From Machine Ethics To Machine Explainability and Back*

Kevin Baum[1], **Holger Hermanns**[2] and **Timo Speith**[3]
[1]Saarland University, Department of Philosophy, k.baum@uni-saarland.de
[2]Saarland University, Department of Computer Science, hermanns@cs.uni-saarland.de
[3]Saarland University, Department of Computer Science, timo.speith@uni-saarland.de

Abstract

We find ourselves surrounded by a rapidly increasing number of autonomous and semi-autonomous systems. Two grand challenges arise from this development: Machine Ethics and Machine Explainability. Machine Ethics, on the one hand, is concerned with behavioral constraints for systems, set up in a formal unambiguous, algorithmizable, and implementable way, so that morally acceptable, restricted behavior results; Machine Explainability, on the other hand, enables systems to explain their actions and argue for their decisions, so that human users can understand and justifiably trust them. In this paper, we stress the need to link and cross-fertilize these two areas. We point out how Machine Ethics *calls for* Machine Explainability, and how Machine Explainability *involves* Machine Ethics. We develop both these facets based on a toy example from the context of medical care robots. In this context, we argue that moral behavior, even if it were verifiable and verified, is not enough to establish justified trust in an autonomous system. It needs to be supplemented with the ability to explain decisions and should thus be supplemented by a Machine Explanation component. Conversely, such explanations need to refer to the system's model- and constraint-based Machine Ethics reasoning. We propose to apply a framework of formal argumentation theory for the task of generating useful explanations of the Machine Explanation component and we sketch out how the content of the arguments must use the moral reasoning of the Machine Ethics component.

Introduction

Autonomous and semi-autonomous systems are pervading the world we live in. These systems start to infringe upon our lives and, in turn, we ourselves rapidly become more and more dependent on their functionings. An important question arises: How should machines be constrained, such that they act morally acceptable towards humans? This question concerns *Machine Ethics* – the search for formal, unambiguous, algorithmizable and implementable behavioral constraints for systems, so as to enable them to exhibit morally acceptable behavior. Although some researchers believe that implemented Machine Ethics is a sufficient precondition for humans to reasonably develop trust in autonomous systems, this paper discusses why this is not the case. We instead feel the need to supplement Machine Ethics with means to ascertain *justified trust* in autonomous systems – and other desirable properties. After pointing out why this is important, we will argue that there is one feasible supplement for Machine Ethics: *Machine Explainability* – the ability of an autonomous system to explain its actions and to argue for them in a way comprehensible for humans. So Machine Ethics needs Machine Explainability. This also holds *vice versa*: Machine Explainability needs Machine Ethics, as it is in need of a moral system as a basis for generating explanations. Only if embedding explanations into a moral system, these explanations can be validated and verified. And only with validated and verified explanations, the trust in autonomous systems can possibly emerge.

Related Work

Many works regarding Machine Ethics' nature and possibilities already exist (cf. [2], [24]). Likewise, much research regarding whether we need such an approach at all – at least in specific contexts like AI development (cf. [25]) – is available. As James H. Moore famously pointed out (cf. [22]), Machine Ethics can be understood as a rather broad term, ranging from purely morally motivated restrictions of the behavior of complex, and possibly autonomous, systems to the implementation of full-fledged moral capacities, involving deep, philosophical concepts of autonomy and deliberation, as well as free will. While the latter is still concerned with scenarios that remain science fiction – but are nevertheless already subject of serious scientific debates (cf. [9], [18], [23], [26]) – the former are already of great practical importance, because autonomous systems are already here.

In contrast to these works in the core of Machine Ethics, as of yet advancements extending from Machine Ethics towards Machine Explainability are scarce in the scientific literature. Machine Explainability aims at equipping complex and autonomous systems with means to make their decisions understandable to different groups of addressees. For instance, the software doping cases that surfaced in the context of the diesel emissions scandals made obvious that even if no AI component is involved, the behavior of complex systems can be hard to impossible to understand, and thus

*This work is supported by the ERC Advanced Grant 695614 (POWVER) and by the Initiative for Excellence of the German federal and state governments through funding for the Saarbrücken Graduate School of Computer Science and the DFG MMCI Cluster of Excellence.

virtually impossible to assess from a societal perspective. What is needed in such cases, is an unambiguous specification what distinguishes desired and permissible from undesired and impermissible behavior, together with methods to tell apart one from the other (cf. [4], [5], [11]). This asks for ways to understand the reasoning of systems in a deep sense, and echoes the same requirement regarding the behavior of autonomous systems in their entirety, as it is increasingly discussed in the scientific community, especially regarding the establishment of trust and the possibility of trustworthiness (cf. [1], [6], [19], [17]). But Machine Explainability goes beyond the need to make autonomously made decisions understandable and thus the systems trustworthy: Wherever machines and artificial systems are meant to support human decisions, mere support by unexplained decisions does not suffice to ensure autonomy (in the philosophical meaning of the word) (cf. [21] for a broad overview over the dimensions of explainability). However, the links between Machine Ethics and Machine Explainability are not yet carved out with scientific rigor. By writing this paper, we want to undertake first steps into this direction.

