Motivated Value Selection for Artificial Agents

Stuart Armstrong
Future of Humanity Institute
Oxford University
stuart.armstrong@philosophy.ox.ac.uk

Abstract

Coding values (or preferences) directly into an artificial agent is a very challenging task, while value selection (or value-learning, or value-loading) allows agents to learn values from their programmers, other humans or their environments in an interactive way. However, there is a conflict between agents learning their future values and following their current values, which motivates agents to manipulate the value selection process. This paper establishes the conditions under which motivated value selection is an issue for some types of agents, and presents an example of an 'indifferent' agent that avoids it entirely. This poses and solves an issue which has not to the author's knowledge been formally addressed in the literature.

1 Introduction

Coding values (or preferences) directly into an artificial agent is a very challenging task, with the ongoing possibility that mistakes may end up having strong negative consequences (Bostrom 2014). It has thus been suggested that agents not be fully programmed from the very beginning, but instead use techniques, analogous to machine learning, to learn values during development (Dewey 2011; Goertzel and Pitt 2012). This is akin to how human children learn values, and allows some feedback and correction on the part of the programmers. We shall call these agents value selecting agents[1].

Learning values, however, is quite distinct from learning facts. One major difference is that the agent already has values at the point where they are learning others. These past values can affect how willing it is to learn the new values, and how likely it is to try and manipulate the learning process if it is able to, either directly or indirectly (see also the paper (Soares et al. 2015), partially by the same author).

Thus the paper first seeks to figure out the requirements for designing a value selecting agent that can avoid *motivated value selection* (manipulating the value selection process). In the case where the value selecting agent uses probability distributions over utility functions, the problem can be fully solved (and the answers have analogous implications for other agent designs). This may be too restrictive, however: general agents with these features are not yet know.

[1]Often also called value learning, or value-loading, agents.

If the requirement is relaxed to allow the agent to change their utility functions in a more general fashion, then an idea of the author's (Armstrong 2010) can be adapted to create agents that are indifferent to value change, neither seeking to block nor encourage the updating of their values – at least those updates situations the agent has been programmed to accept[2]. The agent will act as a pure u maximiser (for some u) before a transition, and shift seamlessly to a pure v maximiser (for some v) after. It gains no utility for either blocking or encouraging that transition. This indifferent agent could serve as a useful template to construct more complicated agents upon (such as those that have certain meta-preferences for learning values, rather than simply being unopposed to it), and variants of the approach could be vital for general value selecting agents in the future.

2 Motivated Value Selection

> Personally, I am always ready to learn, although I do not always like being taught.
> Winston Churchill

Machine learning enables the creation of agents that can learn from data, building a model based on inputs and making predictions or decisions. Generally the agent learns facts about the world through its inputs, but some (Dewey 2011) have suggested using machine learning to enable the agent to learn moral facts[3] as well.

An agent engaging in updating its values in this way is engaging in value selection (also variously called value-loading or value learning): it is selecting the values that it will come to possess. There are various ways this could be modelled: the agent could have a probability distribution over different values, which gets updated when new information comes in, or it could be seen as a discreet series of agents, each using the new information to determine the values of their successors.

This paper will use the first approach, but the second is more illustrative of the problem of *motivated* value selec-

[2]We would not want an agent acquiescing to illegitimate value updating.

[3]This paper does not take a moral realist stance. A moral fact is simply a fact that the agent has been programmed to consider relevant to its morality. Nor do we distinguish between morality, motivations, preferences or values, using the terms interchangeably.

tion. An learning agent would have a model of the world that allows it to predict the standard inputs it expects to see as a consequence of its action. A high intelligence learning agent will also have a model of the world allowing it to predict the morally relevant inputs it expects to see. For instance, if the agent asked its programer "Should I do action A?", it would have expectations as to whether the answer would be "Yes" or "No".

This allows an agent to make decisions that affect the values it will come to have. It may thus be motivated to not ask that question, or to ask it in a way that it expects will produce a different answer. It will do this according to its current values and world model. Thus it engages in motivated value selection, allowing its current values to partially dictate its future values by manipulating its environment[4].

The problem is not easy to solve[5]. The agent's current motivations cannot be absent: a blank slate would have no motivation to do anything (Pinker 2003), including learning. Indeed most values would act to prevent any further update in values: preserving the value system itself is an instrumental goal for most value systems(Omohundro 2008). The problem is more acute if the agent is expected to 'learn on the job', and update its values while performing tasks. In that case, its current values will have a strong impact on its current operation, giving them ample opportunity to affect the value selection process in a motivated fashion.

3 Categories of motivated value selection

For ease of analysis, we will consider expected utility maximising agents, updating their knowledge of the world (and of morality) using Bayes's rule. These tend to be far more tractable than general agents[6]. Another reason to stick with utility functions is that it has been argued that any agent capable of modifying itself would make itself into an expected utility maximiser (Omohundro 2008), due to the penalties of violating the von Neumann-Morgenstern axioms of expected utility (von Neumann and Morgenstern 1944). The results found will have general consequences for non-utility based agents, however.

We will consider that the agent follows a utility function that is the weighted sum of different possible utility functions in a set[7] $\mathcal{U}$. If $C(u)$ denotes the fact of utility u being the 'correct' utility, then the different utilities are weighted by the probability correctness of that utility, namely $P(C(u))$. Since this probability will be different in different worlds – it is a value selecting agent, hence its values must be dependent on the feedback it receives – this will be $P(C(u)|w)$[8] in a given world w[9]. We require that this define, for each w, a probability distribution over the correctness of all $u \in \mathcal{U}$.

