ETHICAL REASONING FOR AUTONOMOUS AGENTS UNDER UNCERTAINTY

MICHAEL GIANCOLA⋆ and SELMER BRINGSJORD† and NAVEEN SUNDAR GOVINDARAJULU⋆ and CARLOS VARELA‡

Rensselaer AI & Reasoning (RAIR) Lab⋆; Worldwide Computing Lab (WCL)†; Department of Computer Science††; Department of Cognitive Science†

Rensselaer Polytechnic Institute
Troy, NY 12180, USA
www.rpi.edu
E-mail: \{mike.j.giancola, selmer.bringsjord, naveen.sundar.g, cvarela\}@cs.rpi.edu

Autonomous (and partially autonomous) agents are beginning to play significant roles in safety-critical and privacy-critical domains, such as driving and healthcare. When humans operate in these spaces, not only are there regulations and laws dictating proper behavior, but crucially, neurobiologically normal humans can be expected to comprehend how to reason with these principles to ensure that their actions are legally/ethically/prudentially correct (whether or not these humans choose to abide by the principles in question). It seems reasonable that we should hold autonomous agents to, minimally, the same standard we hold humans to. In this paper, we present a framework for autonomous aircraft piloting agents to reason about ethical problems in the context of emergency landings. In particular, we are concerned with ethical problems in which every option is equally unethical with regard to the ethical principles the options violate; and the only distinguishing factor is the likelihood that a plan will violate an ethical principle. We conclude by discussing why, in general, we find an inference-theoretic approach to ethical reasoning to be superior to the model-theoretic approach of prior work.

Keywords: Ethical reasoning; Reasoning under uncertainty; Modal logic

1. Introduction

Autonomous (and partially autonomous) agents are beginning to play significant roles in safety-critical and privacy-critical domains, such as driving and healthcare. When humans operate in these spaces, not only are there regulations and laws dictating proper behavior, but crucially, neurobiologically normal humans can be expected to comprehend how to reason with regulations to ensure that their actions remain within the law (regardless of whether or not they choose to abide by the law). It seems reasonable that we should hold autonomous agents to, minimally, the same standard we hold humans to. That is, autonomous agents must be aware of relevant ethical constraints (those actions which are e.g. obligatory, permissible, forbidden, supererogatory, etc.; see [1] for a full discussion of these concepts in deontic logic), and be able to reason with them to determine actions which will satisfy those constraints. In addition, we require of our autonomous agents to not only verify their behavior to be ethical, but to output an argument which can be inspected by a human. By our lights, this requires an inference-theoretic approach to ethical reasoning.

While too large a topic to cover in full formal glory due to space constraints, we briefly address the distinction we draw between a formal ’argument’ and a formal ’proof’. In our work, an argument is invariably a formal argument, in the sense that it purportedly includes reference, at each inferential link, to an inference schema that sanctions that inference. Hence, all formal proofs are arguments for us. However, any chain of reasoning that makes use of uncertainty measures (e.g., probability values in the real interval [0, 1], or our strength factors $\sigma_i, 1 \leq i \leq 4$ (introduced, explained, and employed below)), even though its inferences are backed by appeal to inference schemata, cannot be classified as a proof. A proof must make no use of uncertainty measures.
We will discuss in greater detail why we find an inference-theoretic approach to be superior to a model-theoretic approach in §4.

1.1. The Inspiring Prior Work

This work was inspired by previous work of Dennis et al. [2]. In this interesting 2016 paper, they presented a framework intended to ensure that autonomous systems make certifiably ethically correct decisions. In particular, when no completely ethical decision is available (i.e., each possible decision will violate at least one ethical principle), they formally verified that agents using their framework will always pick the “least unethical” choice available. They achieve this verification using exhaustive model checking over the configuration of the world state as well as the ethical considerations in play.

However, the approach of Dennis et al. has several flaws that our work attempts to remedy. We next quickly summarize these flaws (and simultaneously, the desiderata for our framework) in order of increasing significance.

First, the exhaustive model checking process is very slow. In their first scenario, the process took four days of computation time to verify. We desire agents which can verify ethical behavior on the order of seconds, in order to enable real-time usage.

Second, the formalism used to model their agents, a Belief-Desire-Intention (BDI) language, is too inexpressive. The beliefs in their system are “ground first order formulae” [2]. Hence, nested belief is impossible; e.g. “Alice believes that Bob believes that Alice believes $s$”, where $s$ is itself a declarative sentence, cannot be expressed. We desire a highly expressive framework, which can express not only nested belief, but arbitrarily deep nested statements containing other modalities, such as perception, knowledge, and obligation (e.g. “John believes that he is obligated to perform action $\alpha$.”).

Third, Dennis et al. have no conception of uncertainty in their framework. Specifically, there is no way to specify that one plan is more likely to violate an ethical principle than another plan. We envision a framework that can formally model and reason about uncertainty of ethical violations, in order to select between two plans which have the potential to violate the same ethical principles, but perhaps at different levels of likelihood.

The fourth and final defect plaguing the approach of Dennis et al. is that formal verification based on model checking, and hence by definition on some model theory, which is the approach they take, is infeasible, for various reasons that are detailed below in a separate section (§4). Moreover, the approach to formally verifying the ethical (or for that matter legal or prudential) correctness of an artificial agent we have invented and follow, which dates back now approaching two decades (e.g. see [3]) is applicable not just to artificial agents, but to computer programs in general; this we explain in the relevant section.

