Toward the Engineering of Virtuous Machines

Naveen Govindarajulu
Selmer Bringsjord and **Rikhiya Ghosh**
Rensselaer AI & Reasoning Lab
Rensselaer Polytechnic Institue, (RPI)
Troy, New York 12180, USA
www.rpi.edu

Vasanth Sarathy
Human Robot Interaction Laboratory
Tufts University
Medford, MA 02155, USA
www.tufts.edu

Abstract

While various traditions under the 'virtue ethics' umbrella have been studied extensively and advocated by ethicists, it has not been clear that there exists a version of virtue ethics rigorous enough to be a target for machine ethics (which we take to include the engineering of an ethical sensibility in a machine or robot itself, not only the study of ethics in the humans who might create artificial agents). We begin to address this by presenting an embryonic formalization of a key part of any virtue-ethics theory: namely, the learning of virtue by a focus on exemplars of moral virtue. Our work is based in part on a computational formal logic previously used to formally model other ethical theories and principles therein, and to implement these models in artificial agents.

Introduction

What is virtue ethics? One way of summarizing virtue ethics is to contrast it with the two main families of ethical theories: **deontological ethics** ($\mathcal{D}$) and **consequentialism** ($\mathcal{C}$). Ethical theories in the family $\mathcal{C}$ that are utilitarian in nature hold that actions are morally evaluated based on their **total utility** (or total *dis*utility) to everyone involved. The best action is the action that has the highest total utility. In stark contrast, ethical theories in $\mathcal{D}$ emphasize **inviolable principles**, and reasoning from those principles to whether actions are obligatory, permissible, neutral, etc.[1] In a departure from both $\mathcal{D}$ and $\mathcal{C}$, ethical theories in the virtue-ethics family $\mathcal{V}$ are overall distinguished by the principle that the best action in a situation, morally speaking, is the one that a **virtuous person** would perform (Annas 2011). A virtuous person is defined as a person who has learned and internalized a set of habits or traits termed **virtuous**. For a virtuous person, virtuous acts become second-nature, and hence are performed in many different situations, through time.

While there has been extensive formal and rigorous modeling done in $\mathcal{D}$ and $\mathcal{C}$, there has been little such work devoted to formalizing and mechanizing $\mathcal{V}$. Note that unlike $\mathcal{D}$ and $\mathcal{C}$, it is not entirely straightforward how one could translate the concepts and principles in $\mathcal{V}$ into a form that is precise enough to be realized in machines. Proponents of $\mathcal{V}$ might claim that it is not feasible to do so given $\mathcal{V}$'s emphasis on persons and traits, rather than individual actions or consequences. From the perspective of machine ethics, this is not satisfactory. If $\mathcal{V}$ is to be on equal footing with $\mathcal{D}$ and $\mathcal{C}$ for the purpose of building morally competent machines, AI researchers need to start formalizing parts of virtue ethics, and to then implement such formalization in computation.

We present one such formalization herein; one that uses learning and is based on a virtue-ethics theory presented by Zagzebski (Zagzebski 2010). The formalization is presented courtesy of an expressive computational logic that has been used to model principles in both $\mathcal{C}$ and $\mathcal{D}$ [e.g. (Govindarajulu and Bringsjord 2017a; Govindarajulu et al. 2017)].[2] The formalization answers, abstractly, the following two questions:

Questions

($\mathbf{Q_1}$) When can we say an agent is virtuous?

($\mathbf{Q_2}$) What is a virtue?

The plan for the paper is as follows. First, we briefly look at why virtuous machines might be useful, and then we briefly cover related work that can be considered as formalizations of virtue ethics. Next, we present an overview of virtue ethics itself, and specifically show that an emphasis on moral exemplars makes good sense for any attempt to engineer a virtuous machine. We next present one version of virtue ethics, $\mathcal{V}_z$ (Zagzebski's version of virtue ethics), that we seek to formalize fully. Then, our calculus and the formalization itself ($\mathcal{V}_z^f$) are presented. We conclude by discussing future work and remaining challenges.

Why Virtuous Robots?

Note that we do not advocate that machine ethicists pursue virtue ethics over other familiies of ethical theories. Our goal in the present paper is merely to formalize one version of virtue ethics within the family $\mathcal{V}$. That said, why might virtue ethics be considered over consequentialism or deontological ethics for building morally competent machines? To partially answer this question, we take a short digression into a a series of conditions laid out by Alfano, and characterized as identifying the core of virtue ethics:

[1] Both the families $\mathcal{C}$ and $\mathcal{D}$ are crisply explained as being in conformity with what we say here in e.g. (Feldman 1978).

[2] See (Bringsjord, Arkoudas, and Bello 2006) for an introduction to the logicist methodology for building ethical machines.

Hard Core of Virtue Ethics (partially quoting (Alfano 2013))

(2) **stability** If someone possesses a virtue at time t_1, then *ceteris paribus* she will possess that virtue at a later time t_2.