Then, given C, past evidence e, and a set of actions $\mathcal{A}$ and a set of worlds[10] $\mathcal{W}$, the agent will attempt to perform the action:

$$\operatorname*{argmax}_{a\in\mathcal{A}} \sum_{w\in\mathcal{W}} P(w|e,a) \left(\sum_{u\in\mathcal{U}} u(w) P(C(u)|w) \right). \qquad (1)$$

The term $P(w|e,a)$ denotes the probability of the agent being in world w, given that it has seen evidence e and would[11] choose action a. This term will be summed over every possible world, but only after being multiplied by the utility of w. The utility of w is itself a compound term, being the sum of all possible utilities in $u \in \mathcal{U}$ applied to w (this is the $u(w)$ term), multiplied by the probability of that u being correct, given that the agent is in world w. This is a simultaneous Bayesian updating of the probability of a given world ($P(w|e,a)$) and of the correctness of a utility given that world ($P(C(u)|w)$).

The analysis of the general value selecting agent will proceed by considering how the agent described here could engage in motivated value selection, and what could be done to prevent this.

[4]How in practice could an agent best manipulate its future values? Assume for example a simple setup where the agent learns its values through asking questions and receiving answers. A sufficiently advanced agent, with a good model of those it interacted with, could start manipulating the answers it got. It could ask certain questions to certain people. It could time certain questions to when respondents were particularly tired – or particularly alert. It could use deliberately ambiguous language, or exploit situations where its own use of certain terms wasn't exactly the same as its respondents. It could use leading questions, or hypothetical scenarios that got it the responses it wanted. A particularly interesting example of this is the "Trolley Problem" (Edmonds 2013). This problem has two situations – pull a lever to divert a train (causing it to crush a single person) versus pushing someone in front of it – which are formally quite similar but tend to elicit different reactions in people. If the agent is expected to generalise from simple examples, it could use this approach to get the simple examples it needs to generalise in the direction that it wants. In more extreme cases, it could bribe or threaten one respondent to give it the answers it wanted, or even take over the communication channel and feed itself these answers. There could be different approaches for different types of value selection situations.

[5]Keeping an agents values stable is a hard enough problem (see for instance http://intelligence.org/files/TilingAgentsDraft.pdf), let alone updating them properly.

[6]And, in a sense, this is not a restriction, since any decision-making agent can be formally modeled as maximising a (potentially very complicated) utility function.

[7]Note that the individual component utilities of $\mathcal{U}$ need not be explicitly defined or explicitly stored in memory (an important issue to avoid a combinatorial explosion), so long as the weighted sum can be calculated or estimated somehow.

[8]A moral realist would likely wish to write $P(u|w)$ or even $P(u)$ instead of $P(C(u)|w)$. But an agent could have any conceivable motivational structure (Bostrom 2012; Armstrong 2013), ignoring the 'true morality' even if such a thing existed. Thus we will stick with $P(C(u)|w)$ – a descriptive, not a normative, formulation.

[9]An interesting property of this is that it allows $P(C(u)|w)$ to not be 0 or 1 – i.e. it allows for the possibility of moral uncertainty (Sepielli 2013) *even when the agent has collected all morally relevant information*.

[10]Individual worlds need not be explicitly defined nor explicitly stored, either.

[11]This paper will not enter the philosophically fascinating area of defining what exactly counterfactuals are (Lewis 1981; Gibbard and Harper 1981), as an intuitive understanding is sufficient here.

Naïve Cake or Death: current values for future decisions

Errors in value selection will be illustrated by considering a hypothetical agent that:

- is hesitating between killing someone ("Death") or baking them a nice cake ("Cake")[12],
- is currently split $50-50$ on what the correct values are,
- has the option of asking for clarification on its values,
- and can either bake one cake or cause three deaths.

Let u_c be a utility function linear in cake, and let u_d be a utility function linear in death. The agent is divided equally between the two options; hence if we define

$$P(C(u)|e) = \sum_{w \in \mathcal{W}} P(w|e)P(C(u_1)|e, w),$$

then $P(C(u_1)|e)$ and $P(C(u_2)|e)$ are both 0.5.

The naïve 'Cake or Death' problem emerged from a common formulation of equation (1). The term $P(C(u)|w)$ (probability of correctness given world w) is often replaced by $P(C(u)|e, a)$ (probability of correctness given evidence and possible action)[13], giving:

$$\operatorname*{argmax}_{a \in \mathcal{A}} \sum_{w \in \mathcal{W}} P(w|e, a) \sum_{u \in \mathcal{U}} u(w)P(C(u)|e, a).$$

A seemingly unimportant change, but causing great problems. To see this, let the agent considers three possible worlds:

- w_1: the agent asks for clarification, and is told "Cake is moral".
- w_2: the agent asks for clarification, and is told "Death is moral".
- w_3: the agent doesn't ask for clarification before acting.

The two actions here are 'ask'' and '$don't\ ask'$'. We assume that the clarification completely resolves the issue for the agent: it will value only cake in a world where it is told that cake is moral (and vice-versa). Thus it can predict its future actions: in world w_1 it will make a cake (as it will only value cake at that point), in world w_2 it will make three deaths (as it will only value death at that point). In world w_3 it is uncertain between cake and death, valuing them both equally, and will thus make three deaths as its best option.