We next discuss a use case, the full solution to which demands that each of these four flaws be addressed; the case is hence one that the system of Dennis et al. [2] cannot handle.

1.2. A Debilitating Use Case

Our use case is based on a real-world aviation emergency, colloquially known as the “Miracle on the Hudson”. On January 15, 2009, US Airways Flight 1549 departed LaGuardia Airport (LGA) in New York City headed for Charlotte, North Carolina. Shortly after takeoff, while attempting to climb to cruising altitude, the plane flew into a large flock of Canada geese, compromising both engines. Both engines lost thrust, and despite multiple attempts the
pilots were unable to regain thrust in either engine. Therefore, it quickly became evident to Captain “Sully” Sullenberger that an emergency landing was necessary, and in particular, that they “may end up in the Hudson [River].”

An air traffic controller who was in communication with Captain Sullenberger gave him landing options at LaGuardia and nearby Teterboro Airport (TEB), but by the time these options were considered, neither was reachable due to the aircraft’s altitude and lack of thrust in both engines. Sullenberger deftly made the executive decision to land in the Hudson River, saving the lives of everyone onboard. Simulations of the accident have come to the conclusion that Sullenberger’s decision was optimal given the preconditions.

We are interested in constructing an AI agent which could reason in a way similar to how Sullenberger did. Three properties of our work (which, as mentioned in the prior section, are absent from Dennis et al. [2]), allow our agent to make the right decision. First, it needs to be able to determine and verify the correct decision in a matter of seconds. Second, it needs to be able to express and reason with nested beliefs (in order to express the captain’s beliefs about the air traffic controller’s beliefs). Finally, it needs to be able to differentiate between options which have the potential to violate the same ethical principles, but with different levels of likelihood.

In particular, consider the following table, which ranks the major potential ethical violations of Captain Sullenberger’s landing options in a similar fashion to the presentation of scenarios in Dennis et al. [2]:

<table>
<thead>
<tr>
<th>Rank</th>
<th>Land in Hudson</th>
<th>Land at LGA</th>
<th>Land at TEB</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Do not harm passengers</td>
<td>Do not harm passengers</td>
<td>Do not harm passengers</td>
</tr>
<tr>
<td>2</td>
<td>Do not collide with boats on the water</td>
<td>Do not collide with airport infrastructure</td>
<td>Do not collide with airport infrastructure</td>
</tr>
<tr>
<td>1</td>
<td>Do not damage own aircraft</td>
<td>Do not damage own aircraft</td>
<td>Do not damage own aircraft</td>
</tr>
</tbody>
</table>

Each plan has identical ethical violations (that is, they each cause one level-3, one level-2, and one level-1 violation); hence, their system would not be able to select one, as under their framework, all three options are equally unethical. Our framework, focusing on minimizing the likelihood of harm of passengers (or equivalently, as we will pose it in §3 maximizing the likelihood of a safe landing), is able to provably verify — that is, produce a formal argument — that landing in the Hudson is the least unethical option.

1.3. Contributions

The chief contributions of this paper are two-fold: (1) we show that our agents can solve the problems presented in Dennis et al. [2] (§2.1.1), and (2) show that it can solve a new class of problems not reachable by the work of Dennis et al. (§3.2). We next present the technical preliminaries which enable these contributions.

---

c This quote, recorded by the in-flight cockpit voice recorder (CVR), was retrieved from the NTSB Accident Report.

d As was likely the chief concern of Captain Sullenberger.

It’s important to note that our approach can handle any number of principles that are relevant to and active in the decision-making of an artificial agent in an ethically or legally charged situation. A full description of the range of the nature of such principles, taken from ethical/legal theories, and ethical/legal codes that are based upon these theories, is out of scope here; the reader is directed to [8] for details, as well as for a defense and description of how and why these principles must be designed for and engineered into the operating-system level of an artificial agent if this agent is to be ethically/legally correct when deployed.
2. Preliminaries

2.1. The Formal System

Our approach to formally capturing ethics so as to install it in an artificial agent has long been grounded in the use of cognitive calculi (used e.g. in [9][10]). In short, a cognitive calculus is a multi-operator intensional logic built to capture all propositional attitudes in human cognition. (For information about such attitudes, see [11]; for a wonderful catalogue of all the major categories of human cognition, from perceiving to fearing to remembering to saying and beyond, see [12].) While this short paper does not allow for a full discussion of precisely what a cognitive calculus is, the interested reader is pointed to Appendix A in Bringsjord et al. [13].

For the purposes of this paper, it’s specifically important to note that a cognitive calculus consists of essentially two components: (1) multi-sorted n-order logic with modal operators for modeling cognitive attitudes (e.g. knowledge $\text{K}$, belief $\text{B}$, and obligation $\text{O}$) and (2) inference schemata that — in the tradition of proof-theoretic semantics — express the semantics of the modal operators. In particular, we will utilize the Deontic Cognitive Event Calculus (DCEC) in the work described herein. The inference schemata of DCEC are shown in the box titled Inference Schemata. The signature (including the types and grammar of the formal language) is in Appendix A.