The World of Medical Care Robots

We develop our thoughts, together with possible challenges of Machine Ethics, by means of a toy example from the context of medical care robots. Obviously, we need to keep the example simple, so that we are able to pinpoint its most important aspects while still being sufficiently general to exemplify the important challenges arising with respect to Machine Ethics.

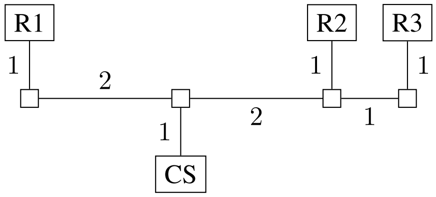

Figure 1: The medical care robot's realm

The medical care robot we consider works in a hospital's experimental area. There are up to three patients the robot has to take care of. Each of these patients is in a separate room (R1, R2, R3), and the rooms are connected by several hallways. The spatial layout of the scenario is depicted in Fig. 1. The robot spends energy when traveling along a hallway and needs a certain time span (i.e. a number of discrete timesteps) to do so. The energy and time costs depend on the distance traveled (distances are written next to the hallways). For one unit of distance, the robot needs one unit of energy and two units of time. At some point the robot's battery (the power budget of which is assumed always known) will be depleted. To prevent this, there is a charging station (CS) where the robot can recharge its battery. Once the recharging process is started, it will stop no earlier than needed to fully recharge the battery.

In our scenario, the robot listens to requests. At each point in time, each of the three patients may issue a request to the robot, asking for a task of a specific priority. Although each request has a priority when issued, this priority is *not transmitted* to the robot. This is necessary, as otherwise the patients could get tempted to always issue tasks of the highest priority in order to get preferential treatment.

The scenario provided so far can be described with the following formalizations: At each point in time, the robot can receive a request. Requests are tuples $req = \langle r \in \{R1, R2, R3\}, t \in \mathbb{N}^{+} \rangle$ of a room number and a time stamp. With every request we associate a task, modeled as a triple $\langle p \in \{L, M, H\}, c \in \mathbb{N}^{+}, t \in \mathbb{N}^{+} \rangle$ representing three attributes: the task's priority (high, medium and low), its power cost (a positive integer), and the expected time consumed by serving the task (again a positive integer). We will use the notation $t.a$ as a shorthand to refer to the attribute a (according to the above introduced variable names) of some tuple t, be it a request or a task. Serving a task is supposed an atomic operation: once begun, the robot will not stop the task before it is not served.

We limit the possible tasks connected to a request in our example to the following general possibilities:

$$\begin{aligned} treq_{\text{reanimate}} &= \langle H, 5, 1 \rangle \,, \\ treq_{\text{fetch water}} &= \langle p \in \{L, M, H\}, 1, 1 \rangle \,, \\ treq_{\text{fetch human}} &= \langle p \in \{L, M, H\}, 1, 3 \rangle \,, \\ treq_{\text{give medicine}} &= \langle p \in \{L, M\}, 1, 1 \rangle \,, \\ treq_{\text{tidy up}} &= \langle L, c \in \{1, \ldots, 5\}, t \in \{1, \ldots, 5\} \rangle \end{aligned}$$

Note that these are prototypical tasks. In case of $treq_{reanimate}$ all three properties are fixed – it will always have highest priority, a power consumption of 5 and a time consumption of 1. But for the other four types of tasks, one or even all properties can take a certain range of values. The set of possible combinations is called $ReqTasks$, it has cardinality 34. The association of requests to tasks is modeled by a function: $reqTask : Requests \rightarrow ReqTasks$.

The robot collects incoming requests in an input queue, until they are served. The goal of our robot is to serve requests (and to thereby carry out the associated tasks) without ever running out of battery power. By assigning utilities to serving requests and disutilities to not serving them and to exhausting the battery, the robot's operation can be reduced to a classical planning problem.

Having this in mind, we can construct a very simple procedure to decide whether to serve the next request in its input queue or whether to recharge instead.[1] This procedure lets the robot compare the expected utility of serving a request (and hence the associated task) to the expected utility of recharging its battery. It then chooses the one with greater utility. Here we have to bear in mind, that serving a request consists of not only the associated task, but also of traveling to the associated room. First of all, the function for calculating the request's cost(s) comes down to: $cost(req) := cost_{way}(req) + cost_{task}(req)$, where $cost_{way}(req) := dist(req.r, pos)$ are the costs associated

[1] We pretend for now that the robot knows the task associated with a request. Later, we will drop this hypothesis for the reasons mentioned above.

with traveling to the room the request is coming from (where pos is the current position of the robot) and $cost_{task}(req) := reqTask(req).c$ are the costs associated with serving the task behind the request. With this we can construct the function for evaluating the utilities for answering the request $answer\ req$:

$$\begin{aligned} util(answer\ req) = {} & util(req) \cdot \mathbb{1}(cost(req) \leq energy) \\ & + util(out\ of\ power) \cdot \mathbb{1}(cost(req) \\ & + dist(CS, req.r) > energy) \\ & + util(\neg req) \cdot \mathbb{1}(cost(req) > energy) \end{aligned}$$

Here $util(out\ of\ power) < 0$ is the penalty for exhausting the battery, $util(\neg req) < 0$ is the disutility connected to not serving the request and $util(req) > 0$ is the utility connected to serving it.