(3) **consistency** If someone possesses a virtue sensitive to reason r, then *ceteris paribus* she will respond to r in most contexts.

(7) **explanatory power** If someone possesses a virtue, then reference to that virtue will sometimes help to explain her behavior.

(8) **predictive power** If someone possesses a high-fidelity virtue, then reference to that virtue will enable nearly certain predictions of her behavior; if someone possesses a low-fidelity virtue, then reference to that virtue will enable weak predictions of her behavior.

Particularly, we feel that if the conditions of **stability**, **consistency**, **explanatory power**, and **predictive power** hold, then virtuous agents or robots might be easier for humans to understand and interact with (compared to consequentialist or deontological agents or robots). This is but our initial motivation; we now present an overview of virtue ethics, in order to show that our focus specifically on learning of virtuous behavior from moral exemplars is advisable.

Surveying Virtue Ethics

See (Scheutz and Malle forthcoming) for a general introduction to the field of moral robots. We begin our survey by reporting that Hurka (Hurka 2000) presents an ingenious formal account involving a recursive notion of goodness and badness. The account starts with a given set of primitive good and bad states-of-affairs. Virtues are then defined as love of good states-of-affairs or hatred of bad states-of-affairs. Vice is defined as love of bad states-of-affairs or hatred of good states-of-affairs. Virtues and vices are then themselves taken to be good and bad states-of-affairs, resulting in a recursive definition (see Figure 2) that is attractive to AI researchers and computer scientists. But despite this, and despite our sense that the main problems with Hurka's account are rectifiable (Hiller 2011), we feel that Hurka's definition might not capture central aspects of virtue (Miles 2013). More problematic is that it must be shown that Hurka's account is different from rigorous and formal accounts of $\mathcal{C}$, which after all are themselves invariably based upon good and bad states-of-affairs. Moreover, it is not clear to us how Hurka's account is amenable to automation. Therefore, we now proceed to step back and survey the overarching family $\mathcal{V}$ of virtue ethics, to specifically pave a more promising AI road: a focus on moral exemplars.

Virtue Ethics: Overview to Exemplarism

The core concepts of consequentialist ethical theories (i.e. of members of $\mathcal{C}$), at least certainly in the particular such theory known as *utilitarianism*, are doubtless at minimum relatively familiar to most of our readers. For instance, most in our audience will know that utilitarianism's core tenet is that actions are obligatory just in case they have the consequence

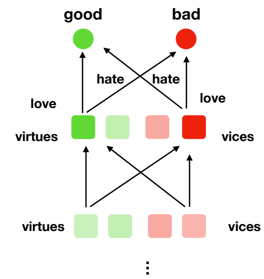

Figure 1: **Hurka's Account** Virtues (vices) are defined recursively as love of good (bad) states-of-affairs or hate (love) of bad states of affairs.

of maximizing happiness, and are forbidden exactly when they fail to so maximize. A parallel state-of-affairs holds for at least basic knowledge of deontological ethical theories (= family $\mathcal{D}$): most readers have for instance some familiarity with Kant's moral system in family $\mathcal{D}$, and specifically with his famous "categorical imperative," which, paraphrasing, says that, unconditionally, one must always act in such a way that this behavior could be universalized.[3] In addition, generally people are familiar with the core tenet of divine-command ethical theories (i.e. of members of $\mathcal{DC}$), which is (approximately) that actions are obligatory for humans if and only if God commands that these actions be performed (a particular member of $\mathcal{D}$is specified in (Quinn 1978)). However, in our experience the epistemic situation is radically different when it comes to the family of ethical theories *virtue ethics* (= $\mathcal{V}$). For while it's true that generally educated people can be assumed to be acquainted with the concept of virtue, and with many things long deemed to be virtues (e.g. bravery), an understanding of virtue ethics at the level of ethical *theory* cannot be assumed. We therefore now provide a rapid (and admittedly cursory) synopsis of $\mathcal{V}$, by drawing from (Vallor 2016), and to some degree from (Annas 2011). It will be seen that $\mathcal{V}$ makes central use of exemplars, and of learning and development that revolves around them. Hence we shall arrive at a convenient entry point for our AI work devoted to trying to design and build a virtuous machine.

Obviously we cannot in the span of the space we have at hand do full justice to the book-length treatment of $\mathcal{V}$ that is (Vallor 2016). But we can quickly establish that our technical work, in its focus on the cultivation of virtue for a machine via learning from exemplars, is not merely based on a single, idiosyncratic member of $\mathcal{V}$, and on one peripheral aspect of this member. On the contrary, study of the work of Vallor and other scholars concerned with a characterization of the family $\mathcal{V}$ confirms that our exploitation specifically

[3]This imperative is first set out in — as it's known in abbreviation — *Groundwork*; see (Kant 19971785). It's generally thought by ethicists, and this may be convenient for machine/AI ethics, that Kant had in mind essentially a decision procedure to follow in the attempt to behave in an ethically correct manner. For a lucid and laconic overview of this point, see (Johnson 20042016); and cf. (Powers 2006).

of Zagzebski's (Zagzebski 2010) focus, from the standpoint of the field of ethics itself, is a worthy point of entry for AI researchers.

