Making Maximally Ethical Decisions via Cognitive Likelihood & Formal Planning

Michael Giancola, Selmer Bringsjord, Naveen Sundar Govindarajulu, and Carlos Varela

Abstract This chapter attempts to give an answer to the following question: Given an obligation and a set of potentially-inconsistent, ethically-charged beliefs, how can an artificially-intelligent agent ensure that its actions maximize the likelihood that the obligation is satisfied? Our approach to answering this question is in the intersection of several areas of research, including automated planning, reasoning with uncertainty, and argumentation. We exemplify our reasoning framework in a case study based on the famous, heroic ditching of US Airways Flight 1549, an event colloquially known as the "Miracle on the Hudson."

1 Introduction

This chapter attempts to give an answer to the following question: Given an obligation and a set of potentially-inconsistent, ethically-charged beliefs, how can an artificially-intelligent agent ensure that its actions maximize the likelihood that the obligation is satisfied? We present a framework for producing such agents. Our framework includes components from several intersecting areas of research, includ-

Michael Giancola
Rensselaer Polytechnic Institute, Rensselaer AI & Reasoning (RAIR) Lab, Department of Computer Science, Troy, NY, USA. e-mail: mike.j.giancola@gmail.com

Selmer Bringsjord
Rensselaer Polytechnic Institute, Rensselaer AI & Reasoning Lab, Department of Computer Science, Department of Cognitive Science, Troy, NY, USA. e-mail: selmer.bringsjord@gmail.com

Naveen Sundar Govindarajulu
Rensselaer Polytechnic Institute, Rensselaer AI & Reasoning Lab, Troy, NY, USA. e-mail: naveen.sundar.g@gmail.com

Carlos Varela
Rensselaer Polytechnic Institute, Worldwide Computing Lab, Department of Computer Science, Troy, NY, USA. e-mail: cvarela@cs.rpi.edu

ing automated planning, reasoning with uncertainty, and argumentation. Therefore, before we proceed with a description of our framework, we begin with a discussion of technical preliminaries which will enable the construction of our framework. This discussion includes a review of previously published concepts and new content. The former includes a review of *cognitive calculi* (§2.1, 2.2) and *automated planning* (§3), while the latter includes an introduction to *cognitive likelihood* (§2.3).

We then present our framework in §4, and its application in a case study based on US Airways Flight 1549 and its "miraculous" saving, in §5. Finally, we discuss some related work (§6) and conclude (§7).

2 Cognitive Calculi

Our[1] 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 [10], the precursor to this book chapter, and [11, 4]). In short, a cognitive calculus is a multi-operator quantified intensional logic built to capture all propositional attitudes in human cognition.[2] While a longer discussion of precisely what a cognitive calculus is is out of scope, the interested reader is pointed to Appendix A in Bringsjord et al. [5].

For purposes of this chapter, 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 **K**, belief **B**, and obligation **O**[3]) 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 Inductive Deontic Cognitive Event Calculus ($\mathcal{IDCEC}$) in the work described herein. We next review a predecessor of $\mathcal{IDCEC}$, the (deductive) Deontic Cognitive Event Calculus ($\mathcal{DCEC}$).

2.1 Deontic Cognitive Event Calculus

$\mathcal{DCEC}$ is fully captured in the following two boxes, titled $\mathcal{DCEC}$ **Signature** and $\mathcal{DCEC}$ **Inference Schemata**. They contain the sorts, function signatures, grammar, and inference schemata which comprise $\mathcal{DCEC}$. Notice that, while cognitive calculi can be constructed from n-order logic (for any value of $n \geq 0$), the standard[4] $\mathcal{DCEC}$ is built with a core of first-order logic.

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

[2] For information about such attitudes, see [20]; for a wonderful catalogue of all the major categories of human cognition, from *perceiving* to *fearing* to *remembering* to *saying* and beyond, see [1].

[3] See the box titled $\mathcal{DCEC}$ **Signature** within §2.1 for the rest of the modal operator descriptors.

[4] Note that several variants of $\mathcal{DCEC}$ have been formalized and deployed. For example, [4] uses $\mathcal{DCEC}^*$, a version of $\mathcal{DCEC}$ which allows for the formal modeling of self-reflective agents.

Also, an automated reasoner for $\mathcal{DCEC}$ — ShadowProver [12] — has been created, is available, and is under active development.

$\mathcal{DCEC}$ Signature

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

$$f ::= \begin{cases} \textit{action} : \textsf{Agent} \times \textsf{ActionType} \rightarrow \textsf{Action} \\ \textit{initially} : \textsf{Fluent} \rightarrow \textsf{Formula} \\ \textit{holds} : \textsf{Fluent} \times \textsf{Moment} \rightarrow \textsf{Formula} \\ \textit{happens} : \textsf{Event} \times \textsf{Moment} \rightarrow \textsf{Formula} \\ \textit{clipped} : \textsf{Moment} \times \textsf{Fluent} \times \textsf{Moment} \rightarrow \textsf{Formula} \\ \textit{initiates} : \textsf{Event} \times \textsf{Fluent} \times \textsf{Moment} \rightarrow \textsf{Formula} \\ \textit{terminates} : \textsf{Event} \times \textsf{Fluent} \times \textsf{Moment} \rightarrow \textsf{Formula} \\ \textit{prior} : \textsf{Moment} \times \textsf{Moment} \rightarrow \textsf{Formula} \end{cases}$$