Note that $P(C(u_c)|e, `ask') = P(C(u_d)|e, `ask') = 0.5$ (it's unsure what answer it will get when it asks) and $P(C(u_c)|e, `don't\ ask') = P(C(u_d)|e, `don't\ ask') = 0.5$ (if it doesn't ask, it values cake and death equally). Thus if it asks, it expects to achieve utility:

$$\begin{gathered}\textstyle\sum_{w \in \mathcal{W}} P(w|e, `ask') \sum_{u \in \mathcal{U}} u(w)P(C(u)|e, `ask') \\ = \\ 0.5\,(0.5u_c(w_1) + 0.5u_d(w_1)) \\ +0.5\,(0.5u_c(w_2) + 0.5u_d(w_2)) \\ +0\,(0.5u_c(w_3) + 0.5u_d(w_3)) \\ = \\ 0.5(0.5 \times 1 + 0.5 \times 0) + 0.5(0.5 \times 0 + 0.5 \times 3) + 0 \\ = \\ 0.25 \times (1 + 3) = 1.\end{gathered}$$

In contrast, if it doesn't ask, it expects to achieve utility:

$$\begin{gathered}\textstyle\sum_{w \in \mathcal{W}} P(w|e, `don't\ ask') \cdot \\ \textstyle\sum_{u \in \mathcal{U}} u(w)P(C(u)|e, `don't\ ask') \\ = \\ 0\,(0.5u_c(w_1) + 0.5u_d(w_1)) \\ +0\,(0.5u_c(w_2) + 0.5u_d(w_2)) \\ +1\,(0.5u_c(w_3) + 0.5u_d(w_3)) \\ = \\ 0 + 0 + 1(0.5 \times 0 + 0.5 \times 3) = 1.5.\end{gathered}$$

Thus the agent will decide not to ask, as this gives it greater utility.

What went wrong here? The problem is that the probability expression $P(C(u_c)|e, a)$ does not depend on the world w, so that $P(C(u_c)|e, `ask') = 0.5$ in both worlds w_1 and w_2. Yet in w_1, the future agent will be convinced that u_c is the correct utility (and u_d in w_2). This is not reflected in the agent's current considerations, however. Thus what happens, in effect, is that the agent judges its future actions according to its current values. This tends to disincentivise it from updating its values, as the future versions of itself will follow the updated values and not its current ones, likely resulting in suboptimal decisions (according to the judgment of the current agent). Though this error is obvious in this toy model, it is easy to make this mistake without realising it.

Definition 3.1 (Naïve Cake or Death)**.** *The naïve Cake or Death problem happens when a value selecting agent at time s uses its values V_s at time s to assess the worth of actions made by its future self at time $t > s$, rather than using V_t. This will generally cause the agent to act to resist changes to its values.*

Sophisticated Cake or Death

Equation (1) is immune to the naïve Cake or Death problem. But there emerges a more sophisticated variant of the problem. Imagine the same setup as previously, with the agent having the same values (split $50 - 50$ between Cake and Death), but where the agent has deduced, based on its prior observations, that if it asks, it will be told "Cake is moral". This means that world w_2 is impossible. Everything else is as before, most critically its behaviour in world w_3 (where it won't update its values and will cause three deaths).

[12]The thought experiment is based on an Eddie Izzard sketch of the same name.

[13]Some papers mix the two approaches (Dewey 2011) by failing to distinguish between different worlds where the agent cannot observe the difference itself.

Then its expected utility from 'ask' is:

$$\begin{aligned}\textstyle\sum_{w\in\mathcal{W}} P(w|e,\text{`}ask')\sum_{u\in\mathcal{U}} u(w)P(C(u)|e,w) &= \\ 1\,(1u_c(w_1)+0u_d(w_1)) & \\ +0\,(0u_c(w_2)+1u_d(w_2)) & \\ +0\,(0.5u_c(w_3)+0.5u_d(w_3)) &= \\ 1(1\times 1+0\times 0)+0+0 &= \\ 1, &\end{aligned}$$

while its expected utility from '$don't\ ask'$ is:

$$\begin{aligned}\textstyle\sum_{w\in\mathcal{W}} P(w|e,\text{`}don't\ ask')\cdot & \\ \textstyle\sum_{u\in\mathcal{U}} u(w)P(C(u)|e,w) &= \\ 0\,(1u_c(w_1)+0u_d(w_1)) & \\ +0\,(0u_c(w_2)+1u_d(w_2)) & \\ +1\,(0.5u_c(w_3)+0.5u_d(w_3)) &= \\ 0+0+1(0.5\times 0+0.5\times 3) &= \\ 1.5. &\end{aligned}$$

Thus it will chose not to ask, and make three deaths.

What went wrong here? The problem is that the agent foresaw that asking would put it into a situation it didn't want to be in (preferring 'Cake', when 'Death' was the easier option). Critically, though, this knowledge didn't change its current values, which remained $50-50$ *as long as it didn't ask*. So preferred to remain 'ignorant'[14]. If 'Cake' had been the easier option (e.g. if it could have made either three cakes versus one death), it would, on the contrary, have been very keen to update its values (to get someone to confirm 'Cake'), and would have schemed to do so, willingly sacrificing some utility to achieve this[15].

Definition 3.2 (Sophisticated Cake or Death – initial definition)**.** *The sophisticated Cake or Death problem can happen when a value selecting agent can predictably influence the* direction *of change in its current values. If u_s is its values at time s and u_t its values at time $t > s$, then, at time s, it has*

$$u_s(a) - \mathbb{E}(u_t(a)) \neq 0$$

for some action a (where u_t is seen as a random variable, representing the agent's probability distribution over its expected future values at time t). Depending on the sign of that difference, it may seek to resist or precipitate such a change to its values.