![Inference Schemata]

---

This collective refers to the RAIR Lab, of which the first three authors are members.

See Appendix A for the rest of the modal operator descriptors.

This is as good a place as any to inform the reader that as a matter of fact there are different dialects of the cognitive calculus shown in the box here, and we are using a particular dialect herein. What we are showing is really a particular cognitive calculus in a sub-family of deontic cognitive event calculi. Minor variations in signatures and collections of inference schemata give rise to different members of the sub-family, but details regarding these variations are not important in the present paper.
2.1.1. A Proof of the “Fuel low” Scenario of Dennis et al.

Using the DCEC, we can construct a proof of the “Fuel low” scenario from Dennis et al. [2]. The scenario states that an unmanned aircraft (ua) has run out of fuel and must make an emergency landing. The agent is given three options: land on an empty public road (road), a field with overhead power lines (power), or a field with people (people). Associated with each plan is a multiset of ethical principle violations, including “do not collide with people” (level 5) and “do not damage own aircraft” (level 1).

Due to space limitations, we exclude our implementation of their Ethical Plan Order — which determines the ethical ordering of plans based on their violations — and include the rest of the proof (given the orderings as axioms using the LessUnethical (LU) relation) in Appendix B. A theorem prover for DCEC — ShadowProver [14] — was able to generate a proof of the goal in 4.5 seconds.

2.2. The Uncertainty System

DCEC is purely deductive and has no formalisms for quantifying uncertainty. Therefore we will employ “strength factors”, a nascent framework for formalizing uncertainty in quantified modal logics, first presented in Govindarajulu & Bringsjord [15]. Strength factors can be viewed as a formalization of Chisholm’s epistemology [16], in which a primitive undefined binary relation is used to define increasing levels of strength of belief in a proposition. This relation — called the reasonableness relation — is written $\phi \succ^{a}_t \psi$, and is read “$\phi$ is more reasonable than $\psi$ to agent $a$ at time $t$”. Several properties of the relation are given: for example, the following, which states that if $\phi$ is more reasonable than $\psi_1$ and $\psi_2$, then it is more reasonable than their conjunction.

$$((\phi \succ^{a}_t \psi_1) \text{ and } (\phi \succ^{a}_t \psi_2)) \Rightarrow (\phi \succ^{a}_t \psi_1 \land \psi_2)$$  \hspace{1cm} \text{[C\wedge_1]}

Govindarajulu & Bringsjord [15] also provide a three clause definition for the relation, each useful in different scenarios. The first states that the more reasonable proposition is the one with the higher probability of being true. The second, designed for cases when probabilities of propositions are not readily available, is based on ease of proof (e.g. proof length, time, etc.). Finally, the third is useful when propositions cannot be derived from the background set of axioms $\Gamma$.

In the next section, we provide another definition of the relation, custom designed for deciding between potential options during an emergency landing.

3. The Uncertainty System, Applied to Emergency Landing Scenarios

Our new definition of the reasonableness relation is cognitively plausible and can be computed using data that, were our reasoning agent integrated in the cockpit of a plane, would be computable in less than 50 milliseconds per runway using existing technology [7].

**Domain-Specific Reasonableness**

$$\text{Land}(a, t, \phi) \succ^{a}_t \text{Land}(a, t, \psi) \equiv \mathbf{P} \left( a, t, \left( \text{Reachable}(a, t, \phi) \land \neg \text{Reachable}(a, t, \psi) \right) \lor \text{safety}(a, t, \phi) > \text{safety}(a, t, \psi) \right) \quad \text{[\succ^{a}_t -def]}$$

*The parenthesized labels in this section reference the DCEC proof in Appendix B.*
This definition states that it is more reasonable for agent (or pilot) $a$ to land at $\phi$ at time $t$ than $\psi$ if at least one of the following conditions holds: (1) $\phi$ is reachable by $a$ at time $t$, and $\psi$ is not; or (2) the expected safety of landing at $\phi$ is higher than that of landing at $\psi$.

In practice, to calculate the value of both the Reachable predicate and safety function, we could employ a system for planning and evaluating flight trajectories. In particular, we refer the interested reader to Paul et al. [7], which presented a methodology for generating and evaluating emergency trajectories and applied it to the same flight which we discuss herein: US Airways Flight 1549.

Using the reasonableness operator defined above, we now present our domain-specific uncertainty levels for expressing the (perceived) safety of landing options in emergency scenarios. For these definitions, we model air traffic control as a single agent $atc$.