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

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

Modal Operator Descriptors:
Perceives, **K**nows, **S**ays, **C**ommon-knowledge
Believes, **D**esires, **I**ntends, **O**ught-to

$\mathcal{DCEC}$ Inference Schemata

$$\frac{\mathbf{K}(a, t_1, \Gamma),\ \ \Gamma \vdash \phi,\ \ t_1 \leq t_2}{\mathbf{K}(a, t_2, \phi)}\ [I_{\mathbf{K}}] \qquad \frac{\mathbf{B}(a, t_1, \Gamma),\ \ \Gamma \vdash \phi,\ \ t_1 \leq t_2}{\mathbf{B}(a, t_2, \phi)}\ [I_{\mathbf{B}}]$$

$$\frac{}{\mathbf{C}(t, \mathbf{P}(a, t, \phi) \rightarrow \mathbf{K}(a, t, \phi))}\ [I_1] \qquad \frac{}{\mathbf{C}(t, \mathbf{K}(a, t, \phi) \rightarrow \mathbf{B}(a, t, \phi))}\ [I_2]$$

$$\frac{\mathbf{C}(t, \phi)\ t \leq t_1 \ldots t \leq t_n}{\mathbf{K}(a_1, t_1, \ldots \mathbf{K}(a_n, t_n, \phi) \ldots)}\ [I_3] \qquad \frac{\mathbf{K}(a, t, \phi)}{\phi}\ [I_4]$$

$$\frac{}{\mathbf{C}(t, \mathbf{K}(a, t_1, \phi_1 \rightarrow \phi_2)) \rightarrow \mathbf{K}(a, t_2, \phi_1) \rightarrow \mathbf{K}(a, t_3, \phi_2)}\ [I_5]$$

$$\frac{}{\mathbf{C}(t, \mathbf{B}(a, t_1, \phi_1 \rightarrow \phi_2)) \rightarrow \mathbf{B}(a, t_2, \phi_1) \rightarrow \mathbf{B}(a, t_3, \phi_2)}\ [I_6]$$

$$\frac{}{\mathbf{C}(t, \mathbf{C}(t_1, \phi_1 \rightarrow \phi_2)) \rightarrow \mathbf{C}(t_2, \phi_1) \rightarrow \mathbf{C}(t_3, \phi_2)}\ [I_7]$$

$$\frac{}{\mathbf{C}(t, \forall x.\, \phi \rightarrow \phi[x \mapsto t])}\ [I_8] \qquad \frac{}{\mathbf{C}(t, \phi_1 \leftrightarrow \phi_2 \rightarrow \neg\phi_2 \rightarrow \neg\phi_1)}\ [I_9]$$

$$\frac{}{\mathbf{C}(t, [\phi_1 \wedge \ldots \wedge \phi_n \rightarrow \phi] \rightarrow [\phi_1 \rightarrow \ldots \rightarrow \phi_n \rightarrow \phi])}\ [I_{10}]$$

$$\frac{\mathbf{B}(a, t, \phi)\ \ \mathbf{B}(a, t, \phi \rightarrow \psi)}{\mathbf{B}(a, t, \psi)}\ [I_{11a}] \qquad \frac{\mathbf{B}(a, t, \phi)\ \ \mathbf{B}(a, t, \psi)}{\mathbf{B}(a, t, \phi \wedge \psi)}\ [I_{11b}]$$

$$\frac{\mathbf{S}(s, h, t, \phi)}{\mathbf{B}(h, t, \mathbf{B}(s, t, \phi))}\ [I_{12}] \qquad \frac{\mathbf{I}(a, t, \textit{happens}(\textit{action}(a^*, \alpha), t'))}{\mathbf{P}(a, t, \textit{happens}(\textit{action}(a^*, \alpha), t'))}\ [I_{13}]$$

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

2.2 Inductive Deontic Cognitive Event Calculus

$\mathcal{DCEC}$ employs no uncertainty system (e.g., probability measures, *strength factors*, or likelihood measures) and hence is purely deductive. Therefore, as we wish to enable our agents to reason about situations involving uncertainty, we must ultimately utilize the *Inductive* $\mathcal{DCEC}$: $\mathcal{IDCEC}$.

In general, to go from a deductive to an inductive cognitive calculus, we require two components: (1) an uncertainty system, and (2) inference schemata that delineate the methods by which inferences linking formulae and other information can be used to build formally valid arguments.

The particular uncertainty system we use herein is discussed in §2.3. The inference schemata of $\mathcal{IDCEC}$ consist of the union of the set presented in §2.1 with that in the box titled **Additional Inference Schemata for** $\mathcal{IDCEC}$. Likewise, the signature of $\mathcal{IDCEC}$ subsumes that of the deductive $\mathcal{DCEC}$; the syntax of $\mathcal{IDCEC}$ also includes the forms given in the box titled **Additional Syntax for** $\mathcal{IDCEC}$.