To begin, Vallor, drawing on and slightly adapting Van Norden's (Van Norden 2007) sets out a quartet of commonalities that at least seem to be true of all members of $\mathcal{V}$, and the second one is: "A conception of moral virtues as cultivated states of character, manifested by those exemplary persons who have come closest to achieving the highest human good" (¶5, §2.2).[4] But given our specific efforts toward engineering a virtuous machine, it is important to note that Vallor specifically informs us about the key concepts of exemplars in the particular members of the $\mathcal{V}$family; to pick just one of many available places, she writes:

> Buddhism's resonances with other classical virtue traditions do not end here. As with the central role granted by Confucian and Aristotelian ethics to 'exemplary persons' (the *junzi* and *phronimoi* respectively), *bodhisattvas* (persons actively seeking enlightenment) generally receive direction to or assistance on the path of self-cultivation from the community of exemplary persons to which they have access. In Buddhism this is the monastic community and lay members of the *Sangha* . . . [¶5, §2.1.3, (Vallor 2016)]

We said above that we would also draw, albeit briefly, from a second treatment of $\mathcal{V}$, viz. (Annas 2011), in order to pave the way into our AI-specific, exemplar-based technical work. About this second treatment we report only that it is one based squarely on a "range of development" (¶3, §Right Action in Ch. 3), where the agent (a human in her case) gradually develops into a truly virtuous person, beginning with unreflective adoption of direct instruction, through a final phase in which "actions are based on understanding gained through experience and reflection" (ibid.). Moreover, Annas explicitly welcomes the analogy between an agent's becoming virtuous, and an agent's becoming, say, an excellent tennis-player or pianist. The idea behind the similarity is that "two things are united: the *need to learn* and the *drive to aspire* (emphasis hers; ¶4 Ch. 3). In addition, following Aristotle on $\mathcal{V}$ (e.g. see (Aristotle 2000) 1103), no one can become a master tennis-player or pianist without, specifically, playing tennis/the piano with an eye to the mastery of great exemplars in these two domains.

In order to now turn to specific AI work devoted to engineering a virtuous machine, we move from completed consideration of the general foundation of $\mathcal{V}$, and its now-confirmed essential use of moral exemplars, to a specific use of such exemplars that appears ripe for mechanization.

Exemplarist Virtue Theory

Exemplarist virtue theory ($\mathcal{V}_z$) builds on the **direct reference theory** (DRT) of semantics. Briefly, in DRT, given a word or term w, its meaning $\mu(w)$ is determined by what the word points out, say p, and not by some definition d. For example, for a person to use the word *"water,"* in a correct manner, that person neither needs to possess a definition of water nor needs to understand all the physical properties of water. The person simply needs to know which entity the word *"water"* picks out in common usage.

In $\mathcal{V}_z$, persons understand moral terms, such as *"honesty,"*, in a similar manner. That is, moral terms are understood by persons through direct references instantiated in **exemplars**. Persons identify moral examplars through the emotion of **admiration**. The emotions of admiration and contempt play a foundational role in this theory. $\mathcal{V}_z$ posits a process very similar to scientific or empirical investigation. Exemplars are first identified and their traits are studied; then they are continously further studied to better understand their traits, qualities, etc. The status of an individual as an exemplar can change over time. Below is an informal version that we seek to formalize:

Informal Version $\mathcal{V}_z$

$\mathbf{I}_1$ Agent or person a perceives a person b perform an action α. This observation causes the emotion of admiration in a.

$\mathbf{I}_2$ a then studies b and seeks to learn what traits (habits/dispositions) b has.

The Goal

From the above presentation of $\mathcal{V}_z$, we can glean the following distilled requirements that should be present in any formalization.

$\mathcal{V}_z^f$ Formalization Components

($\mathbf{R}_1$) A formalization of emotions, particularly admiration.

($\mathbf{R}_2$) A representation of traits.

($\mathbf{R}_3$) A process of learning traits (and not just simple individual actions) from a small number of observations.

Building the Formalization

For fleshing out the above requirements and formalizing $\mathcal{V}_z$, we will use the **deontic cognitive event calculus** ($\mathcal{DCEC}$), a computational formal logic. This logic was used previously in (Govindarajulu and Bringsjord 2017a; Govindarajulu et al. 2017) to automate versions of the Doctrine of Double Effect ($\mathcal{DDE}$), an ethical principle with deontological and consequentialist components. $\mathcal{DCEC}$ has also been used to formalize *akrasia* (the process of succumbing to temptation to violate moral principles) (Bringsjord et al. 2014). Fragments of $\mathcal{DCEC}$ have been used to model highly intensional reasoning processes, such as the false-belief task (Arkoudas and Bringsjord 2008).[5]

[4] In her book, Vallor gives her own more detailed and technologically relevant list of seven core elements that can be viewed as common to all members of $\mathcal{V}$ (or two what she refers to as "traditions" within virtue ethics). We do not have the space to discuss this list, and show that it fits nicely with our technical work's emphasis on exemplars and learning therefrom.