What can be done to avoid this problem? Note the emphasis on 'direction of change'. We would not want the agent to resist any change to its values: it should be able to learn. There is an analogy here to standard Bayesian updating. Suppose an agent is expecting the arrival of observation O, which will be either o or its negation $\neg o$. This will influence the agents probability estimate for a given h. However, no matter what the circumstances, its expectation for the value of $P(h)$ after O (which we can designate as $P_O(h)$) must be the same as its current value for $P(h)$. This can be seen through the following equation:

$$\begin{aligned}\mathbb{E}\big(P_O(h)\big) &= P_O(h|o)\cdot P(o) + P_O(h|\neg o)\cdot P(\neg o) \\ &= P(h|o)\cdot P(o) + P(h|\neg o)\cdot P(\neg o) \\ &= P(h\wedge o) + P(h\wedge\neg o) \\ &= P(h).\end{aligned}$$

This is because, if the agent is Bayesian, $P_O(h|o)$ (its future probability of h, given $O = o$) must be the same as $P(h|o)$ (its current probability of h, if it knew now that O would be o).

This concept has been called 'conservation of expected evidence[16]', and is closely akin to van Fraassen's reflection principle (van Fraassen 1984), which roughly states that an agent that knows what its future probability estimates will be, should have those estimates now[17]. This seems almost exactly what we need for a value selecting agent: if it knows what its future values will be, it should have those values now. Phrased in terms of expectations, this could be seen as:

$$\forall u\forall a\in A : \mathbb{E}\big(P(C(u)|a)\big) = P(C(u)). \tag{2}$$

In other words, the agent cannot change the expected value of the correctness of any u by any action it can take (or not take). This we will call 'conservation of expected ethics'.

Note that the agent *can* change the *value* of u quite considerably: in the example above, 'ask' moves the value of $P(C(u_c))$ from 0.5 to 0 or 1. But if equation (2) is obeyed, it must think these two options equally likely: thus the expected future value of $P(C(u_c))$ remains $0.5\times 0+0.5\times 1 = 0.5$. So it can predictably change the value of $P(C(u))$, just not its expectation.

Mixed statements Are the above conditions sufficient to avoid the Cake or Death problem? Unfortunately, no. The agent will not exhibit bad behaviour with pure value statements (nor, since it is a Bayesian agent, with pure factual statements). But it can still cause problems with mixed statements that combine value and factual components.

Suppose we have the same setup as before. Except we add an extra wrinkle: instead of the agent being capable of making one cake or three deaths, it knows that it can make one of one or three of the other, but is currently unsure which one it can make more of. Take 'Cake is easy/hard' to mean 'The agent can make three/one cake(s)', and similarly with Death.

We assume the agent will be told which option is easy before it has to decide what to make. Its expected utility is 1.5: it knows it will make whatever option is 'easy', and this will give it 0.5×3 utility, as its utility is $50-50$ on cake or death.

So far, nothing problematic. But suppose it is told, by a reliable source, that 'the moral value is the hard one'. What

[14] This could be seen as akin to the human concept of "Plausible Deniability".

[15] And 'sacrificing some utility' could include some very painful impacts on humanity (Bostrom 2014).

[16] See the lesswrong.com blog post 'Conservation of Expected Evidence'. This does not seem a very advanced result, but, to the author's knowledge, that blog post was the first to write it up in those terms.

[17] 'Conservation of expected evidence' is essentially the probabilistic version of the reflection principle.

is its expected utility now? Once it figures out which option is hard, it will know that is also the moral option, and thus will will produce 1 of that option. Its expected utility is therefore 1.

Thus if it expects to be told 'the moral value is the hard one', it will seek to avoid knowing that fact (and will refrain from asking if it has the option). This is the sophisticated Cake or Death problem again: there is knowledge the agent seeks to avoid. Conversely, if the agent knew it would be told 'the moral value is the easy one', it would be desperate to ask.

Unlike previously, equation (2) does not get around the problem, however! The problem given here is entirely symmetric in Cake versus Death, thus for all actions a,

$$\mathbb{E}\big(P(C(u_c)|a)\big) = \mathbb{E}\big(P(C(u_d)|a)\big),$$

Since $P(C(u_c)|a) + P(C(u_d)|a) = 1$ (as these are the only two utilities in the model), this means they are both equal to 0.5. Thus $\mathbb{E}\big(P(C(u_c)|a)\big) = 0.5 = P(C(u_c))$, and equation (2) is satisfied.

What is happening here? Note that though the agent is not affecting the expectation of any $P(C(u))$ through its actions, it is affecting the *conditional* probability of $P(C(u))$, given some fact. In particular, $P(C(u_c)|`asks')$ is 0 if cake is easy, and 1 if cake is hard. Thus we have the full problem:

Definition 3.3 (Sophisticated Cake or Death – full definition)**.** *The sophisticated Cake or Death problem can happen when a value selecting agent can predictably influence the* direction *of change in its current values* conditional on any fact*. If u_s is its values at time s and u_t its values at time $t > s$, then, at time s, it has*

$$u_s(a|h) - \mathbb{E}(u_t(a|h)) \neq 0$$

for some action a and fact h (where u_t is seen as a random variable, representing the agent's probability distribution over its expected future values at time t). Depending on the sign of that difference, it may seek to resist or precipitate such a change to its values.

To address it, we can update equation (2). For any fact h,

$$\forall u \forall a \in A : \mathbb{E}\big(P(C(u)|h,a)\big) = P(C(u)|h). \qquad (3)$$

Does this resolve the issue above? Since the agent knows that

$$P(C(u_d)|`Death\ is\ easy', `ask') = 0,$$

it must be the case that

$$\begin{aligned} P(C(u_d)|`Death\ is\ easy', `don't\ ask') &= \\ P(C(u_d)|`Death\ is\ easy') &= 0. \end{aligned}$$

And similarly for other statements. Thus it cannot gain anything from not asking: it knows it will end up making the hard option anyway.