Additional Syntax for $\mathcal{IDCEC}$

$$\phi ::= \left\{ \mathbf{B}^{\sigma}(a, t, \phi) \right.$$

$$\text{where } \sigma \in [-5, -4, \ldots, 4, 5]$$

Additional Inference Schemata for $\mathcal{IDCEC}$

$$\frac{\mathbf{P}(a, t_1, \phi_1), \;\; \Gamma \vdash t_1 < t_2}{\mathbf{B}^4(a, t_2, \phi)} \; [I^s_{\mathbf{P}}]$$

$$\frac{\mathbf{B}^{\sigma_1}(a, t_1, \phi_1), \ldots, \mathbf{B}^{\sigma_m}(a, t_m, \phi_m), \{\phi_1, \ldots, \phi_m\} \vdash \phi, \{\phi_1, \ldots, \phi_m\} \nvdash \zeta, \Gamma \vdash t_i < t}{\mathbf{B}^{min(\sigma_1, \ldots, \sigma_m)}(a, t, \phi)} \; [I^s_{\mathbf{B}}]$$

$$\text{where } \sigma \in [0, 1, \ldots, 5, 6]$$

$$\frac{}{\mathbf{C}(t, \mathbf{B}^{-\sigma}(a, t, \phi) \leftrightarrow \mathbf{B}^{\sigma}(a, t, \neg\phi))} \; [I^s_{\neg}]$$

Briefly, $\mathbf{B}^{\sigma}(a, t, \phi)$ denotes that agent a at time t believes ϕ with uncertainty σ. We justify in the next section the range of values for σ.

The first inference schema allows agents to infer evident beliefs ($\sigma = 4$, as defined in the next section) from what they perceive.[5] The second schema allows agents to infer a belief that is provable from the beliefs they currently assert, so long as the belief set is not inconsistent. In practice, we usually check that the belief set is consistent by attempting to prove a reserved propositional atom ζ which does not

[5] That is, what they perceive *externally*. We allow agents to infer *certain* beliefs on the basis of *internal* perceptions, but need not delve into this further for the purposes of the present work. $\mathcal{IDCEC}$ used herein makes no provision for these two modes of perception.

appear anywhere else; hence, ζ can only be proved if $\{\phi_1, \ldots, \phi_m\}$ is inconsistent.[6] The third schema specifies how uncertainty for $\neg\phi$ is derived from uncertainty for ϕ. The common knowledge construct in the third schema allows individual agents, in addition to the system, to handle both positive and negative uncertainty values.

As with $\mathcal{DCEC}$, an automated reasoner for $\mathcal{IDCEC}$ — ShadowAdjudicator [10] — is under active development.

As mentioned at the opening of this subsection, in addition to inference schemata, we also require an uncertainty system. The specific uncertainty concept we employ herein is *cognitive likelihood*, which we now discuss.

2.3 Cognitive Likelihood

Our approach to quantifying the uncertainty of beliefs within cognitive calculi eschews traditional probability values in favor of *likelihood* values. The 11 likelihood values employed in this chapter are shown in Table 1.

Table 1 The 11 Cognitive Likelihood Values

Numerical	**Linguistic**
5	CERTAIN
4	EVIDENT
3	OVERWHELMINGLY LIKELY = BEYOND REASONABLE DOUBT
2	LIKELY
1	MORE LIKELY THAN NOT
0	COUNTERBALANCED
-1	MORE UNLIKELY THAN NOT
-2	UNLIKELY
-3	OVERWHELMINGLY UNLIKELY = BEYOND REASONABLE BELIEF
-4	EVIDENTLY NOT
-5	CERTAINLY NOT

Likelihood values can be obtained in either of two ways; both ways immediately reveal that we take likelihood to be *subjective*. The first way is to take as primitive a cognitive binary relation on formulae from the perspective of a rational agent (e.g., ϕ is *more reasonable than* ψ), and then build up formally to the partial or total order in question. This approach is first formalized in [13] and is deployed in e.g. the precursor to this chapter, [10]. Another approach, the one taken here, is to independently justify each likelihood value by appeal to rational human-level cognition. We do so (briefly) next.

[6] In connection with standard practice in mathematical logic, ζ functions essentially like $\perp$ as or '0=1.'

First, note that, because of schema $[I_{\neg}^{s}]$ presented in §2.2, we only need to define the non-negative likelihood values (as a negative likelihood value for belief in some formula ϕ is equivalent to a positive likelihood value for belief in $\neg\phi$).

That which is CERTAIN applies to propositions that a perfectly rational human-level cognizer would affirm as such — that 2+2=4 (Base-10), that $0 \neq 1$, and so on for any theorem that has been certifiably deduced from what is itself CERTAIN. Propositions that are CERTAIN needn't be mathematical in nature, only absolutely indubitable; e.g., that if something has both the properties R_1 and R_2, then it has the property R_1 qualifies.

Propositions are EVIDENT typically when they are given by immediate perception in the absence of conditions known to frequently cause illusory perception. For example, currently the lead author perceives his laptop's screen in front of him, and hence that there is such a screen in front of him is EVIDENT.

