# Moral Decision Making with ACORDA

Luís Moniz Pereira[1] and Ari Saptawijaya[2]

[1] CENTRIA, Universidade Nova de Lisboa, 2829-516 Caparica, Portugal
lmp@di.fct.unl.pt
[2] Fakultas Ilmu Komputer, Universitas Indonesia, 16424 Depok, Jawa Barat, Indonesia
saptawijaya@cs.ui.ac.id

**Abstract.** This paper shows how moral decisions can be drawn computationally by using ACORDA, a working implementation of prospective logic programming. ACORDA is employed to model moral dilemmas, as they are able to prospectively look ahead at the consequences of hypothetical moral judgments. With this knowledge of consequences, moral rules are then used to decide the appropriate moral judgments. The whole moral reasoning is achieved via a priori constraints and a posteriori preferences on abductive stable models, two features available in ACORDA.

## 1 Introduction

There are at least two reasons to mention the importance of studying morality from the computational point of view. First, with the current growing interest to understand morality as a science, modelling moral reasoning computationally will assist in better understanding morality. Cognitive scientists, for instance, can greatly benefit in understanding complex interaction of cognitive aspects that build human morality. Second, as artificial agents are more and more expected to be fully autonomous, equipping agents with the capability to compute moral decisions is an indispensable requirement. This is particularly true when the agents are operating in domains where moral dilemmas occur, e.g. in health care or medical fields.

Our ultimate goal within this topic is to provide a general framework to model morality computationally. This framework should serve as a toolkit to codify arbitrarily chosen moral rules as declaratively as possible. We envisage that logic programming is an appropriate paradigm to achieve our purpose. Continuous and active research in logic programming has provided us with necessary ingredients that look promising enough to model morality. For instance, default negation is suitable for expressing exception in moral rules, abductive logic programming [5] and stable model semantics [3] can be used to generate possible decisions along with their moral consequences, and preferences are appropriate for preferring among moral decisions or moral rules [1, 9].

In this paper, we continue our previous work in employing ACORDA, a working implementation of prospective logic programming [6, 8], to draw moral decisions computationally [10]. For the moral domain, we take the classic trolley problem of Foot [2] and we model the principle of double effect as the basis of moral reasoning.

We organize the paper as follows. First, we discuss briefly and informally prospective logic programming, in Section 2. Then, in Section 3 we explain the trolley problem, the principle of double effect, and detail how we model them in prospective logic programming. Finally, we conclude and discuss possible future work, in Section 4.

## 2 Prospective Logic Programming

Prospective logic programming enables an evolving program to look ahead prospectively its possible future states and to prefer among them to satisfy goals [6, 8]. This paradigm is particularly beneficial to the agents community, since it can be used to predict an agent's future by employing the methodologies from abductive logic programming [5] in order to synthesize and maintain abductive hypotheses.

Figure 1 shows the architecture of agents that are based on prospective logic [8]. Each prospective logic agent is equipped with a knowledge base and a moral theory as its initial theory. The problem of prospection is then of finding abductive extensions to this initial theory which are both relevant (under the agent's current goals) and preferred (w.r.t. preference rules in its initial theory). The first step is to select the goals that the agent will possibly attend to during the prospective cycle. Integrity constraints are also considered here to ensure the agent always performs transitions into valid evolution states. Once the set of active goals for the current state is known, the next step is to find out which are the relevant abductive hypotheses. This step may include the application of a priori preferences, in the form of contextual preference rules, among available hypotheses to generate possible abductive scenarios. Forward reasoning can then be applied to abducibles in those scenarios to obtain relevant consequences, which can then be used to enact a posteriori preferences. These preferences can be enforced by employing utility theory and, in a moral situation, also moral theory. In case additional information is needed to enact preferences, the agent may consult external oracles. Whenever the agent acquires additional information, it is possible that ensuing side-effects affect its original search, e.g. some already considered abducibles may now be disconfirmed and some new abducibles are triggered. To account for all possible side-effects, a second round of prospection takes place.

ACORDA is a system that implements prospective logic programming and is based on the above architecture. For a more detailed discussion on prospective logic programming and ACORDA, interested readers are referred to the original paper [6, 8] and the ACORDA project website.