[5] $\mathcal{DCEC}$ is both *intensional* and *intentional*. There is a difference between intensional and intentional systems. Broadly speaking, extensional systems are formal systems in which the references and meanings of terms are independent of any context. Intensional systems are formal systems in which meanings of terms are dependent on context, such as the cognitive states of agents, time, etc. Modal logics used for modeling beliefs, desires, and intentions are considered intensional systems. Please see the appendix in (Govindarajulu and Bringsjord 2017a) for a more detailed discussion.

Overview of $\mathcal{DCEC}$

$\mathcal{DCEC}$ is a quantified multi-operator[6] modal logic (also known as sorted first-order multi-operator modal logic) that includes the event calculus, a first-order calculus used for commonsense reasoning over time and change (Mueller 2014). This calculus has a well-defined syntax and proof calculus; see Appendix A of (Govindarajulu and Bringsjord 2017a). The proof calculus is based on natural deduction (Gentzen 1935), and includes all the introduction and elimination rules for first-order logic, as well as inference schemata for the modal operators and related structures. As a sorted calculus, $\mathcal{DCEC}$ can be regarded analogous to a typed programming language. We show below some of the important sorts used in $\mathcal{DCEC}$. Among these, the $\textsf{Agent}$, $\textsf{Action}$, and $\textsf{ActionType}$ sorts are not native to the event calculus.

Sort	Description
Agent	Human and non-human actors.
Time	The Time type stands for time in the domain;
Event	Used for events in the domain.
ActionType	Abstract actions instantiated at particular times by actors.
Action	Events that occur as actions by agents.
Fluent	representing states of the world

Note: actions are events that are carried out by an agent. For any action type α and agent a, the event corresponding to a carrying out α is given by $action(a, \alpha)$. For instance, if α is *"running"* and a is *"Jack"* , $action(a, \alpha)$ denotes *"Jack is running"*.

Syntax The syntax has two components: a first-order core and a modal system that builds upon this core. The figures below show the formal language and inference schemata of $\mathcal{DCEC}$. Commonly used function and relation symbols of the event calculus are included. Any formally defined calculi (e.g. the venerable *situation calculus*) for modeling commonsense and physical reasoning can be easily switched out in-place of the event calculus.

The modal operators present in the calculus include the standard operators for knowledge $\mathbf{K}$, belief $\mathbf{B}$, desire $\mathbf{D}$, intention $\mathbf{I}$, obligation $\mathbf{O}$ etc. For example, consider $\mathbf{B}(a, t, \phi)$, which says that agent a believes at time t the proposition ϕ. Here ϕ can in turn be any arbitrary formula.

[6]The full catalogue of available operators exceeds those for belief, desire, and intention, and *a fortiori* exceeds the available operators in any standard modal logic designed to formalize e.g. only either alethic, epistemic, or deontic phenomena.

Syntax (fragment)

$$S ::= \textsf{Agent} \mid \textsf{ActionType} \mid \textsf{Action} \sqsubseteq \textsf{Event} \mid \textsf{Moment} \mid \textsf{Fluent}$$

$$f ::= \begin{cases} action : \textsf{Agent} \times \textsf{ActionType} \rightarrow \textsf{Action} \\ holds : \textsf{Fluent} \times \textsf{Moment} \rightarrow \textsf{Formula} \\ happens : \textsf{Event} \times \textsf{Moment} \rightarrow \textsf{Formula} \end{cases}$$

$$t ::= x : S \mid c : S \mid f(t_1, \ldots, t_n)$$

$$\phi ::= \begin{cases} q : \textsf{Formula} \mid \neg\phi \mid \phi \land \psi \mid \phi \lor \psi \mid \forall x : \phi(x) \mid \\ \mathbf{P}(a, t, \phi) \mid \mathbf{K}(a, t, \phi) \mid \\ \mathbf{C}(t, \phi) \mid \mathbf{S}(a, b, t, \phi) \mid \mathbf{S}(a, t, \phi) \mid \mathbf{B}(a, t, \phi) \\ \mathbf{O}(a, t, \phi, (\neg)happens(action(a^*, \alpha), t')) \end{cases}$$

Inference Schemata The figure below shows a fragment of inference schemata for $\mathcal{DCEC}$. $I_{\mathbf{B}}$ is an inference schema that let us model idealized agents that have their knowledge and belief closed under the $\mathcal{DCEC}$ proof theory. While normal humans are not deductively closed, this lets us model more closely how deliberative agents such as organizations and more strategic actors reason. (Some dialects of cognitive calculi restrict the number of iterations on intensional operators.) I_{12} states that if an agent s communicates a proposition ϕ to h, then h believes that s believes ϕ. I_{14} dictates how obligations propagate to intentions.