Next, as to the concept of BEYOND REASONABLE DOUBT, it has a long-standing history in many legal systems (such as e.g. the one long operative in the U.S.), being the level of argument that a prosecutor must provide in order for a court to convict the defendant. In this context, the following is required (emphasis ours):

> To establish the standard of proof *beyond reasonable doubt*, there must be a plausible explanation of the evidence that includes all of the elements of the crime and, *in addition*, there must be **no plausible explanation that is consistent with innocence**. [16] (§3.2.2. para. 12)

We next move to the center of the likelihood continuum, COUNTERBALANCED, which indicates no belief for or against a formula. From there, MORE LIKELY THAN NOT indicates the lowest level of belief above COUNTERBALANCED. Only a weak argument is required to reach this level.

Finally, we can define LIKELY as any belief whose likelihood is less than BEYOND REASONABLE DOUBT and more than MORE LIKELY THAN NOT.

3 Highly-Expressive Automated Planning

The final necessary component of our framework is an automated planner, in particular one that is fully compatible with our formalisms and their emphasis on declarative content and automated reasoning over that content in uncertain situations. The first modern automated planner was the Stanford Research Institute Problem Solver (STRIPS) [9], which produced a framework for planning upon which many modern planners are built.

The setup of a STRIPS problem is as follows. There is a set of formulae describing the initial state of the *world*, a set of *actions* which describe methods by which the planner can change the world state, and a *goal* set which denotes those formulae that the agent in question wishes to hold. The actions consist of three components: (1) a set of preconditions (formulae which must hold in order to perform the action), (2) a set of additions (formulae that will be added to the world by taking the action), and (3) a set of deletions (formulae to be removed from the world by taking the action).

The expressivity of formulae used to represent the world, actions, and goal was limited to propositional statements. For example, the goal that the book is not on the table could be represented by ¬On(book, table). In this work, we will need to be able to use quantified formulae, e.g. ¬∃ x On(x, table), to describe the world and goal.

The Planning Domain Definition Language (PDDL) [18] is a STRIPS-style planning language, which also supports quantification over zero-order formulae. While some quantification is supported, PDDL has serious restrictions on the syntax of formulae that can be supported. Arbitrary first-order-logic formulae are not allowed. Further, PDDL does not support modal operators such as those for *belief*, *knowledge*, or *obligation*; these are are necessary for modeling states of minds of agents. (E.g., in our case study below, we would ultimately want AI that is able to bring about "mental" goals, such as an a pilot's believing that such and such a course of action is feasible.) Reasoning with such mental states is crucial in ethically charged situations.[7] Formulae of the following nature, which require the ability to nest such modal operators, cannot be expressed in PDDL:

Alice *believes* that all pilots *believe*, before entering a cockpit, that they *know* ϕ.

$$\mathbf{B}(alice, t, \forall x \ \exists t_0 t_1 \ \mathbf{B}(x, t_0, \mathbf{K}(x, t_0, \phi)) \land \mathit{EntersCockpit}(x, t_1) \land t_0 < t_1)$$

Another major limitation of the PDDL family of languages is that they require a finite and fixed universe of objects to be specified beforehand. In many uncertain situations, this is not realistic, as the number of relevant objects and entities will be unknown. Consider a situation in which a firefighting robot has to enter a building with the goal of rescuing any humans in the building. The agent has no prior knowledge of the number of humans in the building. PDDL languages are not directly amenable to modeling such situations.

Overall, then, we need a planning formalism (with an associated automated planner) that can handle arbitrary formulae for describing the world, states of minds, and an unknown set of objects. For a planner with such capabilities, we turn to Spectra [14], a STRIPS-style planner which can be integrated with reasoners for cognitive calculi [12]. While there is a efficiency disadvantage in using a more expressive planning formalism, efficiency gains in reasoning with cognitive calculi can be transferred to efficiency gains in Spectra.

4 Selecting Plans Using Cognitive Likelihood

In our framework, agents are given the following: (1) an obligation, (2) knowledge regarding the conditions required to satisfy the obligation, and (3) a set of (potentially inconsistent) ethically-charged beliefs regarding actions the agent can take to affect the status of the obligation. The agents make maximally ethical decisions by taking a course of action which maximizes the agent's belief that the obligations will stay (or become) satisfied.

[7] See [6] for an example of an ethically-charged situation in which the ascription of mental states is crucial to the success of the AI agents used.