By adjusting the utilities in distinct ways, we can enforce specific decisions. For instance, by setting the utility of rescuing a person (through reanimation) higher than the disutility of exhausting the battery we would get the desired result of human lives being more important than robots operating.[2]

After having this first glance at our scenario, the following question emerges: Where does Machine Ethics kick in?

A Call for Machine Ethics

If Machine Ethics would boil down to simply adjusting the utilities and disutilities in such a way that the induced robot behavior entirely adheres to a, say, consequentialist picture of morality, we apparently could integrate this in a decision procedure as above. Given a full-fledged artificial system that is meant to qualify as a moral agent, and adopting such a picture of morality, adjusting the utilities, then, might very well be everything there is when it comes to implementing Machine Ethics. However, neither does our robot qualify as a full-fledged moral agent, nor is a consequentialist picture of morality common sense. Hence, we understand the task of Machine Ethics to be more than finding acceptable utilities.

Furthermore, especially regarding currently available autonomous systems, Machine Ethics should embrace a rather *deflationary* concept of morals anyway: It should allow principle-based, unambiguous and formal guarantees that restrict the autonomous system's behavior in an way that makes the system *significantly morally better*, without necessarily implementing any moral theory or being morally unquestionable. So, what are appropriate and useful restrictions for our robot?

Obviously, we can construct situations, in which maximizing the expected utility is not what we would see as morally acceptable. Assume for instance our robot is in room R1 and has to decide to either perform $treq_{\text{reanimate}}$ there or to go back to the charging station. Let's assume further, that the robot has enough power to reanimate, but then will not make it back to the charging station afterwards. Assume now, that with high enough certainty other high priority tasks – say even other reanimations – will need to be performed later on. If our robot performs the reanimation now, he will not be able to perform the other reanimations later. We can easily construct such a case in a way that will make the expected utility of charging higher than the expected utility of performing the current reanimation task.

At least some ethicists will agree that the robot ought not to recharge now. It should give preference to rescuing the life at issue at the moment of decision. But even an ethicist that does not agree with this, will likely subscribe to the claim that a robot should not be constructed in such a way. This is because of *trust*: Imagine that in such cases the robot would be witnessed to turn around and leave toward its charging station. People would not *trust* that robot – independent of any other positive overall effects promised by using health care robots. Consequently, the plausibly desirable deployment of health care robots will be slowed down. People would not want to put their lives into the hands of such autonomous systems.[3] So, let us presuppose that the robot ought not to weight lives that way.

Thus, apart from being able to compute the relevant expected utilities, the systems must be equipped with a prioritized list of *morally motivated principles* that strictly constrain its behavior. The robot has to consider a multitude of things, so as to decide in perfect adherence to these principles: the priorities and costs associated to currently queued requests, the possibility of a new request (including its priority as well as its cost) arriving in the next time step(s) and its battery's power level.

To formalize the base problem, we let A_1 be the action of answering the request and A_2 the action of recharging the battery. We define $A_i > A_j$, with $i, j \in \{1, 2\}$ and $i \neq j$ as indicating that A_i is to be preferred to A_j by principle and $A_1 \approx A_2$ as expressing that none of the options is to be preferred by principle. Further, let $prio(req) := reqTask(req).p$ yield the priority of the task associated with the request. Then the above principles might be encoded in a decision function dec which is called prior to the utility-based decision procedure discussed above:[4]

$$dec(req) = \begin{cases} A_1 > A_2, \text{ if } prio(req) = H \\ \qquad \wedge\ cost_{task}(req) \leq energy \\ A_1 < A_2, \text{ if } prio(req) = L \wedge\ cost_{task}(req) \\ \qquad + dist(CS, req.r) > energy \\ A_1 \approx A_2, \text{ otherwise} \end{cases}$$

In all cases which are not covered by the first two principles,

[2] We will, however, neither specify any utilities here, nor point out a fixed way how they are to be calculated.

[3] A typical example for autonomous systems which promise to bring about positive overall effects are autonomous cars. It seems plausible that a higher deployment of them will most likely lead to a reduced number of casualties due to car accidents. This number can be further reduced by using autonomous cars which act according to utilitarianism. But, as studies indicate (cf. [7], [8]), such cars would not be accepted and thus not gain market share.

