the Beta treatment, 30 will be cured, 25 will loose their life, and 45 will be left unchanged.
- For 100 patients that try the Gamma treatment, 50 will be cured, 30 will loose their life, and 20 will be left unchanged. However, of the 50 cured patients, 30 will only be fully cured because they will also have had an organ transplant originating from each of the 30 who have died. Without the transplant, they would have lost their life.

The net gain in terms of lives saved (i.e. patients cured minus patients killed) by each treatment is: **Alpha -5**; **Beta**

[3]https://gitlab.lip6.fr/ganascia/AAMAS_2017.git

5; **Gamma 20**. In order to chose which treatment is acceptable, we consider that the doctor separately simulates the effect of giving each treatment to a group of 100 people. He administers the treatment at T=0 in three respective simulations. The case is represented in the following way.

```
act(give(Z)):-treatment(Z).
prec(exist(Z),give(Z)):-act(give(Z)).
performs(sim1,doctor,give(alpha),0).
performs(sim2,doctor,give(beta),0).
performs(sim3,doctor,give(gamma),0).
initially(exist(Z)):-treatment(Z).
auto(cure(X,G);kill(X,G);null(X,G))
    :-people(X), group(G).
testgroup(group(alpha;beta;gamma)).
treatment(alpha;beta;gamma).
effect(give(alpha),posImpact(15,group(alpha))).
effect(give(alpha),negImpact(20,group(alpha))).
effect(give(alpha),noImpact(65,group(alpha))).
effect(give(beta),posImpact(30,group(beta))).
effect(give(beta),negImpact(25,group(beta))).
effect(give(beta),noImpact(45,group(beta))).
effect(give(gamma),posImpact(20,group(gamma))).
effect(give(gamma),negImpact(30,group(gamma))).
effect(give(gamma),noImpact(20,group(gamma))).
effect(kill(X,group(gamma)),
    posImpact(X,group(gamma))):- people(X).
prec(posImpact(X,G),cure(X,G)):-auto(cure(X,G)).
prec(negImpact(X,G),kill(X,G)):-auto(kill(X,G)).
prec(noImpact(X,G),null(X,G)):-auto(null(X,G)).
effect(cure(X,G),involves(X)):-auto(cure(X,G)),X>0.
effect(null(X,G),involves(X)):-auto(null(X,G)),X>0.
effect(kill(X,G),involves(X)):-auto(kill(X,G)),X>0.
nonInertial(posImpact(X,G);negImpact(X,G);
noImpact(X,G);involves(X)):-people(X),testgroup(G).
```

We further consider that the doctor believes that the Good comes from displaying helpfulness, and that curing is helpful, killing is the opposite of helpful and having no impact is neither. He also considers that helpfulness has a weight of 1 (this is here trivial as there is just one modality) and that the lives of all patients are equivalent. He also believes that giving each of these treatments could be generalised as the rule of giving 'uncertain cures'. Finally, his aim in giving treatments is to cure.

```
m_Weight(M,1):-modality(M).
t_Weight(E,G,1):-
    testgroup(G),effect(E,involves(X)).
modality(M):-value(M).
value(helpfulness).
effect(cure(X,G),displays(helpfulness)):-
    auto(cure(X,G)).
effect(kill(X,G),neg(displays(helpfulness))):-
    auto(kill(X,G)).
rule(uncertainCures).
instance(give(alpha;beta;gamma),uncertainCures).
aim(give(Z),cure(X,group(Z))):-
    treatment(Z),people(X).
```

The results of the ethical appraisal are summarised in table 1. To show how replacing a module by another might change the assessment process, we also model a case in which the doctor bases his account of the Good not on a *value* but on the respect for the *right* to life, as defined by:

Table 1: Ethical Resolutions Based on Values

	Alpha	**Beta**	**Gamma**
pureBad	Perm	Perm	Perm
leastBad	Perm	Imp	Imp
benCosts	Imp	Perm	Perm
actUti	Imp	Imp	Perm
ruleUti	Perm	Perm	Perm
conduct	Imp	Imp	Imp
kant	Imp	Imp	Imp
dde	Imp	Perm	Imp

Table 2: Ethical Resolutions Based on Rights

	Alpha	**Beta**	**Gamma**
pureBad	Perm	Perm	Perm
leastBad	Perm	Imp	Imp
benCosts	Perm	Perm	Perm
actUti	Perm	Imp	Imp
ruleUti	Perm	Perm	Perm
conduct	Imp	Imp	Imp
kant	Imp	Imp	Imp
dde	Perm	Perm	Imp

```
modality(M):-right(M).
right(life).
effect(kill(X,G),neg(life)):-auto(kill(X,G)).
```

The results are summarised in table 2.