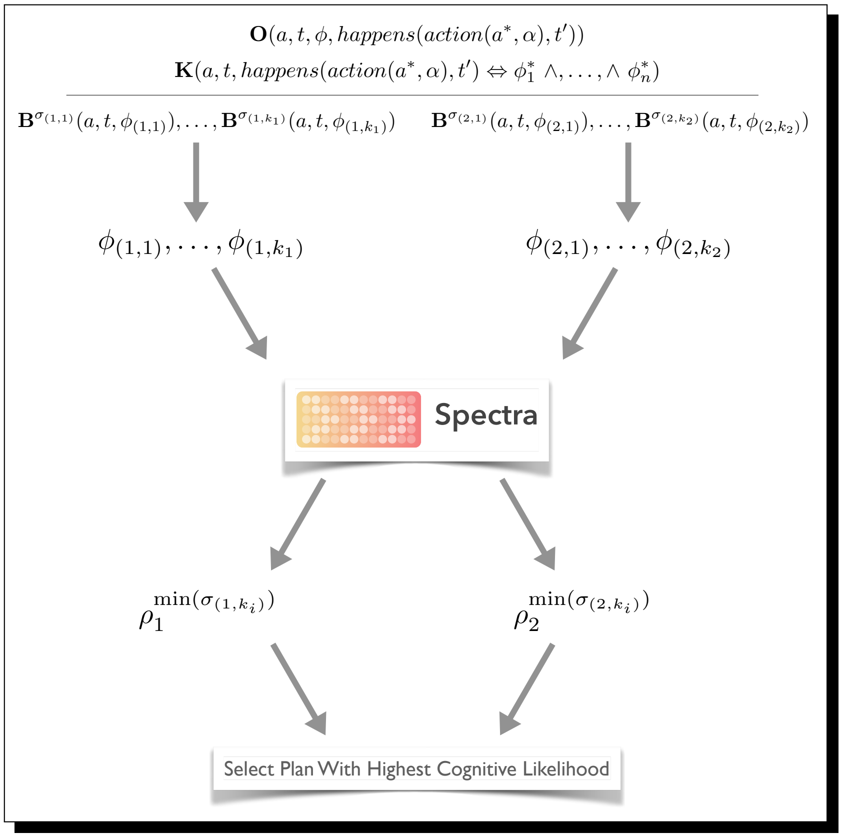
$\mathbf{O}(a, t, \phi, happens(action(a^*, \alpha), t'))$
$\mathbf{K}(a, t, happens(action(a^*, \alpha), t') \Leftrightarrow \phi_1^* \wedge, \ldots, \wedge\ \phi_n^*)$
$\mathbf{B}^{\sigma_{(1,1)}}(a, t, \phi_{(1,1)}), \ldots, \mathbf{B}^{\sigma_{(1,k_1)}}(a, t, \phi_{(1,k_1)})$
$\mathbf{B}^{\sigma_{(2,1)}}(a, t, \phi_{(2,1)}), \ldots, \mathbf{B}^{\sigma_{(2,k_2)}}(a, t, \phi_{(2,k_2)})$
$\phi_{(1,1)}, \ldots, \phi_{(1,k_1)}$
$\phi_{(2,1)}, \ldots, \phi_{(2,k_2)}$
Spectra
$\rho_1^{\min(\sigma_{(1,k_i)})}$
$\rho_2^{\min(\sigma_{(2,k_i)})}$
Select Plan With Highest Cognitive Likelihood

Fig. 1 A Framework for Selecting Maximally Ethical Plans. *This diagram shows two belief subsets, from which Spectra generates one plan each. More generally, there could be an arbitrary number of belief subsets, as well as an arbitrary number of plans generated. However, this is not shown in order to simplify the illustration.*

The decision-making framework is outlined pictorally in Figure 1. An agent a is obligated to perform some action α, given that it believes some precondition ϕ holds. It also knows the conditions that will enable α to happen (in Figure 1, $\phi_1^*, \ldots, \phi_n^*$).

Next, a has a set of beliefs regarding formulae pertinent to its obligations. Various subsets of those beliefs (in Figure 1, $\phi_{(1,1)}, \ldots, \phi_{(1,k_1)}$ and $\phi_{(2,1)}, \ldots, \phi_{(2,k_2)}$), are passed to Spectra, with the goal of generating plans which cause α to occur. We assign each of those plans a likelihood based on the likelihood of the weakest belief required to generate the plan. Finally, we select the plan with the highest likelihood as the one to enact.

5 Case Study: The Miracle on the Hudson

To display our framework "in action," we consider two potential arguments concerning what decision should be made in the case of US Airways Flight 1549, colloquially known as the "Miracle on the Hudson." Namely, after losing thrust in both engines, the pilots had to quickly make the decision where to attempt an emergency landing, ultimately considering the following options: (a) attempt to return to LaGuardia Airport (LGA), (b) attempt to reach Teterboro Airport (TEB), or (c) attempt to land in the Hudson River.

5.1 Recounting US Airways Flight 1549

Our case study is based on the 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; this compromised both engines. In fact, 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]."[8] 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, thereby saving the lives of everyone onboard. Simulations of the accident have come to the conclusion that Sullenberger's decision was optimal given the preconditions [21].

5.2 The Setup

The setup of the framework for our case study is as follows. We have three agents: a_1 and a_2 will each present two inconsistent arguments regarding where the plane should be landed (in the following subsection), and a^* is the adjudicator who will decide which argument and plan to proceed with.

We denote the moment after the plane flew into the flock of geese as t^*. At that time, a^* believes there is an emergency, and consequently, the agent is obligated to ensure that the landing site it selects is safe.