Inference Schemata (fragment)

$$\frac{\mathbf{B}(a, t_1, \Gamma), \ \Gamma \vdash \phi, \ t_1 < t_2}{\mathbf{B}(a, t_2, \phi)} \ [I_{\mathbf{B}}] \qquad \frac{\mathbf{S}(s, h, t, \phi)}{\mathbf{B}(h, t, \mathbf{B}(s, t, \phi))} \ [I_{12}]$$

$$\frac{\mathbf{B}(a, t, \phi) \quad \mathbf{B}(a, t, \mathbf{O}(a, t, \phi, \chi)) \quad \mathbf{O}(a, t, \phi, \chi)}{\mathbf{K}(a, t, \mathbf{I}(a, t, \chi))} \ [I_{14}]$$

We also define the following inference-schemata-based relationships between expressions in our calculus.

Generalization of Formulae. The generalization of a set of formulae Ψ, is a set of formulae Φ from which any element of Ψ can be inferred: $\Phi \vdash \bigwedge \Psi$. This is denoted by $g(\Psi) = \Phi$.

Generalization of Terms: A term x is a generalization of a term y if given any first-order predicate P, from $P(x)$ we can derive $P(y)$: $\{P(x)\} \vdash P(y)$. This is denoted by $g(y) = x$.

Semantics

$\mathcal{DCEC}$ uses *proof-theoretic* semantics (Gentzen 1935; Francez and Dyckhoff 2010), an approach commonly associated with natural deduction inference systems. Briefly, in this approach, meanings of modal operators are defined via functions over proofs. Specifying semantics then reduces to specifying inference schemata.

Events, Fluents, and Utilities

In the event calculus, fluents represent states of the world. Our formalization of admiration requires a notion of utility for states of the world. Therefore, we assign utilities to fluents through a utility function: $\mu : \textsf{Fluent} \times \textsf{Time} \rightarrow \mathbb{R}$. An event can initiate one or more fluents. Therefore, events can also have a utility associated with them. For an event e at time t, let e^t_I be the set of fluents initiated by the event, and let e_T^t be the set of fluents terminated by the event. If we are looking up till horizon H, then $\nu(e,t)$, the total utility of event e at time t, is:

$$\nu(e,t) = \sum_{y=t+1}^{H} \left(\sum_{f \in e_I^t} \mu(f,y) - \sum_{f \in e_T^t} \mu(f,y) \right)$$

With the calculus given above, we now move on to specifying parts of the formalization $\mathcal{V}_z^f$, that is, $\mathbf{R}_1$, $\mathbf{R}_2$, and $\mathbf{R}_3$.

Defining Admiration

We start with $\mathbf{R}_1$ and formalize admiration in $\mathcal{DCEC}$. To acheive this, we build upon the **OCC model**. There are many models of emotion from psychology and cognitive science. Among these, the OCC model (Ortony, Collins, and Clore 1988) has found wide adoption among computer scientists. Note that the model presented by (Ortony, Collins, and Clore 1988) is informal in nature and one formalization of the model has been presented in (Adam, Herzig, and Longin 2009). This formalization is based on propositional modal logic, and while comprehensive and elaborate, is not expressive enough for our modelling, which requires at the least quantification over objects.

In OCC, emotions are short-lived entities that arise in response to *events*. Different emotions arise based on: (i) whether the *consequences* to events are positive (desirable) or negative (undesirable); (ii) whether the event has occured; and (iii) whether the event has consequences for the agent or for another agent. OCC assumes an undefined primitive notion of an agent being *pleased* or *displeased* in response to an event. We represent this notion by a predicate Θ in our formalization. In OCC, admiration is defined as "*(approving of) someone else's praiseworthy action*." We translate this definition into $\mathcal{DCEC}$ as follows. An agent a is said to admire another agent b's action α, if agent a believes the action is a good action. An action $action(b,\alpha)$ is a considered a good action if $\nu(action(b,\alpha),t) > 0$. In OCC, agents can admire only other agents and not themselves. This is captured by the inequality $a \neg = b$

($\mathbf{R}_1$) Admiration in $\mathcal{DCEC}$

$$\begin{gathered} holds(admires(a,b,\alpha),t) \\ \leftrightarrow \\ \left(\begin{gathered} \Theta(a,t') \quad \wedge \\ \mathbf{B}\left(a,t,\begin{bmatrix} (a \neq b) \wedge (t' < t) \\ \wedge\, happens(action(b,\alpha),t') \wedge \\ \nu(action(b,\alpha),t) > 0 \end{bmatrix}\right) \end{gathered} \right) \end{gathered}$$

Defining Traits

To satisfy $\mathbf{R}_2$, we need to define traits. We define a *situation* $\sigma(t)$ as simply a collection of formulae that describe what fluents hold at a time t, along with other event-calculus constraints and descriptions (sometimes we use $\sigma(t)$ to represent the conjunction of all the formulae in $\sigma(t)$.)