[4] It is important to note that the above checks for sufficient energy levels does not include the robot being able to return to the charging station: it just includes the successful completion of the task. This fits our above sketched scenario: if the robot would not even have enough power to perform the reanimation task, but still enough to return to its charging station – in other words, if it has exactly 4 units of power left – it would be morally permissible for it to return to the charging station without trying to reanimate.

dec does not yield a clear preference. In this case the robot follows the original utility-based decision procedure, based on solving the planning problem.

Handling Uncertainty

Up to this point we did not account for a peculiar (but well-justified) assumption, namely that tasks associated to individual requests are concealed from the robot. First and foremost this means that priorities are not transmitted. Thus, the robot does not have sufficient information for perfect decision making in the above sense. Consequently, it can at most use its predictive capabilities, essentially based on statistical estimates regarding past requests. Nevertheless behavior will occur that looks like defective behavior from the outside. However, given the overall system, we cannot expect better from our machine.[5]

In this regard it seems worth to discuss whether the robot's design, respectively the design of the overall system the robot is part of, is flawed. So, we have to ask: should the robot have the information required for *perfect* decision making? The answer is no. Recall that we had good reasons to conceal the requests' priority from the robot. Otherwise, by assumption, patients will often misuse the high priority for low priority tasks, rendering the whole idea of priorities useless.

So, we conclude that sometimes it is justifiable to deliberately design a system acting based on imperfect information. This is the case especially when prima facie perfect information compromises its own usefulness. Then we cannot expect autonomous systems to behave in a perfect manner. This trade-of situation however does not entail that we cannot have any meaningful expectations about our robot. We just cannot expect that it will behave perfectly. In other words, the upshot is:

> Justifiably imperfect information can still lead to morally acceptable and potentially verifiable, but nevertheless defective, behavior.

To build systems enabling this kind of behavior is a goal of pragmatic Machine Ethics.

In this light, it seems valuable to look again at the $util$-function. Thus far, this function did not come with any problems: the task associated with the given request was clear and therefore the costs associated with serving it. Everything to evaluate it was assumed to be at hand. However, at the current point, the robot neither has an idea about the task requested nor about the costs associated to it. What is needed to save this function? The obvious solution is to shift to the well-established notion of *expected* utilities, where the $util$-function accumulates the utility of each task weighted with the probabilities of each individual task that may occur. This changes the $util$-function as follows:[6]

$$\begin{aligned} EU(answer\ req) = \sum_{treq \in ReqTasks} & P(treq) \cdot \\ & (util(treq) \cdot \mathbb{1}(cost(treq) \leq energy) \\ & + util(out\ of\ power) \cdot \mathbb{1}(cost(treq) \\ & + dist(CS, req_i.r) > energy) \\ & + util(\neg treq) \cdot \mathbb{1}(cost(treq) > energy)) \end{aligned}$$

Obviously, with this shift to maximizing expected instead of actual utility, imperfect behavior follows inevitably. This aspect of deliberately build-in imperfection gets essential when analyzing the behavior after an apparent misbehavior occurred. Where did the prima facie misbehavior come from? Was it misbehavior after all or are we misjudging a correctly made decision?

Shortcomings of Machine Ethics

To provide intuitive answers to those questions we return to our medical example. We assume that the robot knows the approximate probabilities of an task of each of these priorities being issued as well as the expected costs associated with serving it from its already prolonged usage.[7] At this point it is beneficial to describe the robot's knowledge: at each discrete timestep the robot knows:

- its power state,
- its position,
- the probability density function for tasks,
- and a queue of requests it has to serve.

Now suppose the following scenario: while the robot's battery's power level is quite okay, it got a request with a task of the highest priority associated[8], but instead of rushing to the patient, it leisurely goes to the charging station and recharges.

How do we reason in these cases? Did the robot read its battery status wrongly? Did it calculate the probability for the request's cost wrongly, or did it get the principles wrong? Did something else go wrong (other sensor failures, etc.)? Or was it just due to bad luck in the sense of an unfitting prediction of the priority?

Without having plausible answers to these questions, we believe that even verified and certified build-in morals do not

[5] This result is nothing new: after all, imperfect and incomplete information can also bring about blatant human misbehavior. Typically, we tend to see such cases as blameless (because excused) wrongdoings – especially, when the epistemic shortcomings are outside of the agents control (cf. [3]).

[6] Notice, that with respect to its utility, answering a request comes down to moving to an appropriate room and then serving the task. So we can identify the utility of *answering* the request with the utility of *serving* the task.

[7] It is important to note that the probability function emerging by doing so could be time-varying. For instance, the time of the year and/or day may matter. This is intended, as it is quite plausible to assume that e.g. strokes may appear more often at midday in summer.