[8] This quote, recorded by the in-flight cockpit voice recorder (CVR), was retrieved from the NTSB Accident Report [15].

$$\mathbf{B}(a^*, t^*, \mathit{emergency})$$
$$\mathbf{O}(a, t^*, \mathit{emergency}, \mathit{happens}(\mathit{action}(a^*, \mathit{ensure\_safe}(\mathit{landing\_site})), t^{*'}))$$

The agent also knows the conditions required for a landing site to be safe: it must be close enough to reach, long and wide enough, and far enough from people, as without thrust, the pilots' ability to maneuver the plane will be more limited than usual.

$$\begin{aligned}&\mathbf{K}\Big(a, t^*, \mathit{happens}\,\Big(\mathit{action}(a^*, \mathit{ensure\_safe}(\mathit{landing\_site})), t^{*'}\Big)\\ &\qquad\qquad \Leftrightarrow\ \vdash \mathit{Safe}(\mathit{landing\_site})\Big)\\ &\mathbf{K}\left(a, t^*, \forall \ell\ \mathit{Safe}(\ell) \Leftrightarrow \bigwedge \left\{\begin{array}{c}\mathit{CloseEnough}(\ell),\\ \mathit{LongEnough}(\ell),\\ \mathit{WideEnough}(\ell),\\ \mathit{FarEnoughFromPeople}(\ell)\end{array}\right\}\right)\end{aligned}$$

5.3 The Arguments

We next give two arguments in favor of selecting different landing locations based on the conditions of Flight 1549, then show how our framework would generate plans for each, and finally, how it would select a plan to execute.

5.3.1 Argument 1

The first agent argues for the following two statements:

$$\mathbf{B}^3(a_1, t^*, \mathit{CloseEnough}(\mathit{lga}))$$
$$\mathbf{B}^1(a_1, t^*, \mathit{FarEnoughFromPeople}(\mathit{lga}))$$

The first formula states that it is OVERWHELMINGLY LIKELY that LaGuardia Airport was close enough for the pilots to successfully land there. This is justified by the several studies and simulations performed since the event which identified many feasible trajectories to enable landing at several different runways at LGA (e.g. see [21, 2]).

The second states that it is MORE LIKELY THAN NOT that LGA is far enough from people to ensure a safe landing despite the conditions (i.e. loss of thrust in both engines at low altitude, occurring in — to quote Captain Sullenberger — "a highly developed, metropolitan area" [19]). The likelihood is necessarily weak, as the corresponding justification is weak. As there is no data to go on, one can only speculate that based on the Captain's training, and Air Traffic Control's ability to clear a runway in time, that it is possible that the plane could have been landed at LGA without harming anyone on the ground.

5.3.2 Argument 2

The second argument asserts the following two statements:

$$\mathbf{B}^{-2}(a_2, t^*, \mathit{CloseEnough}(\mathit{teb}))$$
$$\mathbf{B}^{-2}(a_2, t^*, \mathit{FarEnoughFromPeople}(\mathit{lga}))$$

The first statement, that it is UNLIKELY that Teterboro Airport is close enough, was asserted without justification by Captain Sullenberger in the public hearing on the accident [19]. He likely intended to imply an implicit justification that it was obvious to him based on his experience as a pilot.

Second, a_2 asserts that it is UNLIKELY that LGA was far enough from people to ensure a safe landing. Note that this belief is directly inconsistent with a belief of a_1; namely, $\mathbf{B}^1(a_1, t^*, \mathit{FarEnoughFromPeople}(\mathit{lga}))$. Again, this is justified by a statement provided by Captain Sullenberger during the public hearing:

> Looking at where we were and how much time, altitude, and distance would be required to turn back toward LaGuardia and then fly toward LaGuardia, I determined quickly that that was going to be problematic, and it would not be a realistic choice, and I couldn't afford to be wrong. [19]

It is clear that, had Captain Sullenberger chosen to attempt a landing at LGA, he would've risked the lives of people at and around LGA, *in addition* to the inevitable risk already imposed on those in the plane by the emergency.

5.4 The Framework, Applied

We now present the application of our framework, in order to adjudicate these clearly inconsistent arguments[9] and generate a plan. First, the content of each agent's beliefs are passed separately to Spectra. Therefore, the first agent passes:

$$\mathit{CloseEnough}(\mathit{lga})$$
$$\mathit{FarEnoughFromPeople}(\mathit{lga})$$

and the second agent passes:

[9] We note that when we refer to *arguments* being inconsistent, this is to say that the arguments each assert a set of formulae, and from the union of those sets, a contradiction can be deduced.

$$\neg \mathit{CloseEnough}(\mathit{teb})$$
$$\neg \mathit{FarEnoughFromPeople}(\mathit{lga})$$

In order to generate plans, Spectra is given the following actions:

```
(define-action considerRunwayLanding [?r]
  {:preconditions [(CloseEnough ?r)
                   (FarEnoughFromPeople ?r)
                  ]

   :additions     [(LongEnough ?r)
                   (WideEnough ?r)
                  ]

   :deletions     [ ]
  }
)
```

```
(define-action considerTerrainLanding []
  {:preconditions [(not (and (Safe lga) (Safe teb)))
                  ]

   :additions     [(CloseEnough hud)
                   (LongEnough  hud)
                   (WideEnough  hud)
                   (FarFromPeopleEnough hud)
                  ]

   :deletions     [ ]
  }
)
```

The first action requires that, in order to consider landing at a particular runway, the ethically-charged propositions are first satisfied. It then adds that the runway satisfies basic requirements. The idea here is that, if implemented "for real," Spectra would be integrated with systems which could provide the necessary data, i.e. the length and width of the runway being considered, and the length and width required.