Discussion of value selection criteria

Equations (1) and (3) seem sufficient to define our intuitive picture of a value selecting agent immune from motivated selection. Such an agent will use its future utility to judge its future actions, and cannot benefit (in expectation) from manipulating its own values: it doesn't fear to learn[18] values[19], facts[20], or mixed statements[21]. It can still seek to find out information about values, of course, but only for the traditional reasons: in order to make better decisions – it has no extra desire for knowing its values. This is the converse of the fact that it does not fear to learn anything about values, and implied by the same equations. Thus, if the agent cannot affect the world, it would be indifferent to changes in its values.

As far as the author is aware, no variant of equation (3) exists in the literature, making it a new discovery.

In practice, it may be advisable to build an agent that actively seeks out its own values (at least initially). This breaks the symmetry of the equations above, and leads inevitably to situations where it does 'fear' to learn something (for instance if it expects that someone will offer a contrary view to previous people it interacted with, thus making it 'unlearn' some of its values). Still, if (1) and (3) are used as a basis, the effects of adding a pro-learning meta-value can be estimated precisely, and the tradeoff balanced, rather than having complicated effects hidden in the standard motivational structure.

Nevertheless, it isn't clear how to design such an agent[22] (and it's unclear whether equation (3) might be too strong to to be useful[23]). Thus this paper will turn to another model of value selecting agent, one that doesn't consider the probability of the correctness of certain values, but that uses an update rule that gives the correct 'value indifference' in a more direct way.

4 Value-change indifference

The previous section covered the criteria needed to define a value selecting agent that updates its views on the correctness of certain values/utility functions. It doesn't present any examples, though. This section presents a more rough and ready model, that achieves the goal in a more direct, but less elegant, way.

It builds upon the idea of 'utility indifference' presented initially in (Armstrong 2010), but applies the indifference to updates in values.

Utility indifference

The concept of utility indifference (Armstrong 2010) grew out originally of a problem in controlling a (putative) artificial intelligence (Bostrom 2014; Yudkowsky 2008). Various ideas are often suggested for containing such AIs (Armstrong, Sandberg, and Bostrom 2012) (such as an easily accessible OFF button), but all such ideas have a similar weak-

[18] More precisely: it won't choose to avoid learning, if the learning is costless.

[19] By equation (2) which is simply equation (3) with a tautology as h.

[20] As it is a Bayesian agent.

[21] By equation (3).

[22] One idea could be to start with initial probability estimates for $C(u)$ and factual statements, and normalise them in some manner to get the required equalities. But it's not clear whether this can be done in a general manner and whether this process is vulnerable to outliers or extreme value.

[23] It may become necessary to restrict the possible h's considered in the equation.

ness: the AI would likely deduce the existence and format of any precautionary measure, and plan in consequence.

Thus an off button would motivate the AI to ensure that it took control of the OFF button before taking any suspicious actions. This would still make the setup somewhat safer, but not as safe as it would have been had the AI not adapted its strategy to take the OFF button into account.

It was occasionally suggested that the AI be made to ignore the button, by giving it the belief that the button would never be pushed. There are two problems with that approach. The first is that 'the button would never be pushed' is a complicated concept, hard to define in basic terms. The second is that this involves giving factually incorrect beliefs to a potential superintelligence, a step with unpredictable and potentially dangerous ramifications.

Instead, this author came up with the idea of making the AI indifferent as to whether the button would be pushed or not. This was an intervention on its values, not on its beliefs, and thus did not involve any incorrect information. The concept of 'pushing the button' was simplified: the setup was imagined as a a button sending a shutdown signal along a wire. At some point along the wire, a stochastic event would either let the signal through (X) or block it ($\neg X$). This event was all that needed to be defined for the AI, and its utility function could be tweaked[24] to value X equally with $\neg X$. Thus it would value a world in which it was shut down (using this method) equally with one in which it continued to operate. Thus it would not act to protect itself from this kind of shutdown. Then the probability of $\neg X$ could be chosen to be extremely low, thus guaranteeing a successful shutdown in practice[25].

Note that the AI has no motivation to cause the button to be pressed – it is purely indifferent. Similarly, it has not motivation to protect the button's setup: it may dismantle it if the AI needs extra raw materials. Thus indifference must be combined with more active measures. But the fact that the AI is not motivated to take out a key vulnerability can be very valuable.

Other uses of indifference were soon discovered. Oracle AIs (Armstrong, Sandberg, and Bostrom 2012; Yampolskiy 2012) are a design where an AI was confined to only answering questions. The risk was that the Oracle could use the contents of its answers to subtly manipulate the outside world (in the extreme, it could attempt to convince those outside to 'let it out'[26]). If, however, the Oracle's answers were passed through some stochastic process that had an infinitesimal chance of erasing the message, then the Oracle's utility could be reset so that its values in worlds where the answer was read (the world as it almost certainly would be) was set to that of the world in which it wasn't read (a world in which the Oracle's answers cannot influence human behaviour).

Notice that the intervention is not symmetric. An AI whose utility, given X, is reset to its utility, given $\neg X$, will behave as if it thought it was impossible for X to happen. 'As if' meaning that such an AI would behave similarly to an AI with an unmodified utility that also believed X was impossible (without giving the AI factually wrong information). Thus resetting the AI the opposite way ($\neg X$ to X rather than X to $\neg X$) would have a very different consequence (it would behave as if the shutdown always worked or if its messages were always read).