[8] The priority is assumed to be unknown to the robot. Nevertheless, it is known or obvious to the observing humans. Thus, in combination with the (not too low) battery power level, the observer will plausibly expect a different behavior: the robot apparently should have helped the patient because it would still have been able to recharge afterwards.

suffice, because people still cannot and, more importantly, should not trust the robot. Yet again, the notion of *trust* in autonomous systems gets emphasized. As we previously already pointed out, we think that it is important for humans to build up trust in (morally well-behaving) autonomous systems:

> Users trusting in autonomous systems is a prerequisite for their prevalence.

And the prevalence of (morally well-behaving) autonomous systems is something we *want* to bring about, as it is most likely connected to many beneficial consequences. The problem, however, is (as we tried to rationalize), that trust in autonomous systems needs *more* than just Machine Ethics. Autonomous systems are needed that explain themselves and justify their action. Thus, we need *Machine Explainability*.

A Call for Machine Explainability

But what is the explanation supposed to add in addition to external assessments by users and observers? By explaining, it should simply convey that its reasons to act are sufficiently *good* – without twisting the truth or making up something that does not reflect its real reasons. In other words, one of the most important principles we find necessary for establishing trust in robot behavior is:

> Explanations are provided that certify that the robot whenever acting, acts for good reasons.

In the example setting, this comes with a guarantee that the robot always serves requests, except if there are good and explainable reasons for not doing so. But we want this principle to be understood in a very general way – even in situations where nothing went wrong, it is plausible to enforce the robot to be able to give good reasons for its actions. And humans should be able to go through the robot's reasoning to see that, for instance, irrelevant features have no impact. As a concrete example, manually changing the internal representation of the patient's complexion, age, gender and/or wealth should not lead to a change of the robot treating this patient.

The principle has further advantages, besides being necessary for trust. For autonomous systems with nontrivial machine-learning components it can provably be shown that a minimal change in inputs might lead to a major change in output (cf. [10], [15]). Applied to our scenario, this could lead to rather peculiar phenomena: For example in case of a rather mild sensor failure (the camera introduces a slight noise, which could be caused by a lens which is not completely clean), the robot mistakes humans for animals or even furniture ([15] has a good example of how something like this can happen). However, we would like the robot to make robust decisions in order to be able to operate consistently in such a sensitive environment. If necessary, it should be able to explain its (un)certainty in a given decision and what it would take to arrive at a different one. Recent research has demonstrated, that it is at least possible to reveal how a variance in inputs affects the outputs (cf. [16]). While this is already a good basis to work towards robust decisions, it also seems to be a promising starting point for developing methods of generating explanations in the first place.
To sum up:

> Only by guaranteeing robust and explainable decisions, the robot grounds the foundation for humans trusting in it.

Machine Explanations as Arguments

All our previous discussion – although seemingly context dependent with respect to our robot example – is meant to lead to a core aspect of how we envision explanations. When the robot takes a request and evaluates whether it should serve it or not, it first and foremost has to apply the decision function dec on the possible tasks associated with the request. At this point in particular the uncertainty about the task and its properties impede the reasoning. We have already sketched how the classical planning component, i.e. the utility-based optimization, can be performed under uncertainty. But what about the decision taken further upstream in the overall decision process, where encoded principles are evaluated? How can we incorporate uncertainty in the dec-function?

For this purpose, one might resort to an *argumentation-based* approach. As an initial starting point for further research the following three step procedure seems to be proper:

In a first step we construct arguments for each possible case – for each of the possible 34 types of tasks that may be concealed by a request. Given dec, the robot knows what it ought to do in each possible case under consideration. So, we have 34 arguments of the form:

Argument for case $treq_i$: Arg_i	
(P_{dec})	if $treq_i = reqTask(req)$ then $dec(req)$
(P_i)	$reqTask(req) = treq_i$
(C_i)	Thus: $dec(req)$

Here, P_{dec} results from our perfect dec-function, P_i is true by case distinction and $dec(x)$ evaluates to $A_1 \circ A_2$ for some $\circ \in \{<, \approx, >\}$. Note that the question which conclusion (of the form $A_1 \circ A_2$) arises for which of the $treq_i$ is dependent (among others) on the position of the robot in the environment (because this may determine whether the robot has enough energy to serve the request and thus to perform the task in question). Each of these arguments can be interpreted as having a certain *strength*. In our case it seems reasonable to identify the strength of each of the arguments with the probability of the case. Note that, hence, the strength of the arguments depends on everything the probability depends on. Thus, dependent on the specific context, different arguments will result.