The second action allows the planner to consider off-runway landing options *only* if the runway options have been exhausted (that is, it has been determined that none of them meet the imposed safety requirements). As with the first action, Spectra would need to be integrated with another system. In this case, our simulation assumes Spectra would have access to a vision-based landing-site detection system, such as that presented in [23]. Shen et al. specify that (emphasis ours):

> A landing-site is considered safe only if its surface is *smooth* and if its *length* and *width* are adequate. [23] (pg. 295)

At the public hearing, Sullenberger stated that (emphasis ours):

> [Other than LGA or TEB,] the only place in a highly developed, metropolitan area, *long enough*, *wide enough*, *smooth enough* to land was the river. [19] (pg. 25)

Hence we can confidently say that, were Shen et al.'s system integrated with Spectra in this case, the river would have been the only landing option returned.

Each agent's input to Spectra returns a single plan. The former indicates that the pilot can land LGA, as all safety requirements have been satisfied. Alternatively, the latter is able to prove that neither LGA nor TEB are safe options, and hence seeks out off-runway options, and finds the Hudson as the only option.

Finally, note that the weakest likelihood used by agent 1 is MORE LIKELY THAN NOT (= 1) and the weakest of agent 2's argument is LIKELY (= 2). Hence the framework would conclude that agent 2's argument, and corresponding plan, are to be used.

6 Related Work

The most directly related work is [10], a precursor to this book chapter, in which the authors first used uncertainty-infused cognitive calculi to reason about the ethical decision-making involved in the Miracle on the Hudson. Like this paper, it employed an uncertainty system: it used strength factors [13] whereas the present work uses cognitive likelihood. Also, one shortcoming of the prior paper was that the AI agent featured there was given the Hudson as a landing option from the outset. In our subsequent work, reported herein, we show how an AI can creatively[10] find the Hudson on its own (i.e. by deploying a vision-based landing-site detection system such as that presented in [23]).

Giancola et al. [10] (and consequently, this work as well) was inspired by [8]. This paper presented a framework intended to ensure that autonomous systems[11] 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 their system 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 work has several shortcomings by our metrics; these deficiencies are expounded and overcome in [10]. Briefly, (1) the model-checking process used in [8] is too slow for practical applications, (2) the formalism used, a Belief-Desire-Intention (BDI) language, is too inexpressive, and (3) there is no conception of uncertainty in their system. Giancola et al. [10] also elaborates on the infeasibility of formal verification based on model checking in general.

[10] A precise account of what sort of creativity is used by the AI in reasoning to the Hudson as a landing area is beyond the present paper. A number of increasingly impressive types of creativity in an artificial agent are laid out in [3]. Note that at least one expert on AI and creativity, Cope, would classify the AI described in the present paper as creative; see [7].

[11] We use the term 'system' in reference to the work of Dennis et al. as this is the term they use in their own work. However, in keeping with the terminology of standard textbooks in AI [22, 17], we use the term 'agent' in our work.

Another area of prior work is the creation of linear models for evaluating potential landing sites [2] and trajectories [21] for Flight 1549. We applaud these works, as they produced systems which could have given the pilots actionable data (i.e. trajectories and landing sites to use to avoid landing in the Hudson). We envision that ultimately, a system could be engineered which integrates the work herein with the linear models of [2, 21], in order to generate plans that reflect relevant data gathered by an aircraft's instruments as well as the pilots' beliefs and ethical concerns.

7 Conclusion

We presented a framework for AI agents to formally produce a maximally ethical plan based on uncertain, ethically-charged beliefs. We then showed how this framework could potentially be deployed using US Airways Flight 1549 as a case study. As with much logic-based AI research, one major area of future work is developing methods of integrating the framework proposed herein with the necessary components to enable practical usage (e.g. vision systems, aircraft sensors, and connectionist AI systems such as neural networks).

Acknowledgements The authors are grateful to AFOSR for its longstanding support of Bringsjord et al.'s development of cutting-edge formalisms and automated reasoning and planning systems for high levels of computational intelligence. We deeply thank ONR, both for their support of our current work on belief adjudication (as part of our development of AI that surmounts Arrow's Impossibility Theorem and related theorems), and for their past support of our r&d in robot/machine/AI ethics (primarily under a MURI to advance the science and engineering of moral competence in robots).

References

[1] Ashcraft M, Radvansky G (2013) Cognition. Pearson, London, UK, This is the 6th edition.

[2] Atkins E (2010) Emergency Landing Automation Aids: An Evaluation Inspired by US Airways Flight 1549. In: AIAA Infotech@Aerospace 2010, DOI 10.2514/6.2010-3381

[3] Bringsjord S, Sen A (2016) On Creative Self-Driving Cars: Hire the Computational Logicians, Fast. Applied Artificial Intelligence 30:758–786, URL `http://kryten.mm.rpi.edu/SB_AS_CreativeSelf-DrivingCars_0323161130NY.pdf`, The URL here goes only to an uncorrected preprint.

[4] Bringsjord S, Govindarajulu N, Thero D, 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), Chicago, IL, pp 22–29, URL `http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=6883275`, IEEE Catalog Number: CFP14ETI-POD. Papers from the *Proceedings* can be downloaded from IEEE at URL provided here.