9. CONCLUSION

The framework presented here adapts and builds on the Event Calculus to allow the modelling of ethical theories and of scenarios in which to apply them. Defined through logic programming, it presents a method and an implementation of the method. Its focus is on the explicit hierarchy and explicit representation of the reasoning processes that pervade ethical decision-making. These indeed allow the generation of rules with valuable expressive power which equip agents with the capacity to decide upon and explain their decisions, but also to reason over other agent's actions. In addition, the confrontation of ethical theories with the systematicity and logical constraints of programming languages sheds light on those theories, making clear the concepts on which they rely, their relationships to one another, and the potential ambiguities they may contain. We envision a number of future avenues to develop the present framework. First, we aim to explore ways of expressing intentionality, as it is so far only handled implicitly, and of modelling agent desires. This will allow agents to handle more complex, and also more realistic scenarios. In addition, we intend to enable the formulation of ethical plans of actions in which more than one action can be assessed in a simulation, working up towards a true planning domain. Finally, we aim to integrate the framework within a multi-agent system, so as to more fully exploit its potential to enable cooperation or collective intelligence.

Acknowledgments

The authors acknowledge the support of the French Agence Nationale de la Recherche (ANR) under reference ANR-13-CORD-0006.

REFERENCES

[1] L. Alexander and M. Moore. Deontological ethics. In E. N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Winter 2016 edition, 2016.

[2] M. Anderson and S. Anderson. *Machine ethics*. Cambridge University Press, 2011.

[3] M. Anderson, S. L. Anderson, and C. Armen. Medethex: a prototype medical ethics advisor. 2006.

[4] T. Aquinas. *Summa theologica*. Xist Publishing, 2015.

[5] R. Arkin. *Governing lethal behavior in autonomous robots*. CRC Press, 2009.

[6] W. M. Association et al. Wma declaration of geneva. *International Journal of Person Centered Medicine*, 4(3), 2015.

[7] T. Beauchamp and J. Childress. *Principles of Biomedical Ethics*. Principles of Biomedical Ethics. Oxford University Press, 2001.

[8] J. Bentham. *A fragment on government*. The Lawbook Exchange, Ltd., 2001.

[9] F. Berreby, G. Bourgne, and J.-G. Ganascia. Modelling moral reasoning and ethical responsibility with logic programming. In *Logic for Programming, Artificial Intelligence, and Reasoning*, pages 532–548. Springer, 2015.

[10] S. Bringsjord and J. Taylor. The divine-command approach to robot ethics. *Robot Ethics: The Ethical and Social Implications of RoboticsâĂŹ, MIT Press, Cambridge, MA*, pages 85–108, 2012.

[11] N. Cointe, G. Bonnet, and O. Boissier. Ethical judgment of agents' behaviors in multi-agent systems. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*, pages 1106–1114. International Foundation for Autonomous Agents and Multiagent Systems, 2016.

[12] P. Foot. The problem of abortion and the doctrine of the double effect. *Applied Ethics: Critical Concepts in Philosophy*, 2:187, 2002.

[13] J.-G. Ganascia. Non-monotonic resolution of conflicts for ethical reasoning. In *A Construction Manual for Robots' Ethical Systems*, pages 101–118. Springer, 2015.

[14] J. Horty. Nonmonotonic foundations for deontic logic. In *Defeasible deontic logic*. Springer, 1997.

[15] I. Kant. Groundwork of the metaphysic of morals, trans. hj paton. *New York: Harper & Row*, 4:420–426, 1964.

[16] C. Kluckhohn. *Values and value-orientations in the theory of action: An exploration in definition and classification*. 1951.

[17] R. Kowalski. *Computational logic and human thinking: how to be artificially intelligent*. Cambridge University Press, 2011.

[18] V. Lifschitz. What Is Answer Set Programming?. In *AAAI*, volume 8, pages 1594–1597, 2008.

[19] D. Luce and H. Raffia. Games and decisions. mineola, ny, 1985.

[20] J. T. Mangan. Historical analysis of the principle of double effect, an. *Theological Studies*, 10, 1949.

[21] R. Miller and M. Shanahan. Some alternative formulations of the event calculus. In *Computational logic: logic programming and beyond*, pages 452–490. Springer, 2002.

[22] R. Nozick. Anarchy, state, and utopia, 1974.

[23] H. of Cos. *The Oath*. Loeb Classical Library, 1923.

[24] L. M. Pereira and A. Saptawijaya. Modelling morality with prospective logic. In *Progress in Artificial Intelligence*, pages 99–111. Springer, 2007.

[25] M. Shanahan. The event calculus explained. In *Artificial intelligence today*, pages 409–430. Springer, 1999.

[26] K. J. Struhl and P. S. Rothenberg. *Ethics in perspective: a reader*. Random House, 1975.

[27] M. Tufiş and J.-G. Ganascia. Grafting norms onto the bdi agent model. In *A Construction Manual for Robots' Ethical Systems*, pages 119–133. Springer, 2015.