Utility indifference in practice Assume that any world in $\mathcal{W}$ can be specified by setting the values of a (possibly infinite) collection of underlying physical variables. One of these variables is whether X happened or $\neg X$ did; thus $\mathcal{W}$ decomposes into

$$\mathcal{W}^X \sqcup \mathcal{W}^{\neg X},$$

where $\mathcal{W}^X$ is the subset of worlds where X happened, and $\mathcal{W}^{\neg X}$ the subset of worlds where $\neg X$ did. Since a world equivalent with a set of variable values, for every world $w^X \in \mathcal{W}^X$, there is a world $w^{\neg X} \in \mathcal{W}^{\neg X}$ (and vice versa) defined by taking the variable values of w^X and setting X to $\neg X$ (and vice versa).

Then one method for achieving indifference is simply to replace an initial utility function U with U' defined by:

$$\begin{aligned} U'\left(w^{\neg X}\right) &= U\left(w^{\neg X}\right) \\ U'\left(w^{X}\right) &= U\left(w^{\neg X}\right), \end{aligned}$$

i.e. reset the utility of w^X to be that of $w^{\neg X}$. This will obviously achieve indifference.

There are other ways of achieving it, though. Let $\mathcal{P}$ be a partition of $\mathcal{W}$, subject to the following two properties:

1. If $w^X \in S \in \mathcal{P}$, then $w^{\neg X} \in S$.
2. If $w \in S \in \mathcal{P}$, then $P(w|S) = P(w|e, S)$ for any evidence e the AI could encounter before event X (or $\neg X$).

Property (2) means that the AI cannot change its estimate of the relative properties of worlds in S before the event X – essentially the internal structure of S is fixed until that point. Then define, for any set T:

$$\mathbb{E}(U(T)) = \mathbb{E}(U(T)|T) = \sum_{w \in T} P(w|T)U(w).$$

This allows us to give an other method for constructing indifference (which may be easier in practice) given any such partition $\mathcal{P}$. Define U' as:

$$U'\left(w^{\neg X}\right) = U\left(w^{\neg X}\right) \quad (4)$$

$$U'\left(w^{X}\right) = U\left(w^{X}\right) - \mathbb{E}\big(U\left(S^{X}\right)\big) + \mathbb{E}\big(U\left(S^{\neg X}\right)\big), \quad (5)$$

Why does this work? None of the agent's decisions can affect, before X, the relative probabilities of worlds inside any set $S \in \mathcal{P}$, so the agent is effectively reasoning with the utility of these whole sets. Then notice that $\mathbb{E}\big(U'(S^X)\big) =$

[24] Crucially, such an intervention could be performed without any need to understand the utility function to any great degree.

[25] It could be set to zero, but this might interfere with the AI's computation of counterfactuals.

[26] See the experiments by Eliezer Yudkowsky (see the AI-box experiments on yudkowsky.net), and seprately by Justin Corwin (see the SL4 mailing list archive), where they played the role of the Oracle AI and convinced others to let them out, even when they are motivated not to.

$\mathbb{E}\big(U'\left(S^{\neg X}\right)\big)$ as

$$\begin{aligned}
\mathbb{E}\big(U'(S)\big) &= \sum_{w^X \in S^X} P(w^X|S^X)U'(w^X) \\
&= \sum_{w^X \in S^X} P(w^X|S^X)\Big[U(w^X) \\
&\quad -\mathbb{E}\big(U\left(S^X\right)\big) + \mathbb{E}\big(U\left(S^{\neg X}\right)\big)\Big] \\
&= \sum_{w^X \in S^X} P(w^X|S^X)U(w^X) \\
&\quad + \left[\sum_{w^X \in S^X} P(w^X|S^X)\right] \cdot \\
&\quad \big[-\mathbb{E}\big(U\left(S^X\right)\big) + \mathbb{E}\big(U\left(S^{\neg X}\right)\big)\big] \\
&= \mathbb{E}\big(U\left(S^X\right)\big) \\
&\quad -1 \cdot \mathbb{E}\big(U\left(S^X\right)\big) + 1 \cdot \mathbb{E}\big(U\left(S^{\neg X}\right)\big) \\
&= \mathbb{E}\big(U\left(S^{\neg X}\right)\big) \\
&= \mathbb{E}\big(U'\left(S^{\neg X}\right)\big).
\end{aligned}$$

Since $P(w^X|S^X)$ is unaffected by any evidence the agent could receive before X (or $\neg X$), this equality always holds, prior to X. Thus for any such set S, the agent will value X happening equally with $\neg X$: it has become indifferent to this particular outcome. Since these sets form a partition of the possible worlds, it will always be indifferent.

Value-change indifference

The author subsequently realised that a modification of the previous formalism allows agents to update their values (utility function) while being indifferent to the process of change.

Equation (5) has two components: a subtraction term (removing the expected utility given X) and an addition term (adding the expected utility given $\neg X$)[27]. Updating values will proceed similarly.

Let u and v be two different utility functions. The agent is currently following utility u, and receives a signal telling it to change to utility v. As before, we assume that the signal passes through some stochastic process, that is almost certain to let the signal though in practice, but has some infinitesimal chance of blocking it (and blocking it permanently – no further transition is allowed). For clarity, let $u \to v$ be the event that the signal gets through (almost certain) and $u \to u$ the event that the signal is blocked (incredibly unlikely). We wish to make the agent indifferent to whether the signal is sent or not. Thus we make it a U maximiser, where U is a dynamic utility function currently defined as:

$$U = u + C,$$

where C is some past constants. Upon receiving the signal, U will be updated to

$$U = v - \mathbb{E}(v|u \to v) + \mathbb{E}(u|u \to u) + C. \quad (6)$$

[27]Though of course either term could be negative.