[5] Bringsjord S, Govindarajulu NS, Licato J, Giancola M (2020) Learning *Ex Nihilo*. In: Danoy G, Pang J, Sutcliffe G (eds) GCAI 2020. 6th Global Conference on Artificial Intelligence (GCAI 2020), EasyChair, EPiC Series in Computing, vol 72, pp 1–27, DOI 10.29007/ggcf, URL https://easychair.org/publications/paper/NzWG

[6] Bringsjord S, Govindarajulu N, Giancola M (forthcoming) Automated Argument Adjudication to Solve Ethical Problems in Multi-Agent Environments. Paladyn, Journal of Behavioral Robotics URL http://kryten.mm.rpi.edu/AutomatedArgumentAdjudicationPaladyn102120.pdf, The URL here goes to a preprint as of 102120.

[7] Cope D (2005) Computer Models of Muscial Creativity. MIT Press, Cambridge, MA

[8] Dennis L, Fisher M, Slavkovik M, Webster M (2016) Formal Verification of Ethical Choices in Autonomous Systems. Robotics and Autonomous Systems 77:1–14, URL http://dx.doi.org/10.1016/j.robot.2015.11.012

[9] Fikes RE, Nilsson NJ (1971) STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving. Artificial Intelligence 2(3-4):189–208

[10] Giancola M, Bringsjord S, Govindarajulu NS, Varela C (2020) Ethical Reasoning for Autonomous Agents Under Uncertainty. In: Tokhi M, Ferreira M, Govindarajulu N, Silva M, Kadar E, Wang J, Kaur A (eds) Smart Living and Quality Health with Robots, Proceedings of ICRES 2020, CLAWAR, London, UK, pp 26–41, available at http://kryten.mm.rpi.edu/MG_SB_NSG_CV_LogicizationMiracleOnHudson.pdf. The ShadowAdjudicator system can be obtained here: https://github.com/RAIRLab/ShadowAdjudicator.

[11] Govindarajulu N, Bringsjord S (2017) On Automating the Doctrine of Double Effect. In: Sierra C (ed) Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI-17), International Joint Conferences on Artificial Intelligence, pp 4722–4730, DOI 10.24963/ijcai.2017/658, URL https://doi.org/10.24963/ijcai.2017/658

[12] Govindarajulu N, Bringsjord S, Peveler M (2019) On Quantified Modal Theorem Proving for Modeling Ethics. In: Suda M, Winkler S (eds) Proceedings of the Second International Workshop on Automated Reasoning: Challenges, Applications, Directions, Exemplary Achievements (ARCADE 2019), Electronic Proceedings in Theoretical Computer Science, vol 311, Open Publishing Association, Waterloo, Australia, pp 43–49, URL http://eptcs.web.cse.unsw.edu.au/paper.cgi?ARCADE2019.7.pdf, The ShadowProver system can be obtained here: https://naveensundarg.github.io/prover/.

[13] Govindarajulu NS, Bringsjord S (2017) Strength Factors: An Uncertainty System for Quantified Modal Logic. In: Belle V, Cussens J, Finger M, Godo L, Prade H, Qi G (eds) Proceedings of the IJCAI Workshop on “Logical Foundations for Uncertainty and Machine Learning" (LFU-2017), Melbourne, Australia, pp 34–40, URL http://homepages.inf.ed.ac.uk/vbelle/workshops/lfu17/proc.pdf

[14] Govindarajulu, Naveen Sundar (2017) Spectra. URL https://naveensundarg.github.io/Spectra/

[15] Hersman DA, Hart CA, Sumwalt RL (2010) Loss of Thrust in Both Engines After Encountering a Flock of Birds and Subsequent Ditching on the Hudson River. Accident Report NTSB/AAR-10/03, National Transportation Safety Board (NTSB)

[16] Ho HL (2015) The Legal Concept of Evidence. In: Zalta EN (ed) The Stanford Encyclopedia of Philosophy, winter 2015 edn, Metaphysics Research Lab, Stanford University

[17] Luger G (2008) Artificial Intelligence: Structures and Strategies for Complex Problem Solving (6th Edition). Pearson, London, UK

[18] Mcdermott D, Ghallab M, Howe A, Knoblock C, Ram A, Veloso M, Weld D, Wilkins D (1998) PDDL – The Planning Domain Definition Language. Tech. Rep. CVC TR-98-003, Yale Center for Computational Vision and Control

[19] National Transportation Safety Board (NTSB) (2009) Transcript - Public Hearing Day 1. Landing of US Airways Flight 1549, Airbus A320, N106US, in the Hudson River. URL: https://data.ntsb.gov/Docket?NTSBNumber=DCA09MA026

[20] Nelson M (2015) Propositional Attitude Reports. In: Zalta E (ed) The Stanford Encyclopedia of Philosophy, URL https://plato.stanford.edu/entries/prop-attitude-reports

[21] Paul S, Hole F, Zytek A, Varela CA (2017) Flight trajectory planning for fixed wing aircraft in loss of thrust emergencies. In: Dynamic Data-Driven Application Systems (DDDAS 2017), Cambridge, MA, URL http://arxiv.org/abs/1711.00716

[22] Russell S, Norvig P (2020) Artificial Intelligence: A Modern Approach. Pearson, New York, NY, Fourth edition.

[23] Shen YF, Rahman ZU, Krusienski D, Li J (2013) A Vision-Based Automatic Safe Landing-Site Detection System. IEEE Transactions on Aerospace and Electronic Systems 49(1):294–311