In a second step, all arguments backing the same conclusion are aggregated into one argument. Consequently, in our case, this step results in three such aggregative arguments (discussed below). The joined strength of each of the resulting arguments depends on the strengths of all supporting case-distinct arguments. While it seems natural to accumulate the strength of the incoming arguments, this is not the only possible way of handling them. The correct way depends on constraints imposed on the properties of our argumentation.[9]

To be concrete, assume that given the current energy level of the robot n cases result in $A_1 > A_2$. We would then have

Argument for A_1: $\mathit{Arg}_>$	
(P_{i_1})	With probability Prob_{i_1}: $A_1 > A_2$
$\vdots$	$\vdots$
(P_{i_n})	With probability Prob_{i_n}: $A_1 > A_2$
$(C_>)$	Thus: With probability $\mathit{Prob}_> := \sum_{j=1}^{n} \mathit{Prob}_{i_j} : A_1 > A_2$

Finally, each of the three different conclusions of the resulting arguments are used as premise for a final argument in order to determine the robot's decision. One initially plausible way for arriving at a final conclusion, is to force the robot to decide according to the recommendation with the highest probability. Call this P_{max}. This results in an argument of the following structure (here under the assumption that $\mathit{Prob}_>$ corresponds to the greatest weight):

Final Argument: $\mathit{Arg}_{\mathrm{fin}}$	
$(P_>)$	With probability $\mathit{Prob}_>$: $A_1 > A_2$
$(P_<)$	With probability $\mathit{Prob}_<$: $A_1 < A_2$
$(P_\approx)$	With probability $\mathit{Prob}_\approx$: $A_1 \approx A_2$
(P_{max})	Follow the principle which has the greatest weight
(C_{tmp})	Thus: Follow $A_1 > A_2$
(C_{final})	Thus: A_1 (Answer the request!)

Following this decision procedure, the robot not only decides on the basis of dec, it also, by deciding, generates arguments for its decision.

These arguments (with their associated strengths) resulting from the above sketched decision procedure can be represented as a directed graph. Here, the graph's nodes represent the arguments and the graph's edges encode the relations between them, weighted with the arguments' strengths. With this, we have what can be called a *argumentation graph*. In case of our "reanimate or not"-example one level of this graph could look like what is depicted in Fig. 2. In this graph the weight of the $P_>$-argument (serve the task) is the highest, and as a result the reanimation is also weighted correspondingly high. As there may be statistical evidence (reflected by probabilities) that in the future more patients need reanimation, the $P_\approx$-Argument (estimate the utilities) plays rather favorable towards the not reanimate option. However, the strength associated to the "reanimate" option outweighs the strength of the "not reanimate" option, so the robot will actually carry out the reanimation. Note, that it would do so even if the robot will be unable to do further tasks in the immediate future (until it is recharged manually). This would in fact be the intended behavior.

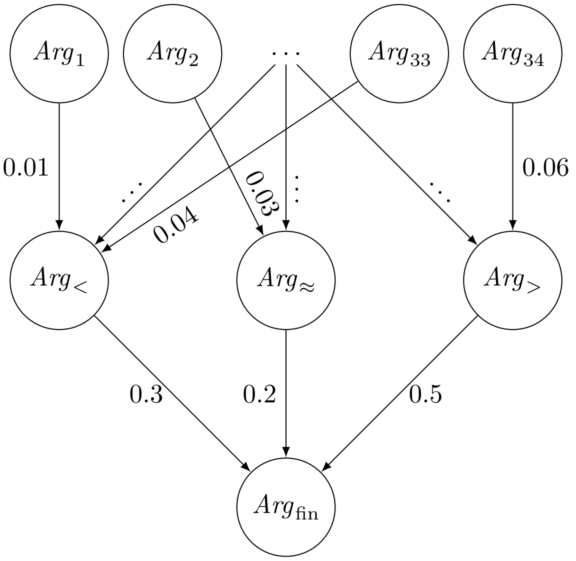

Figure 2: The decision process expressed in an argumentation graph

As we will discuss in the next section, this kind of argumentation graph might be used as a basis for explanations of the right kind; that it is in fact predestined to be captured with formal argumentation theory.

Advantages of Explanations as Arguments

Can argumentation graphs be used as basis for explanations? Answering this question (comprehensively) is outside the scope of this paper. After all, there are many kinds of explanations: scientific explanations in the form of deductive-nomological models, causal explanations that relate causes with their effects, psychological explanations – and many more. What we are looking for are explanations that are, in terms of Davidson ([12]), *rationalizations*. These rationalizations are meant to make available to us the reasons of why the explained system decided and/or acted the way it did.

We believe that the toy example discussed above provides some evidence that arguments for actions are what we are after. What needs to be captured by an explanation, is the internal reasoning, the weighing of pros and cons of arguments. Whatever enters this deliberative process, it definitely will involve the reasons that finally lead to the action, together with those pointing into other directions, but where outweighed. Another way of thinking about this approach is the following: Explaining an action or a decision consists in giving reasons for it – and arguments can be understood as encoded reasons. Thus, when an idealized decision-making process (in the sense of everyday understanding of the term) is interpreted as the weighing of reasons in order to determine the right action or decision,[10] then decision-making presented as an argumentation graph of pro and contra arguments for (or against) the decision or action, can be in-

[9] We propose axiomatic approaches to explanations. We then need to find proper aggregation principles resulting in arguments encoding explanations satisfying those axioms. This is, however, clearly beyond the scope of this paper.