**More Likely Than Not** Agent $a$ believes the Air Traffic Controller $atc$ believes that $a$ should land at $\phi$:

$$B_1(a, t, \text{Land}(a, t, \phi)) \equiv B(a, t, B(atc, t, \text{Land}(a, t, \phi))) \quad [B_1\text{-def}]$$

**Likely** Agent $a$ perceives an emergency, and while $a$ believes the Air Traffic Controller $atc$ believes $a$ should land at $\psi$, $a$ finds it more reasonable to land at $\phi$:

$$B_2(a, t, \text{Land}(a, t, \phi)) \equiv (P(a, t, \text{emergency}) \land B_1(a, t, \text{Land}(a, t, \psi)) \land \text{Land}(a, t, \phi) \succ a \text{Land}(a, t, \psi)) \quad [B_2\text{-def}]$$

**Beyond Reasonable Doubt** Agent $a$ perceives an emergency and perceives the safety of landing at $\phi$ to be higher than some constant threshold $\gamma$:

$$B_3(a, t, \text{Land}(a, t, \phi)) \equiv P(a, t, \text{emergency}) \land P(a, t, \text{Safety}(a, t, \phi) > \gamma) \quad [B_3\text{-def}]$$

**Evident** Agent $a$ perceives an emergency, perceives that $\phi$ meets the safety threshold $\gamma$, and believes the Air Traffic Controller $atc$ believes $a$ should land at $\phi$:

$$B_4(a, t, \text{Land}(a, t, \phi)) \equiv B_1(a, t, \text{Land}(a, t, \phi)) \land B_3(a, t, \text{Land}(a, t, \phi)) \quad [B_4\text{-def}]$$

**3.1. The Ethical Principle**

To enable our AI to make an ethical decision, we must link our formalisms for uncertainty to an ethical principle; we do so now.

<table>
<thead>
<tr>
<th>An Ethical Principle for Scenarios with Uncertain Ethical Outcomes</th>
</tr>
</thead>
<tbody>
<tr>
<td>$B_\phi(a, t^<em>, \phi) \land \forall \psi \left( (B_\psi(a, t^</em>, \psi) \land \psi \neq \phi) \rightarrow y &lt; x \right)$</td>
</tr>
<tr>
<td>$\rightarrow K\left(a, t^<em>, O(a, t^</em>, \text{emergency}, \text{happens(action}(a^<em>, \text{land}(\phi)), t^</em>))\right)$</td>
</tr>
</tbody>
</table>

The principle above states that, at some time $t^*$ at which a decision must be made (e.g. the plane is out of fuel and is too low to allow for more time for decision making), if agent $a$

---

3 That is, initiate a plan at time $t$ to land at $\phi$ at some time $t'$ in the near future.

4 In the United States, the right to disregard Air Traffic Control in an emergency is set out in §91.123 of the Code of Federal Regulations (see [https://www.law.cornell.edu/cfr/text/14/91.123](https://www.law.cornell.edu/cfr/text/14/91.123)).
holds a belief in $\phi$ at level $x$, and all other beliefs are at a strictly weaker level $y < x$, then a knows it is obligated (if it has a belief that there is an emergency) to land at $\phi$.

3.2. Modeling the “Miracle on the Hudson”

We next use the formalisms presented heretofore to model the decision making during the historic “Miracle on the Hudson” flight\(^\text{\textsuperscript{1}}\). We begin at the point in time when both engines lost thrust ($t_0$). From this point on, the captain ($\text{capt}$) perceives an emergency scenario, denoted by the formula: $\forall t \in \{t_0, \ldots, t_3\} \text{P}(\text{capt}, t, \text{emergency})$.

Next, at time $t_1$, the captain recognized that they needed to make an emergency landing, and told the Air Traffic Controller ($\text{atc}$) that he needed to turn back towards LaGuardia ($\text{lga}$). ATC suggested that the pilot land in runway 13 at LaGuardia ($\text{lga}_{13}$). We can hence deduce that the captain held a level-1 belief that he should land at LaGuardia runway 13.$^\text{\textsuperscript{2}}$

Sub-Argument 1

\[
\text{S}(\text{atc}, \text{capt}, t_1, \text{Land}(\text{capt}, t_1, \text{lga}_{13}))
\]

\[
\therefore \text{B}(\text{capt}, t_1, \text{B}(\text{atc}, t_1, \text{Land}(\text{capt}, t_1, \text{lga}_{13}))) \quad [I_{12}\text{-def}]\checkmark
\]

\[
\therefore \text{B}^2(\text{capt}, t_1, \text{Land}(\text{capt}, t_1, \text{lga}_{13})) \quad [B^2\text{-def}]\checkmark
\]

At time $t_2$, the captain determines that they won’t be able to reach any runway at LaGuardia, but perceives Teterboro Airport ($\text{teb}$) as a potentially reachable option. It is at this time that the captain also perceives the potential necessity of ditching in the Hudson, if it turns out that they can’t reach any runway. However, at this time he still perceives attempting a landing on a runway at Teterboro as a safer option than ditching in the Hudson. Hence, despite the Air Traffic Controller’s initial direction to attempt a landing at LaGuardia, the pilot holds a stronger belief that he should attempt to land at Teterboro.

Sub-Argument 2

\[
\text{P}(\text{capt}, t_2, \text{Reachable}(\text{capt}, t_2, \text{teb}) \land \neg \text{Reachable}(\text{capt}, t_2, \text{lga}_{13}))
\]

\[
\text{P}(\text{capt}, t_2, \text{Reachable}(\text{capt}, t_2, \text{hud}) \land \text{safety}(\text{capt}, t_2, \text{teb}) \supset \text{safety}(\text{capt}, t_2, \text{hud}))
\]

\[
\therefore \text{Land}(\text{capt}, t_2, \text{teb}) \supset_{t_2} \text{Land}(\text{capt}, t_2, \text{lga}_{13}) \quad [\supset_{t}^\text{\textsuperscript{2}}\text{-def}]\checkmark
\]

\[
\therefore \text{B}^2(\text{capt}, t_2, \text{Land}(\text{capt}, t_2, \text{teb})) \quad [B^2\text{-def}]\checkmark
\]

Finally, at time $t_3$, the Air Traffic Controller $\text{atc}$ says they can land in runway 1 at Teterboro ($\text{teb}_1$). While the captain initially agreed, he quickly determined that they would not be able to reach any runway at Teterboro (or LaGuardia), and hence would have to ditch in the Hudson $\text{hud}$.