($\mathbf{R}_2$) Trait

An agent a has a situation σ and action type α as an *m-trait* $\langle \sigma, \alpha \rangle$ if there are at least m situations $\{\sigma_1, \sigma_2, \ldots, \sigma_m\}$ in which *instantiations* of α are performed, and σ is the generalization of the situations.

A trait $\langle \sigma, \alpha \rangle$ can be represented as single formula:

$$\tau \equiv \sigma \wedge happens(action(\alpha,a),t)$$

We introduce a new modal operator $\mathbf{Trait}$ that can then be applied to the collection of formulae τ denoting a trait. $\mathbf{Trait}(\tau,a)$ says that agent a has trait τ. The following inference schema then applies to $\mathbf{Trait}$:

($\mathbf{R}_2$) Inference Schema for Trait

$$\frac{\left\{ \begin{matrix} \sigma_i, happens(action(\alpha_i,a),t_i) \\ g(\sigma_i(t)) = \sigma,\ g(\alpha_i) = \alpha \end{matrix} \right\}_{i=1}^{n}}{\mathbf{Trait}(\tau,a)}\ [I_{\mathbf{Trait}}]$$

Defining Learning of Traits

To address $\mathbf{R}_3$ we need a definition of what it means for an agent to learn a trait. We start with a learning agent l. An agent e is identified as an exemplar by l *iff* the emotion of admiration is triggered n times or more by e in l. This is written down in $\mathcal{DCEC}$ as follows (note that admiration can be triggered by different actions):

Exemplar Defnინition

$$Exemplar(e,l) \leftrightarrow \exists^{!n} t. \exists \alpha. holds(admires(l,e,\alpha),t)$$

Once e is identified as an exemplar, the learner then identifies one or more traits of e by observing e over an extended period of time. Let l believe that e has a trait τ; then l incorporates τ as its own trait:

($\mathbf{R}_3$) Learning a Trait

$$LearnTrait(l,\tau,t) \leftrightarrow \exists e \begin{bmatrix} Exemplar(e,l) \wedge \\ \mathbf{B}\Big(l,t,\mathbf{Trait}\big(\tau,e\big)\Big) \end{bmatrix}$$

$$LearnTrait(l,\langle\sigma,\alpha\rangle,t) \rightarrow \big(\sigma \rightarrow happens(action(l,\alpha),t)\big)$$

Example For instance, if the action type *"being truthful"* is triggered in situations: *"talking with alice,"*, *"talking with bob"*, *"talking with charlie"*; then the trait learned is that *"talking with an agent"* situation should trigger the *"being truthful"* action type.

A Note on Learning Methods

When we look at humans learning virtues by observing others or by reading from texts or other sources, it is not entirely clear how models of learning that have been successful in perception and language processing (e.g. the recent successes of deep learning and statistical learning) can be applied. Learning in $\mathcal{V}$-relevant situations is from one or few instances or in some cases through instruction, and such learning may not be readily amenable to models of learning which require a large number of examples.

The abstract learning method that we will use is **generalization**, defined previously. See one simple example immediately below:

Example 1

$$\frac{\begin{array}{r}\Gamma_1 = \{\mathit{talkingWith}(\mathit{jack}) \rightarrow \mathit{Honesty}\} \\ \Gamma_2 = \{\mathit{talkingWith}(\mathit{jill}) \rightarrow \mathit{Honesty}\}\end{array}}{\textbf{generalization}\ \Gamma = \{\forall x.\mathit{talkingWith}(x) \rightarrow \mathit{Honesty}\}}$$

One particularly efficient and well-studied mechanism to realize generalization is **anti-unification**, which has been applied successfully in learning programs from few examples.[7] In anti-unification, we are given a set of expressions $\{f_1, \ldots, f_n\}$ and need to compute an expression g that when substituted with an appropriate term θ_i gives us f_i. For example, if we are given $\mathit{hungry}(\mathit{jack})$ and $\mathit{hungry}(\mathit{jill})$, the anti-unification of those terms would be $\mathit{hungry}(x)$.

In higher-order anti-unification, we can substitute function symbols and predicate symbols. Here P is a higher-order variable.

$$\begin{array}{ll}\textbf{Example2} & \textbf{Example3} \\ \mathit{likes}(\mathit{jill}, \mathit{jack}) & \mathit{likes}(\mathit{jill}, \mathit{jack}) \\ \mathit{likes}(\mathit{jill}, \mathit{jim}) & \mathit{loves}(\mathit{jill}, \mathit{jim}) \\ \hline \mathit{likes}(\mathit{jill}, x) & P(\mathit{jill}, x)\end{array}$$

Defining Virtuous Person and Virtues

With the formal machinery in place we finally present formalizations that answer $\mathbf{Q}_1$ and $\mathbf{Q}_2$ posed at the outset. An n-virtuous person or agent s is an agent that is considered as an exemplar by n agents:

($\mathbf{Q}_1$) Virtuous Person

$$\mathbf{V}_n(s) \leftrightarrow \exists^{\geq n} a : \mathit{Exemplar}(s, a)$$

An n-virtue is a trait possesed by at least n virtuous agents:

($\mathbf{Q}_2$) Virtue

$$\mathbf{G}_n(\tau) \leftrightarrow \exists^{\geq n} a : \mathbf{Trait}(\tau, a)$$

Implementation & Simulation

We have extended *ShadowProver*, a quantified modal logic prover for $\mathcal{DCEC}$ used in (Govindarajulu and Bringsjord 2017a) to handle the new inference schemata and definitional axioms given above. We now show a small simulation in which an agent learns a trait and uses that trait to perform an action. Assume that we have a marketplace where things that are either old or new can be bought and sold. A seller can either honestly state the condition of an item $\{\mathit{new}, \mathit{old}\}$ or falsely report the state of the item. Agent a has two items x and y. x is new and y is old. a is asked about the state of the items, and a responds accurately. We have an agent d that observes agent a correctly report the state of the items. d also has beliefs about a's state of mind. We also have that the agent d considers a to be an exemplar. When all this information is fed into the prover along with the definitions above, d learns a trait representing a form of honesty, shown below:

$$\left\langle \begin{array}{l}\mathbf{B}(d, t, \mathit{holds}(x, t) \wedge \nu(\mathit{utter}(x), t) > 0), \\ \mathit{utter}(x)\end{array} \right\rangle$$

When d is queried about the state of an item u, d responds accurately (input and output shown in Figure 2). The prover responds with the required output in 3.6 seconds.[8]

```
{ ;; A's state of mind.
  P1 (Believes! I now (and (Knows! a t1 (holds (state x new) t1))
                           (Knows! a t2 (holds (state y old) t2))))
 ;; D observes a's utterances
  P2 (Perceives! a t1 (happens (action a (utters (state x new))) (next t1)))
  P3 (Perceives! a t2 (happens (action a (utters (state y old))) (next t2)))
  Background (Believes! I t0 (holds (state u old) now))
  Admire (Exemplar a d)}
-----------------------------------------------------------------------------
(happens (action d (utters  (state  u old))) (next now))}
```

Figure 2: Simulation Input and Output Formulae

Conclusion & Future Work

We have presented an initial formalization $\mathcal{V}_z^f$ of a virtue ethics theory $\mathcal{V}_z$ in a calculus that has been used in automating other ethical principles in deontological and consequentialist ethics. Many important questions have to be addressed in future research. Among them are questions about the nature and source of the utility functions that are used in the definitions of emotions. Lacking in our above model is an account of uncertainties and how they interact with virtues. We plan to leverage an account of uncertainty for a fragment of $\mathcal{DCEC}$ presented in (Govindarajulu and Bringsjord 2017b). In future work, we will compare learning traits with work on learning norms (Sarathy, Scheutz, and Malle 2017). The notion of learning we have presented here is quite abstract. In order to handle more complex traits, more sophisticated learning frameworks may have to be considered. Finally, we need to apply this model to more realistic examples and case studies, and implement our theories in realistic robotics architectures (Sarathy et al. 2016). The way forward to the production of virtuous machines is thus challenging, but we are confident that the foundation is now in place for their eventual arrival.

Acknowledgments. Support from ONR (to pursue morally competent robots) and AFOSR (to pursue forms of logicist AI made possible by high-expressivity calculi) has in part enabled the research and engineering that underlies the present paper, and we are most grateful.

[7]This discipline, known as **inductive programming**, seeks to build precise computer programs from examples (Nienhuys-Cheng and De Wolf 1997). See (Muggleton et al. 2018) for an application in generating human comprehensible programs.

[8]See: https://github.com/naveensundarg/prover/releases/tag/virtue-ethics-simulation.

References

Adam, C.; Herzig, A.; and Longin, D. 2009. A Logical Formalization of The Occ Theory of Emotions. *Synthese* 168(2):201–248.

Alfano, M. 2013. Identifying and Defending the Hard Core of Virtue Ethics. *Journal of Philosophical Research* 38:233–260.

Annas, J. 2011. *Intelligent Virtue*. Oxford, UK: Oxford University Press. Kindle edition used for AIES 2019.

Aristotle. 2000. *Nicomachean Ethics*. Cambridge, UK: Cambridge University Press. The editor and translator is Roger Crisp. Aristotle wrote the work around 340 BC.

Arkoudas, K., and Bringsjord, S. 2008. Toward Formalizing Common-Sense Psychology: An Analysis of the False-Belief Task. In Ho, T.-B., and Zhou, Z.-H., eds., *Proceedings of the Tenth Pacific Rim International Conference on Artificial Intelligence (PRICAI 2008)*, number 5351 in Lecture Notes in Artificial Intelligence (LNAI), 17–29. Springer-Verlag.