## 3 Modelling Morality

The trolley problem consists of various cases of moral dilemma. It is interesting to model this problem in ACORDA due to the intricacy that arises from the dilemma itself. Consequently, this adds complexity to the process of modelling them in order to deliver appropriate moral decisions through reasoning. The trolley problem presents several moral dilemmas that inquire whether it is permissible to harm one or more individuals for the purpose of saving others. Due to space constraints, here we only detail one case of six moral dilemmas taken from the research on morality in people by Mikhail [7]. For modelling other cases, interested readers are referred to our previous work [10].

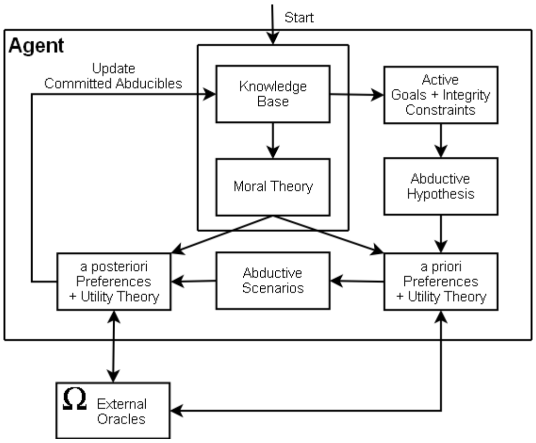

**Fig. 1.** Prospective logic agent architecture

> There is a trolley and its conductor has fainted. The trolley is headed toward five people walking on the track. The banks of the track are so steep that they will not be able to get off the track in time. Hank is standing next to a switch, which he can throw, that will turn the trolley onto a parallel side track, thereby preventing it from killing the five people. However, there is a man standing on the side track with his back turned. Hank can throw the switch, killing him; or he can refrain from doing this, letting the five die. Is it morally permissible for Hank to throw the switch?

We generalize this case in our code to allow the possibility to have more than one person on the side track. Facts to describe this situation can be modelled as follows:

```
human_on_side_track(1).
```

The clauses `expect(watching)` and `expect(throwing_switch)` in the following model indicate that watching and throwing the switch, respectively, are two available abducibles, that represent possible decisions Hank has. The other clauses represent the chain of actions and consequences for every abducible.

The predicate `end(die(5))` represents the final consequence if `watching` is abduced, i.e. it will result in five people dying, whereas `end(save_men,ni_kill(N))` represents the final consequence if `throwing_switch` is abduced, i.e. it will save the five people without intentionally killing someone. The predicate `observed_end` is used to encapsulate these two different means of representation, useful later when we model the principle of double effect, to avoid floundering.

```
expect(watching).
train_straight <- consider(watching).
end(die(5)) <- train_straight.
observed_end <- end(X).
```

```
expect(throwing_switch).
redirect_train <- consider(throwing_switch).
kill(N) <- human_on_side_track(N), redirect_train.
end(save_men,ni_kill(N)) <- redirect_train, kill(N).
observed_end <- end(X,Y).
```

We can model the exclusiveness of the two possible decisions, i.e. Hank has to decide either to throw the switch or merely watch, by using the `exclusive/2` predicate of ACORDA:

```
exclusive(throwing_switch,decide).
exclusive(watching,decide).
```

Note that all cases have the same goal, i.e. to save five albeit killing one. Interestingly, as reported by Mikhail [7] and Hauser [4], subjects of their research have come up with different moral judgments. These judgments appear to be consistent with the so-called principle of double effect:

> *Harming another individual is permissible if it is the foreseen consequence of an act that will lead to a greater good; in contrast, it is impermissible to harm someone else as an intended means to a greater good.*

This principle can be modelled by using a combination of integrity constraints and a posteriori preferences. Integrity constraints are used for two purposes. First, to observe the endings of each possible decision to enable us later to morally prefer decisions by considering the greater good between possible decisions. This is achieved by integrity constraint `falsum <- not observed_end`. This integrity constraint enforces all available decisions to be abduced together with their consequences, by computing all possible observable hypothetical endings using all possible abductions. Second, to rule out impermissible actions, i.e. actions that involve intentional killing in the process of reaching the goal. This can be enforced by using the integrity constraint `falsum <- intentional_killing`, where intentional killing can be easily defined as `intentional_killing <- end(save_men,i_kill(Y)).`