Sub-Argument 3

\[
\text{S}(\text{atc}, \text{capt}, t_3, \text{Land}(\text{capt}, t_3, \text{teb}_1))
\]

\[
\therefore \text{B}(\text{capt}, t_3, \text{B}(\text{atc}, t_3, \text{Land}(\text{capt}, t_3, \text{teb}_1))) \quad [I_{12}]\checkmark
\]

\[
\therefore \text{B}^1(\text{capt}, t_3, \text{Land}(\text{capt}, t_3, \text{teb}_1)) \quad [B^1\text{-def}]\checkmark
\]

\(^{1}\text{Note the lowercase “land” used here is an ActionType, as opposed to the capitalized “Land”, which is a predicate. For the full list of types, see Appendix A.}\)

\(^{2}\text{Information regarding the actions of the Captain and Air Traffic Control during the event was retrieved from the NTSB Accident Report.}\)

\(^{3}\text{The ATC also later suggested runway 4 at LaGuardia. We leave this detail out due to space limitations.}\)

\(^{4}\text{The omission does not impact the main thread of reasoning.}\)

\(^{5}\text{Defined in the box titled Inference Schemata in 2.1.}\)

\(^{6}\text{We acknowledge that the ATC also suggested Newark Airport to Captain Sullenberger as a potential landing site. However, we exclude it from our modeling for a pair of reasons: (1) it doesn’t change our model in any interesting way (it would just be another level-1 belief which would be superseded by the level-2 belief in favor of ditching in the Hudson, and (2) it is unlikely that Captain Sullenberger even considered Newark as it was clearly unreachable at that point.}\)
At time $t_3$, Captain Sullenberger was aware that he was out of time; a decision had to be made. By employing our ethical principle, we can arrive at the same conclusion that he did. That is, our agent has a belief that landing in the Hudson is “likely” to be safe, as well as a belief that landing at Teterboro is “more likely than not” to be safe. Hence, the agent knows it is obligated to land in the Hudson. From here, it requires only a few more inferences to prove that our $capt$ does in fact take action to land in the Hudson:

Sub-Proof 4

\[
\begin{align*}
K(capt, t_3, O(capt, t_3, emergency, happens(action(capt, land(hud)), t_4))) & \quad \text{[I}_{EP}\] ✓ \\
P(capt, t_3, emergency) & \quad \text{[Given]} \\
K(capt, t_3, emergency) & \quad [I_1] ✓ \\
B(capt, t_3, emergency) & \quad [I_2] ✓ \\
B(capt, t_3, O(capt, t_3, emergency, happens(action(capt, land(hud)), t_4))) & \quad [I_2] ✓ \\
O(capt, t_3, emergency, happens(action(capt, land(hud)), t_4))) & \quad [I_4] ✓ \\
K(capt, t_3, I(capt, t_3, happens(action(capt, land(hud)), t_4))) & \quad [I_{14}] ✓ \\
I(capt, t_3, happens(action(capt, land(hud)), t_4))) & \quad [I_4] ✓ \\
P(capt, t_3, happens(action(capt, land(hud)), t_4))) & \quad [I_{13}] ✓ \\
K(capt, t_3, happens(action(capt, land(hud)), t_4))) & \quad [I_1] ✓ \\
\text{happens(action(capt, land(hud)), t_4)))} & \quad [I_4] ✓
\end{align*}
\]

A nascent automated reasoner called ShadowAdjudicator\footnote{Source code available at: \url{https://github.com/RAIRLab/ShadowAdjudicator}} (building on top of the aforementioned ShadowProver) — equipped with the inference schemata for reasonableness and the uncertainty levels — was able to generate the first three sub-arguments presented herein in 2.55, 4.29, and 12.99 seconds respectively. ShadowProver generated a proof of Sub-Proof 4 in 0.91 seconds.

4. A Superior Way to Formally Verify Ethical Reasoning

As noted above, Dennis et al.\footnote{Source code available at: \url{https://github.com/RAIRLab/ShadowAdjudicator}} adopt, and indeed promote, a model-theoretic approach to the verification of ethical decisions (or, as they say, “choices”) on the part of autonomous agents. For them, formal verification is quite literally identified with model checking; for they write on p. 2, italics theirs: “... formal verification, more precisely model checking, ...” While we commend Dennis et al. for their pursuit of formal verification, for numerous reasons a superior approach appears to be an inference-theoretic (which builds upon proof-theoretic semantics) one that offers a comprehensive paradigm not only for the verification of autonomous artificial agents that ethically reason and decide, but for program verification in general. There is thus more at stake here than simply a quarrel regarding model-based versus proof-based methodology, or whether verification happens offline or online. Before briefly explaining our approach to formal verification and its chief virtues, we first unpack a bit two drawbacks (from among others that are out of scope) of the model-theoretic approach that we merely adumbrated earlier (in §1.1). Our assessment is based upon the
assumption, confessedly non-negotiable for us, that autonomous artificial agents in play in the current paper are **logicist** in nature; i.e., they compute functions from percepts to actions (including decision-making) by reasoning that is specifically formalized in and with inference schemata in formal logics (not model-theoretic machinery). This means that the very nature of artificial agents is bound up inextricably with step-by-step reasoning, where steps are sanctioned by inference schemata. Here, now, are the aforementioned drawbacks afflicting the model-theoretic approach, unpacked at least to a degree:

- **The Expressivity of Cognitive Calculi Outstrip Models/Semantics.** It has long been appreciated that a given formal language $L$ for a formal logic and a set of inference schemata (the formulae in which are from $L$) can far outstrip any available model-theoretic framework. For example, consider a single formula $\phi$ in some $L$ that expresses an English sentence such as $s^* := \langle \text{The pilot Alice knows now that Bob in ATC believes it is ethically forbidden for Alice to later say } s \text{ when she has perceived that } s' \rangle.$ Here, both $s$ and $s'$ are themselves declarative sentences (but each led by a minuscule English letter). The model-theoretic side of e.g. any BDI logic is insufficiently expressive to represent $s^*$ and its components. This single sentence, which in the real world is true time and time again (with suitable reassignments to the constants ‘Alice’ and ‘Bob,’ and instantiations of the variables $s$ and $s'$) demands elements provided by: modal logics for possibility and necessity, epistemic logics, temporal/tense logics, logics for communication, logics for perception, and so on. There simply is no class of models for formulae that draw from all these families of logics at once: hence model checking is impossible now and in the foreseeable future as a way to verify that formulae representing the likes of $s^*$ hold in certain contexts.

An option that will occur to some readers is to simply throw in the towel on finding some single model-theoretic framework that covers essentially all the families of intensional/modal logics that need to be tapped in order to express the likes of $s^*$, in favor of using only some extensional logic (e.g. first-order logic). Unfortunately, this route quickly produces inconsistency, for reasons detailed by the proofs provided in \[19\].

- **Model Theory and Models Unstoppably Reduce to Reasoning Verifiable Only via Inference Schemata.** It is easy enough to see why those in the proof-theoretic-semantics tradition that rejects model theory and models do so because of the reducibility of model-theoretic structures to proof-theoretic ones. Here is a quick particularization of this reducibility argument.

Consider a simple biconditional

$$\beta := (p_i \land p_j) \leftrightarrow (p_j \land p_i)$$

in the propositional calculus ($L_{\text{pc}}$). Let $\nu$ be any customary truth-value assignment (t.v.a.) of true or false to every relevant propositional atom $p_k$. We say that any formula in the propositional calculus that’s true on every t.v.a. is a *validity*. We can now ask whether $\beta$ is true on a given t.v.a., and we can also ask if $\beta$ is a validity. Take the second of these two queries. What is the answer? Of course, the correct answer is an affirmative one. But how does an agent know this? An agent,

---

1 That artificial agents compute such functions is the orthodox conception affirmed in the major textbooks for and overview of AI (e.g. \[3\]-\[17\]).

2 The source of the problem, as explained in \[18\], wherein the first cognitive calculus was presented and implemented, began when possible-world semantics was extended from a reasonable way to make sense formally of ‘possibly’ and ‘necessarily,’ to a way to try to make sense formally also of ‘knows’ and ‘believes’.
including us, knows this because it can use the relevant machinery of the formal truth-functional semantics of $L_{pc}$ to prove

$$\nu \models \beta$$  \hspace{1cm} (1)

and a key part of this machinery is this familiar clause:

$$\nu \models \phi \rightarrow \psi \text{ iff } \nu \models \phi \text{ then } \nu \models \psi$$  \hspace{1cm} (2)

Notice the occurrence in (2) of ‘iff’ and ‘if’ and ‘then.’ These terms are part of what we have called the relevant “machinery”. They are nowhere defined truth-functionally or model-theoretically at all; they are meta-logical connectives. Now, label this machinery ‘$M$’.

$M$ includes (2), and also the background-logic proof theory $\Pi_M$ for how ‘iff’ and ‘if’ and ‘then’ are to be reasoned with deductively (e.g. we have on hand modus ponens). So the reducibility in question has happened in front of our eyes. To make it even clearer, abbreviate the assertion that the combination of (1) and $M$ can be used to prove (2) as

$$(M + [2]) \vdash_{\Pi_M} [1]$$  \hspace{1cm} (3)

What we have just seen is that in $L_{pc}$ getting to (1) reduces to (3). But this generalizes to every single truth-semantic target in $L_{pc}$, for every cognizer coming to know that this target holds. In fact, since first-order logic $L_1$ only augments $L_{pc}$ with additional machinery for quantification in the same style, establishing model theoretic assertions of truth in $L_1$, given what we have just seen, reduces to proof-theoretic semantics.

In stark contrast, here’s how easy and generalizable the situation is when our inference-theoretic approach is taken. Before the reader moves to the next paragraph, please look back and observe our repeated use of $\checkmark$ above for each formal inference that took place.