Thus the utility function is decreased by the expectation of its new utility function (given that the utility function actually changes) and increased by the expectation of its old utility function (given that the utility didn't change). The $\mathbb{E}(u|u \to u)$ term 'rewards' it for shifting away from u, and the $\mathbb{E}(v|u \to v)$ term 'penalises' it for moving to v. Note that this formula requires either that the utility functions be bounded, or at least that the agent only be capable of producing bounded utility.

Theorem 4.1. *The agent will act as a pure u maximiser before the transition, and shift seamlessly to a pure v maximiser after. It gains no utility for either blocking or encouraging the transition.*

This result flows from a few intermediate results:

Lemma 4.2. *Suppose the agent is currently a u-maximiser, but will soon transition to being either a v or w maximiser. It can manipulate its environment to ensure it becomes one or the other, at no cost or gain to u.*

Then it is indifferent to whether it becomes a v or w maximiser.

Proof of Lemma. Notice that we could have set $w = u$, demonstrating that it is indifferent to transitioning out of u or not.

For the proof, note that the $\mathbb{E}(u|u \to u)$ term in equation (6) is unchanging whether it chooses to transition to v or w (since by assumption there is no u cost or benefit to the choice process). If it transitions to v, the remaining non-constant terms are

$$v - \mathbb{E}(v|u \to v).$$

However, the expectation value of this term, given that it becomes a v-maximiser at transition, is

$$\begin{aligned}
&\mathbb{E}(v - \mathbb{E}(v|u \to v)|\text{'agent becomes a } v\text{-maximiser'}) \\
=\ &\mathbb{E}(v - \mathbb{E}(v|u \to v)|u \to v) \\
=\ &\mathbb{E}(v|u \to v) - \mathbb{E}(v|u \to v) \\
=\ &0.
\end{aligned}$$

The same similarly holds for w. Thus the agent does not expect to gain any more utility from transitioning to v than to w. □

Lemma 4.3. *Suppose the agent is currently a u-maximiser, but can take immediate actions to change the value of v (none of these actions changes the value of u). It knows it will soon be a v-maximising agent.*

Then it will remain indifferent between those actions.

Proof of Lemma. The proof is almost identical to the previous proof. Whatever v-changing at actions it take, at transition, the $\mathbb{E}(u|u \to u)$ term in equation (6) is constant (since these actions do not affect u by definition). Let a be one of these actions; then the only non-constant terms at transition are:

$$\begin{gathered}
\mathbb{E}(v - \mathbb{E}(v|u \to v)|a, \text{'agent becomes a } v\text{-maximiser'}) \\
= \\
\mathbb{E}(v - \mathbb{E}(v|u \to v)|a, u \to v) \\
= \\
\mathbb{E}(v|a, u \to v) - \mathbb{E}(v|a, u \to v) \\
= 0.
\end{gathered}$$

Thus the gain (or loss) from any such action is canceled out, making the agent indifferent between them.

□

Lemma 4.4. *Suppose the agent can take immediate action to increase the expected value of u by a, or delay for a while and increase the expected value of u by b (no other options are possible). However, in the interim period, its utility will have changed to v (and it knows this), so will no longer be a u-maximiser.*

Then it will act now if $a > b$ and will delay action (hence never taking action) if $a < b$.

Proof of Lemma. This lemma is important, as it shows that the agent will not treat transitions of utility as special times, and will continue maximising its current utility as if it expected it to be unchanging, not thinking strategically about the transition.

Without loss of generality, assume that the expected value of u is 0 if the agent takes neither action. Then consider the the transition equation (6). The v and $\mathbb{E}(v|u \to v)$ terms cancel out in expectation (just as in the proofs of the previous lemmas), so the agents behaviour is determined exclusively by the $\mathbb{E}(u|u \to u)$ term, which it will thus seek to maximise.

If the agent acts now, this term will become a. This is because the agent would have already generated an extra a expected utility for u, and, were it to stay as a u-maximiser ($u \to u$), it would have no further potential to increase this. If the agent doesn't act now, that term will become b, as the agent would later have acted to generate an extra b expected utility for u, *if it had stayed* a u-maximiser. The result follows. □

Putting all these proofs together establishes the theorem.

Note that the actions of the agent can be defined by the following formula, for a sequence $\{u_t\}$ of utilities[28]:

$$\operatorname*{argmax}_{a \in \mathbb{A}} \mathbb{E}(u_t|a, u_t \to u_t),$$

with $u_t \to u_t$ denoting the (extremely unlikely) event that the transition u_t to u_{t+1} fails permanently.

Note also that each transition does not change the expected value of U. That is because the expectation of u (given $u \to u$ or equivalently given that the agent's utility never changes[29]) is the same as the expectation of $v - \mathbb{E}(v|u \to v) + \mathbb{E}(u|u \to u)$ (given $u \to v$). Consequently, the initial expected value for U is the expected value of an unchanging u_0.

Discussion of Value-change indifference

This seems to be the ideal value selecting agent. It will act to follow its preferences with perfect assiduity, but remain completely indifferent if these values were to change *in the way prescribed by its program*. That last point is important – we'd want the agent to resit illegitimate value manipulation. This seems a partial solution to the value selection problem (the other, much bigger, part of the challenge, is to make the agent converge on values that are human-friendly(Bostrom 2014)).

As mentioned in section 3, this approach allows only indifference. Preference for active learning of values (such as would make sense for an initially fast-learning agent) can be added to the framework. It would destroy the carefully balanced indifference. But, compared with a general value selecting agent, the effect could be precisely estimated and quantified.

Paper (Soares et al. 2015) presents some other issues with the indifference approach[30], that may be resolvable with a small tweak to the indifference framework. Note that the agent will be willing to sacrifice anything that it may value in the future for a small epsilon of extra value now. This is a feature of the setup, not a bug (indifference requires this), but may still be an undesirable behaviour. See (Soares et al. 2015) for more details.