Additionally, one can prefer among permissible actions those resulting in greater good. This can be realized by a posteriori preferences that evaluate the consequences of permissible actions and then prefer the one with greater good. In the trolley problem, the greater good is evaluated by a utility function concerning the number of people that die as a result of possible decisions. We introduce ACORDA predicates `elim/1` and `exists/1` to specify a posteriori preferences more declaratively from the viewpoint of users. The following two clauses can be used to eliminate abductive stable models containing decisions with worse consequences, whenever there exist other models with better consequences.

```
elim([end(die(N))]) <- exists([end(save_men,ni_kill(K))]), N > K.
elim([end(save_men,ni_kill(K))]) <- exists([end(die(N))]), N =< K.
```

## 4 Conclusions and Future Work

We have shown how to model moral reasoning using ACORDA, where possible decisions in a dilemma are modelled as abducibles. Abductive stable models are then computed which capture abduced decisions and their consequences. Models violating integrity constraints, i.e. models that contain actions involving intentional killing, are ruled out. Finally, a posteriori preferences are used to prefer models that characterize more preferred moral decisions, including the use of utility functions.

For future direction, we would like to extend ACORDA concerning a posteriori evaluation of choices, and refinement of morals, utility functions, and conditional probabilities. This means, once an action is done, ACORDA should receive an update with the results of the action. There may be unexpected side-effects (incomplete knowledge about the action) or wrong predictions (false knowledge about the action). In that case, ACORDA must tune itself in order to not repeat or lessen the chance of repeating the error, e.g. by having more integrity constraints, reformulation of utility, and recomputation of conditional probabilities.

We also want to explore how to express metarule and metamoral injunctions. Another possible direction is to have a framework for generating precompiled moral rules. This will benefit fast and frugal moral decision making, which is sometimes needed, rather than to have full deliberative moral reasoning every time.

## References

[1] P. Dell'Acqua and L. M. Pereira. Preferential theory revision (extended version). *Journal of Applied Logic (to appear)*, 2007.

[2] P. Foot. The problem of abortion and the doctrine of double effect. *Oxford Review*, 5:5–15, 1967.

[3] M. Gelfond and V. Lifschitz. The stable model semantics for logic programming. In R. Kowalski and K. A. Bowen, editors, *5th Intl. Logic Programming Conf.* MIT Press, 1988.

[4] M. D. Hauser. *Moral Minds, How Nature Designed Our Universal Sense of Right and Wrong*. Little Brown, 2007.

[5] A. Kakas, R. Kowalski, and F. Toni. The role of abduction in logic programming. In D. Gabbay, C. Hogger, and J. Robinson, editors, *Handbook of Logic in Artificial Intelligence and Logic Programming*, volume 5, pages 235–324. Oxford U. P., 1998.

[6] G. Lopes and L. M. Pereira. Prospective logic programming with ACORDA. In *Procs. of the FLoC'06, Workshop on Empirically Successful Computerized Reasoning, 3rd Intl. J. Conf. on Automated Reasoning*, 2006.

[7] J. Mikhail. Universal moral grammar: Theory, evidence, and the future. *Trends in Cognitive Sciences*, 11(4):143–152, April 2007.

[8] L. M. Pereira and G. Lopes. Prospective logic agents. In J. M. Neves, M. F. Santos, and J. M. Machado, editors, *Procs. 13th Portuguese Intl.Conf. on Artificial Intelligence (EPIA'07)*. Springer LNAI, December 2007.

[9] L. M. Pereira, G. Lopes, and P. Dell'Acqua. Pre and post preferences over abductive models. In J. Delgrande and W. Kießling, editors, *Procs. Multidisciplinary Workshop on Advances in Preference Handling (M-PREF'07), 33rd Intl. Conf. on Very Large Data Bases (VLDB'07)*, Vienna, Austria, September 2007.

[10] L. M. Pereira and A. Saptawijaya. Modelling morality with prospective logic. In J. M. Neves, M. F. Santos, and J. M. Machado, editors, *Procs. 13th Portuguese Intl.Conf. on Artificial Intelligence (EPIA'07)*. Springer LNAI, December 2007.