Bringsjord, S.; Arkoudas, K.; and Bello, P. 2006. Toward a General Logicist Methodology for Engineering Ethically Correct Robots. *IEEE Intelligent Systems* 21(4):38–44.

Bringsjord, S.; Govindarajulu, N. S.; Thero, D.; and Si, M. 2014. Akratic Robots and the Computational Logic Thereof. In *Proceedings of ETHICS • 2014 (2014 IEEE Symposium on Ethics in Engineering, Science, and Technology)*, 22–29. IEEE Catalog Number: CFP14ETI-POD.

Feldman, F. 1978. *Introductory Ethics*. Englewood Cliffs, NJ: Prentice-Hall.

Francez, N., and Dyckhoff, R. 2010. Proof-theoretic Semantics for a Natural Language Fragment. *Linguistics and Philosophy* 33:447–477.

Gentzen, G. 1935. Investigations into Logical Deduction. In Szabo, M. E., ed., *The Collected Papers of Gerhard Gentzen*. Amsterdam, The Netherlands: North-Holland. 68–131. This is an English version of the well-known 1935 German version.

Govindarajulu, N. S., and Bringsjord, S. 2017a. On Automating the Doctrine of Double Effect. In Sierra, C., ed., *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, 4722–4730. Preprint available at this url: https://arxiv.org/abs/1703.08922.

Govindarajulu, N. S., and Bringsjord, S. 2017b. Strength Factors: An Uncertainty System for a Quantified Modal Logic. Presented at Workshop on Logical Foundations for Uncertainty and Machine Learning at IJCAI 2017, Melbourne, Australia.

Govindarajulu, N. S.; Bringsjord, S.; Ghosh, R.; and Peveler, M. 2017. Beyond the doctrine of double effect: A formal model of true self-sacrifice. International Conference on Robot Ethics and Safety Standards.

Hiller, A. 2011. The Unusual Logic of Hurka's Recursive Acount. *Journal of Ethics and Social Philosophy* 6(1):1–6.

Hurka, T. 2000. *Virtue, Vice, and Value*. Oxford, UK: Oxford University Press.

Johnson, R. 2004/2016. Kant's Moral Philosophy. In Zalta, E., ed., *The Stanford Encyclopedia of Philosophy*.

Kant, I. 1997/1785. *Practical Philosophy*. Cambridge, UK: Cambridge University Press. This volume, edited by Mary Gregor, collects all of Kant's major writings on moral and political philosophy together, and includes what has traditionally taken to be the definitive source of Kant's views on ethics, viz. *The Groundwork of the Metaphysics of Morals*, first published in 1785 in the German (as *Grundlegung zur Metaphysik der Sitten*).

Miles, J. 2013. Against the Recursive Account of Virtue. *Theoretical & Applied Ethics* 2(1):83–92.

Mueller, E. 2014. *Commonsense Reasoning: An Event Calculus Based Approach*. San Francisco, CA: Morgan Kaufmann.

Muggleton, S. H.; Schmid, U.; Zeller, C.; Tamaddoni-Nezhad, A.; and Besold, T. 2018. Ultra-strong machine learning: comprehensibility of programs learned with ilp. *Machine Learning* 107(7):1119–1140.

Nienhuys-Cheng, S.-H., and De Wolf, R. 1997. *Foundations of Inductive Logic Programming*, volume 1228. Springer Science & Business Media.

Ortony, A.; Collins, A.; and Clore, G. L. 1988. *The Cognitive Structure of Emotions*. Number 0521353645. Cambridge [England] ; New York : Cambridge University Press.

Powers, T. 2006. Prospects for a Kantian Machine. *IEEE Intelligent Systems* 21:4.

Quinn, P. 1978. *Divine Commands and Moral Requirements*. Oxford, UK: Oxford University Press.

Sarathy, V.; Wilson, J. R.; Arnold, T.; and Scheutz, M. 2016. Enabling Basic Normative HRI in a Cognitive Robotic Architecture. *arXiv preprint arXiv:1602.03814*.

Sarathy, V.; Scheutz, M.; and Malle, B. F. 2017. Learning Behavioral Norms in Uncertain and Changing Contexts. In *Cognitive Infocommunications (CogInfoCom), 2017 8th IEEE International Conference on*, 000301–000306. IEEE.

Scheutz, M., and Malle, B. F. forthcoming. *Moral Robots*. New York: NY: Routledge/Taylor & Francis. URL: http://research.clps.brown.edu/SocCogSci/Publications/Pubs/ScheutzMalle_inpress_NeuroethicsMoralRobots.pdf.

Vallor, S. 2016. *Technology and the Virtues: A Philosophical Guide to a Future Worth Wanting*. Oxford, UK: Oxford University Press.

Van Norden, B. 2007. *Virtue Ethics and Consequentialism in Early Chinese Philosophy*. Cambridge, UK: Cambridge University Press.

Zagzebski, L. 2010. Exemplarist Virtue Theory. *Metaphilosophy* 41(1-2):41–57.