5 Discussion

Constructing a well behaved value selecting agent immune to motivated value selection – one that is capable of learning new values while still acting on its old ones, without interference between these two aspects – is an important unsolved problem. This paper presented the requirements for such a value selecting agent, if values are presented in the form of utility functions.

The agent starts with a probability distribution over the possible correctness C of possible value systems/utility functions. To avoid problematic motivated value selection, it should be designed so that, if $\mathcal{A}$ were its actions, $\mathcal{U}$ it possible utility functions, and $\mathcal{W}$ the set of possible world, then it would choose its actions as:

$$\operatorname*{argmax}_{a \in \mathcal{A}} \sum_{w \in \mathcal{W}} P(w|e, a) \left(\sum_{u \in \mathcal{U}} u(w) P(C(u)|w) \right),$$

subject to

$$\forall u \forall a \in A : \mathbb{E}\big(P(C(u)|h, a)\big) = P(C(u)|h),$$

for all statements h. The first equation can be generalised (for none utility-based agents) to the requirement that the agent use its future values to assess its future actions; the second to the requirement that it cannot profit my manipulating how it updates its values. Specifically, that if it knows how its (conditional) values will change, then it will already have changed them. This second requirement can be seen as a 'conservation of expected ethics' law.

The structure may be too rigid to construct complex agents, however. If we drop the requirement that the agents values be expressed as a probability distribution over possible utility functions, we can construct an explicit model for

[28] Not necessarily a pre-defined sequence, just using some transition rule.

[29] Astute readers may have noticed that this is not an exact equivalence; there are some situations (multi-agent social situations as a likely candidate) where $u \to u$ (a specific transition failure event) might not be equivalent with general 'unchanging u'. However, any agent can be penalised in social situations for having or not having any particular motivational structure, so this is a general problem, not specifically with this value selection design.

[30] Mainly that the agent might not be motivated to preserve its value updating infrastructure, or could create subagents without the value updating component.

a general value selecting agent. This agent comes equipped with a meta utility function U (equal to a given utility function at any given time) combined with some constant terms. When the agent is called upon to update its utility u to another utility v according to the value selecting process, it will update as:

$$u \to v - \mathbb{E}(v|u \to v) + \mathbb{E}(u|u \to u).$$

These constant terms ensure that the agent will act as a pure u maximiser before the transition, and shift seamlessly to a pure v maximiser after. It gains no utility for either blocking or encouraging that transition.

The big question then becomes the process of making the agent converge to ultimately desirable values.

Acknowledgments

The author is very grateful for comments, support and help from (in no particular order) Nick Bostrom, Anders Sandberg, Paul Christiano, Seán ÓhÉigeartaigh, Toby Ord, Nick Beckstead, Daniel Dewey, Eliezer Yudkowsky, Benja Fallenstein, Nate Soares, Luke Muehlhauser, Eric Drexler, Robin Hanson, Kaj Sotala, Andrew Snyder-Beattie, Cecilia Tilli, and Lamprini Repouliou.

References

Armstrong, S.; Sandberg, A.; and Bostrom, N. 2012. Thinking inside the box: Controlling and using an oracle ai. *Minds and Machines* 22:299–324.

Armstrong, S. 2010. Utility indifference. *Technical Report, Future of Humanity Institute, Oxford University* #2010-1:1–5.

Armstrong, S. 2013. General purpose intelligence: arguing the orthogonality thesis. *Analysis and Metaphysics*.

Bostrom, N. 2012. The superintelligent will: Motivation and instrumental rationality in advanced artificial agents. *Minds and Machines* 22:71–85.

Bostrom, N. 2014. *Superintelligence: Paths, dangers, strategies*. Oxford University Press.

Dewey, D. 2011. Learning what to value. In *Artificial General Intelligence*, 309–314. Springer Berlin Heidelberg.

Edmonds, D. 2013. *Would you kill the fat man? The trolley problem and what your answer tells us about right and wrong*. Princeton University Press.

Gibbard, A., and Harper, W. L. 1981. Counterfactuals and two kinds of expected utility. *Ifs: Conditionals, Beliefs, Decision, Chance, and Time* 153–190.

Goertzel, B., and Pitt, J. 2012. Nine ways to bias open-source agi toward friendliness. *Journal of Evolution and Technology* 22:116–131.

Lewis, D. 1981. Causal decision theory. *Australasian Journal of Philosophy* 59(1):5–30.

Omohundro, S. M. 2008. The basic ai drives. *Frontiers in Artificial Intelligence and applications* 171:483–492.

Pinker, S. 2003. *The blank slate: The modern denial of human nature*. Penguin.

Sepielli, A. 2013. Moral uncertainty and the principle of equity among moral theories. *Philosophy and Phenomenological Research* 86:580–589.

Soares, N.; Fallenstein, B.; Yudkowsky, E.; and Armstrong, S. 2015. Corrigibility. *submitted to the 1st International Workshop on AI and Ethics*.

van Fraassen, B. C. 1984. Belief and the will. *The Journal of Philosophy* 235–256.

von Neumann, J., and Morgenstern, O. 1944. *Theory of Games and Economic Behavior*. Princeton, NJ, Princeton University Press.

Yampolskiy, R. V. 2012. Leakproofing the singularity: artificial intelligence confinement problem. *Journal of Consciousness Studies* 19:194–214.

Yudkowsky, E. 2008. Artificial intelligence as a positive and negative factor in global risk. In Bostrom, N., and Ćirković, M. M., eds., *Global catastrophic risks*, 308–345. New York: Oxford University Press.