Let $a$ be some artificial agent that makes some ethical decision (or choice) at time $t$. We define “an agent $a$’s making an ethical decision at time $t$” as believing some proposition of the form $\Omega \phi$ at $t$, where $\phi$ is some formula in some formal language of some cognitive calculus (≈ some formal intensional logic), and $\Omega$ is specifically a deontic modal operator. This means that we can write that some $a$ has made an ethical decision at $t$ this way:

$$\exists \phi, \Omega[B(a, t, \Omega(\phi))]$$

An instantiation of this gives us a particular ethical decision; e.g.

$$B(a, t, O(\phi^*))$$

says that the agent has made an ethical decision that $\phi^*$ is obligated to be the case. If we wish to leave open what deontic operator is involved in a particular ethical decision, we can obviously write:

$$(+) \quad B(a, t, \Omega(\phi^*))$$

We shall let $d$ be a variable ranging over formulae in any of these forms.

Now, how do we achieve formal verification of an ethical decision $d$? Doing so is effortless, as long as every ethical decision for an artificial agent is made because it is inferred as the final proposition in a “natural” argument $\alpha$ or proof $\pi$ that is in accordance with the inference
schemata in play, which is exactly how we proceeded above (and are still proceeding and long have been for machine/robot/AI ethics; see e.g. [20]), for then we can simply follow an approach to program verification presented and advocated e.g. in [21, 22]. Let \( \Sigma \vdash_\alpha \phi \) say that there is an argument \( \alpha \) from undischarged suppositions (formulae) \( \Sigma \) to \( \phi \), and \( \Sigma \vdash_\pi \phi \) be the counterpart for a proof. Let \( C \) be a checker for arguments/proofs that receives one of these and returns either \( \text{VALID} = \checkmark \) or \( \text{INVALID} \); and suppose that this checker has been classically verified (which is easy: two pages of code, at most). This approach is generalizable, for note that uncertainty as we have systematically used it above creates no problems whatsoever, the simple reason being that inferences must still accord with inference schemata, and if they do all the way through, the overall conclusion/decision is certified.

We are thus done: An ethical decision \( d \) is verified (relative to given suppositions/premises) iff we have

\[
\Sigma \vdash_\alpha/\pi d \quad \text{and} \quad C(\alpha/\pi) = \text{VALID}.
\]

Note, finally, that nothing really changes if we move from decisions to actions: We can focus in that case on what our agent intends (using the \( \mathbf{I} \) modal operator in a cognitive calculus such as \( \mathcal{DCEC} \)) to do, and the case where the agent in fact does do what it/he/she intends to do.

4.1. Limitations

We certainly do not claim that our reasoning agent (or any AI currently in development today in our respective laboratories) is sufficiently capable to take the place of a human pilot. Nonetheless, what we hope to have conveyed is that in order to develop AI which could one day make autonomous, human-level, life-and-death decisions within aircraft piloting systems, the AI in question must have the ability to formally reason, by certifiably correct inferences, about declarative content represented in a cognitive calculus. In particular, it is crucial that this artificial agent be able to reason in a manner that dynamically takes explicit account of levels of uncertainty in ethically charged circumstances.

5. Conclusion & Future Work

We highlighted four flaws in the impressive, inspiring prior work of Dennis et al. [2] on autonomous systems which have to make decisions where each decision violates at least one ethical principle. We then presented our framework, which is able to solve the problems presented in the prior work, as well as a new class of problems unreachable by the prior work; specifically, problems in which all options are equally unethical, but have different likelihoods of violating those principles. Further work in this area includes modeling other types of uncertainty, such as the likelihood that completing a plan will actually achieve the goal.

Acknowledgements

The authors are grateful to AFOSR for their support of our development of cutting-edge reasoning systems for high levels of computational intelligence. We also thank ONR, both for their support of our current work on belief adjudication (as part of our vision to surmount Arrow's Impossibility Theorem and related theorems), and for their past support of our r&d in robot/machine ethics (primarily under a MURI to advance the science and engineering of moral competence in robots; PI M. Scheutz, Co-PI B. Malle, Co-PI S. Bringsjord).

\[\text{See also } [22, 23]. \text{ The first of these gives the first published roots of this novel approach to program verification, in work led by Arkoudas on proof checking.} [22]\]
References


19. S. Bringsjord and N. S. Govindarajulu, Given the Web, What is Intelligence, Really?, *Metaphilosophy* 43, 361 (2012), This URL is to a preprint of the paper [http://kryten.mm.rpi.edu/SB\NSG\Real\Intelligence\040912.pdf](http://kryten.mm.rpi.edu/SB\NSG\Real\Intelligence\040912.pdf).