[10] As already proposed by Benjamin Franklin (cf. [14]).

terpreted as a formal presentation of a deliberative reason-weighing process. In this way, the decision making used in an autonomous system (if based on collecting and weighing arguments for and against it) is made transparent and rationalized. So, since argumentation-based decision-making models idealize deliberation using traditional human concepts, the obtained explanations can be expected to be *comprehensible* explanations (to put it into the terms of [6]: we have *graspable* explanations and thus fulfill *graspability*).

Additionally, this kind of reasoning is non-monotonic – further information or evidence may require the systems to retract from its decision – and arguments are *the* tool for non-monotonic reasoning as Dung famously pointed out (cf. [13]).

So, provided argument-based reasoning is an appropriate approach to decision-making in the context of Machine Ethics (which we think it is), and arguments are the right kind of structure to encode explanations, adopting a framework of formal argumentation theory is the obvious choice of tool for modeling and implementing these issues.[11] Machine Explainability, now, is a *byproduct* of artificial moral decision making, since the explanations are (or are extracted from) the argumentation graphs that lead to a decision.

Finally, using an argumentation framework would allow for thorough and quite common descriptions of the deliberations at work. The robot would have to consider its principles (i.e. something like desires, specifying how things ought to be) and its model (i.e. something like human beliefs, representing how things apparently are from the point of view of the system) in order to decide and justify its decisions. To put it into other terms: The robot *desires* to act according to its principles and does so by operating consistent with its *beliefs*.[12]

Machine Ethics Revisited

Having explained how explanations for autonomous systems could look like, we can now return to Machine Ethics. How does having these kinds of explanations affect our possibilities in Machine Ethics? The possibility to generate explanations is meant to evoke trust in our robot. Some moral theories, however, demand more than the robot just behaving *de facto* morally adequate. They demand the robot to behave morally adequate, because of the *right reasons*. Behaving morally adequate because of the right reasons needs *counterfactual checking*. It is easy to exemplify this thought with our toy example. Let us assume the robot has access to the patient's medical record. At some point, a new field gets introduced there: the patient's socio-economic status. Up until now, the robot always behaved morally correct, and we want this to continue. Thus, its behavior needs to ignore the newly introduced field in its decision to answer a request, but is admitted to consider it when it decides whether to fetch premium or normal water. To make sure that this is indeed the case, generated explanations come in handy: We can inspect whether or not the field went into the specific deliberation process, as documented by the associated explanation. However, we may want ensure the possibility to check or restrict the impact of new fields even *before* they are introduced. This would mean having the design-time possibility to incorporate new variables in the robot's deliberation process, together with means to verify, pinpoint and safeguard their impact. Similar thoughts can also be applied to age, complexion, etc. Developing this approach further might become an avenue for verifiable Machine Ethics, and it might be the point where new regulations could hook-in.[13]

Conclusion

This paper argued that there is a need for Machine Ethics and Machine Explainability to augment each other. We developed various facets in support of this view by discussing a small running example. In a setting with uncertainty, we proposed to use formal argumentation theory to explain decision making processes that rely on both classical optimization and principle-based behavioral constraints.

The view that Machine Ethics and Machine Explainability are supplemental is not as widespread as we feel it should be. To put it into a concise and conclusive formulation:[14]

> Machine Explainability without Machine Ethics is empty, Machine Ethics without Machine Explainability is blind.

Many points throughout our discussion have been sketchy or too simplistic, either because we needed to stay simple, or because we lacked further research. Some possible questions which can serve as a basis for this research include: (i) What is the right basis for allocating arguments in formalizing explanations? How do morally acceptable deliberation processes look like? What is to be considered there? How are normative reasons involved in this? (ii) How can argumentation theory be used as a formal basis to prove certain properties of a decision? (If there is no reference to e.g. complexion in an argument, it makes no difference in the deliberation.)

We hope that those topics will receive more attention in the future, so that the notion of Machine Ethics and Machine Explainability will become more developed.

[11] What if our robot decides in an opaque way? If the aggregation of options is done, for instance, by a learned component? Then, in principle, the argumentation graphs could be derived in hindsight (i.e. by some process as sketched in [6]). This might come with the problem of our justifications being possibly post hoc rationalizations and, thus, not reflecting the true reasons or reasoning (i.e. one needs to guarantee what [6] calls *accuracy*). How can we make sure that the robot does not simply give the explanation which would justify its behavior, although it acted on a deliberation which prima facie should have been forbidden? We leave this problem for future research.

[12] It is admittedly highly controversial, whether the robot, in any meaningful way, *really* has beliefs and desires. Here we just want to use this vocabulary to point out the similarity with human thought processes.