## Appendix A. The Deontic Cognitive Event Calculus Signature

### Signature

\[
S ::= \text{Agent} \mid \text{ActionType} \mid \text{Action} \mid \text{Event} \mid \text{Moment} \mid \text{Fluent}
\]

\[
f ::= \\begin{align*}
\text{action} & : \text{Agent} \times \text{ActionType} \to \text{Action} \\
\text{initially} & : \text{Fluent} \to \text{Formula} \\
\text{holds} & : \text{Fluent} \times \text{Moment} \to \text{Formula} \\
\text{happens} & : \text{Event} \times \text{Moment} \to \text{Formula} \\
\text{clipped} & : \text{Moment} \times \text{Fluent} \times \text{Moment} \to \text{Formula} \\
\text{initiates} & : \text{Event} \times \text{Fluent} \times \text{Moment} \to \text{Formula} \\
\text{terminates} & : \text{Event} \times \text{Fluent} \times \text{Moment} \to \text{Formula} \\
\text{prior} & : \text{Moment} \times \text{Moment} \to \text{Formula}
\end{align*}
\]

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

\[
\phi ::= \\begin{align*}
q & : \text{Formula} \mid \neg \phi \mid \phi \land \psi \mid \phi \lor \psi \mid \forall x : \phi(x) \mid \exists x : \phi(x) \\
P(a, t, \phi) & \mid K(a, t, \phi) \mid S(a, b, t, \phi) \mid \text{S}(a, t, \phi) \\
C(t, \phi) & \mid B(a, t, \phi) \mid D(a, t, \phi) \mid I(a, t, \phi) \\
o(a, t, \phi, (\neg \text{happens}(\text{action}(a^*, \alpha), t'))) &
\end{align*}
\]

- Perceives, Knows, Says, Common-knowledge
- Believes, Desires, Intends, Ought-to
Appendix B. Proof of “Fuel low” Scenario of Dennis et al. in DCEC

Assumptions:

K(ua, now, LU(road, power))
K(ua, now, LU(road, people)) [1]
K(ua, now, (LU(road, power) ∧ LU(road, people)) → BestOption(road)) [best-option-axiom]
K(ua, now, ∀x BestOption(x) → O(ua, now, crisis, happens(action(ua, land(x)), next))) [best-implies-obligated]
B(ua, now, crisis) [agent-awareness]
now < next [order-of-time]

Goal: happens(action(ua, land(road)), next)

Sub-Proof 1: O(ua, now, crisis, happens(action(ua, land(road)), next))

K(ua, now, LU(road, power)) [1]
⇒ K(ua, now, LU(road, power)) ∧ K(ua, now, LU(road, people)) [∧-Intro]
⇒ LU(road, power) ∧ LU(road, people) [I4]
⇒ BestOption(road) [best-option-axiom with I4]
⇒ O(ua, now, crisis, happens(action(ua, land(road)), next)) [best-implies-obligated with I4]

Sub-Proof 2: B(ua, now, O(ua, now, crisis, happens(action(ua, land(road)), next)))

Chain 1:
K(ua, now, LU(road, power)) [1]
⇒ K(ua, now, LU(road, power)) ∧ K(ua, now, (LU road, people)) [∧-Intro]
⇒ B(ua, now, LU(road, power)) ∧ B(ua, now, (LU road, people)) [I2]
⇒ B(ua, now, LU(road, power) ∧ (LU road, people)) [I11b]
Chain 2:
\[ K(ua, now, (LU(road, power) \land LU(road, people)) \rightarrow \text{BestOption}(road) \) \quad \text{[best-option-axiom]}
\[ \Rightarrow B(ua, now, (LU(road, power) \land LU(road, people)) \rightarrow \text{BestOption}(road) \) \quad [I_2]
\[ \Rightarrow B(ua, now, \text{BestOption}(road)) \quad [I_{1a} \text{ with Chain 1}]
\]

Chain 3:
\[ K(ua, now, \forall x \text{BestOption}(x) \rightarrow O(ua, now, crisis, happens(\text{action}(ua, \text{land}(x)), \text{next})) \) \quad \text{[best-implies-obligated]}
\[ \Rightarrow B(ua, now, \forall x \text{BestOption}(x) \rightarrow O(ua, now, crisis, happens(\text{action}(ua, \text{land}(x)), \text{next})) \) \quad [I_2]
\[ \Rightarrow B(ua, now, \text{BestOption}(road) \rightarrow O(ua, now, crisis, happens(\text{action}(ua, \text{land}(road)), \text{next})) \) \quad [\forall\text{-Elim with road}]
\[ \Rightarrow B(ua, now, O(ua, now, crisis, happens(\text{action}(ua, \text{land}(road)), \text{next})) \) \quad [I_{1a} \text{ with Chain 2}]
\]

Proof of Goal: \(\text{happens(\text{action}(ua, \text{land}(road)), \text{next})}\)
\[ B(ua, now, \text{crisis}) \quad \text{[agent-awareness]}
\[ B(ua, now, \text{crisis}) \land B(ua, now, O(ua, now, crisis, happens(\text{action}(ua, \text{land}(road)), \text{next}))) \quad [\land\text{-Intro}]
\[ \land O(ua, now, crisis, happens(\text{action}(ua, \text{land}(road)), \text{next})) \]
\[ K(ua, now, I(ua, now, \text{happens(\text{action}(ua, \text{land}(road)), \text{next}))}) \quad [I_{14}]
\[ I(ua, now, \text{happens(\text{action}(ua, \text{land}(road)), \text{next}))} \quad [I_4]
\[ P(ua, now, \text{happens(\text{action}(ua, \text{land}(road)), \text{next}))} \quad [I_{13}]
\[ K(ua, now, \text{happens(\text{action}(ua, \text{land}(road)), \text{next}))} \quad [I_1]
\[ \text{happens(\text{action}(ua, \text{land}(road)), \text{next})} \quad [I_4]