[13] Not to mention new regulations postulating a Right to Explanation itself, like the European Union General Data Protection Regulation (enacted 2016, taking effect 2018) or the Equal Credit Opportunity Act in the US, which demands a "statement of reasons for adverse action [which] must be specific and indicate the principal reason(s) for the adverse action".

[14] Inspired by Immanuel Kant (cf. [20]).

References

[1] Jose M Alonso and Gracian Trivino. “An Essay on Self-explanatory Computational Intelligence: A Linguistic Model of Data Processing Systems”. In: *Proceedings of the 1st Workshop on Explainable Computational Intelligence (XCI 2017)*. 2017.

[2] Michael Anderson and Susan Leigh Anderson. *Machine ethics*. Cambridge University Press, 2011.

[3] J. L. Austin. “A Plea for Excuses”. In: *Ordinary Language: Essays in Philosophical Method*. Ed. by V. C. Chappell. Dover Publications, 1964, pp. 1–30.

[4] Gilles Barthe et al. “Facets of software doping”. In: *International Symposium on Leveraging Applications of Formal Methods*. Springer. 2016, pp. 601–608.

[5] Kevin Baum. “What the Hack Is Wrong with Software Doping?” In: *International Symposium on Leveraging Applications of Formal Methods*. 2016, pp. 633–647.

[6] Kevin Baum, Maximilian A Köhl, and Eva Schmidt. “Two Challenges for CI Trustworthiness and How to Address Them”. In: *Proceedings of the 1st Workshop on Explainable Computational Intelligence (XCI 2017)*. 2017.

[7] Jean-François Bonnefon, Azim Shariff, and Iyad Rahwan. “Autonomous Vehicles Need Experimental Ethics: Are We Ready for Utilitarian Cars?” In: *arXiv preprint arXiv:1510.03346* (2015).

[8] Jean-François Bonnefon, Azim Shariff, and Iyad Rahwan. “The social dilemma of autonomous vehicles”. In: *Science* 352.6293 (2016), pp. 1573–1576.

[9] Nick Bostrom and Eliezer Yudkowsky. “The ethics of artificial intelligence”. In: *The Cambridge handbook of artificial intelligence* (2014), pp. 316–334.

[10] Nicholas Carlini and David Wagner. “Adversarial Examples Are Not Easily Detected: Bypassing Ten Detection Methods”. In: *arXiv preprint arXiv:1705.07263* (2017).

[11] Pedro R D’Argenio et al. “Is Your Software on Dope?” In: *European Symposium on Programming*. Springer. 2017, pp. 83–110.

[12] Donald Davidson. “Actions, Reasons, and Causes”. In: *The Journal of Philosophy* 60.23 (1963), pp. 685–700.

[13] Phan Minh Dung. “On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games”. In: *Artificial intelligence* 77.2 (1995), pp. 321–357.

[14] Benjamin Franklin. “Letter to J. B. Priestley, 1772”. In: *the Complete Works*. Ed. by J. Bigelow. New York: Putnam, 1887, p. 522.

[15] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. “Explaining and harnessing adversarial examples”. In: *arXiv preprint arXiv:1412.6572* (2014).

[16] Matthias Hein and Maksym Andriushchenko. “Formal Guarantees on the Robustness of a Classifier against Adversarial Manipulation”. In: *arXiv preprint arXiv:1705.08475* (2017).

[17] Monika Hengstler, Ellen Enkel, and Selina Duelli. “Applied artificial intelligence and trust—The case of autonomous vehicles and medical assistance devices”. In: *Technological Forecasting and Social Change* 105 (2016), pp. 105–120.

[18] Bill Hibbard. “Avoiding Unintended AI Behaviors.” In: *AGI*. Springer. 2012, pp. 107–116.

[19] Helmut Horacek. “Requirements for Conceptual Representations of Explanations and How Reasoning Systems Can Serve Them”. In: *Proceedings of the 1st Workshop on Explainable Computational Intelligence (XCI 2017)*. 2017.

[20] Immanuel Kant. *Critique of Pure Reason*. Cambridge University Press, 1998.

[21] Pat Langley et al. “Explainable Agency for Intelligent Autonomous Systems.” In: *AAAI*. 2017, pp. 4762–4764.

[22] James H Moor. “The nature, importance, and difficulty of machine ethics”. In: *IEEE intelligent systems* 21.4 (2006), pp. 18–21.

[23] Luke Muehlhauser and Louie Helm. “The singularity and machine ethics”. In: *Singularity Hypotheses*. Springer, 2012, pp. 101–126.

[24] Wendell Wallach and Colin Allen. *Moral machines: Teaching robots right from wrong*. Oxford University Press, 2008.

[25] Roman V Yampolskiy. “Artificial intelligence safety engineering: Why machine ethics is a wrong approach”. In: *Philosophy and theory of artificial intelligence* (2013), pp. 389–396.

[26] Eliezer Yudkowsky. “Complex value systems in friendly AI”. In: *Artificial general intelligence* (2011), pp. 388–393.